

DB2 照会報告書作成プログラム



TSO/CICS 用 DB2 QMF インストールおよび管理 の手引き

バージョン 8 リリース 1

DB2 照会報告書作成プログラム



TSO/CICS 用 DB2 QMF インストールおよび管理 の手引き

バージョン 8 リリース 1

お願い

本書、および本書によってサポートされる製品 (プロダクト) をご使用になる前に、 415 ページの『付録 F. 特記事項』に記載されている一般情報を必ずお読みください。

本書は、IBM DB2 Universal Database Server (z/OS 版) (DB2 UDB (z/OS 版)) バージョン 8 リリース 1 (5625-DB2) のフィーチャーである、IBM DB2 照会報告書作成プログラム (QMF (TSO/CICS 版)) バージョン 8 リリース 1 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)

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

原典： GC18-7444-00
DB2 Query Management Facility
Installing and Managing DB2 QMF for TSO/CICS
Version 8 Release 1

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 2003.12

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 1982, 2003. All rights reserved.

© Copyright IBM Japan 2003

目次

本書について	vii
本書の対象読者	vii
前提知識	vii
本書の使用法	viii
各国語機能情報の表し方	viii

第 1 部 TSO/CICS 用 QMF バージョン 8.1 のインストール 1

第 1 章 QMF およびインストール・プロセスの概要 3

QMF の概要	3
QMF が他のデータベースのデータにアクセスする方法	4
データベース・インストール・プロセスの概要	6
DB2 UDB (z/OS 版) の QMF 要件	6
ロング・ネームのサポート	11
QMF インストールのロードマップ	13

第 2 章 インストールの計画 15

ハードウェア要件	15
前提条件のソフトウェア	15
記憶域要件の計画	16
パフォーマンスを向上させるためのモジュールの移動	17
記憶域の見積もり	19
CICS のもとでの QMF の計画	19
DB2 (AIX 版) の QMF	20
ワークシートへの記入	21
QMF インストール・ジョブ情報	22

第 3 章 DB2 アプリケーション・リクエストターとしての QMF の構成 27

QMF 互換モード・インストール	27
QMF 新機能モードのインストール	28
QMF 互換モードへの移行	28

第 4 章 QMF の TSO 用調整 29

TSO ログオン・プロシージャの作成	29
QMF の開始	33

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)	36
---	----

第 5 章 QMF の CICS 用調整 39

CICS の DB2 に対する QMF の記述	39
QMF/GDDM データ・セットの定義およびロード	39
QMF 提供の管理プログラムの変換、アSEMBル、およびリンク・エディット	41
CICS 管理テーブルの更新	41
QMF プロファイルの調整	42
CICS 始動ジョブ・ストリームの更新	43

第 6 章 QMF 互換モード用リモート・サーバーの構成 45

Linux、UNIX、および Windows 版 DB2 アプリケーションとしての QMF 互換モードの構成	45
DB2 UDB (iSeries 版) アプリケーションとしての QMF 互換モードの構成	48
DB2 UDB (z/OS 版) アプリケーションとしての QMF 互換モードの構成	50
すべてのサーバー・タイプの QMF 互換モードへの移行	50

第 7 章 全サーバー・タイプ用の QMF 新機能モードの構成 53

QMF 新機能モードのインストール	53
QMF 新機能モードへの移行	54

第 8 章 QMF インストールのテスト 57

IVP の実行 (TSO)	57
IVP の実行 (CICS)	59
QMF アプリケーション照会およびプロシージャのインストール (TSO)	62
オプション: バッチ・モード IVP の実行	63
インストール後の終結処理	65

第 2 部 TSO/CICS 用 QMF の管理 69

第 9 章 QMF の開始 73

z/OS 上での QMF のセットアップおよび開始	73
第 10 章 開始手順のカスタマイズ	87
各セッション用の仮想記憶域の量の選択	87
プログラム・パラメーターの要約	102
第 11 章 QMF セッション制御機能	105
Q.SYSTEM_INI のインストール	105
Q.SYSTEM_INI プロシージャを実行する時期	105
Q.SYSTEM_INI の使用法	105
ユーザー・セッション・プロシージャの例	106
オブジェクトのリストを表示するプロシージャ	108
セキュリティーおよびセッション・プロシージャの共用	108
診断の考慮事項	108
z/OS での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	109
第 12 章 QMF インストール・ユーザー出口 (DSQUOPTS)	111
z/OS	111
第 13 章 エンド・ユーザーのための QMF サポートの確立	113
TSO/CICS 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	113
SQL 特権の付与と取り消し	124
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	126
拡張オブジェクト・リストの活動化	139
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	146
ユーザーが図表をサポートできるようにする	151
QMF コントロール表を使用した QMF オブジェクトの保守	153
z/OS での DB2 サブシステムの保守	161
DB2 表を使用する表および視点の保守	164
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	165
ユーザー用文書編集インターフェースのカスタマイズ	166
QMF EDIT コマンドのカスタマイズ	173
NLF 環境での英語サポートのイネープリング	175
グローバル変数を使用した通貨記号の定義	176

第 14 章 QMF NLF の計画およびインストール	177
プロファイル表と NLF	177
QMF NLF の計画	177
IBM ソフトウェア配布 (ISD) テープ	179
インストール・プロセス	180
QMF NLF のインストール	181
第 15 章 ユーザーによるオブジェクトの印刷	199
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	199
GDDM サービスを使用して印刷を処理する	200
QMF サービスを使用して印刷を処理する	210
印刷機能キーの同義語の定義	212
オブジェクトの印刷	214
第 16 章 QMF コマンドのカスタマイズ	215
QMF が提供するデフォルトの同義語の使用	215
コマンド同義語表の作成	218
コマンド同義語定義を表へ入力	220
同義語のアクティブ化	226
コマンド同義語表の保守の最小化	227
第 17 章 QMF 機能キーのカスタマイズ	231
カスタマイズしたいキーの選択	231
機能キー表の作成	234
ユーザーの機能キー定義を表へ入力	235
カスタマイズしたいパネルの識別	240
新規機能キー定義のアクティブ化	243
機能キー表の検査と問題診断	244
第 18 章 QMF 書式のためのユーザー独自の編集コードの作成	247
QMF 書式	247
編集コードの選択	248
DATE、TIME、および TIMESTAMP 情報の処理	249
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	250
出力ルーチンとの情報の受け渡し	251
QMF 終了時に制御を出力ルーチンに渡す	256
編集ルーチンを HLASM (ハイレベル・アセンブラ) で作成	256
言語環境プログラム (LE) を使用せずに PL/I で編集ルーチンを作成	263

言語環境プログラム (LE) を使用して PL/I で編集ルーチンを作成	266
編集ルーチンを z/OS 上の CICS 用に PL/I で作成	269
言語環境プログラム (LE) を使用せずに COBOL で編集ルーチンを作成	272
言語環境プログラム (LE) を使用して COBOL で編集ルーチンを作成	276
編集ルーチンを z/OS 上の CICS 用に COBOL で作成	279
2 バイト文字セット・データの処理	283

第 19 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	287
z/OS での管理プログラム出口ルーチンの使 用	287
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	298
QMF が管理プログラム出口ルーチンと呼び 出す方法と時期	302
リソース制御情報を管理プログラム出口に渡 す	310
QMF セッション中のリソース制御情報の保 管	323
ユーザー・アクティビティーの取り消し	324
取り消されたアクティビティーに対するメッ セージ	325
TSO、ISPF、およびネイティブ z/OS バッチ での管理プログラム出口ルーチンのアセンブ ルとリンク・エディット	328
z/OS 上の CICS での管理プログラム出口ル ーチンのアSEMBル、変換、リンク・エディ ット	329
z/OS での DB2 管理プログラムの使用	330

第 20 章 バッチ・プログラムとしての QMF の実行	335
TSO/CICS でのバッチ・プログラムとしての QMF の実行	335
CICS での非対話式トランザクションとして の QMF の実行	355

第 21 章 障害追及と問題診断	359
一般的な問題の障害追及	359
診断援助機能を使用した問題の判別	366
IBM への問題報告	381

第 3 部 付録 385

付録 A. 各種問題	387
問題が発生した場合	387
表示される可能性のあるエラー・メッセージ	387
z/OS 上の CICS での DB2 QMF バージョン	
8.1 製品の制限	392

付録 B. DB2 に常駐している QMF オブジ ェクト	393
QMF プラン	393
QMF パッケージ	393
TSO/CICS での QMF QMF コントロール表 および表スペース	393
QMF 視点	395
TSO/CICS 用の VSAM クラスタ	396
TSO/CICS 用の QMF サンプル表	397

付録 C. QMF ユーザー定義関数	399
APPL_AUTHNAMES	399
CALL_DSQAB1E	400
DSQABA1E	400

付録 D. QMF および GDDM プログラムの CICS への定義のしかた	401
QMF プログラムの CICS への定義のしかた	401
QMF 初期化中の GDDM 定義のロードのし かた	402
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	403

付録 E. QMF リリース間の移行およびフォ ールバック	405
移行とは何か	405
QMF の複数のリリース	405
DB2 サブシステムと移行	406
QMF オブジェクトの移行	409
アプリケーションの移行	410
移行に関するその他の考慮事項	410
フォールバック	412

付録 F. 特記事項	415
商標	417

付録 G. 用語集	419
----------------------------	------------

付録 H. 参考文献	433
-----------------------------	------------

CICS の資料	433	Interactive System Productivity Facility (ISPF)	
COBOL の資料	433	の資料	435
DB2 Universal Database (z/OS 版) の資料	434	OS/390 の資料	435
Document Composition Facility (DCF) の資料	434	OS PL/I の資料	436
Distributed Relational Database Architecture		REXX の資料	436
(DRDA) の資料	435	VM/ESA の資料	436
Graphical Data Display Manager (GDDM) の		VSE/ESA の資料	436
資料	435		
High Level Assembler (HLASM) の資料 . . .	435	索引	437

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 z/OS™ のタイム・シェアリング・オプション / 顧客情報管理システム (TSO/CICS) 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、IBM DB2 Universal Database (z/OS 版) とともに使用する DB2 QMF のインストールおよび管理を担当するシステム・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。本書で「ワークステーション・データベース・サーバー」と呼ぶ場合は、以下のものを指します。

- DB2 DataJoiner®
- DB2 Universal Database バージョン 7 以上

前提知識

ユーザーは、特定の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。

z/OS

z/OS の場合、以下のコンポーネントがあります。

- オペレーティング・システム z/OS。
- タイム・シェアリング・オプション (TSO)。DB2 QMF とその関連製品をサポートする環境です。
- DB2 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)。
- 顧客情報管理システム (CICS)®。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す。CICS/TS® により、DB2 QMF と z/OS 間のインターフェースが提供されます。
- 図形データ表示管理プログラム (GDDM)® 基本製品。印刷要求の有無に関わらず必要です。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します。
- DB2。DB2 QMF 用のデータベース・マネージャーです。
DB2 では、z/OS 上で、バッチ・モードにより、または DB2I (DB2 対話機能) を介して実行できる多くのユーティリティも用意されています。
- SMP/E (拡張システム修正変更プログラム)。

- 高水準アセンブリ言語 (HLASM)。ガバナー出口ルーチンの変更または新規作成に必要です。HLASM は、QMF 書式用のユーザー独自の編集コードを作成するために使用することもできます。
- PL/I。QMF 書式用に 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します。
- VS COBOL II および COBOL。QMF 書式用に COBOL でユーザー独自の編集コードを作成する場合に使用します。
- REXX。DB2 QMF をインストールする EXEC を作成する場合に使用します。

これらの製品についての資料は、433 ページの『付録 H. 参考文献』にリストされています。

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手順に従って 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。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、DB2 QMF 製品そのものを使用して行われます。本書で説明している作業を開始する前に、インストール検査手順 (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、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。

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。例えば、DB2 UDB (z/OS 版) は DB2、DB2 QMF は QMF です。

各国語機能情報の表し方

DB2 QMF では、数種類の異なる言語を使用することができます。それぞれの言語には、各国語機能 (NLF) が提供されます。

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 DB2 QMF の分離したフィーチャーとしてインストールされます。NLF のインストールについては、本書で該当するオペレーティング・システム用のインストールに関するセクションの NLF インストール情報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 DB2 QMF 製品 (英語) およびいずれの NLF でも行うことができます。手順は、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* が含まれているものがありますが、これ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
カナダ・フランス語	C	FRANCAIS CANADIEN
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
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、IBM 3277、3278、および 3279 端末があります。

第 1 部 TSO/CICS 用 QMF バージョン 8.1 のインストール

第 1 章 QMF およびインストール・プロセスの概要	3	QMF のための GDDM の調整	20
QMF の概要	3	DB2 (AIX 版) の QMF	20
QMF が他のデータベースのデータにアクセスする方法	4	ワークシートへの記入	21
リモート作業単位	4	QMF インストール・ジョブ情報	22
DB2 UDB (z/OS 版) 分散作業単位	5	第 3 章 DB2 アプリケーション・リクエストとしての QMF の構成	27
データベース・インストール・プロセスの概要	6	QMF 互換モード・インストール	27
DB2 UDB (z/OS 版) の QMF 要件	6	QMF 新機能モードのインストール	28
DB2 UDB (z/OS 版) 知識の前提条件	7	QMF 互換モードへの移行	28
インストール中に QMF が作成する DB2 UDB (z/OS 版) オブジェクト	7	第 4 章 QMF の TSO 用調整	29
データベース許可 ID Q	8	TSO ログオン・プロシージャの作成	29
QMF をリモート作業単位用に設定する	8	TSO での QMF の開始	29
DB2 間分散作業単位用 QMF のセットアップ	9	TSO ログオン・プロシージャの準備	30
例	9	QMF の開始	33
ロング・ネームのサポート	11	ISPF を使用した QMF の開始	33
互換モードと新機能モード (NFM)	12	TSO での QMF の開始	35
DB2 QMF (TSO/CICS 版) バージョン 8.1 の移行の概説	13	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)	36
QMF インストールのロードマップ	13	第 5 章 QMF の CICS 用調整	39
初期インストールまたは移行	13	CICS の DB2 に対する QMF の記述	39
サーバー・データベースのインストール	14	QMF/GDDM データ・セットの定義およびロード	39
第 2 章 インストールの計画	15	GDDM ADMF データ・セットへの QMF/GDDM マップ・セットのロード	39
ハードウェア要件	15	QMF/GDDM 図表および QMF トレース・データ・セットの作成	40
前提条件のソフトウェア	15	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	41
記憶域要件の計画	16	CICS 管理テーブルの更新	41
TSO 記憶域	16	DCT (宛先管理テーブル)	41
CICS/ESA 領域	16	QMF プロファイルの調整	42
パフォーマンスを向上させるためのモジュールの移動	17	CICS 始動ジョブ・ストリームを更新	43
CICS の場合	18	第 6 章 QMF 互換モード用リモート・サーバーの構成	45
記憶域の見積もり	19		
プログラム・ディレクトリーの読み取りおよびサービスの適用	19		
CICS のもとでの QMF の計画	19		
QMF のための CICS の調整	19		

Linux、UNIX、および Windows 版 DB2 アプリケーションとしての QMF 互換モードの構成	45
QMF 互換モードのインストール	46
DB2 DRDA AS に対する QMF の開始	47
DB2 DRDA AS からの QMF の削除	47
DB2 UDB (iSeries 版) アプリケーションとしての QMF 互換モードの構成	48
DB2 UDB (iSeries 版) サーバーに対する QMF の始動	48
QMF の削除	49
DB2 UDB (z/OS 版) アプリケーションとしての QMF 互換モードの構成	50
すべてのサーバー・タイプの QMF 互換モードへの移行	50

第 7 章 全サーバー・タイプ用の QMF 新機能モードの構成	53
QMF 新機能モードのインストール	53
QMF 新機能モードへの移行	54

第 8 章 QMF インストールのテスト	57
IVP の実行 (TSO)	57
IVP の実行 (CICS)	59
QMF を開始する前に	59
QMF の開始およびテスト	60
QMF アプリケーション照会およびプログラマーのインストール (TSO)	62
オプション: バッチ・モード IVP の実行	63
インストール後の終結処理	65
旧アプリケーション・プランの解放	66
QMF バージョン 7.2 および以前のバージョンの DB2 サブシステムからの削除	67

第 1 章 QMF およびインストール・プロセスの概要

この章では、DB2 QMF ホスト製品を概説し、QMF がインストールされる方法を概説し、QMF がさまざまな DB2 UDB データベースに接続する方法について説明します。

QMF の概要

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、オペレーティング・システム z/OS のもとで稼動し、主に DB2 UDB (z/OS 版) によっ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。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もできます。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 に関係付けられるかを示したものです。

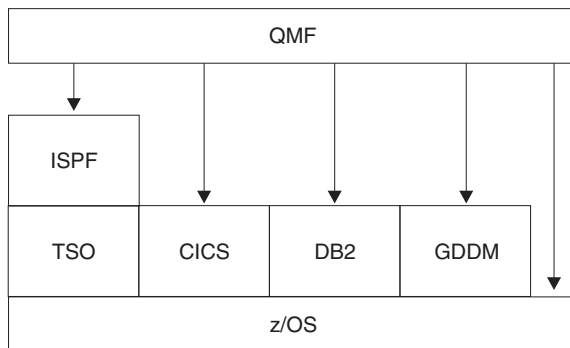


図 1. ホスト専用構成の QMF。

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 (z/OS 版)、DB2 UDB (UNIX、Windows、および OS/2 版)、DB2 Server (VSE または VM 版)、DB2 UDB (iSeries 版)、または DB2 UDB (UNIX 版)、および Windows データベースのいずれか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 (z/OS 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 (z/OS 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 (z/OS 版) 間分散作業単位により、DB2 UDB (z/OS 版) に特定の通信方式を使用して他の DB2 サブシステムにアクセスすることができます。DB2 UDB (z/OS 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クエ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、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。

DB2 UDB (z/OS 版) 分散データ機能 (DDF) は、DB2 (OS/390 版) バージョン 2.3 で始まる 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 (z/OS0 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 コマンドが接続先のリモート・ロケーションを指定するのに使用されている場合は、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。

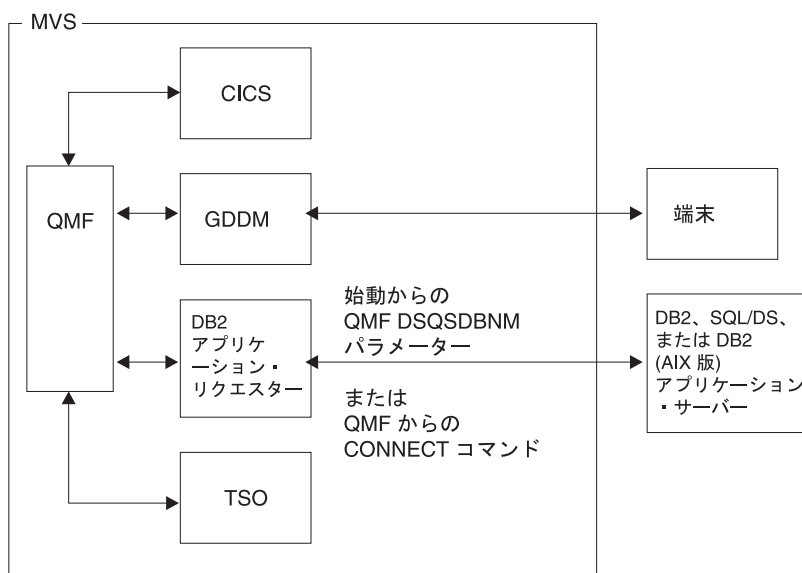


図 2. リモート作業単位を用いた QMF

DB2 UDB (z/OS 版) 分散作業単位

これは分散作業単位の初期の版であり、DB2 (OS/390 版) 2.2 からサポートされるようになりました。これにより、DB2 UDB (z/OS 版) 専用の通信方式を使用して他の DB2 UDB (z/OS 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 (z/OS 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

概要

ロケーションを判別し、そのサブシステムに接続します。 次の図は、DB2 UDB (z/OS 版) 間アクセス接続について示しています。

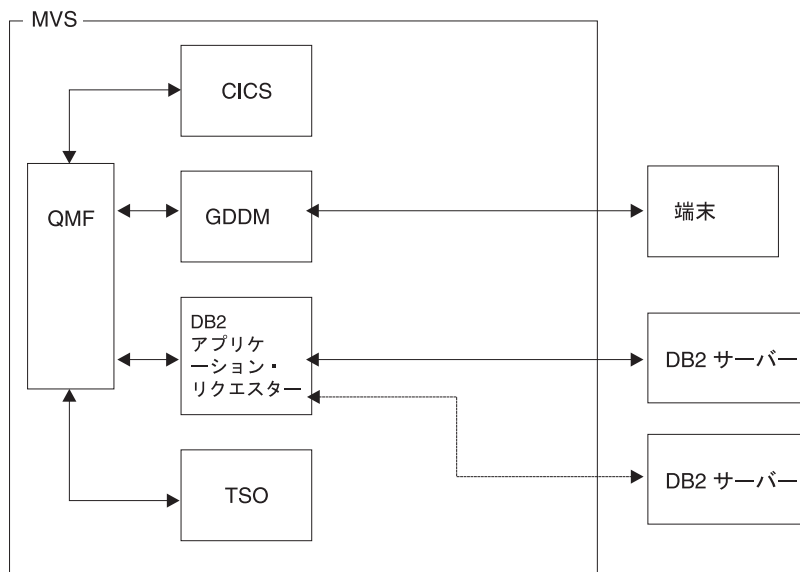


図3. DB2 UDB (z/OS 版) 間接続

データベース・インストール・プロセスの概要

DB2 QMF (TSO/CICS 版) 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 コントロール表、カタログ視点、およびサンプル表

DB2 UDB (z/OS 版) の QMF 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 (z/OS 版) アプリケーション・プログラムです。少なくとも 1 つの DB2 UDB (z/OS 版) サブシステムに QMF を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 をその他の DB2 UDB (z/OS 版) サブシステムにインストールする必要が生じる場合があります。

DB2 UDB (z/OS 版) 知識の前提条件

QMF は DB2 UDB (z/OS 版) のアプリケーションであるため、DB2 UDB (z/OS 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の詳細については、「DB2 UDB (z/OS 版) 解説書」で説明されています。
- 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
これらの用語については、「DB2 (z/OS 版) アプリケーション・プログラミングおよび SQL ガイド」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。これらの用語については、「DB2 UDB (z/OS 版) 管理ガイド」を参照してください。
- DB2 UDB (z/OS 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 (z/OS 版) 管理ガイド」で説明されています。
- QMF のインストールを計画している DB2 サブシステムの ID
サブシステム ID については、「DB2 UDB (z/OS 版) 管理ガイド」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 (z/OS 版) データベース
- ロケーション名
- リモート作業単位

これらの用語に関する定義および情報については、「DB2 UDB (z/OS 版) SQL 解説書」を参照してください。

インストール中に QMF が作成する DB2 UDB (z/OS 版) オブジェクト

QMF が DB2 UDB (z/OS 版) システムにアクセスする場合は、インストール中に QMF 用に作成されたオブジェクト・タイプが使用できます。

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 (z/OS 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 コントロール表
- 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、393 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 (z/OS 版) 許可 ID は、QMF のすべてのコントロール表、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。

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 (z/OS 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 z/OS システム上の追加の DB2 UDB (z/OS 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 (z/OS 版) サブシステム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する場合は、必要なオブジェクトのみをインストールしてください。

重要: 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のものである場合のみ機能します。

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) を使用する場合は、まずどの DB2 UDB (z/OS 版) 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、コントロール表、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 間分散作業単位用 QMF のセットアップ

リモート・データへの DB2 UDB (z/OS 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。リモート作業単位と DB2 UDB (z/OS 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。

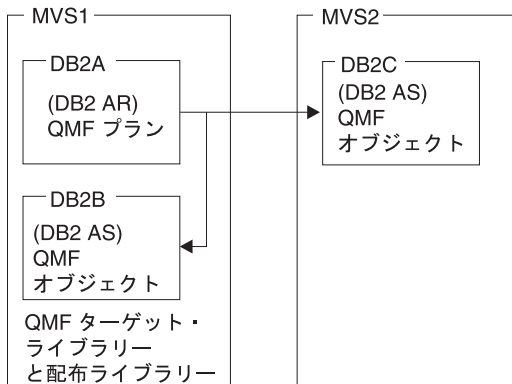
例

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 をインストールする方法を示しています。

システム構成例および要件

- z/OS オペレーティング・システム MVS1 には、2 つの DB2 UDB (z/OS 版) バージョン 8.1 サブシステム DB2A および 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。
- z/OS オペレーティング・システム MVS2 には、1 つの DB2 UDB (z/OS 版) バージョン 8.1 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 z/OS 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、コントロール表、サンプル表、視点、アプリケーション・パッケージ、およびアプリケーション・プランです。

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 (z/OS 版) データベースとして、および 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。

ロング・ネームのサポート

SQL では、最大 128 文字の長さの列、表、およびその他の名前がサポートされてきました。DB2 QMF (TSO/CICS 版) のバージョン 8.1 では、ロング・ネームがサポートされるようになりました。以下の名前が、この機能拡張に係る名前です。

表 2. ロング・ネーム図表

名前	バージョン 7.2 またはそれ以前の互換モード	新機能モード (NFM)
許可 ID	FIXED CHAR(8)	VARCHAR(128)
SQLID	FIXED CHAR(8)	VARCHAR(128)
表、視点、同義語、および関連	VARCHAR(18)	VARCHAR(128)
列 (注 1 を参照)	VARCHAR(18)	VARCHAR(30)
ロケーション (注 2 を参照)	VARCHAR(16)	VARCHAR(16)

注 1: 現在 VARCHAR(16) として記述されている列名は、DB2 カタログでは VARCHAR(128) に拡張されています。ただし、列名は、DB2 UDB (z/OS 版) バージョン 8.1 では 30 バイトのユニコードに限定されます。DB2 QMF (TSO/CICS 版) バージョン 8.1 では、最大 30 文字までサポートされます。

注 2: 現在 VARCHAR(16) として記述されているロケーション名は、DB2 カタログでは VARCHAR(128) に拡張されています。ただし、ロケーション名は、DB2 UDB (z/OS 版) バージョン 8.1 では 16 バイトのユニコードに限定されます。DB2 QMF (TSO/CICS 版) バージョン 8.1 では、最大 16 文字までサポートされます。

DB2 UDB (z/OS 版) バージョン 8.1 が NFM で実行中の場合は、データベース内のロング・ネームがサポートされます。これ以外のモードの場合は、DB2 QMF (TSO/CICS 版) バージョン 8.1 では、データベース内のロング・ネームはサポートされません。

DB2 UDB (z/OS 版) バージョン 8.1 に加えて、リモート・サーバーへの接続時には、以下の名前の長さが DB2 QMF (TSO/CICS 版) バージョン 8.1 でサポートされます。

表 3. リモート・サーバーへの接続時にサポートされるロング・ネーム長

	iSeries	UNIX および Windows	VSE & VM	z/OS
	バージョン 2.0 以前	バージョン 7.1	バージョン 7.2 以下	バージョン 8.1 互換モード以下
スキーマ名	10	8	8	8
列名	10	18	18	18
表名	18	18	18	18

表3. リモート・サーバーへの接続時にサポートされるロング・ネーム長 (続き)

	iSeries	UNIX および Windows	VSE & VM	z/OS
	バージョン 2.0 以上	バージョン 7.1		V8.1 NFM 以上
スキーマ名	10	8		128
列名	30	30		30
表名	128	128		128
		バージョン 7.1		
スキーマ名		30		

互換モードと新機能モード (NFM)

QMF 互換モード・インストールは、以前の QMF レベルのインストールと同様です。QMF 所有者名は 8 文字に制限され、QMF オブジェクト名は 18 文字に制限されます。QMF 互換モード・インストールは、QMF の前リリースと同じサーバー内で共存できます。

QMF バージョン 8.1 新機能モード・インストールでは、データベースで許可されている長さの QMF 所有者およびオブジェクト名が使用できます (最大長 128 文字)。QMF 新機能モード・インストールは、QMF の前リリースと同じサーバー内で共存できません。以下のサーバーは、QMF 新機能モード・インストールを受け入れます。

- NFM の DB2 UDB (z/OS 版) バージョン 8.1 サーバー
- DB2 Universal Database バージョン 7.1 以上
- DB2 (iSeries 版) バージョン 5.1 以上

QMF NFM リクエスターは、DB2 QMF バージョン 8.1 パッケージがサーバーおよびリクエスターにバインドされていれば、QMF 互換モード・サーバーに接続できます。リクエスターおよびサーバーは、同じリリース・レベルのパッケージを持っている必要があります。

ユーザーは、QMF 互換モードから QMF 新機能モードにいつでも移行できます。QMF 新機能モード・インストールは、QMF 互換モード・インストールからの移行、または QMF の前リリースを含んでいないサーバーへの移行のいずれかによって行うことができます。QMF の前リリースから QMF 新機能モードに移行するには、最初に QMF 互換モードに移行して、次に QMF 新機能モードに移行する必要があります。

DB2 QMF (TSO/CICS 版) バージョン 8.1 の移行の概説

QMF バージョン 8.1 は、QMF バージョン 3.3、6.1、7.1、および 7.2 からの移行をサポートします。QMF 移行インストールでは、常に QMF 互換モード・インストールが生成されます。

QMF インストールのロードマップ

この節には、サポートされる各種データベース・サーバーの QMF インストール・タイプおよびパスがリストされています。インストールで作成されるデータベース・オブジェクトの詳細については、393 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

- プログラム・ディレクトリーを読み、SMP/E インストールを指示に従って完了する。
- DB2 UDB (z/OS 版) データベース移行または新規インストールを開始する。

以下の場合には、新規インストールを行ってください。

- QMF の前リリースを含まない DB2 UDB (z/OS 版) への QMF の初めてのインストールの場合。
- QMF 前のリリースを含まない DB2 UDB (z/OS 版) への QMF の唯一のインストールの場合。
- 複数のローカル DB2 UDB (z/OS 版) サブシステムに QMF からアクセスする必要がある場合。追加のローカル DB2 UDB (z/OS 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。

上記のインストール・パスの詳細については、27 ページの『第 3 章 DB2 アプリケーション・リクエストとしての QMF の構成』を参照してください。

ターゲット DB2 UDB (z/OS 版) データベースがすでに QMF バージョン 3.3 以上を含んでいる場合は、移行インストールを行ってください。QMF バージョン 8.1 は、QMF バージョン 3.2 以下からの移行インストールは行いません。バージョン 3.3 以上の QMF 移行インストールでは、常に QMF 互換モード・インストールが生成されます。

QMF 移行インストールの詳細については、27 ページの『第 3 章 DB2 アプリケーション・リクエストとしての QMF の構成』を参照してください。

注: QMF バージョン 8.1 互換モードは、QMF バージョン 3.3 以上からの移行後に、完全な検査を行う必要があります。QMF バージョン 8.1 互換モードから QMF バージョン 8.1 新機能モードへの移行は、注意して行ってください。QMF バージョン 8.1 互換モードを QMF 新機能モード (NFM) に一度移行すると、QMF の以前のどのリリースもその同じサーバーにアクセスできなくなります。QMF NFM への QMF 互換モ

ードからの移行の詳細については、53 ページの『第 7 章 全サーバー・タイプ用の QMF 新機能モードの構成』を参照してください。

サーバー・データベースのインストール

ユーザーのローカル DB2 UDB (z/OS 版) サブシステムからアクセス可能なりモート DB2 データベースに定義されたデータのアクセスを計画している場合は、最初にユーザーのローカル・サブシステムからサーバー・データベースのインストールを行う必要があります。互換モードのインストールおよび移行情報の詳細については、45 ページの『第 6 章 QMF 互換モード用リモート・サーバーの構成』を参照してください。リモート・サーバーへの QMF 新機能モードのインストールについては、53 ページの『第 7 章 全サーバー・タイプ用の QMF 新機能モードの構成』を参照してください。

第 2 章 インストールの計画

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、z/OS および DB2 UDB (z/OS 版) によってサポートされるすべての DASD 装置および 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

下の表に、DB2 QMF バージョン 8.1 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がリストされています。バージョン 8.1 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 4. TSO/CICS 用 DB2 QMF バージョン 8.1 の前提条件ソフトウェア

ソフトウェア	バージョン	部品番号
以下のいずれか 1 つのデータベース・ソフトウェア: DB2 Universal Database (z/OS 版) DB2 Universal Database (OS/390 版) DB2 Universal Database (OS/390 版)	バージョン 8 リリース 1 バージョン 7 リリース 1 バージョン 6 リリース 1	5625-DB2 5675-DB2 5645-DB2
GDDM	バージョン 3 リリース 2	5695-167
CICS/ESA 注: CICS/ESA は CICS ユーザーにのみ必要です。	バージョン 4 リリース 1	5655-018

インストールの計画

TSO/CICS 用 DB2 QMF バージョン 8.1 のオプション機能をサポートするには、以下の最低限度のリリース・レベルのプログラム製品のいずれかが必要です。バージョン 8.1 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。

- DB2 Universal Database (OS/390 版) バージョン 7.1 (拡張 LIST TABLES コマンド用の PTF、 UQ57178、 UQ60456、 UQ60033、 UQ66553 を含む)。部品番号 5675-DB2
- DB2 Universal Database (OS/390 版) バージョン 6.1 (拡張 LIST TABLES コマンド用の PTF、 UQ57177、 UQ60455、 UQ60032 を含む)。部品番号 5645-DB2

記憶域要件の計画

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してください。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、4.5 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、1.0 から 2.0 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 8.0 MB 必要です。

ISPF および GDDM をページング可能リンク・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。

TSO 記憶域

QMF を実行するには、1.0 から 2.0 MB の記憶域スペースが必要です。他のアプリケーションには、追加の記憶域スペースが必要です。たとえば、標準 TSO 環境で ISPF および GDDM を使用して QMF を実行する場合、記憶域スペースは約 8.0 MB 必要です。

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。

DSQCTOPX モジュールは、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。つまり、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/ESA 領域

CICS バージョン 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 システム初期設定テーブル・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 1,536 KB は小さ過ぎるため、QMF のユーザーをサポートできません。QMF を同時に使用するユーザーの数に応じて、EDSASZE を 16 ~ 50 MB の範囲まで拡大してください。16 MB に、QMF を同時に使用するユーザー 1 人につき 2 MB を加算したサイズを使用してください。本件の詳細については、該当する CICS のシステム定義および操作ガイドを参照してください。

パフォーマンスを向上させるためのモジュールの移動

ライブラリー QMF810.SDSQLOAD には、QMF 製品用のロード・モジュールが含まれます。表 5 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 5. 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せん。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。

表 5. PLPA または EPLPA に置くことができるモジュール (続き)

モジュール	説明
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。

表 6 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。

表 6. 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。
DSQCMAPB 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。

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 および QMF のターゲットおよび配布ライブラリーのデータ・セットのすべてのサイズを見積もる方法については、QMF プログラム・ディレクトリーを参照してください。

プログラム・ディレクトリー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、QMF プログラム・ディレクトリーの補足データをお読みください。プログラム・ディレクトリーは QMF の各リリースの間に更新されるため、この資料には、本書に対する変更だけでなく、プログラム一時修正 (PTF) およびプログラム診断依頼書 (APAR) の記述などの有用な情報も含まれています。

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。また、QMF の予防保守計画 (PSP) パッケージを要求してください。QMF PSP パッケージ・サブセットおよびアップグレードの名前は QMF プログラム・ディレクトリーにあります。PSP パッケージには、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、情報をインターネットから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、およびテストを完了している必要があります。

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。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 相互通信の手引き* 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す。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、*GDDM インストールおよびシステム管理* を参照してください。

GDDM 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。

DB2 (AIX 版) の QMF

AIX サーバー用の DB2 と連動させるために 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。

QMF は、分散データ機能 (DDF) を使用して、AIX データベース用の DB2 UDB に常駐する分散データにアクセスします。DB2 UDB (z/OS 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。TSO/CICS 用 QMF から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 (z/OS 版) 管理ガイド*」を参照してください。

DB2 UDB (z/OS 版) では、コミュニケーション・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 (z/OS 版) をサーバーとしてのみ使用したい場合は、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 (z/OS 版) インストレーションの手引き* に説明があります。

AIX サーバー用の DB2 UDB 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 コマンドを実行する必要があります。z/OS DRDA アプリケ

ーション・リクエスター用の DB2 UDB (z/OS 版) と AIX DRDA アプリケーション・サーバー用の DB2 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。

DB2 UDB (z/OS 版) から AIX 用の DB2 UDB 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、45 ページの『第 6 章 QMF 互換モード用リモート・サーバーの構成』を参照してください。

ワークシートへの記入

表 7 に、QMF のインストール時に値を指定する必要がある必須情報が示されています。これらの表をワークシートとして使用してください。

表 7. QMF インストール時の必須情報 (バージョン 8.1 ワークシート、その 1)

必須情報	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 = QMF810)	
ローカル DB2 UDB (z/OS 版) サブシステム ID (デフォルト = DSN)	
ローカル DB2 UDB (z/OS 版) 出口ライブラリー (デフォルト = DSN810.SDSNEXIT)	
ローカル DB2 UDB (z/OS 版) ロード・ライブラリー (デフォルト = DSN810.SDSNLOAD)	
QMF アプリケーション・プラン ID (デフォルト = QMF810)	

必須情報	値
DB2 UDB (z/OS 版) ユーザー・カタログ (ICF) (デフォルト = DSNC8101.USER.CATALOG)	
QMF 表スペース・カタログ別名 (デフォルト = QMFDSN)	
QMF 表ボリューム	
DB2 UDB (z/OS 版) デフォルト句読法	, (コンマ) または . (ピリオド)
前の QMF レベル (移行インストールのみ)	V3R3、V6R1、V7R1、V7R2、または NONE

必須情報	1 次	2 次
以下の情報は、前のレベルの QMF を含まない DB2 UDB (z/OS 版) サーバーに適用されます。QMF では、以下の値が奨励されますが、インストール時にオーバーライドされる場合があります。		

インストールの計画

必須情報	1 次	2 次
QMF control table table space 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) - SAVE DATA (Optional) " (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 (see Note) (200,20) - Q.OBJECT_REMARKSX " (200,20) - Q.OBJECT_OBJDATA " (200,20) - Q.PROFILEX (200,20) - 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)。		
SAVE DATA 表スペースを作成したいか否か	YES または NO	

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

QMF インストール・ジョブ情報

TSO/CICS 用 DB2 QMF バージョン 8.1 での変更点: DSQ1EINS インストール CLIST およびパネルは、製品から除去されました。バッチ・インストール・ジョブの実行依頼が、QMF の唯一のインストール方法です。ジョブ DSQ1EMAP、DSQ1CHRT、DSQ1EJVE および DSQ1EJVC も QMF インストールから除去されました。インストール VSAM パネル・ジョブ DSQ1EPNL の実行についての情報は、プログラム・ディレクトリーのセクション 6.2 に移動しました。

以降の QMF インストールの章では、すべてのジョブ名は QMF810.SDSQSAPE データ・セットのメンバーとして示される場合があります。すべてのジョブについて概要が示されており、調整およびサブミット時に詳細を参照する必要があります。

オーバーライド可能な QMF データベース・インストール・デフォルトはすべて、QMF810.SDSQEXCE(DSQ1DEFS) に示されています。DSQ1DEFS REXX EXEC も、各デフォルトを使用する各ジョブを参照します。これらのデフォルトは、21 ページの表 7

にも示されています。DSQ1DEFS に含まれる変数には、たとえば、QMF コントロール表スペースと索引スペース 1 次と 2 次の数量値、および QMF 記憶域グループの VCAT と VOLUMES パラメーターがあります。すべての QMF インストール・ジョブが DSQ1DEFS のすべての値を使用するわけではありません。ジョブは、参照される各 DSQ1DEFS 値を明確に記述します。ジョブで参照されない DSQ1DEFS 値は無視されません。すべての変数は、**DSQ1DEFS** から削除してはなりません。

QMF のインストール時に DSQ1DEFS 値を変更する場合は、最初にコピーを作成する必要があります。コピーを変更して、QMF インストール・ジョブ DSQDEFS ddcard がコピーを参照するように変更する必要があります。

QMF は、ほとんどのデータベース・インストール・ジョブで、この新規 DSQ1DEFS EXEC を使用します。以下の 2 つの例では、この EXEC の柔軟性と新しいインストール・プロセスについて説明します。

例 1 では、ユーザーがインストール・ジョブ DSQ1TBLJ を実行します。ユーザーは、すべての値について DSQ1DEF で指定された QMF デフォルト値で満足しており、また SSID、LOCATION および VCATNAME をオーバーライドする必要があります。この例では、変更された SYSTSIN ステートメントについて説明します。

```
//DSQ1TBLJ JOBcard
//*comments
//DSQ1TBLJ PROC RGN='2048K',
//      QMFTPRE='QMF810',
//      DB2EXIT='DSN810.SDSNEXIT',
//      DB2LOAD='DSN810.SDSNLOAD'
//*-----
/* CREATE AND LOAD QMF DATABASE AND CONTROL TABLES FOR QMF --
/* COMPATIBILITY MODE.      ---
/* -----
//STEP1 EXEC PGM=IKJEFT01,REGION=&RGN
//STEPLIB DD DSN=&QMFTPRE..SDSQLOAD,DISP=SHR
//      DD DSN=&DB2EXIT.,DISP=SHR
//      DD DSN=&DB2LOAD.,DISP=SHR
//SYSTPRT DD SYSTOUT=*,DCB=BLKSIZE=121
//SYSTEM DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSEXEC DD DSN=&QMFTPRE..SDSQEXCE,DISP=SHR
//DSQDEFS DD DSN=&QMFTPRE..SDSQEXCE(DSQ1DEFS),DISP=SHR
//DSQINDD DD DSN=&QMFTPRE..SDSQSAPE(DSQ1VSTG),DISP=SHR
//      DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLB),DISP=SHR
//      DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLI),DISP=SHR
//      DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLU),DISP=SHR
//      DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLE),DISP=SHR
//      DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLN),DISP=SHR
//      DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLG),DISP=SHR
//      DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLK),DISP=SHR
//      DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBDC),DISP=SHR
//      DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLC),DISP=SHR
//      PEND
```

インストールの計画

```
//DSQTBL EXEC DSQ1TBLJ
//*=====
//* Tailor below:
//*=====
//STEP1.SYSTSIN DD*
%DSQ1INST QMFBSQL SSID(DB2L) LOCATION(MVS1DB2L) +
    VCATNAME(DB2LDSN) VOLUMES('*')
```

例 2 では、ユーザーがインストール・ジョブ DSQ1TBLJ を実行します。ユーザーは、すべての値について DSQ1DEF で指定された QMF デフォルト値で満足しており、また SSID、LOCATION および VCATNAME をオーバーライドする必要があります。DSQ1DEFS ddcard でオーバーライドを行います。コピーは、どのメンバー名に対しても行うことができます。

QMF810.SDSQEXCE(DSQ1DEFS) を QMF810.SESQEXCE(DB2LDEFS) にコピーします。QMF810.SDSQEXCE(DB2LDEFS) を編集して、以下の値に変更します。

```
SSID = "DB2L"
LOCATION = 'MVS1DB2L'
VCATNAME = ""DB2LDSN"
```

コピーして変更したメンバー DB2LDEFS を指すように DSQ1TBLJ DSQDEFS ddcard を変更して、SYSTSIN からオーバーライド・オプションを除去します。DSQ1INST 呼び出しの後、QMFBSQL 値を保持するようにします。

```
//DSQ1TBLJ JOBcard
//*comments
//DSQ1TBLJ PROC RGN='2048K',
//    QMFTPRE='QMF810',
//    DB2EXIT='DSN810.SDSNEXIT',
//    DB2LOAD='DSN810.SDSNLOAD'
//*-----
//* CREATE AND LOAD QMF DATABASE AND CONTROL TABLES FOR QMF ---
//* COMPATIBILITY MODE.          ---
//*-----
//STEP1 EXEC PGM=IKJEFT01,REGION=&RGN
//STEPLIB DD DSN=&QMFTPRE..SDSQLOAD,DISP=SHR
//    DD DSN=&DB2EXT.,DISP=SHR
//    DD DSN=&DB2LOAD.,DISP=SHR
//SYSTSPRT DD SYSOUT=*,DCB=BLKSIZE=121
//SYSTEM DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSEXEC DD DSN=&QMFTPRE..SDSQEXCE,DISP=SHR
//DSQDEFS DD DSN=&QMFTPRE..SDSQEXCE(DB2LDEFS),DISP=SHR <---Modified
//DSQINDD DD DSN=&QMFTPRE..SDSQSAPE(DSQ1VSTG),DISP=SHR
//    DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLB),DISP=SHR
//    DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLI),DISP=SHR
//    DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLU),DISP=SHR
//    DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLE),DISP=SHR
//    DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLN),DISP=SHR
//    DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLG),DISP=SHR
//    DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBLK),DISP=SHR
//    DD DSN=&QMFTPRE..SDSQSAPE(DSQ1TBDC),DISP=SHR
```

```
// DD DSN+&QMFTPRE..SDSQSAPE(DSQ1TBLC),DISP=SHR
// PEND
//DSQTBL EXEC DSSQ1TBLJ
//*=====
//* Tailor below:
//*=====
//STEP1.SYSTSIN DD *
%DSQ1INST QMFBSQL      <-- Modified (Be sure to leave the order
    specified afterDSQ1INST).
```

DB2LDEFS を DB2L サブシステム用に変更した後は、すべての QMF インストール・ジョブで使用できます。たとえば、DSQ1BVW、DSQ1STGJ および DSQ1EIVS でも DB2LDEFS を使用できます。サーバー・インストールごとに DSQ1DEFS を変更して、そのサーバーのすべての QMF インストール・ジョブで使用することもできます。

第 3 章 DB2 アプリケーション・リクエスターとしての QMF の構成

以下のいずれかのインストール・タイプを、それぞれの DB2 UDB (z/OS 版) ローカル・アプリケーション・リクエスターごとに実行する必要があります。

- QMF 互換モードのインストール
- QMF 新規機能モードのインストール
- QMF 互換モードの移行

次の節は、タスク表でジョブ名を参照します。QMF810.SDSQSAPE データ・セットのメンバーであるこれらのジョブには、多くのコメントが付けられています。正常に完了するためには、これらのジョブを調整し、指定の順序で実行する必要があります。

QMF 互換モード・インストール

この一連のステップによって、QMF は、QMF の前リリースがインストールされていない DB2 (OS/390 版) バージョン 6.1 以上のサーバーに、互換モードでインストールされ、準備されます。

表 8. DB2 UDB (z/OS 版) サーバーへの新規 QMF 互換モード・インストールのジョブ・シーケンス

ジョブ名	目的
DSQ1TBAJ	オプション: QMF VCAT 名を作成する。
DSQ1BSQL	QMF インストール・パッケージおよびプランをターゲット・サーバーにバインドする。
DSQ1TBLJ	QMF コントロール表を作成する。
DSQ1BVW	QMF バージョン 8.1 視点を作成する。
DSQ1BPKG	QMF パッケージをバインドする。
DSQ1BINR	QMF アプリケーション・プランをバインドする (DB2 UDB (z/OS 版) ローカル・インストール専用)。
DSQ1STGJ	SAVE DATA 表スペースを QMF IVP 用に作成する。
DSQ1EIVS	QMF サンプル表を作成する。

QMF 新機能モードのインストール

下の表にあるタスクによって、QMF 新機能モードは、QMF の前リリースがインストールされていない DB2 バージョン 8.1 NFM サーバーにインストールされます。

表 9. QMF 新機能モードのインストール

ジョブ	目的
DSQ1TBAJ	オプション - QMF VCAT 名を作成する。
DSQ1BSQL	QMF インストール・パッケージおよびプランをバインドする。
DSQ1BLNI	QMF NFM コントロール表を作成する。
DSQ1BVW	QMF 視点を作成する。
DSQ1BPKG	QMF アプリケーション・パッケージをバインドする。
DSQ1BINR	QMF アプリケーション・プランをバインドする。
DSQ1STGJ	SAVE DATA 表スペースを QMF IVP 用に作成する。
DSQ1EIVS	オプション - QMF サンプル表を作成する。

QMF 互換モードへの移行

下の表のステップによって、DB2 (OS/390 版) バージョン 6 以上のサーバーの QMF バージョン 3.3、6.1、7.1、または 7.2 のインストールは、QMF バージョン 8.1 互換モード・インストールに移行されます。QMF の前リリースは、QMF 互換モードと共存できます。

表 10. QMF バージョン 3.3 以上のバージョンの QMF 互換モードへの移行ジョブ・シーケンス

ジョブ	目的
DSQ1BSQL	QMF インストール・パッケージおよびプランをバインドする。
DSQ1BVW	QMF 視点を作成する。
DSQ1BPKG	QMF アプリケーション・パッケージをバインドする。
DSQ1BINR	QMF アプリケーション・プランをバインドする。

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 29 ページの『第 4 章 QMF の TSO 用調整』を参照してください。CICS の場合は 39 ページの『第 5 章 QMF の CICS 用調整』を参照してください。

第 4 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。

- TSO ログオン・プロシーチャーを作成する。
- QMF を開始する。
- バッチ IVP を実行するために QMF バッチ・ジョブを設定する (オプション)。

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

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、「*ISPF (OS/390 版) OS/390 Dialog Management Services and Examples*」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。

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・プロシージャは、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うユーザーの場合は、追加のリソースが必要になる可能性があります。

ログオン・プロシージャ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・プロシージャの準備

1. QMF810.SDSQSAPE(DSQ1EINV) を編集する。
2. 領域パラメーターを探し出し、それが 16 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。必要ならば、QMF ユーザー出口ライブラリー QMF810.SDSQEXIT を STEPLIB 連結に追加します。これは、QMF810.SDSQEXIT に出口ルーチンが存在する場合のみ必要です。

サンプルには、ISPF、ISPF-PDF、QMF、DB2 UDB (z/OS 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 の複数の並行バージョンを同じ DB2 UDB (z/OS 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 (z/OS 版) データベースで実行する場合は、同じ QMF ロード・ライブラリーを同じプロシージャ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 8 リリース 1.0

QMF810.SDSQLOAD

バージョン 7 リリース 2.0

QMF720.SDSQLOAD

```

バージョン 7 リリース 1.0
      QMF710.SDSQLOAD
バージョン 6
      QMF610.SDSQLOAD
バージョン 3 リリース 3.0
      QMF330.DSLOAD

```

```

//*****
//*          PROGRAM LOAD LIBRARIES                      *
//*****
//STEPLIB DD DSN=QMF810.SDSQEXIT,DISP=SHR      * QMF MODULES *
//          DD DSN=QMF810.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=DSN810 .SDSNEXIT,DISP=SHR   * DB2 MODULES *
//          DD DSN=DSN810 .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。

```

//*****
//*          DATA SETS USED BY TSO                      *
//*****
//SYSPROC DD DSN=SYS2.CLIST,DISP=SHR           * CLIST Library
//          DD DSN=QMF810.SDSQCLTE,DISP=SHR
//SYSEXEC DD DSN=SYS2.EXEC,DISP=SHR
//          DD DSN=QMF810.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。

```

//*****
//*          DATA SETS USED BY ISPF                    *
//*****
//ISPLIB DD DSN=QMF810.SDSQPLBE,DISP=SHR      * Panel libraries
//          DD DSN=ISR.V4R1M0.ISRPLIB,DISP=SHR
//          DD DSN=ISP.V4R1M0.ISPLIB,DISP=SHR
//ISPLIB DD DSN=QMF810.SDSQMLBE,DISP=SHR      * Message Libraries
//          DD DSN=ISR.V4R1M0.ISRMLIB,DISP=SHR
//          DD DSN=ISP.V4R1M0.ISPMLIB,DISP=SHR
//ISPSLIB DD DSN=QMF810.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 および ADMGGMAP ライブラリーが適切に割り振られているか確認する。
- b. 独自の図表書式を保管したいユーザーに、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) DSQIEINV 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=QMF810.SDSQMAPE,DISP=SHR * GDDM Map Group
//ADMCFORM DD DSN=QMF810.SDSQCHRT,DISP=SHR * QMF-Supplied Chart Forms
//DSQUCFRM DD DSN=aaaaaaaa,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。代わりの情報をデータ・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。

```

//*****
//*          DATA SETS USED BY QMF          *
//*****
//DSQPNLE  DD  DSN=QMF810DSQPNLE,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=FBA,LRECL=125,BLKSIZE=1632) * Snap Dump Output
//SYSUDUMP  DD  SYSOUT=A
//DSQSPILL  DD  DSN=&&SPILL,DISP=(NEW,DELETE),      * User's Spill File
//  UNIT=SYSVIO,SPACE=(CYL,(1,1),RLSE),
//  DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)

```

QMF の開始

ログオン・プロシージャーを使用して 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 コマンドを実行する。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 (QMF810) 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 (z/OS 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。

```

Licensed Materials - Property of IBM
5625-DB2 5697-F42 (C) Copyright IBM Corp. 1982, 2003
All Rights Reserved.
IBM is a registered trademark of International Business Machines

-----
QMF ホーム・パネル                      照会報告書作成プログラム
Version 8 Release 1

許可 ID (Authorization ID)              ***** ** ** *****
Q                                         ** ** ** ** ** **
                                         ** ** ** ** ** ** *****
                                         ** ** ** ** ** ** **
接続は                                    ** ** ** ** ** ** **
SQLDS                                     ***** ** ** ** **
                                         **
-----

コマンド行にコマンドを入力するか、該当の機能キーを押してください。
ヘルプを表示するには、ヘルプ機能キーを押すか、HELP コマンドを入力してください。

1= ヘルプ      2= リスト      3= 終了      4= 表示      5= 図表      6= 照会
7= 検索        8= 表編集      9= 書式     10= 手順     11= PROFILE  12= 報告書
了解、コマンドを入力してください。
COMMAND ==>

```

図4. 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。35 ページの図5 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 プロシージャー QMF810.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

次の TSO 呼び出しコマンドを使用して QMF を開始することもできます。

- 全インストールまたはサーバー・インストールの場合:

```
CALL 'QMF810.SDSQLOAD(DSQMF)' 'DSQSSUBS=dbname,DSQSPLAN=planid,...'
```

- リクエスター・インストールの場合:

```
CALL 'QMF810.SDSQLOAD(DSQMF)' 'DSQSSUBS=dbname,DSQSPLAN=planid,  
DSQSDBNM=<location>...'
```

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たい場合は、63 ページの『オプション: バッチ・モード IVP の実行』まで待つ必要があります。このテストを早い段階で実行すると、プロシージャ Q.DSQIEBAT 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 (DSQIEINV) 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(DSQMF) NEWAPPL(DSQE) PARM(M=B,I=Q.DSQIEBAT,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 Q.DSQIEBAT を呼び出します。このプロシージ

ャーが呼び出されると、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。

57 ページの『第 8 章 QMF インストールのテスト』に進みます。

第 5 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済ませておく必要があります。詳細については、「*GDDM Installation and System Management*」および「*DB2 UDB (z/OS 版) 管理ガイド*」を参照してください。

CICS の DB2 に対する QMF の記述

1. DB2 UDB (z/OS 版) および CICS 間接続と、DB2 UDB (z/OS 版) の CICS 用接続機能をインストールします。

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。CICS 環境における DB2 UDB (z/OS 版) の設定について詳しくは、「*CICS DB2 Guide*」を参照してください。

2. CICS RDO 定義が DB2CONN 用に存在していることを確認します。QMF 特定の DB2ENTRY 定義が、DSQ1ECDN ジョブの実行時に作成されます。

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。

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。CICS 用に個別にバインドする必要はありません。

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. QMF810.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しているかを確認する。

```
//DSQ1EADM PROC RGN='2048K',      Job-step region size
//          QMFTPRES='QMF810 ',    QMF prefix name for target libraries
//          GDDMADM='GDDM.ADMF' GDDM ADMF data set name
```

3. QMF810.SDSQSAPE(DSQ1EADM) を実行依頼する。
4. 0 という戻りコードを確認する。戻りコードが 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 8.1 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。

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. QMF810.SDSQSAPE(DSQ1BFRM) を編集する。
2. ジョブのインストリーム・プロシージャー内のインストール・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 QMFTPRES='QMF810 ',    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(QMF810.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. QMF810.SDSQSAPE(DSQ1BFRM) を実行依頼する。
6. 0 という戻りコードを確認する。戻りコードが 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。

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

ジョブ DSQ1EGLK 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・エディットを以下のように行います。

1. ジョブ内のコメントごとに、ジョブ QMF810.SDSQSAPE(DSQ1EGLK) を編集する。
2. ジョブ QMF810.SDSQSAPE(DSQ1EGLK) を実行依頼する。
3. LINKPROG (このジョブには戻りコード 4 が戻される可能性がある) を除くすべてのステップについて、戻りコード 0 を検査する。戻りコードが 0 または 4 でない場合は、問題を訂正してからジョブを再実行します。

CICS 管理テーブルの更新

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)

QMF810.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 DSQ1CDCS 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。

戻りコード 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 以外の戻りコードが表示された場合は、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

DSQ1ECSO 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 では、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) が提供されています。DFHCSDUP を使用して、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。DSQ1ECSO は、QMF トランザクションによって適切な計画名およびグループに関連付けされた DB2ENTRY も定義します。他の QMF - DB2 UDB (z/OS 版) 考慮事項については、39 ページの『CICS の DB2 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、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. QMF810.SDSQSAPE(DSQ1ECSO) 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。
4.

```
//DSQ1ECSO PROC REG=2048K,           Job Step Region
//      QMFTP='QMF810 ',           DSN Prefix for QMF
//      CLOAD='CICS.LOADLIB',       Name of CICS Program Lib
//      CCSD='CICS.DFHCSO',        Name of CICS CSD file
//      OUTC='*'                   Print sysout class
```
5. ジョブを実行依頼し、戻りコード 0 が表示されてジョブが実行されたことを検査する。0 以外の戻りコードが表示された場合は、リスト出力を検査し、エラーを訂正してください。

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 行を使用しません。

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. DB2 用にカスタマイズされた DSNHDECP を含むライブラリーが、STEPLIB によって z/OS からアクセス可能になっていることを確認します。

```
//STEPLIB DD DSN=CICS.SDFHAUTH,DISP=SHR
//          DD DSN=DSN810.SDSNEXIT,DISP=SHR
//          DD DSN=DSN810.SDSNLOAD,DISP=SHR
```

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。

ユーザーの CICS リリースが 4.1 以上の場合、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 (z/OS 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 に入れます。

```
//DFHRPL DD ...
//          DD DSN=QMF810.SDSQLOAD,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 (z/OS 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=QMF810.DSQPNLE,DISP=SHR     QMF Panel File
//DSQDEBUG  DD DSN=QMF810.DSQDEBUG,DISP=SHR    Trace and Error Messages
//DSQUCFRM  DD DSN=QMF810.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。57 ページの『第 8 章 QMF インストールのテスト』に進みます。

第 6 章 QMF 互換モード用リモート・サーバーの構成

この章では、リモート・サーバーでの QMF 互換モードのインストールまたは移行方法について説明します。リモート・サーバーは、ローカルの DB2 UDB (z/OS 版) サブシステムからアクセス可能な DB2 サーバーです。

QMF の前リリースを含まないリモート・サーバーの場合は、以下の節の 1 つに進んでください。

- 『Linux、UNIX®、および Windows 版 DB2 アプリケーションとしての QMF 互換モードの構成』
- 48 ページの『DB2 UDB (iSeries 版) アプリケーションとしての QMF 互換モードの構成』
- 50 ページの『DB2 UDB (z/OS 版) アプリケーションとしての QMF 互換モードの構成』

QMF の前リリースを含むリモート・サーバーの場合は、50 ページの『すべてのサーバー・タイプの QMF 互換モードへの移行』に進みます。

Linux、UNIX®、および Windows 版 DB2 アプリケーションとしての QMF 互換モードの構成

この節では、下記のすべての DB2 製品を集合的に DB2 DRDA AS と呼びます。DB2 分散リレーショナル・データベース方式アプリケーション・サーバー用 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、この節で説明する各ステップを実行する必要があります。必要に応じて、下記のいずれかの製品についてより詳細に解説します。

- Linux、UNIX、および Windows 版 DB2 Universal Database™ バージョン 8.1
- DB2 DataJoiner® バージョン 2.1.1 以上

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、この節で説明する各ステップを実行する必要があります。

QMF を z/OS から DB2 DRDA AS にインストールする前に、DataJoiner 用に以下の準備をする必要があります。

- DB2 DRDA AS 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 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、Microsoft SNA Server (Windows NT 用) のパスワード処理に関する制約事項のため、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 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。

```
"db2 connect to" database-name  
"user" system-id "using" password
```

```
"db2 get authorizations"
```

- **オプション:** 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。

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 ソフトウェア・サポート用に診断情報が追加されています。

QMF 互換モードのインストール

下の表のジョブ・シーケンスに従って、QMF の前リリースが含まれていない Linux 版、UNIX 版、および Windows 版 DB2 バージョン 7.1 以上のサーバーに、互換モードをインストールし、準備します。QMF バージョン 3.3 から QMF 互換モードに移行するには、50 ページの『すべてのサーバー・タイプの QMF 互換モードへの移行』に進んでください。QMF 新機能モードをインストールするには、53 ページの『第 7 章 全サーバー・タイプ用の QMF 新機能モードの構成』を参照してください。

表 11. QMF 互換モードをインストールするためのジョブ・シーケンス

ジョブ名	目的
DSQ1BSQL	QMF インストール・パッケージおよびプランをバインドする。
DSQ1EDJ2	QMF コントロール表を作成する。
DSQ1BVW	QMF バージョン 8.1 視点を作成する。
DSQ1BPKG	QMF パッケージをバインドする。
DSQ1EDJ4	オプション: QMF サンプル表を作成する。

DB2 DRDA AS に対する QMF の開始

QMF を TSO または CICS のもとで開始した場合、QMF を DB2 DRDA AS のもとで開始するには、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。次のようにコマンドを指定してください。

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

ssid はユーザーの DB2 UDB (z/OS 版) サブシステム ID、*location* はユーザーの DB2 DRDA AS ロケーション名です。

これで、57 ページの『第 8 章 QMF インストールのテスト』に進むことができます。

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のみ実行します。

重要: このステップでは、DB2 DRDA AS からすべての QMF コントロール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 は DB2 DRDA AS に接続することができなくなります。

1. QMF810.SDSQSAPE(DSQ1EDX1) を編集する。
2. ジョブのコメントを読んで、必要に応じて値を変更する。
3. *ssid* をユーザーの DB2 UDB (z/OS 版) サブシステム ID に変更する。
4. **オプション:** 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 QMF810.SDSQSAPE(DSQ1EDX1) を実行依頼する。
6. 戻りコードが 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 SYSTEMR を検討する。

戻りコードが 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のみ実行します。

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. QMF810.SDSQSAPE(DSQ1EDX2) を編集する。
2. ジョブのコメントを読んで、必要に応じて値を変更する。
3. *ssid* をユーザーの DB2 UDB (z/OS 版) サブシステム ID に変更する。

互換モード用リモート・サーバーの構成

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 QMF810.SDSQSAPE(DSQ1EDX2) を実行依頼する。
6. 戻りコードが 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 SYSTEMR を検討する。
戻りコードが 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、その後でジョブを再実行します。

DB2 UDB (iSeries 版) アプリケーションとしての QMF 互換モードの構成

下の表のジョブ・シーケンスに従って、QMF の前リリースが含まれていない DB2 UDB (iSeries 版) バージョン 5.1 以上のサーバーに、互換モードをインストールし、準備します。QMF バージョン 3.3 から QMF 互換モードに移行するには、50 ページの『すべてのサーバー・タイプの QMF 互換モードへの移行』に進んでください。QMF 新機能モードをインストールするには、53 ページの『第 7 章 全サーバー・タイプ用の QMF 新機能モードの構成』を参照してください。

表 12. 新しい互換モードの DB2 (iSeries 版) へのインストール・ジョブ・シーケンス

ジョブ名	目的
DSQ1BSQL	QMF インストール・パッケージおよびプランをバインドする。
DSQ1EAS2	QMF コントロール表を作成する。
DSQ1BVW	QMF バージョン 8.1 視点を作成する。
DSQ1BPKG	QMF パッケージをバインドする。
DSQ1EAS4	QMF サンプル表を作成する。

DB2 UDB (iSeries 版) サーバーに対する QMF の始動

QMF を TSO または CICS のもとで始動した場合は、QMF を DB2 Universal Database (iSeries 版) のもとで始動するために、START コマンドの QMF パラメーターを変更します。

次のようにコマンドを指定してください。

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

ssid はユーザーの DB2 Universal Database (z/OS 版) サブシステム ID、location はユーザーの DB2 (iSeries 版) Universal Database ロケーション名です。

QMF の削除

このステップは、QMF をすでに含んでいる DB2 UDB (iSeries 版) 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のみ実行します。

重要: このステップでは、DB2 UDB (iSeries 版) AS からすべての QMF コントロール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 はDB2 UDB (iSeries 版) AS に接続することができなくなります。

1. QMF810.SDSQSAPE(DSQ1EDX1) を編集する。
2. ジョブのコメントを読んで、必要に応じて値を変更する。
3. *ssid* をユーザーの DB2 UDB (z/OS 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 QMF810.SDSQSAPE(DSQ1EDX1) を実行依頼する。
6. 戻りコードが 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 SYSTEMR を検討する。

戻りコードが 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、その後でジョブを再実行します。

DB2 UDB (iSeries 版) 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 UDB (iSeries 版) AS 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。

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

1. QMF810.SDSQSAPE(DSQ1EDX2) を編集する。
2. ジョブのコメントを読んで、必要に応じて値を変更する。
3. <ssid> をユーザーの DB2 UDB (z/OS 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 QMF810.SDSQSAPE(DSQ1EDX2) を実行依頼する。
6. 戻りコードが 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 SYSTEMR を検討する。

戻りコードが 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、その後でジョブを再実行します。

DB2 UDB (z/OS 版) アプリケーションとしての QMF 互換モードの構成

この章では、ローカルの DB2 UDB (z/OS 版) によってアクセスされる DB2 UDB (OS/390 版) バージョン 6.1 以上のサーバーへの QMF のインストールについて説明します。

下の表のジョブ・シーケンスに従って、QMF の前リリースが含まれていない DB2 UDB バージョン 6.1 (OS/390 版) 以上のサーバーに、互換モードをインストールし、準備します。QMF バージョン 3.3 から QMF バージョン 8.1 互換モードに移行するには、『すべてのサーバー・タイプの QMF 互換モードへの移行』に進んでください。QMF バージョン 8.1 新機能モードをインストールするには、53 ページの『第 7 章 全サーバー・タイプ用の QMF 新機能モードの構成』を参照してください。

表 13. QMF の前リリースの含まれない DB2 UDB (z/OS 版) サーバーに互換モードをインストールするジョブ・シーケンス

ジョブ名	目的
DSQ1TBAJ	オプション: QMF VCAT 名を作成する。
DSQ1BSQL	QMF インストール・パッケージおよびプランをバインドする。
DSQ1TBLJ	QMF コントロール表を作成する。
DSQ1BVW	QMF バージョン 8.1 視点を作成する。
DSQ1STGJ	表スペースを QMF IVP 用に作成する。
DSQ1EIVS	オプション: QMF サンプル表を作成する。

すべてのサーバー・タイプの QMF 互換モードへの移行

この節では、QMF バージョン 3.3、6.1、7.1、または 7.2 から QMF 互換モード・インストールへのサーバーの移行について説明します。このインストールでサポートされるサーバーは以下のとおりです。

- DB2 UDB (OS/390 版) バージョン 6.6 以上
- DB2 UDB (Linux、UNIX、および Windows 版) バージョン 8.1 以上
- DB2 UDB (UNIX および Windows 版) バージョン 7.1
- DB2 UDB (iSeries 版) バージョン 5.1 以上
- DB2 Server (VM/VSE 版) バージョン 7.1 以上
- DB2 DataJoiner バージョン 2.1.1 以上

QMF バージョン 3.3 以上がサーバーに存在する場合は、以下のステップを実行して移行を行います。QMF の以前のレベルは、まだ稼動可能であることに注意してください。以下の移行ステップを実行することによって、以前のレベルの QMF と DB2 QMF バージョン 8.1 は、同一のサーバーで共存できます。QMF の前リリースの QMF サ

サンプル表は、DB2 QMF バージョン 8.1 でも有効です。

表 14. QMF バージョン 3.3、6.1、7.1、および 7.2 から QMF 互換モードのリモート・データベース・サーバー (z/OS、Linux、UNIX、Windows 版の DB2 UDB、UNIX 版、Windows 版、OS/2 版の DB2 UDB、iSeries 版の DB2 UDB、および VM/VSE 版の DB2 Server) へのインストールの移行ジョブ・シーケンス

ジョブ名	目的
DSQ1BSQL	QMF インストール・パッケージおよびプランをバインドする。
DSQ1BVW	QMF バージョン 8.1 視点を作成する。
DSQ1BPKG	QMF パッケージをバインドする。

サーバー・レベルを移行するには、上の表にある以前のジョブ・シーケンス・パスも使用します。たとえば、DB2 QMF バージョン 8.1 を DB2 UDB (OS/390 版) バージョン 6 および DB2 UDB (z/OS および OS/390 版) バージョン 7 にインストールする場合です。上の表のステップは、サーバーの移行後に DB2 QMF バージョン 8.1 に対して行う必要がある事項について説明します。これらの同一ステップは、DB2 UDB (z/OS 版) バージョン 8 データベースが互換モードから新機能モードに移行される場合に実行する必要があります。

前リリースの QMF サンプル表は、DB2 QMF バージョン 8.1 でも有効です。そのリリースよりも以前のサンプル表がインストールされていなかった場合に、DB2 QMF バージョン 8.1 とともにインストールするには、下の表の該当するジョブを実行します。以前のサンプル表を削除するジョブも含まれています。削除したサンプル表は、必要であれば再インストールできます。

表 15. サンプル表を削除またはインストールするジョブ

ジョブ名	目的
DSQ1EDSJ	DB2 UDB (z/OS 版) サーバー用サンプル表を削除する。
DSQ1EDX2	DB2 UDB (UNIX、Windows、OS/2 版)、DB2 DataJoiner、および DB2 UDB (iSeries 版) サーバー用サンプル表を削除する。
DSQ1EVS	DB2 UDB (z/OS 版) サーバー用サンプル表をインストールする。
DSQ1EDJ4	DB2 UDB (UNIX、Windows 版)、および DB2 DataJoiner サーバー用サンプル表を削除する。
DSQ1EAS4	DB2 UDB (iSeries 版) サーバー用サンプル表をインストールする。

第 7 章 全サーバー・タイプ用の QMF 新機能モードの構成

ここでは、QMF の前リリースを含まない サポート・サーバー、または QMF 互換モードを含むサポート・サーバーで、QMF 新機能モードを構成する方法について説明します。

QMF 新機能モードをサポートするサーバー・タイプは、以下のいずれかになります。

- DB2 UDB (z/OS 版) バージョン 8.1 (NFM)
- DB2 UDB (Linux、Unix および Windows 版) バージョン 8.1 以上
- DB2 UDB (iSeries 版) バージョン 5.1 以上
- DB2 UDB (UNIX および Windows 版) バージョン 7.1

QMF 新機能モードのインストール

QMF 新機能モード・インストールは、サーバー上の QMF のバージョンのみになりません。

表 16. サーバーへの 新規 NFM インストールのジョブ・シーケンス

ジョブ名	目的
DSQ1TBAJ	オプション: QMF VCAT (DB2 UDB (z/OS 版) サーバーのみ) を作成する。
DSQ1BSQL	QMF インストール・パッケージおよびプランをバインドする。
DSQ1BLNI	QMF NFM コントロール表を作成する。
DSQ1BVW	DB2 QMF V8.1 視点を作成する。
DSQ1BPKG	QMF パッケージをバインドする。
DSQ1BINR	DB2 UDB (z/OS 版) リクエスターのみ: QMF アプリケーション・プランをバインドする。
DSQ1STGJ	DB2 UDB (z/OS 版) サーバーのみ: QMF IVP 用の QMF SAVE DATA 表スペースを作成する。
DSQ1EIVS	DB2 UDB (z/OS 版) サーバーのみ: QMF サンプル表を作成する。
DSQ1EDJ4	DB2 UDB (Unix および Windows 版) サーバーのみ: QMF サンプル表を作成する。
DSQ1EAS4	DB2 UDB (iSeries 版) サーバーのみ: QMF サンプル表を作成する。

QMF 新機能モードへの移行

次の一連のステップを実行することにより、QMF 互換モード・インストールを QMF 新機能モード・インストールに移行します。この移行を実行した後のサーバーで、前リリースの QMF は使用できません。サーバーが QMF バージョン 8.1 互換モードから QMF バージョン 8.1 新機能モードに移行すると、フォールバックはありません。

表 17. QMF 互換モードから QMF 新機能モードへの移行ジョブ・シーケンス

ジョブ名	目的
DSQ1BSQL	QMF インストール・パッケージおよびプランをバインドする。
DSQ1BLNM	QMF コントロール表を QMF NFM に移行する。
DSQ1BROG	DB2 UDB (z/OS 版) サーバーのみ: QMF 表スペースの REORG のサンプル・ジョブ。この機能は、独自のユーティリティーを使用して実行することができます。
DSQ1BINX	QMF 索引を除去および作成する。
DSQ1BVW	DB2 QMF V8.1 視点を作成する。
DSQ1BPKG	QMF パッケージをバインドする。

前リリースの QMF サンプル表は、DB2 QMF バージョン 8.1 でも有効です。前リリースのサンプル表が以前にインストールされておらず、DB2 QMF バージョン 8.1 でそれらをインストールする場合は、下の表で該当するジョブを実行してください。以前のサンプル表を削除するジョブも含まれています。必要に応じて、削除したサンプル表を再インストールすることもできます。

表 18. サンプル表を削除またはインストールするジョブ

ジョブ名	目的
DSQ1EDSJ	DB2 UDB (z/OS 版) サーバー用サンプル表を削除する。
DSQ1EDX2	DB2 UDB (Unix、Windows および OS/2 版)、DB2 DataJoiner、および DB2 UDB (iSeries 版) のサーバー用サンプル表を削除する。
DSQ1EVS	DB2 UDB (z/OS 版) サーバー用サンプル表をインストールする。
DSQ1EDJ4	DB2 UDB (Unix、Windows および OS/2 版) および DB2 DataJoiner のサーバー用サンプル表をインストールする。
DSQ1EAS4	DB2 UDB (iSeries 版) サーバー用サンプル表をインストールする。

新機能モードの構成

第 8 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。

- IVP の実行 (TSO)
- IVP の実行 (CICS)
- 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)
- バッチ・モード IVP の実行 (オプション)
- インストール後の終結処置

IVP の実行 (TSO)

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。DB2 QMF (TSO/CICS 版) インストールをテストするには、QMF を開始し、サーバーまたは DB2 UDB (z/OS 版) データベースに接続します。次に「ヘルプ」パネルが存在していることを確認して、次のコマンドを発行します。

```
LIST TABLES (OWNER=Q)
```

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 (z/OS 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 (z/OS 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 TABLE SPACE 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。

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。

33 ページの『QMF の開始』の説明のように、ログオン・プロシージャーまたは 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
5625-DB2 (C) Copyright IBM Corp. 1982, 2003
All Rights Reserved.
IBM is a registered trademark of International Business Machines
```

```
QMF ホーム・パネル
Version 8 Release 1
```

```
照会報告書作成プログラム
```

```
許可 ID (Authorization ID)
```

```
Q
```

```
接続は
SQLDS
```

```
***** ** ** *****
** ** ** ** ** ** **
** ** ** ** ** ** **
** * ** ** ** ** **
***** ** ** ** **
**
```

コマンド行にコマンドを入力するか、該当の機能キーを押してください。
ヘルプを表示するには、ヘルプ機能キーを押すか、HELP コマンドを入力してください。

```
1= ヘルプ    2= リスト    3= 終了    4= 表示    5= 図表    6= 照会
7= 検索      8= 表編集    9= 書式    10= 手順    11= PROFILE 12= 報告書
了解、コマンドを入力してください。
COMMAND ==>
```

図6. 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 UDB (z/OS 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、387 ページの『付録 A. 各種問題』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。

7. QMF インストール検査プロシーチャーの実行

QMF データ・セットで使用した実際の接頭部で QMF810 を置き換えて、次のコマンドを QMF コマンド行から発行します。

```
IMPORT PROC FROM'QMF810.SDSQSAPE(DSQ1EIVP)'
```

このコマンドは、インストール検査プロシーチャーを表示します。先頭のコメントは、コマンドの実行方法を示しています。IVP を実行する前に、現れるすべての QMF810 を、QMF データ・セットで使用した実際の接頭部で置き換えます。プロシーチャーが正常に実行されない場合は、QMF メッセージおよびメッセージ・ヘルプ・パネルを使用して問題を修正します。

これで、対話モードのインストール検査が完了しました。

IVP の実行 (CICS)

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チャー (IVP) と呼ばれる) を済ませるためのものです。インストールした DB2 QMF (TSO/CICS 版) が正しくインストールされていることを検査するには、QMF を開始してデータベースに接続し、トレース機能がインストールされていることを検証します。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。

QMF インストールのテスト

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。

```
CEMT INQUIRE QUEUE(DSQD)
```

次のような画面が表示されます。

```
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、41 ページの『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、TSO/CICS 製品用に DB2 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をテストします。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、387 ページの『付録 A. 各種問題』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, 2002
 All Rights Reserved.
 IBM is a registered trademark of International Business Machines

QMF ホーム・パネル
 Version 8 Release 1

照会報告書作成プログラム

```

    ***** ** ** *****
    ** ** ** ** ** ** ** ** **
    許可 ID (Authorization ID)
    Q
    ** ** **³³³³³³³³³³³³³³³³
    ** ** **³³³³³³³³³³³³³³³³
    接続は
    SQLDS
    ***** ** ** **³³³³³³³³³³³³³³³³
    **
    
```

コマンド行にコマンドを入力するか、該当の機能キーを押してください。
 ヘルプを表示するには、ヘルプ機能キーを押すか、HELP コマンドを入力してください。

1= ヘルプ	2= リスト	3= 終了	4= 表示	5= 図表	6= 照会
7= 検索	8= 表編集	9= 書式	10= 手順	11= PROFILE	12= 報告書

了解、コマンドを入力してください。
 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, 2002
All Rights Reserved.
IBM is a registered trademark of International Business Machines
+-----+
|                      ヘルプ: 照会報告書作成プログラム                      |
|                                                                 |
| トピックを選択してください。                                     |
|                                                                 |
| 1. What's new in Version 8                                     1 to 7 of 14 |
| 2. プロファイル                                             |
| 3. QMF コマンド                                             |
| 4. 指示照会                                                 |
| 5. SQL (構造化照会言語)                                     |
| 6. 表編集機能                                               |
| 7. 書式                                                       |
|                                                                 |
+-----+
| F1=ヘルプ  F3=終了  F7=逆方向  F8=順方向  F9=キー  F12=取り消し |
+-----+
    
```

了解、ヘルプが実行されました。先に進んでください。

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。

表リスト		
処置	名前	所有者
	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=ヘルプ F4=コマンド F5=記述 F6=最新表示 F7=逆方向 F8=順方向
F9=クリア F10=注釈 F11=分類 F12=取り消し

了解、データベース・オブジェクト・リストが表示されています。

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、一時記憶域キュー DSQD は空です。

QMF アプリケーション照会およびプロシージャのインストール (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・プロシージャを更新します。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・プロシージャを使用します。

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 プロシーチャーを実行することにより、このステップを完了してください。

プロシーチャー

説明

DSQ1ESQD

QMF の旧版からサンプル照会およびサンプル・プロシーチャーを削除します。

DSQ1ESQI

新規のサンプル照会およびサンプル・プロシーチャーを 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・プロシーチャーを削除する。

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 (z/OS 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。

- 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 UDB (z/OS 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 'QMF810.SDSQSAPE(DSQ1ESQD)'
```

ここで、*QMF810* 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は、それに応じて名前を変更してください。

- d. プロシーチャーを実行する。

2. サンプル照会およびサンプル・プロシーチャーを 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 'QMF810.SDSQSAPE(DSQ1ESQI)'
```

ここで、*QMF810* 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。

オプション: バッチ・モード IVP の実行

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、36 ページの『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.DSQIEBAT 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なリソースは、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 を割り振るようにしてください。 QMF PRINT コマンドからの出力は、このファイルに入れます。

2. QMF プロシージャ Q.DSQIEBAT を検査する。

Q.DSQIEBAT 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 QMF810 を使用しない場合は、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。必ず、SHARE=YES を指定してください。 QMF を ISPF ダイアログとして開始した場合は、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(DSQMFE) NEWAPPL(DSQE) PARM(DSQSMODE=B,DSQSRUN=JONES.DSQIEBAT)
```

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。

インストール後の終結処理

QMF の旧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。

重要: 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 8.1 と旧リリースが同じ DB2 UDB (z/OS 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 8.1 と前のリリースが、異なる DB2 UDB (z/OS 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 8.1 と旧リリースが別の DB2 UDB (z/OS 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。66 ページの表 19 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。

重要: バージョン 8.1 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 19. QMF の旧リリースから削除されるライブラリー

V3RxxMy データ・セット	V6R1 データ・セット	V7R1M0 データ・セット	V7R2M0 データ・セット
QMF3xy.ADMFE	QMF610.SDSQCLTE	QMF710.SDSQCLTE	QMF720.SDSQCLTE
QMF3xy.CICS.DFHTEMP	QMF610.SDSQEXCE	QMF710.SDSQEXCE	QMF720.SDSQEXCE
QMF3xy.DSQPMSE	QMF610.SDSQMLBE	QMF710.SDSQMLBE	QMF720.SDSQMLBE
QMF3xy.DSQDBRMD	QMF610.SDSQPLBE	QMF710.SDSQPLBE	QMF720.SDSQPLBE
QMF3xy.DSQSAMPE	QMF610.SDSQSAPE	QMF710.SDSQSAPE	QMF720.SDSQSAPE
QMF3xy.SDSQMAPE	QMF610.SDSQSLBE	QMF710.SDSQSLBE	QMF720.SDSQSLBE
QMF3xy.DSQCLSTE	QMF610.SDSQUSRE	QMF710.SDSQUSRE	QMF720.SDSQUSRE
QMF3xy.DSQEXECE	QMF610.SDSQLOAD	QMF710.SDSQLOAD	QMF720.SDSQLOAD
QMF3xy.DSQUSERE	QMF610.SDSQDBRM	QMF710.SDSQDBRM	QMF720.SDSQDBRM
QMF3xy.DSQPLIBE	QMF610.SDSQMAPE	QMF710.SDSQMAPE	QMF720.SDSQMAPE
QMF3xy.DSQSLIBE	QMF610.SDSQCHRT	QMF710.SDSQCHRT	QMF720.SDSQCHRT
QMF3xy.DSQMLIBE	QMF610.DSQPVARE	QMF710.DSQPVARE	QMF720.DSQPVARE
QMF3xy.DSQLOAD	QMF610.DSQPNLE	QMF710.DSQPNLE	QMF720.DSQPNLE
QMF3xy.DSQDBRM	QMF610.ADSQOBJ	QMF710.ADSQOBJ	QMF720.ADSQOBJ
QMF3xy.DSQTLIBE	QMF610.ADSQDBMD	QMF710.ADSQDBMD	QMF720.ADSQDBMD
QMF3xy.SDSQCHRT	QMF610.ADSQMACE	QMF710.ADSQMACE	QMF720.ADSQMACE
QMF3xy.DSQMACE	QMF610.ADSQPMSE	QMF710.ADSQPMSE	QMF720.ADSQPMSE
QMF3xy.DSQOBJ	QMF610.DSQDEBUG	QMF710.DSQDEBUG	QMF720.DSQDEBUG
QMF3xy.DSQPNLE			QMF720.SDSQEXIT
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 8.1 と旧リリースが同じ DB2 UDB (z/OS 版) サブシステム内にある場合にのみ、このステップを実行します。

1. QMF810.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)。

```
//DSQ1JFPL PROC RGN='2048K',
Job-step region size
// QMFTPRES='QMF810',           QMF prefix
// DB2EXIT='DSN810.SDSNEXIT',  DB2 Universal Database for
//                               z/OS exit library
// DB2LOAD='DSN810.SDSNLOAD'   DB2 Universal Database for
//                               z/OS program library
```

4. QMF810.SDSQSAPE(DSQ1DEL1) を編集する。
5. DSN を DB2 Universal Database (z/OS 版) サブシステムの名前で置き換え、QMF810 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(QMF810)
```

表 20. QMF の各リリースのデフォルト

旧リリース	デフォルト
QMF バージョン 7.2	QMF720
QMF バージョン 7.1	QMF710
QMF バージョン 6.1	QMF610
QMF バージョン 3.3	QMF330

6. ジョブ QMF810.SDSQSAPE(DSQ1JFPL) を実行依頼する。

ジョブが失敗した場合は、エラーを訂正してジョブを再実行します。

QMF バージョン 7.2 および以前のバージョンの DB2 サブシステムからの削除

QMF バージョン 8.1 と以前のリリースが別の DB2 UDB (z/OS 版) サブシステム内にある場合にのみ、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 (z/OS 版) エンティティーを除去します。

重要: このジョブでは、DB2 UDB (z/OS 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 (z/OS 版) サブシステム内に存在しない場合にのみ実行する必要があります。

1. QMF810.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
// QMFTPRES='QMF810',             QMF prefix
// DB2EXIT='DSN810.SDSNEXIT',     DB2 Universal Database for z/OS
//                               exit library
// DB2LOAD='DSN810.SDSNLOAD'      DB2 Universal Database for z/OS
//                               program library
```

4. メンバー QMF810.SDSQSAPE(DSQ1DEL1) を編集する。
5. DSN を DB2 UDB (z/OS 版) サブシステムの名前で置き換え、QMF810 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(QMF810)
```

表 21. QMF の各リリースのデフォルト

旧リリース	デフォルト
QMF バージョン 7.2	QMF720
QMF バージョン 7.1	QMF710
QMF バージョン 6.1	QMF610
QMF バージョン 3.3	QMF330

6. メンバー QMF810.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 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
```

```
QMF810
```

7. メンバー QMF810.SDSQSAPE(DSQ1DEL13) を編集する。
このメンバーには、QMF コントロール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 により管理される場合は、このステップを実行する必要はありません。
8. ジョブ QMF810.SDSQSAPE(DSQ1DELA) を実行依頼する。
ジョブが失敗した場合は、エラーを訂正してジョブを再実行します。

第 2 部 TSO/CICS 用 QMF の管理

第 9 章 QMF の開始	73	診断の考慮事項	108
z/OS 上での QMF のセットアップおよび開始	73	z/OS でのデフォルトのシステム初期化プロ	
z/OS の許可 ID の選択	73	シージャーのインポート	109
QMF をバッチ・ジョブとしてネイティブ			
z/OS で実行するように設定	73		
TSO 上での QMF のセットアップおよび開			
始	74	第 12 章 QMF インストール・ユーザー出	
ISPF 上での QMF のセットアップおよび		口 (DSQUOPTS)	111
開始	77	z/OS	111
CICS 上での QMF のセットアップおよび			
開始	83	第 13 章 エンド・ユーザーのための QMF	
CICS のもとでの QMF の開始の例	84	サポートの確立	113
z/OS 上での QMF データ・セットの検査	84	TSO/CICS 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	113
第 10 章 開始手順のカスタマイズ	87	インストール先システムのためのプロファ	
各セッション用の仮想記憶域の量の選択	87	イル構造の確立	113
z/OS のプログラム・パラメーター	87	新規のユーザー・プロファイルの	
DSQSBSTG (報告書データ用の記憶域の調		Q.PROFILES 表への追加	114
整)	87	固有のプロファイルを持たないユーザーに	
DSQSRSTG (アプリケーションに使用され		よる QMF の使用の防止	115
る予約記憶域の調整)	88	Q.PROFILES 表の読み取り	116
DSQSPILL (エクストラ記憶域の獲得)	90	TSO/CICS 用の正しいプロファイルの提供	120
DSQSIROW (表示用に検索される報告書の		ユーザー・プロファイルの更新	121
行数の制御)	95	Q.PROFILES 表からのプロファイルの削除	123
セッション開始時点での QMF アクティビ		SQL 特権の付与と取り消し	124
ティーのトレース	97	SQL GRANT ステートメントの使用	124
セッション開始時点での QMF アクティ		SQL REVOKE ステートメントの使用	125
ビティーのトレース	100	QMF オブジェクトおよびデータベース・オ	
プログラム・パラメーターの要約	102	ブジェクトに対するアクセスの制御	126
		z/OS でのアクセスの制御	126
第 11 章 QMF セッション制御機能	105	拡張オブジェクト・リストの活動化	139
Q.SYSTEM_INI のインストール	105	デフォルト・オブジェクト・リストの使用	141
Q.SYSTEM_INI プロシージャーを実行する時		ユーザーがデータベース内で表を作成できる	
期	105	ようにする	146
Q.SYSTEM_INI の使用法	105	z/OS での表の作成	147
QMF と共に出荷される例	106	ユーザーが図表をサポートできるようにする	151
ユーザー・セッション・プロシージャーの例	106	TSO および ISPF での図表サポート	151
オブジェクトのリストを表示するプロシー		z/OS 上の CICS での図表のサポート	152
ジャー	108	QMF コントロール表を使用した QMF オブ	
セキュリティおよびセッション・プロシー		ジェクトの保守	153
ジャーの共用	108	Q.OBJECT_DIRECTORY 表の読み取り	153
		Q.OBJECT_DATA 表の読み取り	155
		Q.OBJECT_REMARKS 表の読み取り	156

QMF 照会、書式、およびプロシージャ のリスト	156	ステップ 2 — NLF QMF の TSO 用調整	186
QMF 照会、書式、およびプロシージャ の表示	157	ステップ 3 — NLF QMF の CICS 用調 整	187
照会、書式、およびプロシージャの所有 権の転送	157	ステップ 4 — QMF NLF の Workstation Database Server 用調整 (オプション). . .	189
古くなった 照会、書式、およびプロシ ージャの削除	158	ステップ 5 — QMF NLF の DB2 UDB (iSeries 版) サーバー用調整 (オプション).	191
z/OS データ・セット内の照会、書式、お よびプロシージャのインポート	159	ステップ 6 — 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).	193
z/OS での DB2 サブシステムの保守	161	ステップ 7 — QMF 対話モードでの IVP の実行	193
データ・セットの管理	162	ステップ 8 — 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	193
コントロール表の保守	162	ステップ 9 — バッチ・モード IVP の実 行 (オプション).	195
バッファ・プールの切り替え	164	ステップ 10 — インストール後の終結処 置	195
DB2 表を使用する表および視点の保守 . . .	164	ステップ 11 — 永続ライブラリーの受け 入れ	196
z/OS での DB2 カタログ表の使用	164	ステップ 12 — CDS 間環境の作成	196
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	165		
z/OS でのローカル定義の日付 / 時刻フ ォーマット	165		
CICS z/OS でのローカル定義の日付 / 時 刻フォーマット	166		
ユーザー用文書編集インターフェースのカ スタマイズ	166		
z/OS での文書編集インターフェースのカ スタマイズ	166		
QMF EDIT コマンドのカスタマイズ	173		
z/OS での EDIT コマンド	173		
NLF 環境での英語サポートのイネープリ ング	175		
グローバル変数を使用した通貨記号の定 義	176		
第 14 章 QMF NLF の計画およびインス トール	177	第 15 章 ユーザーによるオブジェクトの印 刷	199
プロファイル表と NLF	177	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	199
QMF NLF の計画	177	CICS に関する考慮事項	200
ハードウェアとプログラム製品の要件 . .	177	GDDM サービスを使用して印刷を処理する QMF が GDDM ニックネームとインター フェースする方法	200
SMP/E 要件	178	z/OS での GDDM サービス	200
IBM ソフトウェア配布 (ISD) テープ . . .	179	QMF サービスを使用して印刷を処理する ネイティブ z/OS バッチ、TSO、および ISPF における印刷で QMF サービスを使 用	210
FMID	179	CICS で印刷を行うために QMF サービス を使用	211
インストール・プロセス	180	印刷機能キーの同義語の定義	212
準備: プログラム・ディレクトリーを読み NLF ワークシートに記入する	180	ネイティブ z/OS バッチ、TSO および ISPF	212
QMF NLF のインストール	181	CICS の印刷機能キーの同義語の定義 . .	213
ステップ 1A QMF コントロール表の更新	182	オブジェクトの印刷	214
ステップ 1B と 1C — QMF NLF サンプ ル表の設定	183		
		第 16 章 QMF コマンドのカスタマイズ	215
		QMF が提供するデフォルトの同義語の使用	215

z/OS でのデフォルトの同義語	215	出力域の特徴を示すフィールド	256
コマンド同義語表の作成	218	QMF 終了時に制御を出口ルーチンに渡す	256
z/OS でのコマンド同義語表の作成	218	編集ルーチンを HLASM (ハイレベル・アセ	
コマンド同義語定義を表へ入力	220	ンブラー) で作成	256
verb の選択	220	ネイティブ z/OS、TSO、または ISPF 用	
オブジェクト名の選択	222	の編集ルーチンを作成	256
同義語定義の選択	222	編集ルーチンを CICS 用にアセンブラー	
同義語のアクティブ化	226	で作成	259
z/OS での同義語のアクティブ化	226	言語環境プログラム (LE) を使用せずに PL/I	
コマンド同義語表の保守の最小化	227	で編集ルーチンを作成	263
1 つの同義語表を全ユーザーに割り当てる	227	LE を使用せずにネイティブ z/OS、TSO、	
同義語表の視点を個々のユーザーに割り当		または ISPF 用の編集ルーチンを作成	263
てる	228	言語環境プログラム (LE) を使用して PL/I	
第 17 章 QMF 機能キーのカスタマイズ	231	で編集ルーチンを作成	266
カスタマイズしたいキーの選択	231	言語環境プログラム (LE) を使用して、ネ	
フルスクリーン・パネルのデフォルト・キ		イティブ z/OS、TSO、または ISPF での	
ー	231	編集ルーチンを PL/I で作成	266
ウィンドウ・パネルのデフォルト・キー	232	編集ルーチンを z/OS 上の CICS 用に PL/I	
機能キー表の作成	234	で作成	269
z/OS での表の作成	234	プログラム例 DSQUXCTP	269
ユーザーの機能キー定義を表へ入力	235	PL/I 編集ルーチンと CICS との対話方法	270
コマンドと機能キーの結び付け	235	プログラムの変換	271
機能キーにラベルを付けて画面の上に配置	237	z/OS でのプログラムのコンパイル	271
キー定義の例	238	プログラムのリンク・エディット	271
カスタマイズしたいパネルの識別	240	変換、コンパイル、およびリンク・エディ	
フルスクリーン・パネル ID	240	ットの JCL ステートメント例 (z/OS 上の	
ウィンドウ・パネル ID	241	CICS)	272
新規機能キー定義のアクティブ化	243	CICS プログラム定義	272
z/OS での定義のアクティブ化	243	言語環境プログラム (LE) を使用せずに	
機能キー表の検査と問題診断	244	COBOL で編集ルーチンを作成	272
第 18 章 QMF 書式のためのユーザー独自の編集コードの作成	247	言語環境プログラム (LE) を使用せずに、	
QMF 書式	247	ネイティブ z/OS、TSO、または ISPF での	
編集コードの選択	248	の編集ルーチンを COBOL で作成	272
DATE、TIME、および TIMESTAMP 情報の		言語環境プログラム (LE) を使用して	
処理	249	COBOL で編集ルーチンを作成	276
データをフォーマット設定するためのユーザー		言語環境プログラム (LE) を使用して、ネ	
一出口ルーチンの呼び出し	250	イティブ z/OS、ISPF、および TSO で編	
z/OS でのユーザー出口ルーチンの呼び出		集ルーチンを COBOL で作成	276
し	250	編集ルーチンを z/OS 上の CICS 用に	
出口ルーチンとの情報の受け渡し	251	COBOL で作成	279
インターフェース制御ブロックのフィールド		COBOL 編集ルーチンと CICS との対話方	
ド	252	法	280
入力域の特徴を示すフィールド	254	COBOL プログラムの変換	281
		プログラム例 DSQUCTC	283
		COBOL 編集ルーチンと QMF との対話方	
		法	283

2 バイト文字セット・データの処理	283
DBCS データの編集コード	283
編集ルーチンが受け取るもの	283
編集ルーチンは必ず正しい結果を戻すこと	284
第 19 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	287
z/OS での管理プログラム出口ルーチンの使用	287
IBM 提供の管理プログラム出口ルーチンの使用	287
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	298
z/OS での管理プログラム出口の変更	298
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	302
z/OS	302
リソース制御情報を管理プログラム出口に渡す	310
DXEGOVA 制御ブロックの構造	310
リソース・コントロール表のアドレッシング	314
DXEXCBA 制御ブロックの構造	316
QMF セッション中のリソース制御情報の保管	323
ユーザー・アクティビティの取り消し	324
z/OS	325
取り消されたアクティビティに対するメッセージ	325
z/OS	325
TSO、ISPF、およびネイティブ z/OS バッチでの管理プログラム出口ルーチンのアセンブルとリンク・エディット	328
管理プログラム出口のアセンブル	328
管理プログラム出口ルーチンのリンク・エディット	328
z/OS 上の CICS での管理プログラム出口ルーチンのアセンブル、変換、リンク・エディット	329
管理プログラム出口のアセンブル	329
z/OS での DB2 管理プログラムの使用	330
リソースのモニター	331
管理プログラム間の相違点	331
最大プロセッサ時間の超過	332
DB2 管理プログラムの QMF への適用	332

第 20 章 バッチ・プログラムとしての QMF の実行	335
TSO/CICS でのバッチ・プログラムとしての QMF の実行	335
TSO	335
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	341
ネイティブ z/OS での QMF バッチの実行	353
CICS での非対話式トランザクションとしての QMF の実行	355
端末からのバッチの実行	355
端末なしでのバッチの実行	356
プロシージャのデバッグ	356
終了戻りコード	357

第 21 章 障害追及と問題診断	359
一般的な問題の障害追及	359
初期化エラーの処理	359
警告メッセージの処理	360
印刷時の GDDM エラーの処理	361
z/OS での印刷時の QMF エラーの処理	363
表示エラーの処理	364
パフォーマンス上の問題の解決	365
診断援助機能を使用した問題の判別	366
症状に応じた適切な診断援助機能の選択	366
QMF メッセージ・サポートを使用した問題の診断	367
QMF トレース機能の使用	369
異常終了の診断	376
QMF 割り込み機能の使用	378
Q.ERROR_LOG 表からのエラー・ログ報告書の使用	380
IBM への問題報告	381
ServiceLink を使用して既に報告されている問題の検索	381
IBM サポート・センターと共に作業する	384

第 9 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。

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、*DB2 QMF アプリケーション開発の手引き* を参照してください。

z/OS 上での QMF のセットアップおよび開始

z/OS では、QMF を TSO、ISPF の下でバッチ・ジョブとして、または CICS の下で稼働させるようにセットアップできます。

z/OS の許可 ID の選択

QMF セッションは、QMF セッションを開始させるユーザーまたはプログラムの許可 ID の下で開始され、稼働します。

1 つの SQL 許可 ID または 1 つの 1 次許可 ID、および 1 つまたは複数の 2 次許可 ID を割り当てることができます。

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 の最大長は、DB2 データベースで許可される長さです。先頭文字は英字である必要がありますが、他の 7 文字は英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 (z/OS 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。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。

QMF をバッチ・ジョブとしてネイティブ z/OS で実行するように設定

エンド・ユーザーがネイティブ z/OS でバッチ・ジョブとして QMF を開始できるようにするために、予備ファイルが存在する個所、パネルが保管される個所、パネルのファイル名、および他の表や QMF オブジェクトの名前とロケーションを定義する JCL を作成します。

QMF コマンドを実行する場合は、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。74 ページの図 7 では、プロシージャの名前は 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ー X を実行します。プロシージャー X が完了すると、QMF は終了します。QMF 戻りコードがレジスター 15 に戻されます。標準 JCL 条件コード・テストを使用して、その戻りコードをテストすることができます。

```
//RUNQMF EXEC PGM=DSQMFE,PARM='M=B,I=X,P=QMF810,S=DSN'
//*****
//*      Program load libraries
//*****
//STEPLIB DD DSN=QMF810.SDSQLOAD,DISP=SHR
//          DD DSN=DSN810.SDSNEXIT,DISP=SHR
//          DD DSN=DSN810.SDSNLOAD,DISP=SHR
//          DD DSN=GDDM.GDDMLOAD,DISP=SHR
//*****
//*      QMF/GDDM maps
//*****
//ADMGMAP DD DSN=QMF810.SDSQMAPE,DISP=SHR
//*****
//*      Data sets used by QMF                *
//*****
//DSQPRINT DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330)
//DSQDEBUD DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=122,BLKSIZE=1210)
//DSQDUMP DD SYSOUT=*,DCB=(RECFM=VBA,LRECL=125,BLKSIZE=1632)
//DSQSPIII DD DSN=&&SPIII,DISP=(NEW,DELETE),
// UNIT=SYSDA,SPACE=(TRK,(100),RLSE),
// DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)
```

図7. ネイティブ z/OS のバッチで QMF プロシージャーを実行するための JCL

ネイティブ z/OS で QMF を実行する場合は、QMF プロシージャーで使用されるデータ・セット名を完全修飾名にする必要があります。TSO の接頭部および接尾部は、ネイティブ z/OS では使用できません。

TSO 上での QMF のセットアップおよび開始

ログオン・プロシージャーでの 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。

ログオン・プロシージャーでの DD ステートメントの使用

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・プロシージャー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ーは端末モニター・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。

QMF を開始するための TSO EXEC を作成する場合は、プログラム・ロード・ライブラリー、モジュール、およびデータ・セットが QMF で使用可能であり、GDDM および DB2 の要件に適合することを確認する必要があります。

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 (IBM ではデフォルトの出口ルーチンも提供) を介して許可 ID を割り当てます。ユーザーの TSO ログオン ID が渡されている場合は、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。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。

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 コマンドでの設定値に影響を受けます。76 ページの図 8 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。

```
//*****
//*          DATA SETS USED BY TSO                               *
//SYSPROC DD DSN=SYS2.CLIST,DISP=SHR                * CLIST Library
//          DD DSN=          QMF810.SDSQCLTE,DISP=SHR
//SYSEXEC DD DSN=SYS2.EXEC,DISP=SHR
//          DD DSN=QMF810.SDSQEXCE,DISP=SHR
//SYSHelp DD DSN=SYS1.HELP,DISP=SHR
//EDT     DD DSN=&EDIT,UNIT=SYSDA,SPACE=(1668,(40,12))
```

図 8. TSO で使用されるデータ・セット

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 'QMF810.SDSQLOAD(DSQMFE)' 'DSQSMODE=I,DSQSSUBS=DB2T'
```

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。ほとんどの場合、DB2 ライブラリーおよび GDDM ライブラリーは、TASKLIB の一部ではありません。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。

DSQQMFE モジュールを使用した QMF の直接開始

READY モードでコマンド行から、あるいは CLIST または EXEC で、DSQQMFE を入力することによって、TSO の下で QMF を開始することができます。

```
DSQQMFE DSQSBSTG=123456,DSQSDBUG=ALL,DSQSIROW=0,DSQSRUN=SAM.PROG1
```

QMF を ISPF からは独立して、TSO で開始するときは、次の戻りコードが有効です。

- 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

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、z/OS 上で 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 DSQSMOD=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。この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。

省略符号は、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

次の例は、開始方法、および ISPF とは独立に作動している QMF にパラメーターを渡す方法を示しています。TSO 用の最初の 2 つのステートメントでは、L2 トレース (DSQSDEBUG=NONE) をオンにし、DSQSBSTG (報告書用の最大記憶域) の値として 50,000 を渡し、DSQSMODE (操作のモード) の値として B (バッチ) を渡します。

- TSO READY モードから開始する場合:

```
DSQQMFE DSQSBSTG=50000,DSQSDEBUG=NONE,DSQSMODE=B
```

- CLIST または EXEC 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せん。

- QMF の開始時に実行させるために、初期プロシージャー Q.IPROC を指定する。
- 変数 &TABLE 用のプロシージャーに、値 Q.STAFF を渡す。

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)
```

ISPF 上での QMF のセットアップおよび開始

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。

- QMF を ISPF の初期ダイアログに追加する。
- 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 (z/OS)。

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。

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。

QMF の開始

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 呼び出しまたは 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)
```

2. QMF を初期ダイアログとして開始する場合は、QMF セッション中に、分割画面から QMF を入力したり、分割画面を作成したりすることはできません。

ISPF メニューからの QMF の開始

QMF を開始するようにメニュー・オプションのセットアップを選択する場合、メニューは QMF を指していなければなりません。79 ページの図 9 は、ISPF マスター・アプリケーション・メニューのサンプル定義を示すもので、このメニューにオプションを追加する方法を示しています。この定義例では、CLIST を介して QMF に到達するために、オプション 2 が追加されています。

```

)BODY
%----- MASTER APPLICATION MENU -----
%SELECT APPLICATION ==>_OPT +
%
%                                +USERID   -
%                                +TIME      -
% 1 +SPF      - SPF PROGRAM DEVELOPMENT FACILITY +TERMINAL -
% 2 +QMF      - RUN QMF UNDER THE ABC SUBSYSTEM  +PF KEYS  -
%
%
%
%
%
%
%
%
%
% 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(ISP@PRIM) NEWAPPL'
              2, 'PGM(DSQMFE) NEWAPPL(DSQE) PASSLIB PARM(DSQSSUBS=ABC)'
              /*                                */
              /* ADD Other APPLICATIONS HERE */
              /*                                */
              P, 'PANEL(ISPOPT)'
              X, 'EXIT'
              ' '
              *, '?' )
)END

```

図9. マスター・アプリケーション・メニュー例

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速くなる可能性があります。TSO ログオン・プロシージャーによってユーザー・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。CLIST に残される機能は 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 サブシステムであり、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ーを使用して 1 つの CLIST を作成する場合もあります。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)'
```

z/OS での LIBDEF ステートメントの使用

ISPF セッション中に、オプションで ISPF LIBDEF ステートメントを使用して、QMF ライブラリーを割り振ることができます。

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、プログラム・ライブラリーを DSQLLIB という固有の QMF DD NAME 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 DD NAME 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('QMF810.SDSQEXIT','QMF810.SDSQLOAD',  
'DSN810.SDSNEXIT','DSN810.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、81 ページの図 10 のような CLIST を作成します。この 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。

```

/*****
/* Allocate QMF and DB2 Programs to DSQLLIB */
/*****
ALLOC FI(DSQLLIB) SHR REUSE
      DA('QMF810.SDSQEXIT',
        'QMF810.SDSL0AD',
        'DSN;810.SDSNEXIT',
        'DSN;810.SDSL0AD')
/*****
/* Allocate QMF libraries used for GDDM */
/*****
ALLOC FI(ADMGGMAP) DA('QMF810.SDSQMAPE') SHR REUSE
ALLOC FI(ADMCFORM) DA('QMF810.DSQCFORM') SHR REUSE
ALLOC FI(DSQUCFRM) DA('QMF810.DSQUCFRM') SHR REUSE
ALLOC FI(ADMGDF) DA('QMF72.CHARTLIB') SHR REUSE
/*****
/* Allocate QMF product data sets */
/*****
ALLOC FI(DSQPRINT) SYSOUT(Z) LRECL(133) RECFM(F B A ) BLKSIZE(1330)
ALLOC FI(DSQPNLE) DA('QMF810.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 ISPLLIB DATASET ID('QMF810.SDSQPLBE')
ISPFEXE LIBDEF ISPLLIB DATASET ID('QMF810.SDSQSLBE')
ISPFEXE LIBDEF ISPLMLIB DATASET ID('QMF810.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 ISPLLIB LIBRARY ID( )
ISPEXEC LIBDEF ISPLLIB LIBRARY ID( )
ISPEXEC LIBDEF ISPLMLIB LIBRARY ID( )
FREE FI(DSQLLIB)
/*****
/* Free QMF product data sets */
/*****
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(DSQUCFRM)
FREE FI(ADMGDF)

```

☒ 10. ISPF LIBDEF CLIST

ISPF でのバッチ・モードによる QMF の開始

QMF をバッチ・モードで開始することにより、リソースおよび時間を節約できる可能性があります。

z/OS では、QMF の開始に 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。次のステートメントのいずれかを、JCL の SYSTSIN データ・セットに指定します。

- CLIST を使用しない場合:

```
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(...DSQSMODE=B,DSQRUN=aa.bbb)
```

- CLIST を使用する場合:

```
ISPSTART CMD(clist_name) NEWAPPL
```

ここで、*clist_name* は QMF を開始する CLIST の名前です。

PARM は、適正な操作モード (DSQSMODE=B) を設定し、実行するプロシージャ (DSQRUN=aaa.bb) を識別し、そのプロシージャ用の変数を組み込むことができます。

PARM の後の省略符号は、必須パラメーター DSQSMODE および DSQRUN 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。プロシージャの名前には、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,DSQRUN=JONES.PROCA)
```

図 11. ユーザー名およびプロシージャ名を指定した、ISPF でのバッチ・モードによる QMF の開始

プロシージャの実行後、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またはコマンドを実行することができます。z/OS で ISPF が停止すると、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: 次の例は、開始方法、および ISPF の下の QMF にパラメーターを渡す方法を示しています。

- ISPF を CLIST (z/OS) 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) に値 150 を渡し、DSQSRSTG (記憶域の予約量) に値 0 を渡します。

- z/OS 上の 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場合:

```
)PROC
      &SEL = TRANS( TRUNC (&OPT, '.')
                  1, 'PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSPILL=NO)'
                  .
                  .
                  .
```

このコードにより DSQSPILL に NO が渡されます。

- CLIST から開始し、初期プロシージャーを指定する場合:

```
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せん。

- QMF の開始時に実行させるために、初期プロシージャー Q.IPROC を指定する。
- 変数 &TABLE 用のプロシージャーに、値 Q.STAFF を渡す。

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
```

CICS 上での QMF のセットアップおよび開始

QMF を CICS 用に調整したら、次のように QMF トランザクション (デフォルトのトランザクションは QMFE) を CICS 画面から開始します。

QMFE parameters

ここで、QMFE はトランザクション 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 コマンドを実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')
```

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 アプリケーションを実行する場所が指定される場合は、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

定するには、該当の端末が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 のもとでの QMF の開始の例

CICS のもとでの QMF の開始の例

次の例では、CICS の下での QMF の開始を示しています。

- 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、次のことを行いません。

- QMF の開始時に実行させるために、初期プロシージャ Q.IPROC を指定する。
- 変数 &TABLE 用のプロシージャに、値 Q.STAFF を渡す。

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)
```

z/OS 上での QMF データ・セットの検査

次に挙げるデータ・セットのリストは、TSO 内で QMF により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。割り振りを変える場合は、呼び出し EXEC を変更する必要があります。

表 22. TSO の QMF で使用されるデータ・セット

データ・セット	目的
DSQPNLE	QMF パネル・ファイル
DSQDUMP	QMF スナップ・ダンプ出力
DSQDEBUG	QMF トレース・ダンプ出力
DSQSPRINT	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

z/OS 上でのプログラム・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 12 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=QMF810.SDSQLOAD,DISP=SHR          * QMF MODULES *
// DD DSN=ISR.V41IM0.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=DSN810.SDSNEXIT,DISP=SHR * DB2 MODULES *
// DD DSN=DSN810.SDSNLOAD,DISP=SHR * DB2 MODULES *
// DD DSN=GDDM230.SADMMOD,DISP=SHR * GDDM MODULES *
```

図 12. 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

z/OS 上での 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

DSQCFRM

ユーザー定義の ICUFORMS

```
//*****
//*      QMF/GDDM DATA SETS                        *
//*****
//ADMGGMAP DD DSN=QMF810.SDSQMAPE,DISP=SHR * GDDM Map Group
//ADMCFORM DD DSN=QMF810.SDSQCHRT,DISP=SHR * QMF-Supplied Chart Forms
//DSQCFRM DD DSN=aaaaaa,DISP=SHR * Saves User-defined ICUFORMS
//ADMCDATA DD DSN=xxxxx,DISP=SHR
//ADMGDF DD DSN=xxxxx,DISP=SHR
//ADMSYMBL DD DSN=xxxxx,DISP=SHR
```

図 13. QMF/GDDM データ・セット

第 10 章 開始手順のカスタマイズ

この章では、QMF セッションをカスタマイズするためにプログラムにパラメーターを渡す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 のパラメーターの引き渡しの詳細については、「DB2 QMF アプリケーション開発の手引き」を参照してください。

各セッション用の仮想記憶域の量の選択

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、拡張ページング可能リンク・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、ページング可能リンク・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 z/OS 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。

z/OS のプログラム・パラメーター

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量
- 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

0 バイト

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 ~ 99,999,999 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。

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、DSQSBSTG が使用されます。TSO、ISPF、またはネイティブ z/OS のデフォルトは 0 です。

TSO 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) を提供します。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。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。ユーザーは、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してください。管理プログラム出口ルーチンの詳細については、287 ページの『第 19 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。

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します。予備ファイルがいっぱいになった場合は、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。

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

短縮形

R

有効な値

0 ~ 99,999,999 バイト

デフォルト

0

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 ~ 99,999,999 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、z/OS 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 z/OS、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。ユーザーが z/OS システム・サービスまたは TSO コマンドを使用し、DSQSBSTG=0 および DSQSRSTG=0 である場合は、障害が起きて異常終了が生じるリスクがあります。その理由は、それらのサービス用の記憶域を QMF は予約していないためです。さらに、ほとんどの普通のユーザーも、インストール先定義の QMF コマンドを実行する際に、気付かずに非 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

YES

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。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。

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にされます。

DSQQMFn 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
- 予備ファイル用に定義された記憶域 (CICS では DFHTEMP、その他では DSQSPILL) がいっぱいになった。

非 CICS ユーザーに関する予備ファイルの割り振り

ログオン・プロシージャ JCL か CLIST における DD ステートメント、または FILEDEF 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。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、固定長レコードを持つ 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、z/OS 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 次エクステントの使用は、予備ファイルが必要なおきの際にのみに限られる。
- 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 次エクステント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。

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。TSO の DASD の記憶域を超過することがあり得ます。

QMF の記憶域要件を満たすためには、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十分な大きさを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。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 (92 ページの表 23 を使用する)。

- 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。
 - 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、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、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 23. 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)
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 VARGRAPHIC 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 24 は、予備ファイルの計算の例を示したものです。

表 24. 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:

$$\frac{600,000 \text{ rows}}{7} * \frac{1 \text{ track}}{10 \text{ blocks}} * \frac{1 \text{ cylinder}}{15 \text{ tracks}} = 571 \text{ cylinders}$$

R=6000、2 バッファー / 行 の場合:

$$6000 \text{ rows} * \frac{2 \text{ blocks/row}}{10 \text{ blocks}} * \frac{1 \text{ track}}{15 \text{ tracks}} * \frac{1 \text{ cylinder}}{15 \text{ tracks}} = 800 \text{ cylinders}$$

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。

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場合、QMF がページの分割を行う必要がある場合。

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 DSQSPILL(YES/NO) を設定する必要があります。当該ジョ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。

「DB2 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: 複数のユーザーが QMF に関して同一のログオン・プロシージャを使用していて、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不十分なことが、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となります。

他のユーザーにとっての予備ファイル問題の発生: 予備ファイルの 2 次割り振りを増やすことで、ユーザーの予備ファイル問題を解決することがありますが、そうした場合、他のユーザーにとっての予備ファイルの問題を生じる可能性があります。したがって、2 次割り振りを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使用されていないボリュームに移動することを検討してください。

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実行しなかった場合です。この間は、他のユーザーが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。

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要因に影響されます。

- 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タスクを行うかどうか。(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。
- 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、スクロールで戻る距離、および報告書の複雑さ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。

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・ロックは解除されます。また、表示された報告書全体をより速くスクロールすることができます。ただし、これによって、最初の報告書画面の表示は遅くなります。ロックの解除により、他のユーザーのパフォーマンスが向上する可能性もあります。

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 ～ 99,999,999 の範囲の任意の数

デフォルト

最初の報告書画面の表示前に検索される最小行である 100

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、91 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、「DB2 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が妨げられる可能性があります。

QMF コントロール表またはシステム・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。

- 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。

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 0 を使用します。

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、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。

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、データベースの行の取り込みを完了します。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。

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・アクティビティの追跡、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追跡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 コマンドを実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。

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 DSQSDBUG 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す。

DSQSDBUG 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。

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、369 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) を使用します。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、369 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 コマンドを実行するので、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。

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・アクティビティの追跡、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追跡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 コマンドを実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。

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。

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、369 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) を使用します。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 という名前の一時的データ・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、369 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

開始手順のカスタマイズ

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 コマンドを実行するので、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。

プログラム・パラメーターの要約

下の表は、パラメーターの名前と短縮形および各パラメーターに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 25. 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	QMF トレースに使用される CICS リソースの名前
DSQSDBQT	—	CICS	QMF トレースに使用される CICS リソースのタイプ
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
DSQPILL	L	TSO、CICS	予備ファイルの使用
DSQSPLAN	P	TSO、CICS	QMF アプリケーション・プランの名前

表 25. プログラム・パラメーター (続き)

名前	短縮形	環境	説明
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
DSQSSPQN	—	CICS	QMF 予備ファイルの名前
DSQSSUBS	S	TSO、CICS	DB2 サブシステムの名前

第 11 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

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、次のように指定できます。

```
SAVE PROC AS Q.SYSTEM_INI (SHARE=YES,COMMENT='QMF System
Initialization Procedure')
```

許可 ID Q のもとでプロシージャを保管するためには、ユーザーは QMF 管理者である必要があります。QMF 管理者は、値が 1 のグローバル変数 DSQAO_QMFADM を持ちます。

Q.SYSTEM_INI プロシージャを実行する時期

Q.SYSTEM_INI プロシージャは、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 プロシージャに使用可能なすべての QMF 機能が、Q.SYSTEM_INI プロシージャ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 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 プロシーチャーの例では、SHARE=YES をすべてのユーザーのデフォルトにします。

DSQSUSER パラメーターを指定すると、QMF により CONNECT コマンドが実行されて、データベースに接続されます。したがって、このパラメーターの規則は、CONNECT コマンドのものと同じです。

- DSQSUSER パラメーターに指定する ID には DB2 CONNECT 権限が必要であり、さもないと QMF セッションが開始されません。SQL GRANT ステートメントを使用して、この権限を付与します。

```
GRANT CONNECT TO userid IDENTIFIED BY password
```

- DSQSUSER に指定する DB2 許可 ID およびパスワードは、DB2 用の CONNECT コマンドの規則に準拠している必要があります。
- SQL 許可 ID およびパスワードは共に、DB2 UDB (z/OS 版) 用の DB2 システム表 SYSIBM.SYSUSERAUTH の中に存在する必要があります。

```
--
-- 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 14. QMF と共に出荷される Q.SYSTEM_INI

ユーザー・セッション・プロシーチャーの例

セッション・プロシーチャーは別のプロシーチャーを呼び出すことができます。呼び出される側のプロシーチャーは、QMF ユーザーが作成、所有、および更新したユーザー・プロシーチャー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チャー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。

```

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 15. ユーザー定義プロシージャを呼び出す 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=MYSAPCE)       -- 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 16. ユーザー・セッション・プロシージャの例: user.USER_INI

オブジェクトのリストを表示するプロシージャ

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 17. 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は、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、制限されているか、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、369 ページの『QMF トレース機能の使用』を参照してください。

z/OS での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

z/OS では、デフォルトの QMF システム初期化プロシージャが提供されています。このプロシージャは DSQ0BINI と呼ばれます。QMF810.SDSQSAPE(DSQ0BINI) で見付けることができます。

サンプルをインストールする前に、システム内にシステム初期化プロシージャがあるかどうかを検査することができます。コマンド `DISPLAY Q.SYSTEM_INI` を使用すると、既にインストールされているものが示されるか、または、初期化プロシージャがインストールされていない場合は、メッセージ "Q.SYSTEM_INI cannot be found" が出力されます。既にシステム初期化プロシージャがあり、それをサンプルで上書きする場合、または、システム初期化プロシージャがなく、サンプルをインストールする場合は、以下の例を続けます。

```
IMPORT PROC FROM'QMF810.SDSQSAPE(DSQ0BINI)'
```

ユーザー独自のバージョンのプロシージャをインポートすることも、デフォルトのプロシージャをインポートすることも、それを保管する前に変更することもできます。あるいは、QMF 内でユーザー独自のプロシージャを作成することもできます。

第 12 章 QMF インストール・ユーザー出口 (DSQUOPTS)

QMF バージョン 8.1 用の新しい QMF インストール・ユーザー出口である DSQUOPTS は、選択されたグローバル変数の初期デフォルト値をオーバーライドするのに使用されます。

DSQUOPTS の第 1 レベルでサポートされるグローバル変数は、DSQEC_DISABLEADM および DSQEC_SHARE です。これらのグローバル変数のいずれか、または両方の初期デフォルト値を、提供された QMF デフォルトとは別の値に設定することができます。

たとえば、DSQEC_DISABLEADM の QMF 初期デフォルト値は 0 です。これは、QMF で QMF 管理者権限検査が行われることを意味します。DSQUOPTS が変更されて DSQEC_DISABLEADM が初期値 1 に設定された場合、QMF 管理者権限検査は行われず、QMF を実行するユーザーが QMF 管理者と見なされることはありません。

QMF インストール・ユーザー出口 DSQUOPTS は、DSQUOPTS アセンブラー・ソースを変更し、モジュールをアセンブルおよびリンク・エディットすることによって変更されます。

z/OS

z/OS の場合、DSQUOPTS アセンブラー・ソースは、QMF810.SDSQUSRE データ・セットのメンバー DSQUOPTS にあります。オーバーライド値の指定方法の詳細については、DSQUOPTS のプロログを参照してください。DSQUOPTS のアセンブルおよびリンク・エディットのサンプル・ジョブは、QMF810.SDSQSAPE データ・セットのメンバー DSQ1UOPT にあります。変更された DSQUOPTS ロード・モジュールは、QMF 出口ライブラリー QMF810.SDSQEXIT に収容されることに注意してください。変更した出口ルーチンを探し出せるように、出口ライブラリーを適切に割り振ることを忘れないようにしてください。DSQUOPTS のデフォルト・バージョンは、QMF810.SDSQLOAD データ・セットに入れて出荷されます。

第 13 章 エンド・ユーザーのための QMF サポートの確立

QMF 機能を使用して、エンド・ユーザー向けのサポートのカスタマイズに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、QMF をセットアップする方法を説明します。

TSO/CICS 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

コード・ページの考慮事項: QMF は、GDDM 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 System Customization and Administration 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 は、以下の QMF リソースを所有、または管理します。

- すべての QMF コントロール表
- サンプル照会
- QMF と一緒に出荷されるサンプル表 (サンプル表の説明については「DB2 QMF 解説書」を参照)
- 139 ページの『拡張オブジェクト・リストの活動化』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。

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、コントロール表 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。116 ページの『Q.PROFILES 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、『新規のユーザー・プロファイルの Q.PROFILES 表への追加』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。

- 使用に備えて 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持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 表エディター (「DB2 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。図 18 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 TSO 環境に作成する SQL の例です。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 18 に示してある値は、使用できるプロファイル値の例です。

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', 'TSO')
```

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'
'TSO')
```

図 18. TSO でのユーザー・プロファイルの作成

注: 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 NULL 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。114 ページの図 18 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 116 ページの『Q.PROFILES 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、プレート照会を設定します。置換変数の使用の詳細については、「DB2 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。図 19 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。表エディターを使用することもできます。これは「DB2 QMF 使用の手引き」で説明されています。

基本 QMF (英語)

ドイツ語 NLF

```
DELETE FROM Q.PROFILES
      DELETE FROM Q.PROFILES
WHERE CREATOR='SYSTEM'
      WHERE CREATOR='SYSTEM'
AND TRANSLATION='ENGLISH'
      AND TRANSLATION='DEUTSCH'
```

図 19. QMF の使用を、固有のプロファイルを持つユーザーに制限

注: 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。

Q.PROFILES 表の読み取り

表 26 は、Q.PROFILES コントロール表の列です。コントロール表の各列は、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、変換版の表 Q.PROFILE を参照してください (n シンボルを、ix ページの表 1 の NLID で置き換えます)。

Q.PROFILES 表には、属性 UNIQUE および CLUSTER を備えた索引 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持つことはできません。

表 26. Q.PROFILES 表の構造

列名	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 (z/OS の場合)
CREATOR	互換モードで実行している場合は CHAR(8)。新機能モードで実行している場合は VARCHAR(128)。	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 SQL 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持たないユーザーは、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、「DB2 QMF 解説書」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>

表 26. Q.PROFILES 表の構造 (続き)

列名	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 (z/OS の場合)
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、「DB2 QMF 解説書」を参照。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>
WIDTH	CHAR (18)	可	<p>機能: 1 ページ当たりの印刷列数を制御する。</p> <p>値: 22 ~ 999。デフォルト = 132。</p>
LENGTH	CHAR (18)	可	<p>機能: 1 ページ当たりの印刷行数を制御する。</p> <p>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。</p>
LANGUAGE	CHAR (18)	可	<p>機能: RESET QUERY コマンドが実行された後で、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。</p> <p>値: SQL (デフォルト)、QBE (例示照会)、または PROMPTED (指示照会)。</p>

QMF サポートの確立

表 26. Q.PROFILES 表の構造 (続き)

列名	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 (z/OS の場合)
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 TABLE SPACE 名と呼ぶ。操作には影響しない。DB2 DataJoiner は表スペースを利用しないので、DB2 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: 任意の有効な表スペース名。</p>
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。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。NLF で変換される値は ALL と NONE だけである。</p>

表 26. Q.PROFILES 表の構造 (続き)

列名	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 (z/OS の場合)
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。</p> <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 コマンドを実行する必要があります。</p> <p>FREE DDNAME(DSQPRINT)</p>
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。 ix ページの表 1 の右側に、この列で使用する必要がある変換済みの名前が示してある。</p>
PFKEYS	<p>互換モードで実行している場合は CHAR(31)。 新機能モードで実行している場合は LONG VARCHAR(261)。</p>	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。</p>

QMF サポートの確立

表 26. Q.PROFILES 表の構造 (続き)

列名	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 (z/OS の場合)
SYNONYMS	互換モードで実行している場合は CHAR(31)。 新機能モードで実行している場合は LONG VARCHAR(261)。	可	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。 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。 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。
RESOURCE_GROUP	CHAR (16)	可	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。 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 SQL 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)。
MODEL	CHAR (8)	可	機能: データ・アクセスのモデルを指定する。 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。
ENVIRONMENT	CHAR (8)	可	機能: 操作環境を指定する。 値: z/OS を介してプロファイルにアクセスする場合、この値は TSO、CICS のいずれか。

TSO/CICS 用の正しいプロファイルの提供

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 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 または TSO から開始される場合、それぞれ CICS、z/OS、または TSO 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つて、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、「DB2 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 表エディターを使用して更新できます。表エディターについては、「DB2 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 表のすべてのフィールドを更新することができます。

既存のユーザー・プロファイルを更新する場合は、図 20 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 20. 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

注: 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ものとします。NONPRIME グループ内のユーザーについては、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 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 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・プロファイルを削除するのは、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、153 ページの『QMF コントロール表を使用した 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、164 ページの『DB2 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 21 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 21. 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。

z/OS でのプロファイルの削除

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。DROP ステートメントについては、「DB2 UDB (z/OS 版) SQL 解説書」に説明があります。ERASE コマンドについては、「DB2 QMF 解説書」で説明されています。

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、操作しているとします。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ものとします。

```
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、ユーザーの現行許可 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。

現行許可 ID ではない修飾名で表、視点、または索引を作成するには、DBA 権限を持つ必要があります。

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 が次のコマンドを実行する必要があるとします。

```
EDIT TABLE ORDER__BACKLOG (MODE=CHANGE
```


ユーザーがこの表の所有者である場合は、次のステートメントを使用して、変更モードで表 ORDER_BACKLOG を編集するために必要な SQL UPDATE 特権を JONES に付与します。

```
GRANT UPDATE ON ORDER_BACKLOG TO JONES WITH GRANT OPTION
```

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、「DB2 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。「DB2 QMF 使用の手引き」に、プロシージャの作成方法の説明があります。

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 22 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 22. 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、該当する「DB2 Universal Database SQL リファレンス」を参照してください。

注: 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。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 23. 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。

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、および表エディター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らない場合があります。

- 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。
- 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ーに対するアクセスを制御することができます。

z/OS でのアクセスの制御

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をアクセスする必要があります。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。インストール時に、QMF プランおよびパッケージは PUBLIC への EXECUTE 権限を付与 (GRANT EXECUTE) されます。必要ならば、取り消して (REVOKE)、特定の認可をユーザー ID/ グループに発行することができます。

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、次のようにユーザー JONES にアクセスを付与します。

```
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となります。

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、およびプロシーチャーの処理後は、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、z/OS 上での 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となります。

他のユーザーが元ユーザーと許可 ID を共有している場合は、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 を共有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。

オブジェクトのアクセスに必要な DB2 特権

QMF 照会、表エディター、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と同じものです。

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、所有するコントロール表を更新します。ユーザーが QMF の管理にまったく関与しない場合は、このようなコントロール表に対する 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 コマンドを実行して、表の内容を表示します。

QMF コマンド、指示照会と QBE 照会、および表エディターに関して必要な DB2 特権は、135 ページの『オブジェクトのアクセスに必要な 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、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
```

被認可者の特権を取り消し、その被認可者が依然としてその特権を持っている場合、該当の被認可者は、特権を別のユーザーから受け取ったこととなります。

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 UDB (z/OS 版) 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 特権を持つことになります。

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 特権を受けるのは、SQL GRANT 照会による場合、z/OS での 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 UDB (z/OS 版) 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 ID がユーザーの現行 SQL 許可 ID でない場合は、その ID にログオンするか、または可能であれば、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、被認可者は 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となります。

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、視点を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 (z/OS 版) SQL 解説書」の CREATE VIEW 照会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。特権の被認可者は、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、それらの表から 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持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。

z/OS での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、誰か別のユーザーにその保守をしてもらうとします。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になります。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、CREATETS 特権 (データベースの表スペースを作成するためのもの)、および CREATETAB 特権 (データベース内に表を作成するためのもの) も含まれます。

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: z/OS の SAVE および IMPORT コマンド: CICS では IMPORT コマンド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。

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 DB2 UDB (z/OS 版) データベース DSNDB04 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。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となります。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。

注: データベースは DB2 UDB (z/OS 版) デフォルトのデータベース (DSNDB04) でもかまいません。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 (DSNDB01、DSNDB03、または DSNDB05) 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。

```
GRANT CREATETAB ON DATABASE &dbname TO &authid
GRANT CREATETS ON DATABASE &dbname TO &authid
GRANT USE OF TABLE SPACE &dbname.&tbspname TO &authid
```

ここで、

&dbname

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 DATABASE2 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 DATABASE2 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。

z/OS での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になります。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。

z/OS での 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 照会でできること、およびできないことについては、「DB2 UDB (z/OS 版) 管理ガイド」を参照してください。「DB2 UDB (z/OS 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。

オブジェクトのアクセスに必要な SQL 特権

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 27 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。

表 27. 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	必要な特権は表エディター・モードによって異なる。
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。

表 27. 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
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。
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。

SELECT、INSERT、UPDATE、または DELETE などの SQL 特権の詳細については、「DB2 UDB (z/OS 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 28 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 28. 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、「DB2 QMF 使用の手引き」を参照してください。

表エディターに必要な SQL 特権: 表 29 を使用して、ユーザーが使用する必要のある表エディターを見つけ、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 29. 表エディター・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エディターの機能	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

表エディターの詳細については、「*QMF 使用の手引き*」を参照してください。

SQL 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 が次のコマンドを実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、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 24. 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、「*DB2 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 プロシージャを使用して、この作業を行うことを考慮する必要があります。QMF 使用の手引き にプロシージャ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、次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、それらのユーザーがそれぞれ他のユーザーに INSERT 権限を付与することを可能にします。

```
GRANT INSERT ON ORDER__BACKLOG TO PUBLIC WITH GRANT OPTION
```

図 25. 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 (z/OS 版) SQL 解説書*」を参照してください。

注: 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。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。

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ー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、次のコマンドでは、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 26. 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、「DB2 QMF 解説書」を参照してください。

オブジェクトの所有者は、次に示すように、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 27. 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、「DB2 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 プロシージャー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。これはデフォルトです。

DB2 QMF バージョン 8.1 の場合、値 0 は、データベース・サーバー DB2 (OS/390 版) バージョン 4 以上と共に使用される場合のみ有効になります。

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。

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、「DB2 QMF 解説書」を参照してください。

拡張オブジェクト・リストの活動化

拡張オブジェクト・リストによって、ユーザーは、グループ ID に属する DB2 表、ユーザーが所有する表、および共通で表示可能な表をリストできます。この場合は、表特権は、ユーザー ID ではなく、グループ ID に付与されます。これらのグループ ID または 2 次許可 ID にアクセスできるすべてのユーザーは、その特権を持ちます。

拡張オブジェクト・リストを使用するには、QMF によって提供されたユーザー定義関数 (UDF) をインストールし、アクティブ化する必要があります。UDF は、DB2 UDB (OS/390 版) V6 またはそれ以降のデータベースにインストールする必要があります。拡張オブジェクト・リストをインストールして活動化するには、以下のステップを行います。

1. UDF 用に環境をセットアップする。このステップでは、WLM 設定済みアドレス・スペースの DB2 ストアード・プロシージャおよび UDF 用に環境をセットアップして維持します。通常システム管理者は、このステップを行います。DB2 ストアード・プロシージャおよび UDF 用 WLM 設定済みアドレス・スペースのセットアップの詳細については、DB2 UDB (z/OS 版) のマニュアルを参照してください。
2. QMF 提供 UDF を実行する WLM 設定済みアドレス・スペースに QMF プログラム DSQABA1E を追加する。DSQABA1E は、QMF ロード・ライブラリー QMF810.SDSQLOAD にあります。WLM 設定済みストアード・プロシージャ・アドレスの STEPLIB 連結にあるロード・ライブラリーに SDSQLOAD から DSQABA1E メンバーをコピーします。これは、アドレス・スペースを開始するために使用する JCL PROC に定義された STEPLIB 連結です。

3. QMF 提供 UDF が実行される WLM 環境名を取得する。これは、WLM 環境を開始するために使用する JCL PROC の APPL ENV= パラメーターで指定されます。
DB2 インストーラーで、DB2 インストール・パネル DSNITIPX のオプション 6 「WLM Environment (WLM 環境)」が指定されている場合は、DB2 はデフォルトを前提にします。これは、マクロ DSN6SYSP で ZPARM "WLMENV" として記録され、DB2 インストール・ジョブ・ストリーム DSNTIJUZ で明確にリストされます。
4. QMF UDF を DB2 に定義する。これは、QMF810.SDSQSAPE ライブラリーに配置されている QMF 提供ジョブ DSQ1BUDF を実行することによって実行される登録アクティビティです。このジョブは、SQL CREATE FUNCTION ステートメントを発行し、実行特権を付与します。実行する前に、ジョブ DSQ1BUDF を調整する必要があります。
5. 登録を検査する。次のステップで「QMF List (QMF リスト)」視点を変更する前に、上記ステップが正常に行えたかどうか検査してください。登録を検査するには、QMF または SPUFI を開始して、次の SQL を実行します。

```
SELECT U.AUTHNAME FROM TABLE( Q.APPL_AUTHNAMES( 'PUBLIC "PUBLIC*"') ) U
```

結果は、上記 SQL ステートメントを実行するユーザーに有効な許可名のリストです。以下はそのリストの形式例です。

```
AUTHNAME
-----
W397754
#DQZA
#J49A
DB2FUNC
QMFDEV
PUBLIC
PUBLIC*
```

6. QMF UDF を実行するように QMF リスト視点を変更します。視点を変更するには、QMF810.SDSQSAPE に配置されている QMF 提供ジョブ DSQ1BUDV を実行します。
7. (オプション) IBM 提供の表視点を調整する場合は、QMF 指定の UDF を使用するように変更された次の SELECT ステートメントを参照してください。この例は、ユーザーのカスタマイズされた視点の変更に役立ちます。

```
SELECT T.CREATOR, T.NAME, ...
FROM SYSIBM.SYSTABLES T
,( SELECT DISTINCT TA.TCREATOR, TA.TTNAME
FROM SYSIBM.SYSTABAUTH TA
WHERE TA.GRANTEETYPE=' '
AND TA.GRANTEE IN
( SELECT U.AUTHNAME
FROM TABLE( Q.APPL_AUTHNAMES( 'PUBLIC
"PUBLIC*"') ) U
) AS UAT ("CREATOR", "NAME")
WHERE T.CREATOR=UAT.CREATOR AND
T.NAME=UAT.NAME AND T.TYPE
IN ('T', 'V')
```


次の表のジョブ・シーケンスに従って、DB2 UDB (z/OS 版) サーバー V8.1 の QMF V8.1 互換モードまたは新機能モードに、「拡張リスト (Enhanced List)」視点をインストールします。

表 30. 「拡張リスト (Enhanced List)」視点をインストールするジョブ・シーケンス

ジョブ名	目的
DSQ1BSQL	現行サーバーにインストール・プログラムをバインドする。
DSQ1BUDF	「Enhanced QMF List (拡張 QMF リスト)」視点機能を作成する。
DSQ1BUDV	「Enhanced QMF List (拡張 QMF リスト)」視点を作成する。

「Enhanced List (拡張リスト)」視点が希望どおりに機能しない場合は、ジョブ DSQ1BVW を実行して「QMF default List (QMF デフォルト・リスト)」視点を復元します。

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 コマンドを実行すると、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、DSQEC_TABS_RDB2、または DSQEC_TABS_SQL 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実行すると、QMF はグローバル変数 DSQEC_COLS_LDB2、DSQEC_COLS_RDB2、または DSQEC_COLS_SQL 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 により提供される視点の完全なリストについては、付録 B を参照してください。QMF は、以下のデフォルトの視点を提供し、DB2 UDB (z/OS 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 UDB (z/OS 版)、およびワークステーション・データベース・サーバーまたは iSeries データベース・サーバー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 28 は、DB2 UDB (z/OS 版) に提供される視点を示します。

```
CREATE VIEW Q.DSQEC__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 28. LIST コマンドの表のリストを提供するデフォルトの視点 (z/OS)

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実行します。

```
SET GLOBAL (DSQEC__TABS__LDB2 = QMFADM.LOCAL__DB2__TABLES
```

視点 Q.DSQEC__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 UDB (z/OS 版) でなければなりません。

```

CREATE VIEW Q.DSQEC_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 29. LIST コマンドの表のリストを提供するデフォルトの視点 (z/OS)

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実行します。

```
SET GLOBAL (DSQEC__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。

視点 Q.DSQEC__ALIASES は、表のリストに関する別名のリスト、あるいは DB2 UDB (z/OS 版)、DB2 ワークステーション、または iSeries サーバー内の別名に関する列情報だけを選択します。

```

CREATE VIEW Q.DSQEC__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 30. 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 (z/OS)

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実行します。

```
SET GLOBAL (DSQEC__ALIASES = QMFADM.DB2__ALIASES
```

```
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 31. DESCRIBE コマンドの列情報を提供するデフォルトの視点 (z/OS)

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実行します。

```
SET GLOBAL (DSQC_COLS_LDB2 = QMFADM.LOCAL_DB2_COLUMNS)
```

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実行します。

```
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。

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実行します。

```
SET GLOBAL (DSQC_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 SYSIBM.SYSTABAUTH 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、145 ページの図 32 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、「DB2 QMF 解説書」を参照してください。

```
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 32. 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

使用される新規の視点名のため、対応する変数に対して SET GLOBAL を実行することを忘れないでください。

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 視点に似ているが、列リストについては SYSIBM.SYSTABLES または 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大域変数の使用の詳細については、「DB2 QMF 使用の手引き」を参照してください。大域変数のリスト、およびアプリケーションでの大域変数の使用については、「DB2 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。

注: QMF により提供される視点の完全なリストについては、393 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。

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、「DB2 QMF 解説書」およびオンライン・ヘルプを参照してください。

- QMF IMPORT TABLE または IMPORT VIEW コマンド。

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、「DB2 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。

z/OS での表の作成

表 31. 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	ユーザーが自分自身で表を作成
<p>ステップ 1</p> <p>最初に DB2 を使用する前に、表スペースを作成し、そのスペースを DB2 に定義する。該当する「DB2 UDB 管理ガイド」を使用すれば、表スペースを作成するために割り当てる権限を決める際に役立ちます。</p>	<p>ステップ 1</p> <p>「DB2 UDB (z/OS 版) 管理ガイド」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。</p>
<p>ステップ 2</p> <p>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 コマンドを実行するか、または IMPORT TABLE コマンドを実行する。表の作成の例については、「DB2 QMF 使用の手引き」を参照してください。</p>	<p>ステップ 2</p> <p>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る。SYSTEM プロファイルは、そのデフォルト値を変更したい場合は更新できます。</p>

QMF サポートの確立

表 31. 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 3</p> <p>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する。CREATE INDEX ステートメントおよび表の論理設計の詳細については、「DB2 UDB (z/OS 版) SQL 解説書」を参照してください。</p>	<p>ステップ 3</p> <p>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する。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。</p>
<p>ステップ 4</p> <p>表にデータを入れる。DB2 UDB (z/OS 版) LOAD ユーティリティー、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 (z/OS 版) ユーティリティーの手引きと解説書」では、LOAD ユーティリティーの使用方法について説明しています。「DB2 QMF 使用の手引き」では、QMF におけるオブジェクトのエクスポートおよびインポートについての説明があります。</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う。これらのコマンドの詳細については、「DB2 QMF 解説書」を参照してください。</p>
<p>ステップ 5</p> <p>表に関する DB2 および SQL 特権を必要なユーザーに付与する。</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 特権を付与する。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>

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 (z/OS 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ガイドラインについては、「DB2 UDB (z/OS 版) 管理ガイド」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 (z/OS 版) SQL 解説書」の CREATE TABLESPACE 照会の説明を参照してください。

表スペースの詳細については、「DB2 UDB (z/OS 版) 管理ガイド」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 (OS/390 版および z/OS 版) 管理ガイド」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

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ます。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。

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 (z/OS 版) 管理ガイド」を参照してください。

ユーザーへの DB2 CREATETAB 権限の付与 (z/OS)

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 33 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。

```
GRANT CREATETAB on database DBASEA TO userid1, userid2, userid3, ...
```

図 33. 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。

表スペースの作成の詳細、および DB2 権限レベルの説明については、「DB2 UDB (z/OS 版) 管理ガイド」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 (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・プロシージャー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、次のように指定できます。

```
ALLOC DSNAME(aaaaaaaa) DDNAME(DSQUCFRM) SHR
```

IBM 提供の図表書式は、ライブラリー QMF810.SDSQCHRT 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、QMF810.SDSQCHRT 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。

z/OS 上の 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。

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 Presentation Graphics Feature Interactive Chart Utility User's Guide*」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せん。

QMF コントロール表を使用した QMF オブジェクトの保守

QMF の照会、書式、およびプロシーチャーを保管する QMF コントロール表を、定期的に圧縮および再編成する必要があります。QMF コントロール表の定期保守には、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。

IBM 提供の QMF コントロール表の完全なリストについては、「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。

QMF の照会、書式、およびプロシーチャーはすべて、次の 3 つの QMF コントロール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 155 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 156 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生じることもあります。

コントロール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、およびプロシーチャーごとに、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。

QMF サポートの確立

Q.OBJECT_DIRECTORY 表は 表 32 に示す構造を持っています。

表 32. 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。
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	オブジェクトを共用していない場合 (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用している場合は NO。
MODEL	CHAR	8	可	この値は常に 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・スタンプ値を示す。この値は SAVE または IMPORT コマンドの後で記録されます。
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 33 に示す構造を持っています。

表 33. 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	VARCHAR	データベース によって決定 される。	不可	オブジェクトの作成者の許可 ID を示 す。
NAME	VARCHAR	データベース によって決定 される。	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)	データベース によって決定 される。	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。 重要: 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。

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。（「DB2 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 34 に示す構造を持っています。

表 34. 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	データベース によって決定 される。	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	データベース によって決定 される。	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	データベース によって決定 される。	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、157 ページの図 34 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 34. 特定のユーザーが所有している書式、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 が照会の場合には、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 列に反映されません。

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 35. 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。

注: 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 Y にリセットできますが、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、158 ページの図 36 に示す照会を使用します。必ず、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

注: 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ようにします。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 SET OWNER = 'newuserid' WHERE OWNER = 'olduserid' AND NAME IN namelist	UPDATE Q.OBJECT_REMARKS SET OWNER = 'newuserid' WHERE OWNER = 'olduserid' AND NAME IN namelist	UPDATE Q.OBJECT_DATA SET OWNER = 'newuserid' WHERE OWNER = 'olduserid' AND NAME IN namelist
---	---	--

図 36. QMF オブジェクトの他のユーザーへの転送

図 36 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、次のように指定できます。

```
('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有する必要がある場合は、157 ページの図 35 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 37 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 コントロール表

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 WHERE OWNER = 'olduserid'	DELETE FROM Q.OBJECT_REMARKS WHERE OWNER = 'olduserid'	DELETE FROM Q.OBJECT_DATA WHERE OWNER = 'olduserid'
---	---	--

図 37. QMF コントロール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 つのコントロール表から該当する行をすべて削除することができます。

z/OS 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 z/OS 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・コントロール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・コントロール表 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 ユーティリティを実行する。
2. 次の照会を実行する。

```
SELECT SPACE
FROM SYSIBM.SYSTABLEPART
WHERE TSNAME='tttttttt' AND DBNAME='DSQDBCTL'
```

ここで、tttttttt 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ならば、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・コントロール表用の表スペースを拡大するには、次のようにします。

1. 表スペースのイメージ・コピーを作成する。
これは、プロシーチ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。

QMF サポートの確立

```
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。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 35. QMF オブジェクトを保管するコントロール表用の表スペース

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 36. DB2 パラレル・エディション VIR2 データベースまたは DB2 Universal Database を使用して QMF オブジェクトを保管するコントロール表のノード・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 コントロール表。	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 コントロール表。	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必要があります。

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 (z/OS 版) ユーティリティ・ガイド および解説書」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かった場合、DB2 QMF バージョン 8.1 が DB2 管理対象の表スペース・データ・セットを作成します。

z/OS での DB2 サブシステムの保守

注: 特に断りがない限り、この節の説明は DB2 UDB (z/OS 版) についてのものです。

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。

ワークステーション・データベース・サーバーのユーザー: 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 権限だけを付与します。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。

DB2 管理のデータ・セット用の記憶域グループ

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 37 に示すように、索引を持つ各コントロール表ごとに、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。

ワークステーション・データベース・サーバーのユーザー: 記憶域グループは適用されません。

表 37. コントロール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

コントロール表のデータ・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。

コントロール表の保守

コントロール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 はコントロール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 で DSN プロセッサを介して発行するか、または DB2I の SPUFI 機能を介して対話式に発行することができます。

ワークステーション・データベース・サーバーのユーザー: 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。

これらの主題に関する情報については、「DB2 UDB (z/OS 版) 管理ガイド」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、コントロール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、他のコントロール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 コマンドを実行する。

DB2-STOP DATABASE コマンドの説明については、「DB2 UDB (z/OS 版) ユーティリティ・ガイドおよび解説書」を参照してください。

コントロール表のモニターと再編成

DB2 システム・カタログを介してコントロール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 (z/OS 版) 管理ガイド」を参照してください。

RUNSTATS ユーティリティの実行: コントロール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、コントロール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 (z/OS 版) 管理ガイド」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。

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、コントロール表が入る表スペース用またはコントロール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、コントロール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。

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 (z/OS 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 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 カタログ表の使用法の詳細については、該当する DB2 UDB SQL 解説書を参照してください。

表または視点の所有権の転送は、非常に困難なタスクになることがあります。

z/OS での DB2 カタログ表の使用

注: 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。

表と視点のリスト

165 ページの図 38 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 DB2 UDB (z/OS 版) の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 38. 特定のユーザーが所有する DB2 表および視点のリスト (OS/390)

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 コマンドを使用します。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、該当する *DB2 UDB 管理ガイド* 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

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。

z/OS でのローカル定義の日付 / 時刻フォーマット

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 (z/OS 版) 管理ガイド*」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 は適切なルーチンと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。

編集ルーチンの使用可能化

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 39 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 39. 編集ルーチンの使用可能化

CICS z/OS でのローカル定義の日付 / 時刻フォーマット

他の QMF 稼働環境で使用可能なローカル定義の日付・時刻編集コード (TTL および TDL) は、CICS 上の QMF では利用不能です。IBM により提供されないこれらの機能を実行する編集出口ルーチンの作成を選択した場合、編集コードとして TTL および TDL を使用できません。代わりに、ユーザーのローカル日付 / 時刻出口ルーチンを識別するために、編集コード Uxxxx または Vxxxx を使用します。

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、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。

z/OS での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 エディター用の IBM 提供のマクロです。

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。

z/OS では、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。

QMF810.DSQ*

ISPF/PDF および PS/TSO エディターは、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、「DB2 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必要があります。

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 z/OS の QMF810.SDSQCLTE のメンバーです。
- それ以外のコンポーネントは z/OS の QMF810.SDSQSAPE のメンバーです。

文書インターフェース・マクロ DSQAED1P 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「DB2 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 QMF810.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 QMF810.DSQCLSTn のメンバー DSQAED1P 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、GETQMFn 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 QMF810.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 'QMF810.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 'QMF810.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ジャ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 QMF810.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ジャ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。

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 z/OS のライブラリー QMF810.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。その代わりに、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 z/OS 上の QMF810.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またはミニディスク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 z/OS 上の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 または DSQAED2P 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、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、およびいくつかの QMF コマンド用のキーワードが含まれます。

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: z/OS のライブラリー QMF810.DSQSAMP n のメンバー DSQA n D0L 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: 各ページ脚注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:
```

DSQABD03: 各ページ見出し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 QMF810.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。z/OS では、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 用の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、QMF810.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=QMF810.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 = QMF810.DSQSAMP&LANGCHAR
SET &BASELIB = QMF810.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 QMF810.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 = 'QMF810.DSQPLIB&LANGCHAR' /* ISPF Panel Library */
SET MNAME = 'QMF810.DSQMLIB&LANGCHAR' /* ISPF Message Library */
SET SNAME = 'QMF810.DSQSLIB&LANGCHAR' /* ISPF Skeleton Library */
SET LNAME = 'QMF810.SDSQLOAD' /* QMF Modules */
ISPEXEC LIBDEF ISPPLIB 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を外してください。

```

/*****/
/* 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 ISPPLIB DATASET ID() */
/* ISPEXEC LIBDEF ISPMLIB DATASET ID() */
/* ISPEXEC LIBDEF ISPSLIB 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 QMF810 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。

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、87 ページの『第 10 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、エディターで QMF の照会およびプロシージャーを変更できます。エディターの 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)。

z/OS での EDIT コマンド

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エディターの使用を前提としています。EDIT TABLE コマンドは、表エディターを呼び出すもので、テキスト・エディターを必要としません。

EDIT コマンドでエディターを使用可能にするには、以下のようになります。

1. エディターを呼び出すための 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. エディター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 QMF810.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 コマンドでエディター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 (173 ページ) 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、エディター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、「DB2 QMF 解説書」を参照してください。

6. 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 コマンドでエディター名として 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) エディターを指定した場合、QMF は QMF アプリケーション ID の DSQE または DSQn (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 エディター画面を持つ必要がある場合、PDF エディターを 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 で実行されるエディターを呼び出します。

175 ページの図 40 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 40. 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せん。

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 verb、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 により制御されます。

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。

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言語を変更するには、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 セッション内から言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 14 章 QMF NLF の計画およびインストール

QMF NLF (各国語機能) 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。

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ます、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。プロシージャーに著しい相違点がある場合は、NLF のインストール・プロシージャー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、179 ページの表 40 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 および CICS の環境では、これらの行はユーザー ID の SYSTEM とともに挿入されます。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 UDB 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

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、19 ページの『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。SMP/E および QMF のターゲットおよび配布ライブラリーのデータ・セットのサイズを見積もる方法については、QMF プログラム・ディレクトリーを参照してください。

QMF NLF の配布ライブラリー

NLF の DB2 QMF バージョン 8.1 配布ライブラリーは次のとおりです。

- QMF810.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。
- QMF810.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 38 に示されています。

表 38. 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
QMF810.ADSQMACn	QMF NLF インストール・プロシージャ	15	13	15
QMF810.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) は下の表のとおりです。

表 39. 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
QMF810.SDSQSAPn	17	15	17
QMF810.SDSQPLBn	1	1	1
QMF810.SDSQCLTn	2	1	2
QMF810.SDSQMLBn	1	1	1
QMF810.SDSQEXCn	1	1	1
QMF810.SDSQUSRn			

IBM ソフトウェア配布 (ISD) テープ

QMF NLF をインストールするには、最初に IBM ISD テープにある情報をお読みください。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 8 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
- DB2 QMF バージョン 8.1 NLF が使用するパネルその他のアイテム

ISD テープは SMP/E (RELFILE) フォーマットです。フォーマットについては、「OS/390 SMP/E 解説書」を参照してください。

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 40 に示されています。

表 40. 言語 ID と FMID

各国語機能	言語 ID	QMF 8.1 FMID
U/C 英語	U	JSQ8851
デンマーク語	Q	JSQ8855
フランス語	F	JSQ8856
ドイツ語	D	JSQ8857
イタリア語	I	JSQ8858
日本語漢字	K	JSQ8859
韓国語 (ハングル)	H	JSQ885A
ブラジル・ポルトガル語	P	JSQ885B
スペイン語	S	JSQ885C
スウェーデン語	V	JSQ885D
スイス・フランス語	Y	JSQ885E
スイス・ドイツ語	Z	JSQ885F
カナダ・フランス語	C	JSQ885G

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。

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。この章で説明するステップを実行する前に、インストールする QMF NLF ごとに SMP/E インストールを完了する必要があります。SMP/E インストールの説明は、QMF プログラム・ディレクトリーに含まれています。

NLF では、DB2 QMF バージョン 8.1 サンプル・ライブラリー QMF810.SDSQSAPE およびロード・モジュール・ライブラリー QMF810.SDSQLOAD を使用する必要があります。

準備: プログラム・ディレクトリーを読み 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 のプログラム・ディレクトリー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。下の表に、QMF NLF のインストール時に用意する必要がある情報を示します。この表はワークシートとして使用できます。

表 41. QMF NLF インストール・パラメーター (バージョン 8.1)

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 = QMF810)	
配布ライブラリー接頭部 (デフォルト = QMF810)	
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 =V8R1)	
ローカル DB2 出口ライブラリー (デフォルト =DSN810.SDSNEXIT)	
ローカル DB2 ロード・ライブラリー (デフォルト =DSN810.SDSNLOAD)	

注: この章で説明される QMF NLF インストールは、QMF 新機能モードまたは QMF 互換モードのインストールのいずれかに該当します。

ステップ 1A QMF コントロール表の更新

- 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 1Aa — 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 3.3 およびそれ以降がこの DB2 サブシステムにある場合は、183 ページの『ステップ 1B — 以前の QMF NLF サンプル表の削除』にスキップする。

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 SYSTERM の出力の中にあります。SYSPPRINT 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 1Aa — 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、以下を実行します。

- NLF 項目を Q.PROFILES 表に追加する。このジョブは QMF810.SDSQSAPh(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 QMF810.SDSQSAPh(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。

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 (**QMF810**)
DB2EXIT
DB2 出口ライブラリーの名前 (**DSN810.SDSNEXIT**)
DB2LOAD
DB2 プログラム・ライブラリーの名前 (**DSN810.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)
- DSQ1nCCS (必須 SQL ステートメントを実行する場合)

SYSTEMM の完了メッセージを検査します。エラーが発生した場合は、SYSTSPRT または SYSPRINT でエラー・メッセージを検査します。

ジョブの再実行: ジョブが失敗した場合は、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 1B と 1C — QMF NLF サンプル表の設定

以下のいずれかの条件に適合する場合は、ステップ 3B と 3C をスキップしてください。

- NLF が大文字機能 (UCF) である。
- QMF NLF の前の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 1B — 以前の QMF NLF サンプル表の削除

QMF NLF の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 8 NLF をインストールする場合は、このステップを実行してください。それ以外の場合は、184 ページの『ステップ 1C — 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 8 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 QMF810.SDSQSAPn(DSQ1nDSJ) です。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。必要な場合は、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・パラメーター値を以下のように変更してください。

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 (**QMF810**)

DB2EXIT

DB2 出口ライブラリーの名前 (**DSN810.SDSNEXIT**)

DB2LOAD

DB2 プログラム・ライブラリーの名前 (**DSN810.SDSNLOAD**)

RGN ジョブ・ステップの領域サイズ (**2048K**)

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 QMF810.SDSQSAPn 内のジョブ DSQ1nDSJ を実行します。SYSTEMM の完了メッセージを検討します。エラーが発生した場合は、SYSTSPRT または SYSPRINT でエラー・メッセージを検査します。

ジョブの再実行: ジョブが失敗した場合は、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、ジョブが除去を試行する表がすでに除去されていた場合は、そのジョブが失敗することがあります。

ステップ 1C — 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 QMF810.SDSQSAPn(DSQ1nIVS) です。必要な場合は、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・パラメーター値を以下のように変更してください。

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 (**QMF810**)

DB2EXIT

DB2 出口ライブラリーの名前 (**DSN810.SDSNEXIT**)

DB2LOAD

DB2 プログラム・ライブラリーの名前 (**DSN810.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 UDB 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 (z/OS 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、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 QMF810.SDSQSAPn 内のジョブ DSQ1nIVS を実行します。SYSTEM の完了メッセージを検討します。エラーが発生した場合は、SYSTSPRT または SYSPRINT でエラー・メッセージを検査します。

ジョブの再実行: ジョブが失敗した場合は、エラーを訂正してジョブを再実行します。

QMF NLF を別のデータベースにインストールする場合は、193 ページの『ステップ 6 — バッチ IVP 実行のための NLF バッチ・ジョブのセットアップ (オプションル)』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、187 ページの『ステップ 3 — NLF QMF の CICS 用調整』を参照してください。

ステップ 2 — 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 用に調整する手順については、29 ページの『第 4 章 QMF の TSO 用調整』に概要を示しています。

- 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=QMF810.DSQMAPn,DISP=SHR
```
 - ISPPLIB 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。


```
//ISPPLIB DD DSN=QMF810.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=QMF810.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=QMF810.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=QMF810.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=QMF810.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、次の例のようになります。


```
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 (DSQMFn 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(DSQMFE) NEWAPPL(DSQE)'
                3, 'PGM(DSQMFn) NEWAPPL(DSQn)'
                /* */
                /* ADD OTHER APPLICATIONS HERE */
                /* */
                P, 'PANEL(ISPOPT)'
                X, 'EXIT'
                ' '
                ' '
                *, '?' )
)END

```

図 41. NLF 用 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 3 — NLF QMF の CICS 用調整

QMF 製品を 39 ページの『第 5 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。バージョン 3.3 から移行する場合は、すべてのステップを実行してください。

ステップ 3A — 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。

```
DSNCRCT TYPE=ENTRY, TXID=QMFn, PLAN=QM720, AUTH=DEPT1
```

QMF には、QMFn.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 3B — 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 QMF810.SDSQSAPn(DSQInADM) 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。必要な場合は、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・パラメーター値を以下のように変更してください。

表 43. DSQInADM 用のインストール・パラメーター

パラメーター名	値の記述	デフォルト
QMFTPRE	QMF ターゲット・ライブラリーの接頭部名	QMF810
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 3C — 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 QMF810.SDSQSAPn(DSQInCSD) 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。必要な場合は、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・パラメーター値を以下のように変更してください。

表 44. DSQInCSD 用のインストール・パラメーター

パラメーター名	値の記述	デフォルト
QMFTPRE	QMF ターゲット・ライブラリーの接頭部名	QMF810
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.SDFHLOAD
CCSD	CICS CSD データ・セットの名前	CICS.DFHCSO

ステップ 3D — 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=QMF810.DSQPNLn,DISP=SHR
```

ここで、*n* は NLF 文字です。

ステップ 3E — IVP の実行

59 ページの『IVP の実行 (CICS)』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 QMF810.SDSQSAPn に
- DSQ1EIVC を DSQ1nIVC に

ここで、*n* は NLF 文字です。

ステップ 4 — 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する場合のみ、このステップに示されている各ステップを実行してください。

QMF NLF を z/OS 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ための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、45 ページの『第 6 章 QMF 互換モード用リモート・サーバーの構成』を参照してください。

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。

ステップ 4A — Workstation Database Server での QMF NLF コントロール表の作成

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. QMF810.SDSQSAPE(DSQ1nDJ2) 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)。

```
//DSQ1TBJ4 PROC RGN='2048K',           Job-step region size
//           QMFTPRES='QMF810',       Prefix for QMF target libraries
//           DB2EXIT='DSN810.SDSNEXIT', Exit DB2 library name
//           DB2LOAD='DSN810.SDSNLOAD' DB2 program library name
```

3. SYSTEM(DSN) の DSN を DB2 UDB (z/OS 版) サブシステム ID に変更する。
4. ジョブ QMF810.SDSQSAPE(DSQ1nDJ2) を実行依頼する。
5. 戻りコード 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、SYSTEMM を検討する。

戻りコードが 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 4B — 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. QMF810.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
//           QMFTPRES='QMF810',       Prefix for QMF target libraries
//           DB2EXIT='DSN810.SDSNEXIT', Exit DB2 library name
//           DB2LOAD='DSN810.SDSNLOAD' DB2 program library name
```

3. SYSTEM(DSN) の DSN を DB2 UDB (z/OS 版) サブシステム ID に変更する。
4. ジョブ QMF810.SDSQSAPE(DSQ1nDJ4) を実行依頼する。
5. 戻りコード 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、SYSTEMM を検討する。

戻りコードが 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。

重要: 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. QMF810.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
//                QMFTPRE='QMF810',     Prefix for QMF target libraries
//                DB2EXIT='DSN810.SDSNEXIT', Exit DB2 library name
//                DB2LOAD='DSN810.SDSNLOAD' DB2 program library name
```

3. SYSTEM(DSN) の DSN を DB2 UDB (z/OS 版) サブシステム ID に変更する。
4. ジョブ QMF810.SDSQSAPE(DSQ1nDX1) を実行依頼する。
5. 戻りコード 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、SYSTEMM を検討する。

戻りコードが 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. QMF810.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
//                QMFTPRE='QMF810',     Prefix for QMF target libraries
//                DB2EXIT='DSN810.SDSNEXIT', Exit DB2 library name
//                DB2LOAD='DSN810.SDSNLOAD' DB2 program library name
```

3. SYSTEM(DSN) の DSN を DB2 UDB (z/OS 版) サブシステム ID に変更する。
4. ジョブ QMF810.SDSQSAPE(DSQ1nDX2) を実行依頼する。
5. 戻りコード 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、SYSTEMM 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 5 — QMF NLF の DB2 UDB (iSeries 版) サーバー用調整 (オプション)

QMF の DB2 UDB (iSeries 版) Database Server 用のサポートはオプションです。DB2 UDB (iSeries 版) Database Server を QMF NLF のアプリケーション・サーバーと

して実行する場合にのみ、ここに示されている各ステップを実行してください。 z/OS から、QMF NLF を DB2 UDB (iSeries 版) Database Server にインストールする前に、QMF 基本製品を DB2 UDB (iSeries 版) Database Server データベースにインストールするために必要なステップを実行したことを確認する必要があります。

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。

DB2 UDB (iSeries 版) サーバーでの QMF NLF コントロール表の作成

1. QMF810.SDSQSAPE(DSQ1nAS2) 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)。

```
//DSQ1nAS2 PROC RGN='2048K', Job-step region size
// QMFTPRE='QMF810', Prefix for QMF target libraries
// DB2EXIT='DSN810.SDSNEXIT', Exit DB2 library name
// DB2LOAD='DSN810.SDSNLOAD' DB2 program library name
```

3. SYSTEM() で DB2 UDB (z/OS 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 QMF810.SDSQSAPE(DSQ1nAS2) を実行依頼する。
6. 戻りコード 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、SYSTEMR を検討する。戻りコードが 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、その後でジョブを再実行します。

DB2 UDB (iSeries 版) サーバーでの QMF NLF サンプル表の作成

1. QMF810.SDSQSAPE(DSQ1nAS4) 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)。

```
//DSQ1nAS4 PROC RGN='2048K', Job-step region size
// QMFTPRE='QMF810', Prefix for QMF target libraries
// DB2EXIT='DSN810.SDSNEXIT', Exit DB2 library name
// DB2LOAD='DSN810.SDSNLOAD' DB2 program library name
```

3. SYSTEM() で DB2 UDB (z/OS 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 QMF810.SDSQSAPE(DSQ1nAS4) を実行依頼する。
6. 戻りコード 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、SYSTEMR を検討する。戻りコードが 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、その後でジョブを再実行します。

ステップ 6 — 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 36 ページの『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てください。

```
ISPSTART PGM(DSQMFn) NEWAPPL(DSQn) PARM(DSQSMODE=B,DSQSRUN=Q.DSQ1nBAT)
```

ステップ 7 — QMF 対話モードでの IVP の実行

IVP の実行については、57 ページの『IVP の実行 (TSO)』および 59 ページの『IVP の実行 (CICS)』を参照してください。ライブラリー QMF810.SDSQSAPn にある NLF IVP (DSQ1nIVP) 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。*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 QMF810 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 QMF810 を変更してください。

```
IMPORT PROC FROM 'QMF810.SDSQSAPn(DSQ1nIVP)'  
RUN PROC
```

ステップ 8 — 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 8A — 既存のサンプル照会およびプロシーチャーの削除』
- 194 ページの『ステップ 8 — 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 8A — 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 を持つ QMF NLF 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 バージョン 8.1 サンプル・ライブラリー QMF810.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。*prefix* は QMF データ・セットの接頭部です。

これらのプロシージャーは、この接頭部が QMF810 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 プロシージ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 QMF810 を変更してください。

```
IMPORT PROC FROM 'QMF810.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、次のステップに進んでください。

ステップ 8 — 各国語サンプル照会およびプロシージ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ジ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 QMF810.SDSQSAPn (DSQ1nSQI) の QMF プロシージャーをインポートおよび実行してください。このプロシージャー (DSQ1nSQI) は、QMF 英語サンプル・ライブラリー (*prefix* .SDSQAPE) から照会をインポートします。 *prefix* は QMF データ・セットの接頭部です。

これらのプロシージャーは、この接頭部が QMF810 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 プロシージ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 QMF810 を変更してください。

```
IMPORT PROC FROM 'QMF810.SDSQSAPn(DSQ1nSQI)'  
RUN PROC
```

ユーザーが Q でない場合は、QMF アプリケーション照会およびプロシージャーのインストール (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・プロシージ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は、エラーを訂正し、以前に作成されたサンプル照会を削除するプロシージャー DSQ1nSQD を実行してください。続いて、プロシージャー DSQ1nSQI を再実行します。

ステップ 9 — 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、63 ページの『オプション: バッチ・モード IVP の実行』を参照してください。DSQQMFE の代わりに各国語プログラム DSQQMFn を使用してバッチ IVP を開始します。このステップでは、QMF バッチ IVP を使用しません。

ステップ 10 — インストール後の終結処置

インストール後の終結処置アクティビティーについては、65 ページの『インストール後の終結処理』を参照してください。

QMF NLF 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 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持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。

重要: QMF バージョン 8.1 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.DSQPLIBn
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 42. QMF NLF 前のリリースから削除されるライブラリー

ステップ 11 — 永続ライブラリーの受け入れ

z/OS 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。

ステップ 12 — CDS 間環境の作成

基本 QMF バージョン 8.1 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録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 QMF810.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 QMF810.SDSQSAPn 内の) に入れられます。

第 15 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。

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

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、プロシージャー、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。

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。

CICS に関する考慮事項

以下の考慮事項は CICS の場合に必要です。

- 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。

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。

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 印刷ファイルを開くことが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時、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。

z/OS での GDDM サービス

以下のサービスは、ネイティブ z/OS バッチ、TSO、ISPF、および CICS に適用されません。

ネイティブ z/OS バッチおよび TSO

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。ニックネームでは、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が定義され、また、ローカル・デバイスとリモート・デバイスの両方が定義可能です。
2. GDDM デフォルト・モジュールの ADMADFT を、ニックネームの指定で更新する。
3. DD 名 ADMDEFS を割り振る。DD 名 ADMDEFS の割り振りについては、208 ページの『ニックネーム・ファイルを割り振る (ネイティブ z/OS バッチ、TSO、および ISPF の場合)』に説明があります。
4. Q.PROFILES 表の該当ユーザー行の PRINTER フィールドを更新する。

CICS

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。

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。ニックネームでは、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が定義され、また、ローカル・デバイスとリモート・デバイスの両方が定義可能です。
2. GDDM デフォルト・モジュールの ADMADFC を、ニックネームの指定で更新する。
3. 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。
4. Q.PROFILES 表の該当ユーザー行の PRINTER フィールドを更新する。

プリンターのための GDDM ニックネームの選択

ここには、GDDM が検索するデータ・セットについての情報があります。

ネイティブ z/OS バッチ、TSO、および ISPF: ネイティブ z/OS バッチおよび TSO 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 ADMDEFS データ・セットを検索し、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が一致するものを探します。

CICS: CICS では、GDDM はデフォルト・モジュールの ADMADTC だけを検索します。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 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示します。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 z/OS 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および ADMOPUT と ADMOPUJ ユーティリティの使用の詳細については、「*GDDM System Customization and Administration*」参照してください。

ニックネーム指定の作成

ここでは、ネイティブ z/OS、TSO、および CICS でニックネームを作成するための説明があります。

ネイティブ z/OS バッチ、TSO、および ISPF: ニックネームを、DD 名 ADMDEFS のデータ・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: 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 43 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 43. 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。
- 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 ADMNICK NAME=MYPRINT,TOFAM=3,DEVTOK=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、204 ページの図 44 と類似した ADMNICK 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,DEVTOK=R87S,TONAME=GRAP
```

図 44. 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

ネイティブ z/OS バッチ、TSO および ISPF: TSO およびネイティブ z/OS バッチでニックネームを作成した後、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。

CICS: 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 (QMF810.SDSQMAPE) が含まれます。

```
//ADMGGMAP DD DSN=QMF810.SDSQMAPE,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 System Customization and Administration*」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。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

この情報を使用して、ネイティブ z/OS バッチおよび TSO 上で、ファミリー 3 GDDM プリンター用のニックネームを定義します。

ネイティブ z/OS バッチ、TSO および ISPF: 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、以下のいずれかと類似した ADMMNICK 指定を使用します。

```
ADMMNICK NAME=370PRINT,TOFAM=3,DEVTOK=R87S,TONAME=370P (CICS)
ADMMNICK NAME=370PRINT,TOFAM=3,DEVTOK=R87S (CMS)
```

図 45. *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 またはネイティブ z/OS バッチでニックネームを作成すると、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: 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、以下のいずれかと類似した *ADMMNICK* 指定を使用します。

```
ADMMNICK NAME=370PRINT,TOFAM=3,DEVTOK=R87S,TONAME=370P (CICS)
ADMMNICK NAME=370PRINT,TOFAM=3,DEVTOK=R87S (CMS)
```

図 46. *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

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 209 ページの図 50 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 プリンター用のニックネーム例 (ネイティブ z/OS バッチ、TSO、または ISPF)

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、以下のいずれかと類似した *ADMMNICK* 指定を使用します。

```
ADMMNICK NAME=3900PRNT,TOFAM=4,DEVTOK=R87S
```

図 47. *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。CSPOOL 処理オプション・セットを所有している場合は、*JES* を介して自動的にファイルを *PSF/OS/390* および *z/OS* にスプールできます。ファミリー 4 印刷の詳細については、「*GDDM System Customization and Administration*」を参照してください。

1 つの定義で複数のニックネームの定義

次に示したように、マルチプリンター・アドレスを定義するために、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。

ユーザーによるオブジェクトの印刷

```
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の定義例については、*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。

- **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,DEV TOK=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,DEV TOK=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,DEV TOK=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,DEV TOK=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、印刷されるページ見出し、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、ページ見出しが抑止されます。

重要: 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 System Customization and Administration*」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 を定義するには、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 デフォルト・モジュールをニックネームで更新

以下の情報を使用して、ネイティブ z/OS バッチ、TSO、および CICS 上で、GDDM デフォルト・モジュールを更新します。

ネイティブ z/OS バッチ、TSO および ISPF: TSO およびネイティブ z/OS 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する。
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・リンク・エディットする。

デフォルト・モジュールについては、*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。

CICS: 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。

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する。
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・リンク・エディットする。

デフォルト・モジュールについては、*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。

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 (ネイティブ z/OS バッチ、TSO、および ISPF の場合)

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・モジュールにアセンブルします。

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。

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教えてください。

ニックネーム・ファイルを割り振る (ネイティブ z/OS バッチ、TSO、および ISPF の場合)

TSO およびネイティブ z/OS 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 にアセンブルされます。

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、以下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 48. 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、以下に示されるような GDDM ニックネーム表を設定します。

```
GDDMPRT ADMNICK TOFAM=3,  FAMILY (SYSTEM PRINTER)           X
          NAME=SYSVRT,    PRINTER NAME (NICKNAME)           X
          DEVTOK=S1403W6, DEVICE TOKEN (1403)              X
          TONAME=SYSP     TONAME MUST MATCH CICS DCT ENTRY
```

図 49. 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 System Customization and Administration で、ニックネームのユーザー・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 50 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 SYSVRT 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 SYSVRT
SYSVRT  DFHDCT TYPE=EXTRA,DESTID=SYSP,DSCNAME=ADMSYSP,RSL=1
```

図 50. TONAME 項目の CICS DCT

以下のように DD 名 ADMSYSP を CICS 始動 JCL に追加する必要があります。

```
//ADMSYSP DD SYSOUT=A
```

209 ページの図 50 に示される TYPE=SDSCI 項目を、DCT 内で他のすべての TYPE=SDSCI 項目の後に追加します。始動 JCL の割り当てステートメントに従い、装置アドレス (SYS097) は、プリンター 04E に対応します。SYSLST を使用する場合は、CICS STATS が QMF 報告書に含まれます。代わりに、代替プリンターを使用してください。

QMF サービスを使用して印刷を処理する

ネイティブ z/OS バッチ、TSO、および CICS で印刷を処理するために、以下の情報を使用してください。

ネイティブ z/OS バッチ、TSO、および ISPF における印刷で 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)
```

CC オプションを YES に設定します。

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。

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。

印刷機能キーの同義語の定義

以下の指示に従って、ネイティブ z/OS バッチ、TSO、および CICS で同義語を定義してください。

ネイティブ z/OS バッチ、TSO および ISPF

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。

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'
```

- 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')
```

- 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を呼び出すために、次の技法があります。

- 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 (QUEUE=QMFREPT,QUEUE=TS)
CICS QMFP (FROM='QMFREPT')
```

- 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')
```

- 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')
```

ユーザーによるオブジェクトの印刷

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。)

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。下の表は、各オブジェクトの要件を要約したものです。

表 45. 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 (z/OS の場合)
図表	可	GDDM ICU は、PRINT コマンドが実行される時必ず制御を得る。	出力は GDDM によって制御される。
書式	可	PRINT コマンドが実行されると、必ず GDDM が制御を得ます。	出力は GDDM によって制御される。
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	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 コマンドが実行されると、必ず GDDM が制御を得ます。	出力は GDDM によって制御される。
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。

第 16 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、z/390 上の CICS、または TSO コマンドに結び付けることを可能にすることにより、QMF コマンドのカスタマイズを支援します。同義語は、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。

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。

ワークステーション・データベース・サーバーのユーザー: インストール後、これらの同義語は Q.COMMAND_SYN_TSO 表に現れます。

z/OS でのデフォルトの同義語

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。

バッチ照会 / プロシージャー

同義語は BATCH です。ユーザーがバッチ・モードで照会またはプロシージャーを実行できるようにします。

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、「DB2 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 コマンドが実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、「DB2 QMF 使用の手引き」および「DB2 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 DPRE、BATCH、LAYOUT、および ISPF は、ISPF の下で QMF が開

始した場合のみ有効です。QMF が ISPF の下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。

ISPF における印刷報告書の表示 (DPRE)

印刷された報告書は、画面上に表示されるものと正確には同じに見えません。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。違いについては、「DB2 QMF 使用の手引き」を参照してください。IBM は、QMF アプリケーション DPRE を提供し、報告書の表示が印刷時と同じに見え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。

TSO の下での DPRE コンポーネントは、データベース内では Q.DSQAER1P という名前のプロシージャーであり、ライブラリー QMF810.SDSQCLTE 内では DSQABR13 という名前の CLIST です。

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、次のコマンドを実行します。

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 コントロール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 TSO 割り振りステートメン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 パネルの変更と暫

定報告書のブラウズを交互に行うことができます。ただし、こ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、明示または暗黙のどちらかで PRINTER=' ' と指定されている 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。

TSO の下では、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 のファンクション・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 QMF810.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ー DSQAER1P 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。

z/OS でのコマンド同義語表の作成

以下のプロシージャーにより、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、220 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。z/OS では、表のための記憶域のコンテナは、表スペースと呼ばれます。下記の例では、z/OS のデフォルトの表スペース名 `TBSPACE1` が使用されています。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。

```
CREATE TABLESPACE DSQTSSN1
  IN DSQDBCTL
  USING STOGROUP DSQSGSYN
  PRIQTY 100
  SECQTY 20
  LOCKSIZE PAGE
  BUFFERPOOL BP0
  CLOSE NO
```

図 51. 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、219 ページの図 52 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 52. 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。

- 図 52 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 という名前の列を持つものとします。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 (z/OS 版) 管理ガイド」を参照してください。

- 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 53 と類似した SQL INSERT ステートメントを使用して、同義語を表へ入力します。表エディターを使用して表を更新することもできます。これは「DB2 QMF 使用の手引き」で説明されています。

```
INSERT INTO COMMAND_SYNONYMS (VERB,OBJECT,SYNONYM_DEFINITION)
VALUES('COMPUTE','MONTHLY_SALES','RUN PROC JONES.SALES_FIGURES')
```

図 53. コマンド同義語定義の作成

これが 226 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 線形手順を実行します。

図 54 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 54. 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 53 および 54 に示す同義語は、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。

verb の選択

どのコマンドの同義語定義にも verb が必要です。オブジェクト名のみがオプションです。

verb は、SYNONYM_DEFINITION 列に保管される QMF RUN コマンド、CICS 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 verb 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。
- verb *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 verb:

無効な verb:

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 verb として基本 QMF verb 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 プロシーチャーは照会を実行し、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。verb - オブジェクトの形の同義語で基本 verb 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 verb を、verb - オブジェクト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オブジェクト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、「DB2 QMF 解説書」の構文図で調べてください。

z/OS の考慮事項

verb は、SYNONYM_DEFINITION 列に保管される QMF RUN コマンド、CICS 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 verb RUN の同義語 COMPUTE を作成することができます。

オブジェクト名の選択

オブジェクト名はコマンド同義語ではオプションです。しかし、オブジェクト名を使用するときは、`verb` とオブジェクト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 ～ 31 文字。
- DB2 表の命名規則に適合しなければならない。
- オブジェクト名がブランク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)。

以下の例は有効なオブジェクトと無効なオブジェクトを示しています。

有効なオブジェクト:

無効なオブジェクト:

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)

『Net Sales』

JONES GROSS (ブランクには二重引用符が必要)

完全修飾表名を使用する場合: 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 オブジェクトは 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 コマンドのオブジェクト名は、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ーです。

z/OS での同義語の選択

SYNONYM_DEFINITION 列には次のものを入れることができます。

- QMF プロシージャー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 線形手順を実行する場合は、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詳しくは、「DB2 QMF アプリケーション開発の手引き」を参照してください。

線形手順を同義語定義で使用: 線形手順は、QMF コマンドを順番に実行する QMF プロシージャです。同義語定義は、いくつかの QMF コマンドの働きをする線形手順を含むことができます。たとえば、図 55 のプロシージャ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 55 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 55 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)
TSO RPTX (FROM=('REPORTX, XYZ'))
```

図 55. 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

このプロシージャを実行する同義語の定義は、図 56 のようになります。

SYNONYM	OBJECT	DEFINITION
VERB		
-----	-----	-----
SHOW	SALES	RUN PROC SALES_PROC

図 56. コマンド同義語を使用して線形手順を実行

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に含まれるその他のオブジェクトは、プロシージャ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。

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オブジェクト（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 57 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。

SYNONYM VERB	OBJECT	DEFINITION
EXECUTE	-	RUN QUERY MYQUERY (&TABLENAME=Q.STAFF)

図 57. 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、QMF はその同義語の右側に入力した情報をすべて変数値として使用します。&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 58 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。

SYNONYM VERB	OBJECT	DEFINITION
SHOW_INFO	-	RUN QUERY STAFFQUERY (&ALL)

図 58. 変数 &ALL をコマンドの同義語定義で使用

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は、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 コマンドの構文の詳細については、「DB2 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 NULL 値で置換します。224 ページの図 58 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 列にキー入力: 同義語定義を同義語表にキー入力するときは、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 verb 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、すべてのオブジェクト名を修飾する。

QMF は、指定されたオブジェクト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、その同義語定義内のオブジェクト名は 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は、「DB2 QMF 解説書」で説明されています。

同義語のアクティブ化

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当てはまります。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については、「DB2 QMF 解説書」を参照してください。

z/OS での同義語のアクティブ化

QMF バージョン 8.1 によって、コマンド同義語表で名前を使用できます。Q.PROFILES 列 SYNONYMS は、VARCHAR(261) になっています。これによって、128 バイトの表所有者 ID および表名が使用できます。

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 59 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='TSO'
      AND ENVIRONMENT='TSO'
```

図 59. 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。

226 ページの図 59 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、INSERT 照会を使用します。

2. 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。

3. 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には、「DB2 UDB (z/OS 版) 管理ガイド」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ふものです。図 60 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。

```
UPDATE Q.PROFILES
  SET SYNONYMS='Q.COMMAND__SYNONYMS'
  WHERE TRANSLATION='ENGLISH' and ENVIRONMENT='TSO'
```

図 60. 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と、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 61 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 61. 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. 229 ページの図 62 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 62. 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 63 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 63. 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

第 17 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。ただし、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。

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。

『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。

表エディター・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

QMF 機能キーのカスタマイズ

キー	実行されるコマンド
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
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

キー	実行されるコマンド
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
List	LIST
Menu	HELP MENU
More Help	MORE HELP
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、「DB2 QMF 解説書」を参照してください。

機能キー表の作成

以下の説明を参考にして、z/OS 上に機能キー表を作成します。

z/OS での表の作成

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. 図 64 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。MY__PFKEYS をユーザー独自の名前で置き換えます。TSO の下では、TBSPACE1 をユーザー独自の表スペースで置き換えます。

```
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 64. 機能キー表の作成

新規の表スペース作成の詳細については、「DB2 UDB (z/OS 版) 管理ガイド」を参照してください。

2. 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について詳しくは、「DB2 UDB (z/OS 版) 管理ガイド」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 という名前の列を持つものとし、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 (z/OS 版) SQL 解説書」を参照してください。

3. 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。

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 表エディターを使用して、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 236 ページの表 46 に示す値を必ず持つようにします。

QMF 機能キーのカスタマイズ

表 46. 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>240 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。241 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い場合は、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。

表 46. 機能キー表をカスタマイズする値 (続き)

列	値	情報
PF_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 47. 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 QMF 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、235 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 47. 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は、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 (235 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。
PF_SETTING	機能キー・ラベルのテキスト	<p>フルスクリーン・パネルでは、QMF はユーザーがこの列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 239 ページの図 65 に示されています。</p> <p>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 239 ページの図 66 および 240 ページの図 67 に示されています。</p>

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で使用してください。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、240 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

239 ページの図 65 に示した SQL 照会を使用して、ホーム・パネル上の F2 を「リスト」から「インポート」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。

```
INSERT INTO MY_PFKEYS (PANEL,ENTRY__TYPE,NUMBER,PF__SETTING)
VALUES('HOME','K',2,'IMPORT')
INSERT INTO MY_PFKEYS (PANEL,ENTRY__TYPE,NUMBER,PF__SETTING)
VALUES('HOME','L',1,'1= ヘルプ 2= インポート 3= 終了 4= 表示 5= 図表 6= 照会')
```

図 65. 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。

コマンド行にコマンドを入力するか、PF キーを使用してください。ヘルプを表示するには、PF1 を押すか、HELP を入力してください。

1= ヘルプ	2= インポート	3= 終了	4= 表示	5= 図表	6= 照会
7= 検索	8= 表編集	9= 書式	10= 手順	11= PROFILE	12= 報告書

了解、カーソルが位置付けられました。
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、PF_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 PF_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。

コマンド行にコマンドを入力するか、PF キーを使用してください。ヘルプを表示するには、PF1 を押すか、HELP を入力してください。

インポート					
7= 検索	8= 表編集	9= 書式	10= 手順	11= PROFILE	12= 報告書

了解、カーソルが位置付けられました。
COMMAND ==>

ウィンドウ・パネルのキーの定義の入力

図 66 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,PF__SETTING)
VALUES('QPTABL','K',3,'CANCEL')
INSERT INTO MY_PFKEYS (PANEL,ENTRY__TYPE,NUMBER,PF__SETTING)
VALUES('QPTABL','L',3,'CancelMe')
```

図 66. 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

図 67 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,PF__SETTING)
VALUES('HEXXXX', 'K', 13, 'CANCEL')

INSERT INTO MY__PFKEYS (PANEL,ENTRY__TYPE,NUMBER,PF__SETTING)
VALUES('HEXXXX', 'L', 13, 'CancelMe')
```

図 67. ヘルプ・パネルまたはプロンプト・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

241 ページの図 68 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 68. 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

パネル ID	表題または説明
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・アクション
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	条件コネクター - 変更内容
QPCDIT	条件コネクター
QPCOCH	列 - 変更内容
QPCODE	列の説明
QPCOFI	列合計機能項目
QPCOFU	列合計機能
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)

パネル ID	表題または説明
QPROCT	行 - 含む
QPROC1	行 - 比較演算子 1
QPROC2	行 - 比較演算子 2
QPROEN	行 - 終了
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	Sort
QPSPEC	指定
QPTABL	表
QPXXXX	PQ ウィンドウ・クラス

新規機能キー定義のアクティブ化

QMF バージョン 8.1 では、機能キー表でロング・ネームを使用できます。

Q.PROFILES 列 PFKEYS は、VARCHAR(261) になっています。これによって、128 バイトの表所有者 ID および表名が使用できます。

z/OS での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ー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、機能キー定義表の名前で更新します。

たとえば、244 ページの図 69 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 = 'TSO')
      AND ENVIRONMENT = 'TSO')
```

図 69. OS/390 で、カスタマイズされた機能キーをユーザーからアクセス可能にする

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、227 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 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、新規の機能キー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` コマンドを実行してグローバル変数 `DSQAP_PFKEY_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は、`Q.PROFILES` の行入力を検査します。

第 18 章 QMF 書式のためのユーザー独自の編集コードの作成

注: この章には、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

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。

DB2 QMF バージョン 8.1 では、新規編集コード 'M' が作成されています。編集コード 'M' は、メタデータが表示され、実際の列データではなく、文字フォーマットで記述子域 (DA) が表示されることを示しています。LOB データ・タイプ CLOB、DBCLOB、BLOB、および定義済みの長さフィールドが、デフォルトで LOB 列に表示されます。実際の LOB データを表示する場合、ユーザーは FORM.MAIN または FORM.COLUMN を変更し、編集コードを 'C' または 'CW' に変更して、文字データを表示することができます。

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、この章の情報を使用して FORM.MAIN および FORM.COLUMN 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。「DB2 QMF 解説書」には、QMF で提供された編集コードが示されています。

また、この章では、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。

QMF では、編集ルーチンが、31 ビットまたは 24 ビットの AMODE または RMODE でサポートされます。ただし、サポートされる言語の一部のバージョンでは 31 ビット・アドレッシングがサポートされません。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。

編集コードの選択

ユーザー編集出口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す。

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 コードを使用してください。

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 48 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 48. 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・タイプ	QMF が変換する長さ
短整数	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

編集コード出口を使用して、日付、時刻、およびタイム・スタンプの値をフォーマットすることもできます。

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 / 時刻の出口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、「DB2 QMF 使用の手引き」を参照してください。これらは DB2 出口であり、QMF 出口ではないことに留意してください。これらの出口の作成方法の詳細については、該当する「DB2 System Administration」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 49 で説明しています。

表 49. 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	yyyy-mm-dd。ここで、 yyyy 年を指定。常に、4 桁の数字。 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。 dd 日を指定。常に、2 桁の数字で、先行ゼロが可能。 ダッシュ (-) は実際のダッシュを意味します。 たとえば、1990-12-12 は、1990 年 12 月 12 日です。
TIME データ	hh.mm.ss。ここで、 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。 mm 分を指定。常に、2 桁の数字で、先行ゼロが可能。 ss 秒を指定。常に、2 桁の数字で、先行ゼロが可能。 ピリオドは実際のピリオドを意味します。 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。

表 49. 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、マイクロ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、252 ページの表 50 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

以下の説明を参考にして、z/OS 上でユーザー出口ルーチンを呼び出します。

z/OS でのユーザー出口ルーチンの呼び出し

図 70 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。

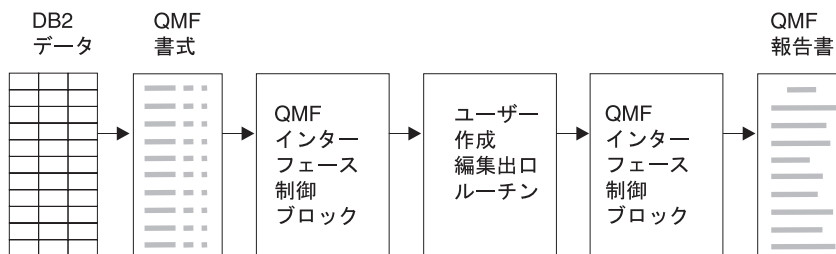


図 70. ユーザー編集ルーチンと QMF (TSO/CICS 版) との関係

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、252 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、サンプル編集出力ルーチンの 6 つの異なるバージョンを QMF810.SDSQSAPE で提供しています。

言語	TSO およびネイティブ z/OS 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。これらのルーチンは、z/OS の QMF810.SDSQSAPE にあります。

QMF では、ユーザー編集ルーチン DSQUEDIT が TSO およびネイティブ z/OS 用に、再入可能モジュール DSQUECIC が CICS 用に提供され、それらは QMF ライブラリー QMF810.SDSQLOAD にあります。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。

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、あるいは見出し、脚注、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。ECSRSLN 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。

ユーザー独自の編集コードの作成

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用しないでください。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、表 50 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 50 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 50. 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
ECSDECPT	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。

表 50. 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
ECSERRET	<p>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、これをゼロ以外の戻りコードに設定する。 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。</p> <p>番号 エラー</p> <p>99101 認識できない編集コード</p> <p>99102 編集コードに不適切な入力データ・タイプ</p> <p>99103 フォーマット設定される項目に無効な入力値</p> <p>99104 フォーマット設定される項目が短過ぎる</p> <p>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)</p> <p>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。</p>
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。
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は、V タイプ・コードのみに適用される。

表 50. 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
ECSINTYP	<p>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ます。</p> <p>384 DATE データ・タイプ</p> <p>388 TIME データ・タイプ</p> <p>392 TIMESTAMP データ・タイプ</p> <p>448 VARCHAR データ・タイプ</p> <p>452 CHAR データ・タイプ</p> <p>456 LONG VARCHAR データ・タイプ</p> <p>464 VARGRAPHIC データ・タイプ</p> <p>468 GRAPHIC データ・タイプ</p> <p>472 LONG VARGRAPHIC データ・タイプ</p> <p>480 FLOAT データ・タイプ</p> <p>484 DECIMAL データ・タイプ</p> <p>496 INTEGER データ・タイプ</p> <p>500 SMALLINT データ・タイプ</p> <p>940 拡張浮動小数点データ・タイプ</p> <p>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。</p>
ECSNAME	<p>制御ブロックの名前である DXEPCS 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。</p>
ECSRQMF	<p>T に設定されると、終了呼び出しが要求される。</p>
ECSRSLEN	<p>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)</p>
ECSTHSEP	<p>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)。</p>
ECSUSERS	<p>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。</p>

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、249 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呼び出すというものです。このような設計にしておく、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、ユーザーの編集出口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。どのように表現されるかは、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 (ECSINTYP フィールドによって決まる)、または、編集コードが U コードであるか V コードであるか (ECSECODE フィールドによって決まる) によります。

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。出力域は一時記憶域であり、32,767 行以下の出力を保持できます。

QMF 終了時に制御を出口ルーチンに渡す

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口ルーチンが必ず制御を受け取るように指示します。編集出口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口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 (ハイレベル・アセンブラー) で作成

ネイティブ z/OS、TSO、および CICS 用の編集ルーチンをアセンブラーで作成することができます。

ネイティブ z/OS、TSO、または ISPF 用の編集ルーチンを作成

アセンブラー用の 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。

257 ページの図 71 は、ネイティブ z/OS、TSO、または ISPF 用のアセンブラー編集出口ルーチンのプログラム構造を示しています。

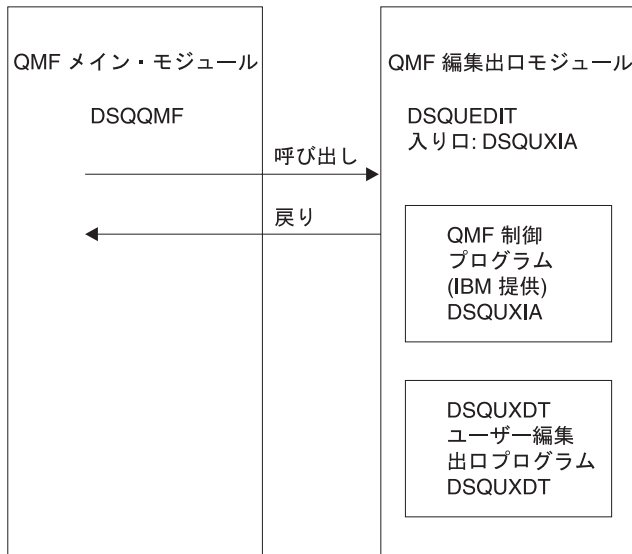


図 71. TSO およびネイティブ z/OS 用のアセンブラー編集出口ルーチンのプログラム構造

プログラム例 DSQUXDTA

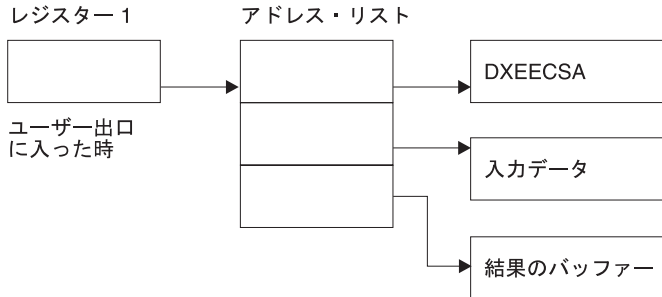
IBM 提供のアセンブラー用サンプル編集プログラム DSQUXDTA は、z/OS の QMF810.SDSQSAPE ライブラリーにあります。このサンプル・プログラムには、ユーザーが要件に合わせて変更できるように、コメントが付けられています。このプログラム例を使用する場合は、ユーザーのプログラム・ライブラリーにコピーし、名前を DSQUXDT に変更してください。このファイルの最下部付近に、DXEECSA 用の COPY ステートメントがあります。DXEECSA は z/OS の DSQUSERE MACLIB のメンバーです。DXEECSA により入力フィールドが定義され、この章で使用する名前がそこに示されています。

アセンブラー編集ルーチンがネイティブ z/OS と対話する方法

ユーザー編集プログラムは、TSO およびネイティブ z/OS では、標準のアセンブラー 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。リンケージは、標準の IBM 呼び出し規則に従います。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。

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 用のアセンブラー DSECT が DXEECSA として QMF に組み込まれており、z/OS ではライブラリー QMF810.SDSQUSRE の中にあります。アセンブラーの 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。

プログラム例では、次のように、ステートメントによってアドレスがレジスター 8、9、および 10 に設定されます。

```
ECSPTR EQU R10
        L      ECSPTR,0(R1)
        USING  DXEECS,ECSPTR
ECSINPT EQU R9
        L      ECSINPT,4(R1)
        USING  ECSINPT,ECSINPT
ECSRSLT EQU R8
        L      ECSRSLT,8(R1)
        USING  ECSRSLT,ECSRSLT
```

USING ステートメントは、DXEECSA で定義された DSECT を参照します。これらは、3 つのパラメーターおよびその入力フィールドのコンポーネントを定義します。

この結果、レジスター 10、9、および 8 はそれぞれ、制御ブロック、フォーマットされる値、およびフォーマット後の結果のために確保された記憶域を指します。

標準的な規則を使用して、各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。

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 DSQUXDT 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。この制御ブロックは、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。

z/OS でのユーザー・プログラムのアセンブルおよびリンク・エディット

アセンブル中は、TSO またはネイティブ z/OS の QMF サンプル・ライブラリー QMF810.SDSQUSRE 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。

QMF モジュール・ライブラリー QMF810.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。

z/OS での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

以下のステートメントの例は、TSO またはネイティブ z/OS 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す。

```
//sampasm    JOB
//STEP1     EXEC PROC=ASMHCL
//* Provide Access to QMF Edit Macro DXEECSA
//C.SYSLIB  DD DSN=QMF810.SDSQUSRE,DISP=SHR
//C.SYSIN   DD *
.
        Your program or copy of QMF sample DSQUXDTA
.
/*
//* Provide Access to QMF Interface Module
//L.QMFLOAD DD DSN=QMF810.SDSQLOAD,DISP=SHR
//L.SYSIN   DD *
        INCLUDE QMFLOAD(DSQUXIA)
        ENTRY DSQUXIA
        MODE  AMODE(31) RMODE(ANY)
        NAME  DSQUEDIT(R)
/*
```

編集ルーチンを CICS 用にアセンブラーで作成

CICS におけるアセンブラー用の 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。

ユーザー独自の編集コードの作成

- CICS プロログ・マクロおよびエピログ・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 DFHEAI0 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。

図 72 は、CICS のアセンブラー編集出口ルーチンのプログラム構造を示しています。

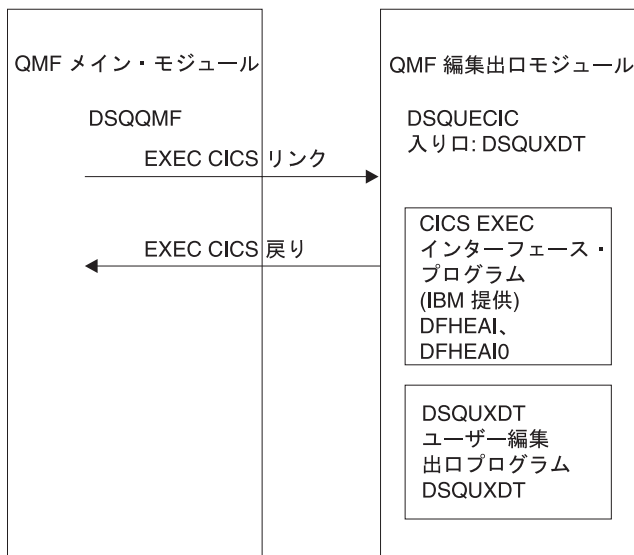
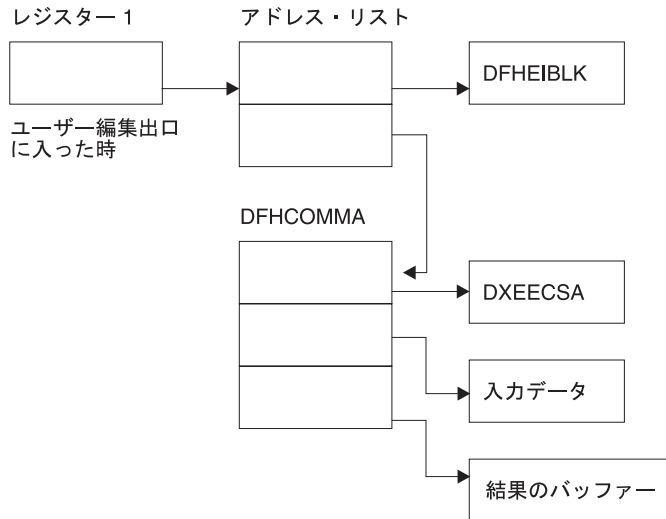


図 72. CICS のアセンブラー編集出口ルーチンのプログラム構造

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。



- 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 に組み込まれており、ライブラリー QMF810.SDSQSAPE の中にあります。アセンブラーの 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、アドレス可能性を設定する標準の 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、さらに標準の CICS エピローグ (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 QMF810.SDSQSUSRE 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共に、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 JCL ステートメント例 (z/OS 上の 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='QMF810.SDSQLOAD'
//TRN.SYSIN DD *
        .
        Your program or modified copy of QMF sample DSQUXCTA
        .
/*
//* Provide access to QMF Edit Macro DXEECSA
//ASM.SYSLIB DD DSN=QMF810.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

DSQUXCTA という名前の、アセンブラーで書かれた IBM 提供の編集プログラムの例が、z/OS の QMF サンプル・ライブラリー QMF810.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、オンラインで参照することも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。

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

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は、QMF により変更されることはありません。

DXEECS 制御ブロックのサンプル・アセンブラー・バージョンとして、QMF によって提供されている DXEECSA ファイルを参照してください。このファイルは、z/OS のライブラリー QMF810.SDSQSAPE にあります。

言語環境プログラム (LE) を使用せずに PL/I で編集ルーチンを作成

言語環境を使用せずにネイティブ z/OS または TSO 用に PL/I で編集ルーチンを作成することができます。

LE を使用せずにネイティブ z/OS、TSO、または ISPF 用の編集ルーチンを作成

TSO、ISPF、およびネイティブ z/OS 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。DXEECS 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。
- 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。

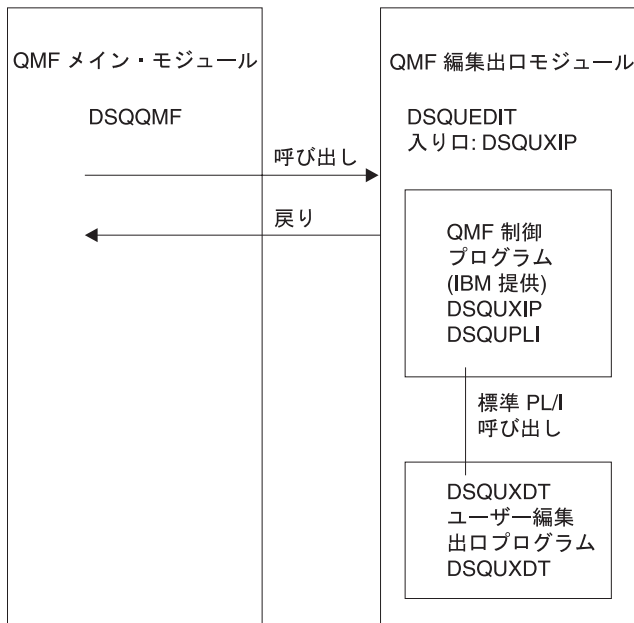


図 73. LE を使用しない PL/I 編集出口ルーチンのプログラム構造

PL/I 編集ルーチンの、ネイティブ z/OS、TSO、または ISPF 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。

```
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 QMF810.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、z/OS の QMF サンプル・ライブラリー QMF810.SDSQSURE 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。

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 QMF810.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 モジュールを入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

以下に示すのは、TSO または z/OS 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 DXEECSF
//PLI.SYSLIB   DD DSN=QMF810.SDSQSURE,DISP=SHR
//PLI.SYSIN    DD *
                .
                Your program or copy of QMF sample DSQUXDTP
                .
/*
/* Provide Access to QMF Interface Module
//LKED.QMFLOAD DD DSN=QM720.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

DSQUXDTP という名前の、PL/I で書かれた 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 QMF810.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、オンラインで参照することも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 PL/I で編集ルーチンを作成

以下の説明を参考にして、ネイティブ z/OS または TSO 用の編集ルーチンを言語環境を使用して作成します。

言語環境プログラム (LE) を使用して、ネイティブ z/OS、TSO、または ISPF での編集ルーチンを PL/I で作成

LE を使用した TSO、ISPF、またはネイティブ z/OS 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。DXEECSF 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。

267 ページの図 74 は、TSO、ISPF、またはネイティブ z/OS 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。

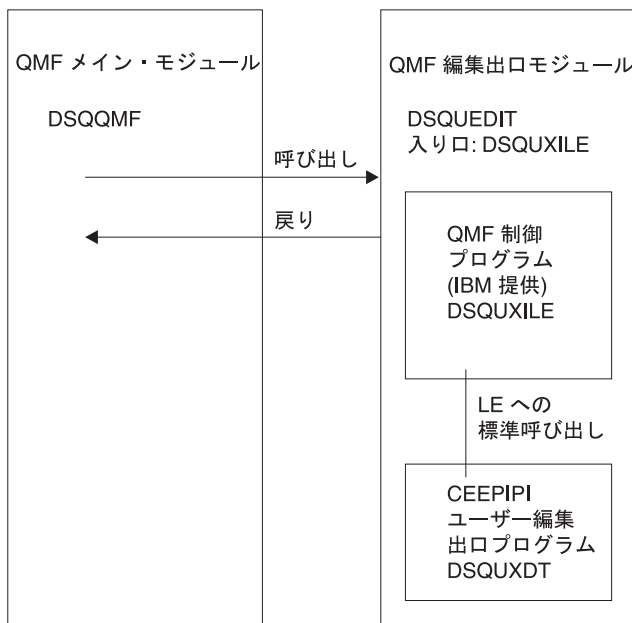


図 74. LE を使用した PL/I 編集出口ルーチンのプログラム構造

PL/I 編集ルーチンが LE を使用したネイティブ z/OS、TSO、または ISPF 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・ステートメントの例です。

```
DSQUXDT:
  PROCEDURE(DXEECSF,ECSINPTF,ECSRSLTF) OPTIONS(REENTRANT);
```

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 QMF810.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、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;
```

ユーザー独自の編集コードの作成

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 QMF810.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

以下に示すのは、TSO または z/OS 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 PLIXCL
//* Provide Access to QMF Edit Macro DXEECS
//PLI.SYSLIB  DD DSN=QMF810.SDSQSRE,DISP=SHR
//PLI.SYSIN   DD *
                .
                Your program or copy of QMF sample DSQUXDTP
                .
/*
/** Provide Access to QMF & LE Interface Module
//LKED.QMFLOAD DD DSN=QMF810.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

DSQUXDTP という名前の、PL/I で書かれた 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 QMF810.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、オンラインで参照することも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。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 z/OS 上の 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。DFHPL10I 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。

図 75 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。

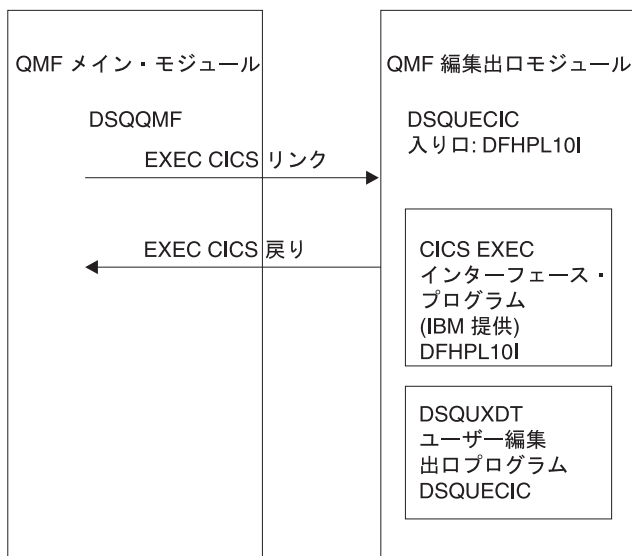


図 75. CICS の PL/I 編集出口ルーチンのプログラム構造

プログラム例 DSQUXCTP

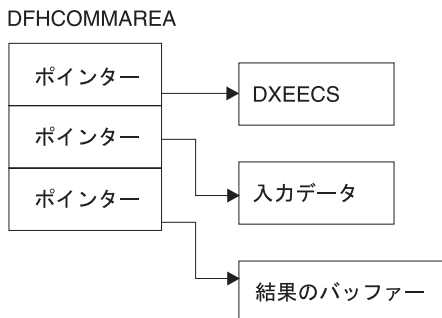
DSQUXCTP という名前の、アセンブラで書かれた IBM 提供の編集プログラムの例が、QMF サンプル・ライブラリー QMF810.SDSQSAPE にあります。このプログラムにはコメントが数多く付いており、オンラインで参照することも印刷することも可能です。また、ユーザーの要件に合わせて変更することもできます。PL/I データ構造は、QMF と共に DXEECSP として出荷され、ライブラリー QMF810.SDSQUSRE の中にあります。この構造をユーザーのプログラムに組み込みます。

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。

図 76. DFHCOMMAREA



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パラメーターとして提供します。

```
DSQUECIC:
    PROCEDURE(DFHCOMM) 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 ARE DESCRIPTION OF EDIT EXIT PARAMETERS */
/*****
DECLARE
    DFHCOMM PTR;
DECLARE
    1 DFHCOMM BASED(DFHCOMM),
      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。

```

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 DXEECSP として出荷され、ライブラリー QMF810.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。

z/OS での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 QMF810.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。

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 とユーザーの編集出口プログラム DSQUXCTP を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。また、DFHPL1OI または DFHPL1I が編集出口モジュールの最初のモジュールになるようにしてください。

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 JCL ステートメント例 (z/OS 上の 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='QMF810.SDSQLOAD'
//TRN.SYSIN DD *
        .
        Your program or modified copy of QMF sample DSQUXCTP
        .
/*
/** Provide access to QMF Edit Macro DXEECSP
//PLI.SYSLIB DD DSN=QMF810.SDSQSRE,DISP=SHR
//LKED.SYSIN DD *
        REPLACE PLISTART
        INCLUDE CICSLOAD(DFHPL10I)
        REPLACE PLISTART
        ORDER DFHPL10I
        ENTRY DFHPL10I
        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。

言語環境プログラム (LE) を使用せずに COBOL で編集ルーチンを作成

ネイティブ z/OS または TSO 用に COBOL で編集ルーチンを作成することができます。

この節では、特に断りのない限り、COBOL とは VS COBOL II、COBOL/370、および COBOL (z/OS 版) を指します。

言語環境プログラム (LE) を使用せずに、ネイティブ z/OS、TSO、または ISPF での編集ルーチンを COBOL で作成

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。DXEECSC 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 77 は、COBOL 編集出口ルーチンのプログラム構造を示します。

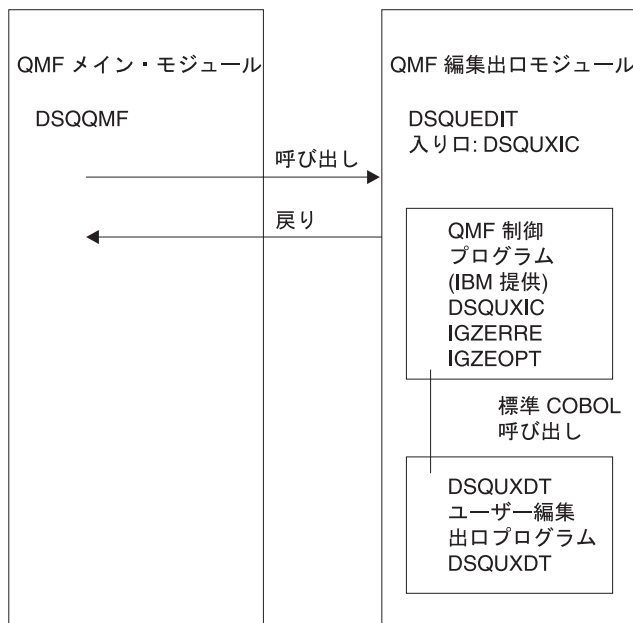


図 77. COBOL 編集出口ルーチンのプログラム構造

プログラム例 DSQUXDTC

DSQUXDTC という名前の、COBOL で書かれた IBM 提供の編集出口プログラムの例が、z/OS 上の QMF サンプル・ライブラリー QMF810.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、オンラインで参照することも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。

COBOL 編集ルーチンの作動方法

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。

ユーザー独自の編集コードの作成

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、z/OS の QMF サンプル・ライブラリー QMF810.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 (z/OS 版)

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 QMF810.SDSQLOAD ライブラリーに入れます。

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 は、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。

z/OS での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 QMF810.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 モジュール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。

z/OS での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

以下に示すのは、TSO またはネイティブ z/OS 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,NLOAD'
//SYSPRINT 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))
//SYSPUNCH 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=QMF810.SDSQUSRE,DISP=SHR
//COB2.SYSIN  DD *
        .
        Your program or copy of QMF sample DSQUXDTC
        .
/*
/* Provide Access to QMF Interface Module
//LKED.QMFLOAD DD DSN=QMF810.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)
```

ユーザー独自の編集コードの作成

```
ENTRY DSQUXIC
MODE AMODE(31) RMODE(ANY)
NAME DSQUEDIT(R)

/*

COBOL/370 または IBM COBOL (z/OS 用) 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))
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYPUNCH 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=QMF810.SDSQUSRE,DISP=SHR
//COBOL.SYSIN DD *

        Your program or copy of QMF sample DSQUXDTC
        .
/*
/* Provide Access to QMF Interface Module
//LKED.QMFLOAD DD DSN=QMF810.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)

/*
```

言語環境プログラム (LE) を使用して COBOL で編集ルーチンを作成

以下の説明を参考にして、ネイティブ z/OS および TSO 用に、言語環境プログラムを使用して COBOL で編集ルーチンを作成します。

言語環境プログラム (LE) を使用して、ネイティブ z/OS、ISPF、および TSO で編集ルーチンを COBOL で作成

ネイティブ z/OS および TSO 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。DXEECSC 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 DSQUXDT。
- LE 事前初期化サービス・プログラム。名前は CEEPIPI。

図 78 は、ネイティブ z/OS および TSO 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。

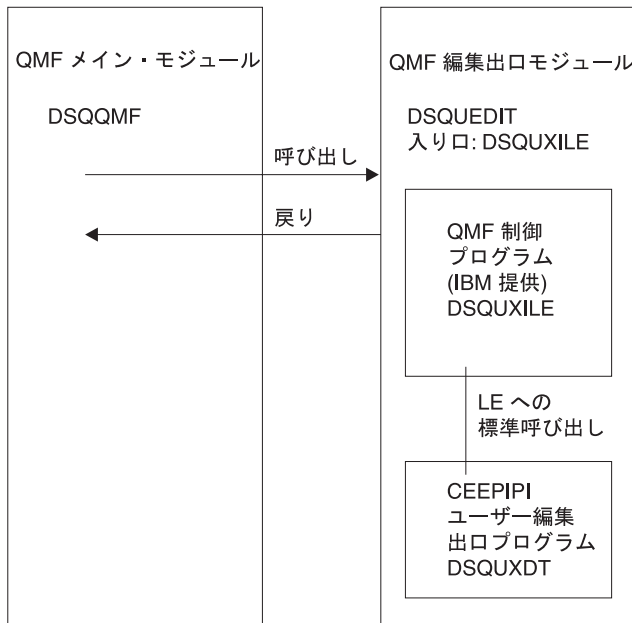


図 78. LE を使用した TSO、ISPF、またはネイティブ z/OS での COBOL 編集出口ルーチンのプログラム構造

QMF で配布されている編集制御ブロック DXEECSC 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、DXEECSC 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 DSQUXDTC

DSQUXDTC という名前の、COBOL で書かれた IBM 提供の編集出口プログラムの例が、z/OS 上の QMF サンプル・ライブラリー QMF810.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、オンラインで参照することも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。

COBOL 編集ルーチンが LE 内でネイティブ z/OS、TSO、または ISPF 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・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 DXEECSC として出荷され、ライブラリー QMF810.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。

DSQUXDT のコンパイル

コンパイル中は、z/OS の QMF サンプル・ライブラリー QMF810.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。

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、および NODYNAM です。

プログラムのリンク・エディット

z/OS 上の QMF モジュール・ライブラリー QMF810.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。

z/OS での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

以下に示すのは、TSO またはネイティブ z/OS 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。

```
//samCOBOL JOB
//STEP1 EXEC PROC=IGYWCL
//* Provide Access to QMF Edit Macro DXEECS
//COBOL.SYSLIB DD DSN=QMF810.SDSQSURE,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=QMF810.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)
/*
```

編集ルーチンを z/OS 上の 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。DXEECS 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 DSQUECIC。

280 ページの図 79 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。

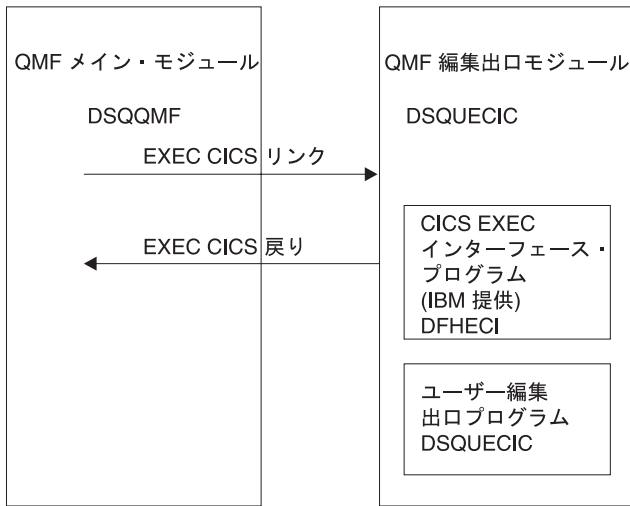
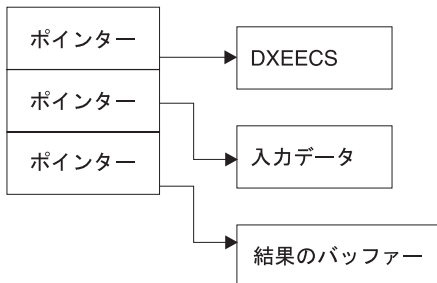


図 79. CICS の COBOL 編集出口ルーチンのプログラム構造

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。

DFHCOMMAREA



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 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。

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、z/OS のライブラリー QMF810.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 QMF810.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・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。

ユーザー独自の編集コードの作成

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 DXEEESC を配布します。IBM より配布される DXEEESC 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 JCL ステートメント例 (z/OS 上の 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='QMF810.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 DXEEESC
//COB.SYSLIB DD DSN=QMF810.SDSQUSRE,DISP=SHR
//LKED.SYSIN DD *
        INCLUDE SYSLIB(DFHECI)
        ORDER DFHECI
        ENTRY DSQUECIC
        MODE AMODE(31) RMODE(ANY)
        NAME DSQUECIC(R)
/*
```

z/OS での 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。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。

プログラム例 DSQUCTC

DSQUXCTC という名前の、COBOL で書かれた IBM 提供の編集プログラムの例が、z/OS の QMF サンプル・ライブラリー QMF810.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、オンラインで参照することも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。

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は、QMF により変更されることはありません。

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) で表示できます。このタイプのデータを処理するルーチンを編集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す。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

ユーザー独自の編集コードの作成

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 文字の文字列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持つことができます。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Stringには DBCS データはありません。たとえば、ECSINPT に次のようなStringが入っています。

```
ccccSodededededededededeSiccSodededededeSi
```

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。

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 文字の文字列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 ECSRSLN フィールドにあります。結果が入りきらない場合は、右側を切り捨てます。切り捨てられる結果で表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。

ユーザー独自の編集コードの作成

第 19 章 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

注: この章には、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。

管理プログラム出力ルーチンにより、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 でデフォルトの限界値が設定された管理プログラム出力ルーチンを提供しています。たとえば、ユーザーがデータベースから検索できる行数を制御できます。ユーザーは、このデフォルトの出力ルーチンの使用、出力ルーチンの変更、またはアセンブラを使用したユーザー独自の出力ルーチンの作成を行うことができます。

z/OS での管理プログラム出力ルーチンの使用

z/OS では、QMF コマンドの実行に費やされる時間のデフォルトの限界値が設定されています。

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 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、「DB2 QMF High Performance Option User's Guide for TSO/CICS」を参照してください。

IBM 提供の管理プログラム出力ルーチンの使用

CICS 用に提供される管理プログラム出力ルーチン (DSQUEGV3) は、データベースから検索できる行数を制御します。TSO、ISPF、およびネイティブ z/OS 用に提供される管理プログラム出力ルーチン (DSQUEGV1) は、データベースから検索できる行数、または QMF コマンドの実行に使用されるプロセッサ時間を制御します。管理プログラム出力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。

管理プログラム出口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、次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 25,000 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 80. 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、ix 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 80 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、次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 81. 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 81 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は、次に示すようなメッセージを受け取ります。

```
Procedure canceled.
```

図 82. 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

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。

TSO、ISPF、およびネイティブ z/OS 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プロンプト値) 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、次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 83. リソース限界に近づいたときに表示されるメッセージ (z/OS)： 図の中のシンボル n は、ix 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。

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 アクティビティを取り消すように管理プログラム出口ルーチンをセットアップするには、以下の手順を行ってください。

1. SQL 照会パネルから、図 84 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 84. 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、290 ページの図 85 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。

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 85. 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。

3. ユーザーに、データベースに再接続して新しい値でアクティブ化するように指示してください。これは、DB2 CONNECT コマンドを使用して行うか、または、ユーザーが現行 QMF セッションを終了してから新規のリソース・グループをアクティブ化して別のセッションを開始することにより可能です。

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は、『管理プログラム出口ルーチンによるリソースの制御方法』を参照してください。その後、293 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による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・コントロール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。

表 51 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、3 つの事前定義された CICS のリソース・オプションを持っています。このグループには、TSO、ISPF、またはネイティブ z/OS バッチ用に、追加の時間オプションがあります。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。

表 51. 全体に共通な 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

GROUP	OPTION	INTVAL	FLOATVAL	CHARVAL
SYSTEM	SCOPE	-	-	Indicate whether governor is active
SYSTEM	ROWLIMIT	100,000	-	Cancel after fetching 100,000 rows
SYSTEM	ROWPROMPT	25,000	-	Prompt user after fetching 25,000 rows

表 52. TSO、ISPF、またはネイティブ z/OS バッチ用の IBM 提供の管理プログラム出口のオプション

GROUP	OPTION	INTVAL	FLOATVAL	CHARVAL
SYSTEM	TIMELIMIT	1440	-	Cancel after 24 minutes CPU
SYSTEM	TIMEPROMPT	360	-	Prompt user after 6 minutes CPU
SYSTEM	TIMECHECK	900	-	15 minute interval between time check

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 = 100,000

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ーを取り消します (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ます。

ROWPROMPT = 25,000

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 z/OS バッチの場合、以下の 3 種類のオプションが追加されています。

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 SRB (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 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ちます。

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・コントロール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 86 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。

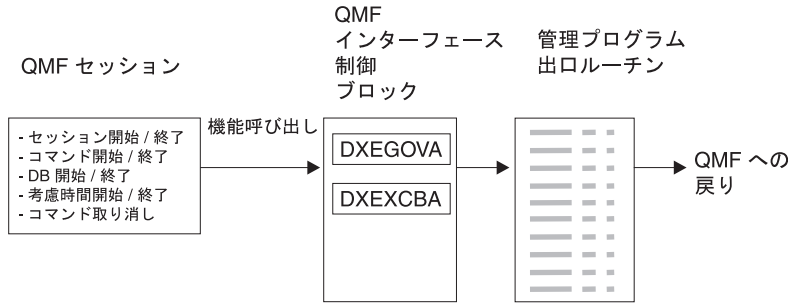


図 86. 管理プログラム出口ルーチンと QMF (TSO/CICS 版) の関係

QMF は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、302 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ます。

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 288 ページの図 81 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せん。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タをリセットし、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、298 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・コントロール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、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、TSO およびネイティブ z/OS バッチの場合、管理プログラムは、プロセッサ時間が 300 秒経過したときに GROUP1 のユーザーにプロンプトを出し、プロセッサ時間が 1,000 秒経過したときにユーザーのアクティビティーを取り消します。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 87 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、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 87. 行限界のプロンプトのアクティブ化

2. 図 88 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 88. 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 89 を照会するプログラムを実行して、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 (295 ページ) 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 89. 時間限界のプロンプトのアクティブ化

4. 図 90 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 90. TSO およびネイティブ OS/390 および z/OS バッチの場合: 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 91 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 91. TSO およびネイティブ z/OS バッチの場合: 時間間隔検査のアクティブ化

6. 図 92 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。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 を持つ必要があります。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。

```
INSERT INTO Q.RESOURCE__VIEW (RESOURCE__GROUP,RESOURCE__OPTION,INTVAL)
VALUES('GROUP1','SCOPE',0)
```

図 92. 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

7. 図 93 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 93. 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、ix 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。これは、DB2 CONNECT コマンドを使用して行うか、または、ユーザーが現行 QMF セッションを終了してから新規の値をアクティブ化して別のセッションを開始することにより可能です。

ユーザー独自のリソース・コントロール表の作成

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 297 ページの表 53 で示されたすべての列が含まれている場合は、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。

図 94 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 を削除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、図 94 および 297 ページの図 95 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。TSO の下では、SPACE1 をユーザー独自の表スペース名で置き換えます。

```
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 94. リソース・コントロール表の作成または Q.RESOURCE_TABLE の名前変更

QMF (TSO/CICS 版) を実行する場合は、視点を削除すると、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削除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 (z/OS 版) SQL 解説書」に記載されています。

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。

図 95 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 95. Q.RESOURCE_VIEW の再定義

視点を作成した後、Q.RESOURCE_VIEW に関する SELECT 特権を PUBLIC に与える必要があります。その後、新規の視点をテストします。SPUFI を使用して視点をテストできます。最後に、QMF 許可プランを再バインドします。

表 53. 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 のユーザー一行の 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	整数値を持つリソース・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。

表 53. Q.RESOURCE_TABLE 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
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。 302 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。 310 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。
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。 324 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。 325 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。

z/OS での管理プログラム出口の変更

TSO の場合、およびネイティブの z/OS 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SEMBルし、リンク・エディットします。

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。329 ページの『z/OS 上の CICS での管理プログラム出口ルーチンのアセンブル、変換、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 54 は、これらのコンポーネントを示しています。名前はインストールした言語（英語または NLF）によって異なります。下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（ix 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 z/OS バッチを表しています。

表 54. 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、ISPF、およびネイティブ z/OS		
DSQUnGV1	QMF810.SDSQLOAD	TSO およびネイティブ z/OS バッチ用のロード・モジュール
DSQUnGV1	QMF810.SDSQUSRn	TSO 用、およびネイティブ z/OS バッチ用の管理プログラム出口ルーチンのソース・コード
DXEUnGV1	QMF810.SDSQUSRn	TSO およびネイティブ z/OS 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を含む
z/OS での CICS		
DSQUnGV3	QMF810.SDSQLOAD	CICS 用のロード・モジュール
DSQUnGV3	QMF810.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	QMF810.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	QMF810.SDSQUSRn	CICS での管理プログラムのプロンプトの BMS マップ
DXEGOVA	QMF810.SDSQUSRn	DXEGOVA 制御ブロックの DSECT
DXEXCBA	QMF810.SDSQUSRn	DXEXCBA 制御ブロックの DSECT

NLF を使用する場合: 各言語環境用のモジュール DSQUnGVx 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 z/OS 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 z/OS 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・コントロール表 (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、およびネイティブ z/OS と管理プログラム出力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 コマンドを実行して、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力ルーチン間では、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。管理プログラム出力ルーチンは、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力ルーチンが呼び出される可能性があるからです。

この章においては、ロード・モジュール・ライブラリー QMF810.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。

図 96 は、管理プログラム出力ルーチンのプログラム構造を示しています。

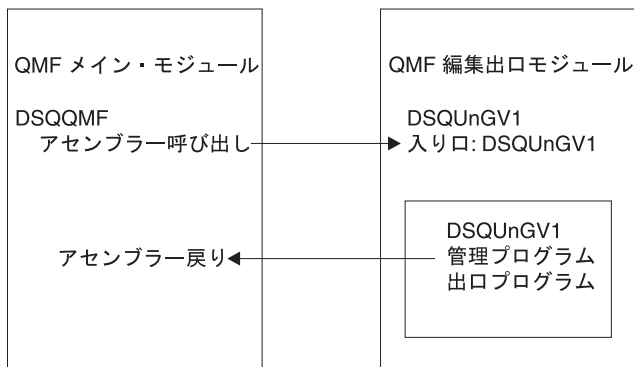


図 96. QMF と管理プログラム出力ルーチンを対話させる TSO またはネイティブ z/OS の処理

CICS の管理プログラム出力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 コマンドを実行して、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、329 ページの『z/OS 上の CICS での管理プログラム出口ルーチンのアセンブル、変換、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。

- 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 提供のプロログおよびエピログ・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。

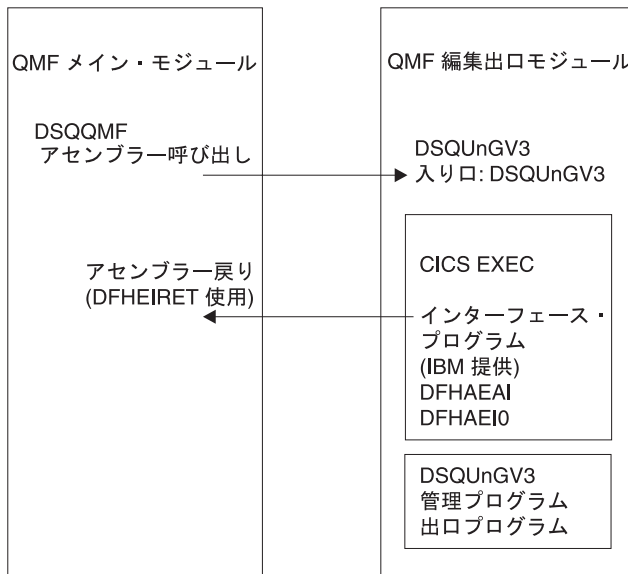


図 97. QMF と管理プログラム出口ルーチンを対話させる CICS 処理

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、DSQU n GV3 です。QMF は、管理プログラム出口ルーチン呼び出すとき、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、329 ページの『z/OS 上の CICS での管理プログラム出口ルーチンのアセンブル、変換、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。

z/OS

z/OS の場合、以下の説明に従ってください。

QMF が管理プログラム呼び出す時点

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 DSQU n GV x 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。

• 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・コントロール表が再度、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り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、365 ページの『パフォーマンス上の問題の解決』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
 - 表エディターの 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
 - 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ずスクロール・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
- 異常終了の開始時

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 GOVFUNCT 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。機能呼び出しの各タイプには、GOVFUNCT フィールドの特別な値があります。これらの値は、305 ページの図 98 に示されています。

管理プログラム出口ルーチンに入ると起きること

QMF は、入り口点 DSQU n GV1 (TSO) または DSQU n GV3 (CICS) 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 DSQU n GV3 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。305 ページの図 98 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。

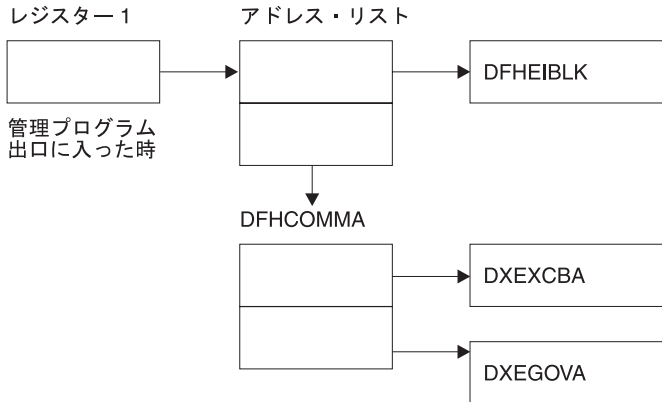


図 98. 管理プログラム出口ルーチンの呼び出し時のレジスター 1 の内容

- 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。

次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。

```

DSQUEGV3 TITLE 'QMF GOVERNOR EXIT ROUTINE'
DFHEISTG DSECT
DSQUEGV3 DFHEIENT CODEREG=(12),DATAREG=(13),EIBREG=(10)
          B      FIDENTRY          BRANCH AROUND CONSTANTS
*
MODNAME  DC    C'DSQUEGV3'          MODULE NAME
          DC    C' '
          DC    C'&SYSDATE '        DATE OF ASSEMBLY
          DC    C'&SYSTIME '        TIME OF ASSEMBLY
          DS    0H
*
FIDENTRY DS    0H
          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 99. 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 99 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 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。

重要: 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 セッションを終了します。

管理プログラムは、307 ページの図 100 と類似したコードで終了します。

```

:
:
*
XR   R15,R15           ZERO RETURN CODE
DFHEIRET RCREG=15
*
    
```

図 100. 管理プログラムの終了コード

入り口点へのブランチ: QMF は、入り口点 DSQUEGV1 (TSO) 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呼び出しします。管理プログラム出口ルーチンへ入るには、以下のようにします。

- レジスター 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。

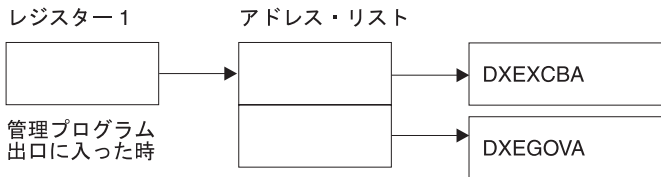


図 101. 管理プログラム出口ルーチンの呼び出し時のレジスター 1 の内容

- レジスター 13 には、QMF の SAVE 域のアドレスが入っています。
- レジスター 14 には、呼び出しからの戻りアドレスが入っています。
- レジスター 15 には、入り口点のアドレスが入っています。これは、DSQUEGV1 です。

管理プログラムが呼び出された後、308 ページの図 102 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、TSO またはネイティブ z/OS 用の IBM 提供の管理プログラムからのものです。

```

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 102. 管理プログラムの開始時点でのサンプル・コード (TSO、ISPF、またはネイティブ z/OS の場合)

図 102 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ジャンプします。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。


```

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 (TSO) または DSQU n GV3 (CICS) 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 GOVFNCT 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 GOVFNCT 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、GOVFNCT 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 103 に示します。CICS で実行している場合は、GOVABEND は呼び出されません。

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 103. DXEGOVA の GOVFNCT フィールドの文字値と数値

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、GOVFNCT 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。

310 ページの図 104 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 104. 機能呼び出しのタイプの識別と該当するアドレスへのジャンプ

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は、次の節 293 ページの『ユーザー独自のリソース限界の定義』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。

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、Q.RESOURCE_VIEW と呼ばれるリソース制御の視点にあります。

311 ページの表 55 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、稼働中のオペレーティング・システムの QMF には適用されない場合があります。

表 55. 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。
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"> • GOVNIT (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)
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。
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・コントロール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。
GOVRESC	10XL128	リソース・コントロール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・コントロール表のユーザー・リソース・グループの 1 つの行に関する情報を含む。 <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。

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. 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	<p>リソース・コントロール表のユーザーの 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。</p> <p>GOVOPTN(CL16) リソース・コントロール表の 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。</p> <p>GOVNULLI(H) INTVAL 列のヌル標識</p> <p>GOVINTVL(F) INTVAL 列の値</p> <p>GOVNULLF(H) FLOATVAL 列のヌル標識</p> <p>GOVFLOAT(D) FLOATVAL 列の値</p> <p>GOVNULLC(H) CHARVAL 列のヌル標識</p> <p>GOVCHLEN(H) CHARVAL 列のデータの長さ</p> <p>GOVCHAR(CL80) CHARVAL 列の値</p> <p>GOVNEXTR(A) 次のリソース表の行のデータ・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。</p> <p>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。</p>
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 照会からの戻りコード。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。

```

***** 00001000
*
* CONTROL BLOCK NAME: DXEGOVA * 00002000
*
* FUNCTION: * 00003000
* * 00004000
* * 00005000
* * 00006000
* THIS IS THE INTERFACE CONTROL BLOCK BETWEEN QMF AND * 00007000
* THE GOVERNOR EXIT ROUTINE. * 00008000
* * 00009000
* STATUS: VERSION 8 RELEASE 1 LEVEL 0 * 00010000
* * 00011000
* INNER CONTROL BLOCKS: NONE * 00012000
* * 00013000
* CHANGE ACTIVITY: NA * 00014000
* * 00015000
* CHANGE DATE: NA * 00016000
* * 00017000
***** 00018000
*
DXEGOVA DSECT 00019000
DS 0D 00020000
GOVNAME DS CL8 -- CONTROL BLOCK IDENTIFICATION 00021000
SPACE 00022000
GOVEXCTL DS XL72 -- EXIT CONTROL 00023000
ORG GOVEXCTL 00024000
GOVFUNCT DS XL1 ----- FUNCTION CODE 00025000
GOVINIT EQU 1 ----- INITIALIZATION OF SESSION 00026000
GOVTERM EQU 2 ----- TERMINATION OF SESSION 00027000
GOVSCMD EQU 3 ----- START COMMAND 00028000
GOVECMD EQU 4 ----- END COMMAND 00029000
GOVCONN EQU 5 ----- CONNECT COMMAND 00030000
GOVSDBAS EQU 6 ----- START DATA BASE 00031000
GOVEDBAS EQU 7 ----- END DATA BASE 00032000
GOVSACTV EQU 8 ----- SUSPEND QMF ACTIVITY 00033000
GOVRACTV EQU 9 ----- RESUME QMF ACTIVITY 00034000
GOVABEND EQU 10 ----- QMF ABEND OPERATION 00035000
GOVPAD10 DS CL7 ----- RESERVED FIELD 00036000
SPACE 00037000
GOVCADDR DS A ---- ADDR TO BRANCH TO FOR CANCELLATION 00038000
SPACE 00039000
GOVOROWS DS F ---- NUMBER OF OPTION ROWS RETRIEVED 00040000
SPACE 00041000
GOVSQLRC DS F ---- RESOURCE TABLE SQL RETURN CODE 00042000
SPACE 00043000
GOVSQLCA DS A ---- ADDRESS OF SQLCA FOR ERROR CONDITION 00044000
SPACE 00045000
GOVGROUP DS CL16 ---- GROUP NAME 00046000
GOVPAD20 DS CL32 ---- RESERVED FIELD 00047000
00048000

```

図 105. DXEGOVA 制御ブロック (1/2)

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 105. DXEGOVA 制御ブロック (2/2)

リソース・コントロール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・コントロール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 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 NULL 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。

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、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。

図 106 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、GOVRESC が GOVRESCT DSECT を指しています。

```

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 106. 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ー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 56 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、稼働中のオペレーティング・システムの 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 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	<p>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は、適用されない。取り得る値は次のとおり。</p> <ol style="list-style-type: none"> 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 <p>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に変わる。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。</p>

表 56. 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
XCBAIACT	CL1	<p>現行のコマンドが対話式に実行しているか否かを示す。</p> <p>1 対話式 0 非対話式 (バッチ)</p> <p>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。</p>
XCBAUTH	CL8	<p>ユーザーの SQL 許可 ID を含んでいる。新機能モードで実行中に、SQL 許可 ID が 8 文字を超えている場合、値は切り捨てられてこのフィールドに配置されます。完全な SQL 許可 ID については、次の XCBAUTHX を参照してください。</p>
XCBAUTHX	CL128	<p>ユーザーの SQL 許可 ID を含んでいる。</p>
XCBCAN	CL1	<p>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。</p>
XCBCLOC	CL18	<p>現行ロケーションの名前が含まれる。</p>
XCBCMDL	F	<p>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。</p>
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	<p>現行コマンドの verb 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。</p>
XCBDDBMG	CL1	<p>データベース・マネージャーを識別する。この値は、DB2 UDB (z/OS 版) の場合は 2 に設定される。</p>
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 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。</p>
XCBERRET	F	<p>デフォルトの取り消しメッセージで使用される戻りコードが含まれる。</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 56. 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
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、「DB2 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。
XCBKPARAM	CL1	DSQSDBCS プログラム・パラメータ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。
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。
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。
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 DSQDEBBUG 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。
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、ix 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。
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。
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。

表 56. 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
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 V7R2.0 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。z/OS DB2 システム表または QMF コントロール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 ERASE TABLE EXPORT TABLE IMPORT TABLE PRINT TABLE 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>
XCBRELN	CL2	QMF のリリース・レベルを識別する。DB2 QMF バージョン 8.1 では、これは 12 になる。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。

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 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
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 z/OS を示す 3 に設定されている。取り得る値は次のとおり。</p> <p>3 TSO、またはネイティブ z/OS バッチ</p> <p>5 CICS</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)。</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、369 ページの『QMF トレース機能の使用』を参照。</p>
XCBUSER	CL8	<p>ユーザーの TSO ログオン ID (TSO の場合)、または JOB ステートメントのユーザー・パラメーター (ネイティブ z/OS バッチの場合) 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。</p>
XCBUSERS	CL2048	<p>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。</p>

DXEXCBA 制御ブロックの構造が次に示されています。

```

***** 00001000
*
* CONTROL BLOCK NAME: DXEXCBA * 00002000
*
* FUNCTION: * 00003000
* * 00004000
* * 00005000
* * 00006000
* THIS IS THE INTERFACE CONTROL BLOCK BETWEEN QMF AND * 00007000
* EXIT ROUTINES. * 00008000
* * 00009000
* STATUS: VERSION 8 RELEASE 1 LEVEL 0 * 00010000
* (Version 8.1 applies only to OS/390 and z/OS * 00011000
* VM and VSE remain at Version 7.2) * 00012000
* INNER CONTROL BLOCSCK: NONE * 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' ----- 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
XCBKPARM EQU C'0' ----- LATIN 00048000

```

図 107. DXEXCBA 制御ブロック (1/3)

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 and z/OS	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
XCBIACT	DS	CL1	-----	CURRENT INTERACT MODE	00060000
XCBIAIAC	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
XCBRGRP	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 107. DXEXCBA 制御ブロック (2/3)

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
XCBERRET	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
XCBAUTHL	DS	H	-- LENGTH OF AUTHORIZATION ID	00126000
XCBAUTHX	DS	CL128	-- AUTHORIZATION ID EXTENDED	00127000
XCBPAD70	DS	CL50	----- RESERVED FIELD	00128000

図 107. DXEXCBA 制御ブロック (3/3)

QMF セッション中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 324 ページの図 108 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。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。

```
LA    WORKPTR,GOVUSERS
      USING WORK,WORKPTR
```

図 108. 管理プログラム・スクラッチパッド域へのアドレス可能度の確立

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

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 DSQUnGV1、DSQUnGV2、または DSQUnGV3 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ブランチします。図 109 は、QMF 取り消しサービスへのアドレス可能度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 109. 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 が取り消し要求を受け入れた場合は 0 の戻りコードを、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 の戻りコードを含んでいます。

TSO またはネイティブ z/OS 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 (TSO およびネイティブ z/OS)、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。z/OS では、ソース・コードは、ライブラリー QMF810.SDSQSRE のメンバー DSQU_nGV1 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くてもあり、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。

z/OS

TSO またはネイティブ z/OS において、タイマーがアクティブの場合、管理プログラム出口ルーチンは非同期コマンドを取り消すことができます。

TSO またはネイティブ z/OS 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。

取り消されたアクティビティーに対するメッセージ

以下の説明を参考にして、z/OS 上にメッセージを表示します。

z/OS

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

エラー戻り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、XCBMGTX 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。

- 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、XCBMGTX 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。

XCBLOGM 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 XCBLOGM 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、XCBMSGNO にメッセージ定義のメッセージ ID で埋め、XCBMGTX 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 XCBMGTX 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、XCBMSGNO 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 XCBMGTX 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と、ユーザーにはメッセージ・ヘルプは提供されません。

QMF が ISPF のもとで呼び出された場合は、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 LOG サービスを用いてこの記録を行うことができます。ISPF LOG サービスの詳細については、該当する「*ISPF* ダイアログ管理サービス」マニュアルを参照してください。

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 DSQDEBUG 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 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、369 ページの『QMF トレース機能の使用』を参照してください。

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。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は、メッセージは画面に表示されません。

管理プログラムへ入ったとき、XCBMGTXТ 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。

- 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、XCBMGTXТ 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。

XCBLOGM 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 XCBLOGM 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。

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 DSQDEBUG 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 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、369 ページの『QMF トレース機能の使用』を参照してください。

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。

TSO、ISPF、およびネイティブ z/OS バッチでの管理プログラム出口ルーチン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言語のみです。アセンブラ言語は、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 用に作成されています。このプログラムは、QMF810.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。

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 QMF810.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 DSQUnGV1 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 QMF810.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、QMF810.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。

この新規モジュールの入り口点は、必ず、DSQUnGV1 でなければなりません。ソース・コードが、DSQUnGV1 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 DSQUnGV1 ラベルで始まっている

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 z/OS 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)
```

z/OS 上の CICS での管理プログラム出口ルーチンのアセンブル、変換、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。

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 はアドレス可能度を確立する標準の 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、QMF810.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。

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 のロード・モジュールが入っている QMF810.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 DSQUnGV3 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 QMF810.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、QMF810.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、DSQUnGV3 でなければなりません。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、リンケージ・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)
```

z/OS での 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 (z/OS 版) 管理ガイド」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、または指示照会の場合があります。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。これらのコマンドは、+495 SQL コードの回答においてプロンプト画面を表示しません。

表エディターの実行

表エディターのサポートでは、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。

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機能は作成しません。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 DSQDEBBUG 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場合があります。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 セッションは終了してしまい、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。

- 列 ASUTIME の最大プロセッサ時間の値
- 列 RLFFUNC に値 '2' を指定する
- 列 RLFCOLLN に集合名 'Q' を指定する

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 が入っています。その他のユーザーの行には、その許可 ID に対して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情報については、「DB2 UDB (z/OS 版) 管理ガイド」を参照してください。

第 20 章 バッチ・プログラムとしての QMF の実行

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。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。

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ジャーを実行することができます。プロシージ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・プロシージ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バッチ・プロシージャーを作成することができます。プロシージャーでは、照会や他のプロシージャーを呼び出したり、他の QMF コマンドを実行することができます。バッチ・プロシージャーの作成の詳細については、*DB2 QMF 使用の手引き* を参照してください。

また、QMF には、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。

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。

TSO/CICS でのバッチ・プログラムとしての QMF の実行

この章では、TSO、ISPF、ネイティブ z/OS、または CICS での QMF バッチ・モードの使用方法を説明します。z/OS での 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。

TSO

z/OS に関する説明の順序は、TSO、ISPF、ネイティブ z/OS、および CICS です。

バッチ・モード (TSO) で操作する権限

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 を判別する方法は次のとおりです。

- 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は、ログオン ID および 1 次許可 ID は 339 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にのみ実行されます。

- 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 QMF 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、*DB2 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 z/OS へのジョブの送信

管理者もしくはユーザーは、実行する QMF 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 QMF 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、*DB2 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。

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 Extensions Command Language Reference*」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調整することが可能であり、呼び出された CLIST にその値が渡されます。

プロシージャーの実行中にエラーが起こると、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 リカバリー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、エディター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。

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。

バッチ・プログラムとしての QMF の実行

- 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。

重要: 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。

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、適切な数の動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、SYSTEM、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 (別の FIB コマンド)。 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(DSQMF) NEWAPPL(DSQE) PARM(DSQSMODE=B,DSQSRUN=LMN.PROCA)
:
/*
```

図 110. 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。

PROFILE PREFIX ステートメント: 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は、QMF は ログオン ID を BATCH と見なします。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、DSQDEBUG 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、87 ページの『第 10 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てします。

```
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。

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、73 ページの『第 9 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。

```
ISPSTART CMD(clist_name) NEWAPPL
```

ここで、clist_name は QMF を開始する 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(DSQMF) 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) の値を指定すればよい場合、十分です。この後処理の指定方法が分からない場合は、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の問題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。

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。

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: 変換された同義語をユーザーに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に変換します。同義語の割り当ての詳細については、215 ページの『第 16 章 QMF コマンドのカスタマイズ』を参照してください。

z/OS での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 を作成しますが、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。TSO 管理者がこの権限をユーザーに付与します。

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 (146 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)。

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いない必要はありません。これらのオブジェクトは、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。

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 に対して変換された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 コントロール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呼び出される必要があります。アプリケーションが呼び出されると、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。

- 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。ユーザーは、JCL あるいは QMF プロシージャに関する知識を必要としません。

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、344 ページの図 111 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 111. 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、カーソルが入力が必要とするフィールドに位置付けられます。表 57 に、必須フィールドの説明があります。

表 57. 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、この名前が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 または 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。

表 57. BATCH アプリケーションの必須入力フィールド (続き)

フィールド	説明
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: 表 58 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。

表 58. 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。	
PROC ARGUMENTS	このフィールドを介して、OBJECT NAME フィールドで指定された REXX プロシージャに引き数を渡すことができる。

表 58. 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
FORM NAME	<p>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、</p> <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、書式はこの名前で作成される。書式が誰か他の人に所有され、共有されている場合は、名前を修飾する必要がある。 • 現行の書式の場合、作成できる名前を指定する。書式はそれぞれの許可 ID で作成されるため、名前は修飾しないで指定する必要がある。 <p>この書式は、プロファイルの設定として CONFIRM=NO を用いて作成される。</p> <p>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。</p>
Current FORM	<p>バッチ書式が現行の書式である場合は、このフィールドに YES を入力する。すると、書式は後にバッチで使用するために作成される。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。</p>
DB2 SUBSYSTEM	<p>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 DSQSSUBS の値と同じ。</p>
DB2 PLAN	<p>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。</p>
LOGON PASSWORD	<p>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。</p>
LOGGING	<p>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。</p>

表 58. 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
SAVE DATA	照会またはプロシージャの執行結果のデータを保管したい場合は、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 z/OS JCL に渡されるので、z/OS 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、z/OS 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。現在そのデータセットが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要。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 以下の場合は、デフォルト値 133 を使用する。
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。

表 58. 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
OUTPUT CLASS	<p>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。</p> <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) <p>インストール先システムでその準備がされている場合は、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。</p>
DISTRIBUTION USERID および NODE	<p>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。</p> <p>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。</p>

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を作成してください。

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。

- QMF810.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。

- QMF810.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。
- QMF810.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。
- QMF810.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をモデル化したもの。
 - DSQABB1P および DSQABB1S は QMF プロシージャをモデル化。ジョブの JCL 実行依頼プロシージャ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)。
- 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。

バッチ・プログラムとしての QMF の実行

- 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。351 ページの図 112 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 112. DSQABB1S SKELETON の変更

352 ページの図 113 のコメントにあるように、DSQABB12 CLIST に 5 つの変更を行います。

```

/*****/ 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__NLFCMD__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 113. DSQABB12 CLIST の変更 (1/2)

```

/*****/
/* 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 113. DSQABB12 CLIST の変更 (2/2)

ネイティブ z/OS 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 z/OS バッチ・ジョブとして QMF を実行することができます。354 ページの図 114 に示す JCL を使用して、ネイティブ z/OS のバッチ・ジョブとして QMF を実行することができます。

```

/*****/                                00299000
//QMFBAT JOB                            00300000
//S1 EXEC PGM=DSQQMFE,PARM='M=B,I=yourQMFproc' 00301000
//*                                       00302000
//* Program libraries required when running in batch 00303000
//*                                       00304000
//STEPLIB DD DSN=QMF810.SDSQLOAD,DISP=SHR      00305000
// DD DSN=DSN.SDSNEXIT,DISP=SHR              00306000
// DD DSN=DSN.SDSNLOAD,DISP=SHR             00307000
// DD DSN=GDDM.ADMLOAD,DISP=SHR            00308000
//*                                       00309000
//* QMF/GDDM maps are required when running in batch 00310000
//*                                       00311000
//ADMGGMAP DD DSN=QMF810.SDSQMAPE,DISP=SHR    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=VBA,LRECL=125,BLKSIZE=1632) 00319000
//DSQSPILL DD DSN=*&SPILL,DISP=(NEW,DELETE), 00320000
// UNIT=SYSDA,SPACE=(TRK,(100),RLSE),        00321000
// DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)    00322000
//*                                       00323000
/*****/                                00324000

```

図 114. QMF をネイティブ z/OS 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 z/OS で実行する場合は、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。
- TSO または ISPF を必要とする QMF 機能は、ネイティブ z/OS で QMF を実行する場合は作動しない。
- デフォルトのユーザー ID の接尾部は使用できない。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。
- 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、355 ページの図 115 に示すように IRXJCL を使用する必要がある。

355 ページの図 115 に示す REXX プログラムでは、QMF 呼び出し可能インターフェースを使用して、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.QMF810.SDSQLOAD,DISP=SHR
//ADMGGMAP DD DSN=QMFDEV.QF720.SDSQMAPE,DISP=SHR
//SYSEXEC DD DSN=ROBIN.QMF810.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 115. QMF をバッチ・モードで開始、実行させる REXX プログラム

CICS での非対話式トランザクションとしての QMF の実行

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してください。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、356 ページの図 116 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。

バッチ・プログラムとしての QMF の実行

まず (QMF プロシージャー内の 「--」 で始まる行を、QMF はコメントとして扱います)。

```
-- 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 116. 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。プロシージャーが正常終了すると、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 プロシージャーをバッチで実行するには、EXEC CICS START コマンドを使用します。次の例では、QMF プロシージャー 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。

プロシージャーのデバッグ

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 プロシージャーをバッチ・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 補助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。

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。一時記憶域がすぐにいっぱいになってしまいます。メッセージ・レベル以外のトレースを行う場合は、一時データ (デフォルト) を使用するようお勧めします。

終了戻りコード

QMF の終了戻りコードは、正常終了の場合は **0**、異常終了の場合は **8** です。

第 21 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ために使用してください。『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、366 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。

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上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。

- 職場のすべての 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 *DB2 QMF* メッセージおよびコードの *DSQDEBUG* 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 *DSQDEBUG* にも何も無い場合は、*ISQL* 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。

z/OS での考慮事項

DSN (DB2) または *ADM (GDDM)* エラー・メッセージが出ていた場合は、*DB2 QMF* メッセージおよびコードでそれらのメッセージを調べる。

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 コマンドを実行した後) 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、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、215 ページの『第 16 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、231 ページの『第 17 章 QMF 機能キーのカスタマイズ』に説明がある。
- 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、220 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、235 ページの『ユーザーの機能キー定義を表へ入力』参照。
- 表のすべての行が固有であること。

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、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。

z/OS の考慮事項

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 およびネイティブ z/OS 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。

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。

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。

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 プリンターのファミリーごとの有効な装置トークンのリストについては、「*GDDM System Customization and Administration*」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、「*GDDM System Customization and Administration*」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。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。

z/OS での GDDM エラーの処理

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。

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。

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 プリンターのファミリーごとの有効な装置トークンのリストについては、「*GDDM System Customization and Administration*」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、「*GDDM System Customization and Administration*」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。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。

z/OS での印刷時の QMF エラーの処理

下の表の情報は、印刷時に発生したエラーを解決する際に役立ちます。

障害追及と問題診断

起こったこと	意味	処理
<p>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。</p> <p>GDDM printer nickname is required for PRINTER.</p>	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。
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。
<p>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。</p> <p>File DSQPRINT did not open.</p>	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

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、DB2 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、247ページの『第 18 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。

パフォーマンス上の問題の解決

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。

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場合があります。たとえば、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な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、ユーザーに十分な量の仮想記憶域を指定するようにしてください。

い。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。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、予備ファイルには各行 4K サイズで最大 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、低下することがあります。

z/OS の考慮事項

DSQSRSTG パラメーターの設定を非常に高い値に設定することが、パフォーマンスの低下の原因になることもあります。

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、コントロール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 (z/OS 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。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。

診断援助機能を使用した問題の判別

359 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 TSO 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

367 ページの表 59 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

生じた場合は、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 59. 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	ダンプ	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ース	X	X	X	X		X	
表示パネル	X	X			X	X	X
文書インターフェース	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
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	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こと、あるいはエラーが発生したことを知らせます。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、DB2 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より詳細な説明が記載されています。

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、表 60 に示します。トレース ID は、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 60. 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 60 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、*DB2 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。 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 戻りコードの処理

場合によっては、QMF が表示するメッセージが SQL 戻り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ます。このメッセージは、SQL 戻りコード -30060 にマップされます (この戻りコードのテキストは次のとおりです)。

RDB AUTHORIZATION FAILURE

SQL 戻りコードについては、DB2 メッセージおよびコード を参照してください。

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。

z/OS でのトレース機能

z/OS でのトレース機能の使用については、以下の説明に従ってください。

トレース・データ・セットの割り振り (TSO): 本書の一部のプロシージャーでは、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 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・セットに記録されます。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 プロシージャー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 117. 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。

QMF の開始時には DSQSDBUG 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。

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。

- 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ます。

- ユーザーは、DISPLAY PROFILE を実行して、プロファイルの TRACE パラメーターを変更することができる。ユーザーがこの設定を永続的なものにする場合は (次の変更まで)、PF2 を押してそれを保管できます。
 - ユーザーは、設定用の SET (T=value コマンドを発行できる。この設定により、ユーザーのプロファイルが一時的に変更されます。この設定を保管するために、ユーザーは、SAVE PROFILE コマンドを発行できます。
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は、トレース・データ・セットにアクセスします。
DSQDEBUG を調べると、エラーの理由を判別するのに役立ちます。
 5. トレース出力を解読します。
DSQDEBUG 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、下の表に示します。

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、表エディター、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、および表エディター・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)
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 118 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 118. 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・セットまたはファイル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、372 ページの図 118 に示されているメッセージは、*DB2 QMF* メッセージおよびコード で次のように示されます。

```
Column &01 is not in table &02.
```

372 ページの図 118 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て、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SEMB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)
```

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・エディター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) DSNAME(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せん。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、375 ページの図 119 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 119. 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せん。

- 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。
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は、プログラム診断依頼書 (APAR) 番号も示されます。

```
** DSQFQWRM: ENTERED FROM DSQFMCTL ***  
V7R2.00 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実行します。

```
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、プログラム・パラメーター DSQSDBG がこのトレースを制御します。

異常終了の診断

インストール先の環境で使用可能な TSO、z/OS、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。図 120 にその例を示します。

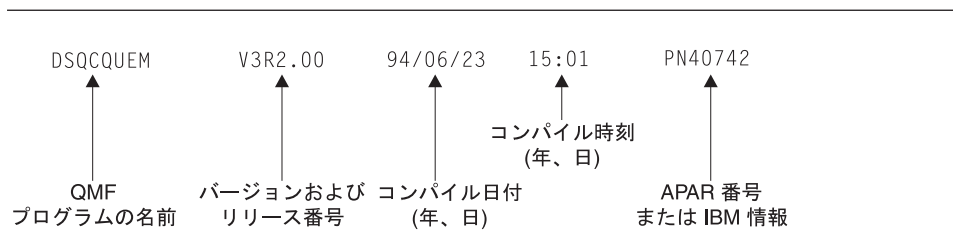


図 120. QMF プログラムを識別するスタンプの例

z/OS 診断機能の使用

異常終了を診断するには、該当する *Tools and Service Aids* に記載されている手順を使用する必要があるか、または、QMF 異常終了ハンドラーを使用することができます。

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は、DXEECS、入力域、および結果域が出力に追加されません。

管理プログラムの場合は、DXEXCBA および DXEGOV が出力に追加されます。

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ることがあります。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。378 ページの図 121 に、使用できる診断表示属性の例を示します。

図 121 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
ERR Last line      : Yes           No | Yes
ERRIntensify      : Yes           No | Yes
ERRColor          : Red           NO | Blue | Red | Pink | Green
                  |               | Turquoise | Yellow | Neutral
ERRHighlight      : Underline     No | Blink | Reverse | Underline
    
```

図 121. CICS 診断表示用の TYPETERM 指定

QMF 割り込み機能の使用

z/OS

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はない限り、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 122. 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 123. 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は、CANCEL を入力します。そうすると、キーボードがアンロックされ、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 124 に示すような診断情報を得たい場合は、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 124. 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 124 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は、CANCEL を入力します。 そうすると、キーボードがアンロックされ、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 コントロール表です。 表の構成を、表 61 に示します。

表 61. 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	いいえ	エラーの発生した日付。 フォーマットは yyyyymmdd。
TIMESTAMP	CHAR	5	いいえ	エラーが発生した時刻。 フォーマットは hh:mm で、hh は時、mm は分。
USERID	VARCHAR	128	いいえ	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	いいえ	エラーと一緒に発行された QMF メッセージ番号。
MSGTEXT	VARCHAR	254	いいえ	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。 その場合は、MSGTEXT 列を除く各列の値は繰り返し出されます。 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およその時刻を知っている必要があります。図 125 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 125. 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。

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。公表されていない問題については、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の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は、直接 384 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組み立てて、SSF を検索してください。QMF 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 (表 62 に示してある番号) で始まります。

表 62. 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

表 62. QMF 基本製品および NLF のリリース番号 (続き)

NLF	ID
イタリア語	657
日本語	658
韓国語	659
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

383 ページの図 126 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。

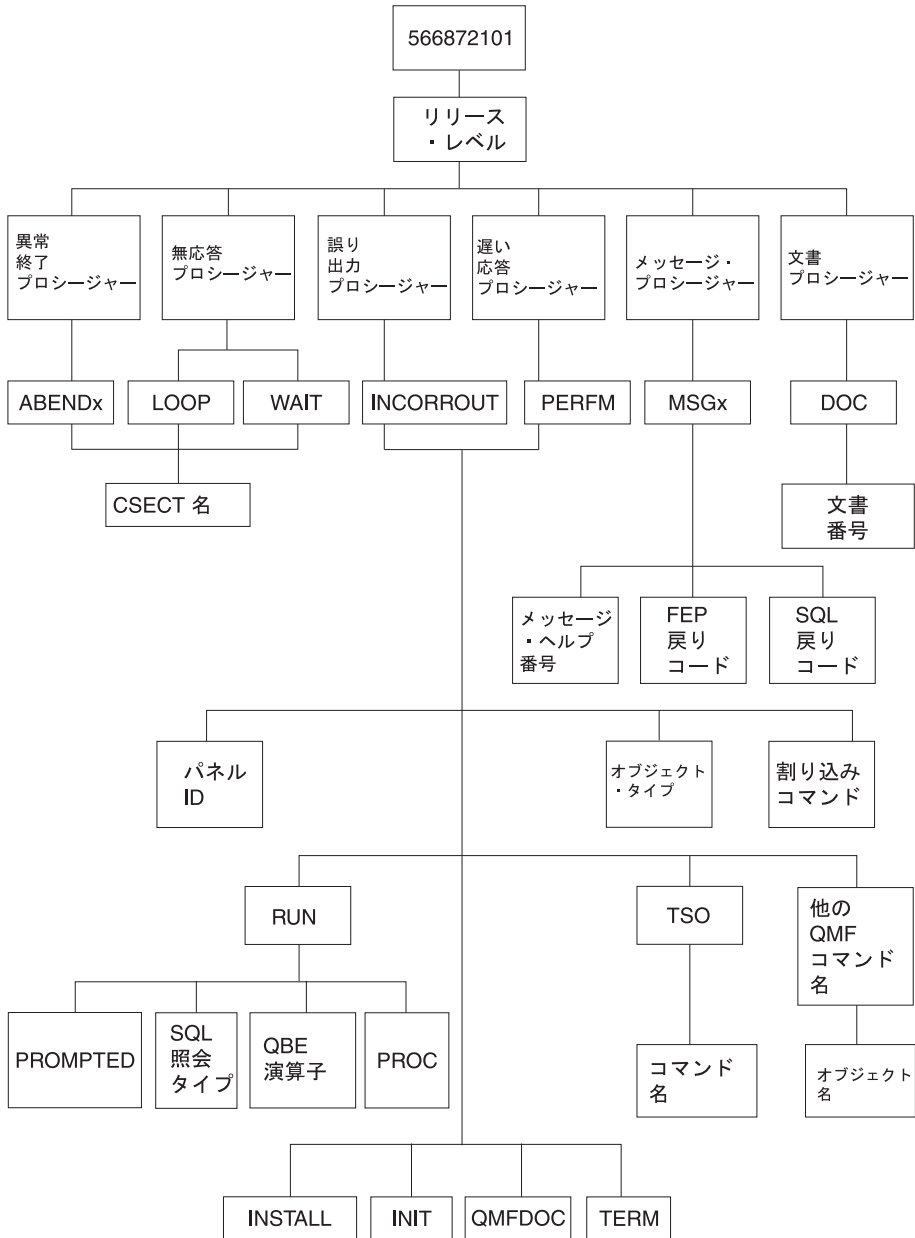


図 126. キーワード・タイプの図表：（この表を上から下へとたどって、キーワードを決定してください。）

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 の問題である場合は、次の検索句を使用してください。

```
566872101 09 ABEND0C4 DSQFDTBL
```

CSECT 名を見つけるには、ABEND CSECT NAME という見出し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、369 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プログラムの CSECT に関するいくつかの情報を提供する必要がある場合があります。多くの場合、このような種類の情報は 369 ページの『QMF トレース機能の使用』で説明されてい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、367 ページの表 59 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。

第 3 部 付録

付録 A. 各種問題

問題が発生した場合

インストール・プロセス中に、無視しても差し支えない通知メッセージがいくつか表示されます。他に、誤り訂正が必要な警告メッセージまたはエラー・メッセージも表示されます。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 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 8.1 LOADLIB とマップ・グループを割り振るようになる。
- 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 (z/OS 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 z/OS LLA REFRESH が行われたかを確認する。
- 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)

問題が発生した場合

- 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を確認する。DD ステートメントについては、43 ページの『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・メッセージを探します。

AEY9 ABEND

DB2 UDB (z/OS 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 proflex が無効。これは、DSQ1TBJ2 で生成される通常のメッセージです。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てください。詳細については、33 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 (z/OS 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 を使用する場合は、次の項目をコーディングしてください。


```
DSNCRCT TYPE=ENTRY, TXID=QMFE, PLAN=QMF810, AUTH=DEPT1
```

この例では、許可 ID は DEPT1 であり、プラン ID は QMF810 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 (z/OS 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

項目 QMFCAT.DSNDBC.DSQDBCTL.PROFILEX.I0001.A001。

IDC3009I

**VSAM カタログの戻りコードは 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。

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 プロシーチャーの 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チャーについては、199 ページの『第 15 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 プロシーチャーの 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、42 ページの『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、問題が生じてでもエラー・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 8 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 QMF810.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、359 ページの『第 21 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、図 127 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。

```
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 127. 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。

z/OS 上の CICS での DB2 QMF バージョン 8.1 製品の制限

z/OS および TSO では使用可能だが、z/OS 上の CICS では使用不能な基本システム・サービスその他のプログラム製品があります。QMF によって提供される機能の中には、それらに依存するものもあります。ISPF は CICS では使用不能です。REXX は CICS では使用不能です。以下の QMF 機能またはプログラムは CICS での実行中に QMF でサポートされません。つまり、これらの機能は ISPF に依存します (いくつかのケースでは、他のサービスにも依存します)。

- 報告書の計算
- 条件付きフォーマット設定
- 列定義
- ロジックを持つプロシージャ

以下の他の製品は CICS では利用不能です。

- リポジトリ・マネージャー
- 文書インターフェース

EDIT PROC および EDIT QUERY コマンドは、CICS では利用不能です。ただし、QMF で DISPLAY コマンドを使用して、プロシージャおよび照会を編集することはできます。

- TSO コマンド
- CONNECT コマンド (他のデータベースに接続するために発行される場合)
- リモート作業単位および分散作業単位
- QMF クライアント/サーバー・コンポーネント

付録 B. DB2 に常駐している QMF オブジェクト

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 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 DB2 QMF バージョン 8.1 を実行するため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・オブジェクトをリカバリー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。

QMF プラン

表 63 に、TSO/CICS 用 DB2 QMF 出荷時のプランを示します。

表 63. QMF プラン

プラン名	バインド・ジョブ	コメント
QMF810	DSQ1BINR	汎用 QMF プラン
DSQIN810	DSQ1BSQL	インストール・ジョブに限って使用される QMF プラン

QMF パッケージ

表 64 に、QMF 出荷時のパッケージを示します。

表 64. QMF パッケージ

パッケージ名	バインド・ジョブ
DSQE*	DSQ1BPKG JCL (z/OS)
リモート・サーバーへの: DSQE*	DSQ1BPKG JCL (z/OS がサポートする任意のサーバー)

TSO/CICS での QMF QMF コントロール表および表スペース

394 ページの表 65 に、QMF 出荷時のコントロール表を示します。

注: iSeries では、これらの QMF DB 記憶域構造が作成される前に、集合 "Q" が作成される必要があります。iSeries には、ノード・グループも表スペースも DB スペースもありません。

表 65. QMF オブジェクト、コントロール表、保管データ表、およびサンプル表

コントロール表名	表スペース	表スペース のサイズ (1K 単位)	表の内容	索引
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。	Q.PROFILEX
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す。	Q.OBJECT_ DIRECTORYX
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。	Q.OBJECT_ OBJDATAX
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。	Q.OBJECT_ REMARKSX
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。	Q.COMMAND_ SYNONYMNSX
Q.RESOURCE_ TABLE	DSQTSGOV	100 - 1 次 20 - 2 次	管理プログラム出力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。	Q.RESOURCE_ INDEX

表 65. QMF オブジェクト、コントロール表、保管データ表、およびサンプル表 (続き)

コントロール表名	表スペース	表スペースのサイズ (1K 単位)	表の内容	索引
Q.ERROR_LOG	DSQTSLOG	100 - 1 次 20 - 2 次	システム・エラー、リソース・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。	なし
Q.DSQ. RESERVED	DSQTSRDO	12 - 1 次、 4 - 2 次	インストール中に QMF が使用する情報を含んでいます。重要: この表を変更しないでください。	なし

QMF 視点

下の表は、QMF 出荷時の視点を示します。

表 66. QMF 出荷時の視点

視点名	視点の元となる表	オペレーティング・システム
Q.DSQEC_ALIASESL	SYSIBM.SYSTABLES	z/OS
	SYSCAT.TABLES	ワークステーション
	QSYS2.SYSTABLES	iSeries
Q.DSQEC_COLS_LDB2L	SYSIBM.SYSCOLUMNS	z/OS
	SYSIBM.SYSTABAUTH	z/OS
	SYSCAT.COLUMNS	ワークステーション
	SYSCAT.TABAUTH	ワークステーション
Q.DSQEC_COLS_RDB2L	QSYS2.SYSCOLUMNS	iSeries
	SYSIBM.SYSCOLUMNS	z/OS
	SYSIBM.SYSTABAUTH	z/OS

表 66. QMF 出荷時の視点 (続き)

視点名	視点の元となる表	オペレーティング・システム
Q.DSQEC_QMFOBJSJL	Q.OBJECT-DIRECTORY	全オペレーティング・システム
	Q.OBJECT_REMARKS	全オペレーティング・システム
Q.DSQEC_TABS_LDB2L	SYSIBM.SYSTABAUTH	z/OS
	SYSIBM.SYSTABLES	z/OS
	SYSCAT.TABAUTH	ワークステーション
	SYSCAT.TABLES	ワークステーション
	QSYS2.SYSTABLES	iSeries
Q.DSQEC_TABS_RDB2L	SYSIBM.SYSTABAUTH	z/OS
	SYSIBM.SYSTABLES	z/OS
Q.RESOURCE_VIEW	Q.RESOURCE_TABLE	全オペレーティング・システム
Q.DSQ_RESERVED_DB	SYSIBM.SYSCOLUMNS	z/OS
	QSYS2.SYSCOLUMNS	iSeries
	SYSCAT.COLUMNS	ワークステーション
Q.DSQ_RESERVED_OBJ	Q.OBJECT_DIRECTORY	全オペレーティング・システム

これらの視点のいくつかは、DB2 システム表に基づいており、QMF の LIST 機能および DESCRIBE 機能で使用されます。

ジョブ DSQ1BVW JCL を実行することにより、z/OS から、サポートされる任意の DB2 データベース上のすべての QMF コントロール表視点を作成 / 再作成することができます。このジョブによって、すべての QMF コントロール表視点の除去 (DROP) および作成 (CREATE) され、必要な権限が付与 (GRANT) されます。

z/OS では、QMF コントロール表視点を DB2 の 2 次許可 ID で使用可能にする場合は、このジョブを実行して、その DB2 データベースに対する QMF 視点をリフレッシュする必要があります。

TSO/CICS 用の VSAM クラスタ

397 ページの表 67 に、QMF 出荷時の VSAM クラスタを示します。

表 67. VSAM クラスター

クラスター名	クラスターを必要とするオブジェクト
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

TSO/CICS 用の QMF サンプル表

表 68 に、サンプル表を示します。

表 68. 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

DB2 に常駐している QMF オブジェクト

付録 C. QMF ユーザー定義関数

APPL_AUTHNAMES

ユーザー定義関数表についての構文記述は次のとおりです。

```
▶▶—APPL_AUTHNAMES(———)——▶▶  
                    └──adjuncts──┘  
  
▶▶—RETURNS TABLE(———)——▶▶  
                    └──authname──┘  
                    └──namekind──┘
```

APPL_AUTHNAMES 関数は、現行アプリケーション・プロセスの DB2 許可 ID を戻します。許可名ごとに行が戻されます。スキーマ名は Q です。

adjuncts VARCHAR(255)

許可名のストリング。それぞれの許可名を、ID または区切り ID として指定します。それぞれの許可名を、1 つまたは複数のブランクで分離します。

```
'SALES "DEPT A1" PAYROLL'
```

これらの 3 つの名前は、現行のプロセスの許可 ID としてまだ定義されていない別個の値を示している場合、関数の出力に追加されます。

この関数の結果は、以下の列を持つ DB2 表です。

- **authname** CHARACTER (8)
現行プロセスの許可 ID の名前。
- **namekind** CHARACTER(1)
AUTHNAME の名前値の種別コード。
 - **1** 1 次許可 ID またはユーザー名
 - **2** 2 次許可 ID またはグループ名
 - **3** 現行許可 ID
これは、CURRENT SQLID が現行プロセスの 1 次 ID でも 2 次 ID でもない場合のみ適用されます。
 - **9** 付加属性名前値
これは、ADJUNCT パラメーターが使用され、そこで指定されている ID が現行プロセスの許可 ID でない場合のみ適用されます。

CALL DSQAB1E

ストアード・プロシージャ・インターフェースについての構文記述は次のとおりです。

▶▶CALL DSQABA1E(—userid—,—groupids—,—sqlid—)▶▶

DSQAB1E ストアード・プロシージャは、現在実行中のプロセスの DB2 許可 ID を戻します。スキーマ名は Q です。

userid VARCHAR(130)

1 次許可 ID がこのパラメーターで戻されます。

groupids VARCHAR (32672)

2 次許可 ID がこのパラメーターで戻されます。

それぞれの許可名は、VARCHAR データ・フォーマットから単一文字列構造に変換されます。呼び出し側プログラムは、個々の許可名を得るために文字列の内容を解釈する必要があります。

sqlid VARCHAR (130)

現行の SQL 許可 ID がこのパラメーターで戻されます。

DSQABA1E

診断ユーザー定義関数についての構文記述は次のとおりです。

▶▶DSQABA1E(—)▶▶

DSQABA1E 関数は、IBM サービス技術員の問題診断を支援する診断情報を戻します。スキーマ名は Q です。

この関数の結果は、データ・タイプ VARCHAR の文字列で、実際の長さは 5,300 バイト以下です。この文字列は、列の設定を WIDTH = 53 および EDIT コードを CW にした QMF 報告書の書式設定に適しています。

付録 D. QMF および GDDM プログラムの CICS への定義のしかた

TSO/CICS 用 QMF では、QMF プログラムを CICS に対して定義したり、GDDM 定義をロードするために必要なジョブ、および QMF パネルの図表フォーマットを提供しています。QMF インストール中に QMF プログラムを定義する方法および GDDM 定義をロードする方法を知る必要がある場合は、この節をお読みください。

QMF プログラムの CICS 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、ix 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。

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
DSQUEGV3
DSQUECIC

CICS 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。

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、402 ページの『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

QMF および GDDM プログラムの CICS への定義のしかた

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。

QMF 初期化中の GDDM 定義のロードのしかた

QMF 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、*GDDM System Customization and Administration* を参照してください。

図表フォーマットの定義方法

QMF のデフォルトのインストールでは、GDDM ファイル ADMF 内に、図表フォーマット、図表データ、および GDF データが保管されます。GDDM 外部デフォルト・モジュール ADMADFC の OBJFILE セクションを変更することにより、この GDDM オブジェクト・ファイルの名前を変更するかまたは追加の GDDM オブジェクト・ファイルを作成して、図表オブジェクトに保管することができます。たとえば、図表フォーマット、図表データ、および GDF データを別のファイルにすることができます。

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-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度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。両方式とも、*CICS/OS390 相互通信の手引き* で説明されています。

付録 E. QMF リリース間の移行およびフォールバック

注: QMF を初めてインストールする場合は、この節をスキップしてください。

QMF バージョン 8.1 を正常にインストールした後で、ユーザーが依然として以前の QMF のリリースを使用していないかどうか確認します。使用していた場合は、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。以下を行ってください。

- ユーザーに DB2 QMF バージョン 8.1 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、新リリース下での 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、本書の「インストール」節のステップを実行する過程のことです。QMF バージョン 8.1 の場合、移行には 2 つのシナリオがあります。

- QMF の前のリリース (バージョン 3.3、6.1、7.1、または 7.2) からバージョン 8.1 への互換モード移行
- QMF バージョン 8.1 互換モードから QMF バージョン 8.1 新機能モードへの移行

この付録では、QMF バージョン 8.1 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は、説明の一部が適用できないことがあります。

QMF の複数のリリース

QMF バージョン 8.1 以前は、QMF の複数のリリースで 1 つの DB2 UDB データベースにアクセスすることができ、すべてのリリースで同じ QMF コントロール表および QMF オブジェクトを使用していました。QMF バージョン 8.1 の場合は、QMF 互換モードでインストールされた QMF バージョン 8.1 のみ以前の QMF リリースと同じ DB2 UDB データベースにアクセスできます。

QMF バージョン 8.1 新機能モードが DB2 UDB データベースにインストールされている場合、QMF の以前のリリースはすべて、そのデータベースにアクセスできません。

注: QMF バージョン 8.1 新機能モードのインストールは QMF バージョン 8.1 互換モード・インストールに接続でき、逆に QMF バージョン 8.1 互換モード・インストールは QMF バージョン 8.1 新機能モードのインストールに接続できます。

DB2 サブシステムと移行

ユーザーを移行させる場合は、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 UDB 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、『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 UDB 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 UDB サブシステム上にある場合は、この節をお読みください。

注: QMF バージョン 8.1 互換モードのインストールのみ、QMF の前リリースと同じ DB2 UDB サブシステムに共存できます。この節では、QMF バージョン 8.1 互換モードの移行のみについて言及しています。

z/OS での 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は、QMF の開始時に DSQSPRID パラメーターの値が TSO ID になっている必要があります。そうでない場合は、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 3.3、6.1、7.1、および 7.2 ユーザー: バージョン 3.3 以上から移行している場合、Q.PROFILES 表に新しい列はありません。

DB2 QMF バージョン 8.1 のもとで、以前のリリースからのオブジェクトを使用可能にする

両方の QMF のリリースが同じ DB2 サブシステム上にある場合、すべての DB2 オブジェクト（たとえば、表および視点）は、前リリースで使用できれば、DB2 QMF バージョン 8.1（互換モードおよび新規機能モード共）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 8.1 のもとでは使用できないものもあります。これについては 409 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 前のリリースのデータベースにある QMF オブジェクトをバージョン 8.1 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 8.1 のデータベース内のオブジェクトも、QMF 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 8.1 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。

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 69 に示します。

表 69. 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は、追加の SYSTEM 行があり、そこでは 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 8.1 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すが、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。

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、407 ページの表 69 を参照してください。

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、243 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。

QMF バージョン 8.1 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

この節は、QMF 互換モードおよび新規機能モードのインストールに適用されます。

DB2 表および QMF オブジェクトは、QMF 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 8.1 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 (z/OS 版) 管理ガイドを参照してください。
2. アンロードした表を、DB2 ロードーを使用してバージョン 8.1 の DB2 サブシステムにロードします。

QMF の 2 つのバージョンが異なる z/OS システム上にある場合は、使用可能なネットワーク機能を使用して、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 8.1 が入っているシステムに送信してください。

QMF 照会、書式、プロシージャー、およびアプリケーションを移行する場合は、必ず次の節 (409 ページの『QMF オブジェクトの移行』) を読んでください。

DB2 QMF 高性能オプション (HPO) をインストールしている場合は、1 つの DB2 サブシステムから他の DB2 サブシステムへの QMF オブジェクトの移行を援助させるために、HPO オブジェクト・マネージャーを使用することができます。

データ・リフレッシャーまたは DB2 QMF 高性能オプションの詳細については、Web サイト <http://www.ibm.com/software/data/qmf> を参照してください。

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 の以前のリリースで作成されたオブジェクトの大部分は、DB2 QMF バージョン 8.1 でも使用することができます。(QMF 前のリリースに戻る移行については、412 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 前のリリースで作成されたすべての照会および書式は、DB2 QMF バージョン 8.1 でも使用することができます。

プロシージャ

バージョン 3.3 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 8.1 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は、それらがバージョン 8.1 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含まれているバージョン 3.3 のプロシージャは、DB2 QMF バージョン 8.1 が ISPF ダイアログとして開始される場合のみ実行されます。英語で書かれ、QMF バージョン 3.3 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 8.1 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 verb として使用される verb を含むコマンドを出している場合は、正しく機能しません。DB2 QMF バージョン 8.1 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。この結果、これらのプロシージャは、DB2 QMF バージョン 8.1 で実行されます。インストール先定義コマンドの詳細については、215 ページの『第 16 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以前の QMF リリースのアプリケーションは、バージョン 8.1 が ISPF ダイアログとして開始される場合のみ実行されます。英語で書かれたコマンドを発行し、以前の QMF のリリースで実行される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 8.1 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

DB2 QMF バージョン 8.1 より前にリンク・エディットされていて、呼び出し可能インターフェースを使用するユーザーの QMF アプリケーションで LIBDEF 機能を使用する場合は、DB2 QMF バージョン 8.1 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。

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。

CICS での管理プログラム

IBM 提供の管理プログラムの使用を予定している場合は、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。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。

TSO およびネイティブ z/OS バッチでのユーザー編集ルーチン

QMF バージョン 8.1 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、247 ページの『第 18 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 3.3 以上から移行している場合は、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。

z/OS 上の CICS でのエクスポート/インポート・サポート

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

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。 こうしておくと、IGG019II 状態で異常終了 (ABENDASRA または ABEND0C4) 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るようにする移行機能が用意されています。 QMF バージョン 8.1 がインストールされている場合は、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 3、6、または 7 からの移行

QMF バージョン 8.1 では、QMF インストール CLIST、DSQ1EINS、および関連する ISPF インストール・パネルは存在しません。 QMF は、バッチ・ジョブでインストールしなければなりません。 QMF インストールにより、以前には非 SMP/E 管理データ・セットだった 3 つの新規 SMP/E ターゲット・ライブラリーが作成されます。

表 70. 以前の非 SMP/E 管理データ・セットおよびそれらを置き換える SMP/E ターゲット・ライブラリー

以前の非 SMP/E 管理データ・セット	新規 SMP/E ターゲット・ライブラリー
QMF.DSQMAPn	QMF.SDSQMAPn
QMF.DSQCHART	QMF.SDSQCHRT
QMF.DSQPVARn	QMF.SDSQPVRn

QMF バージョン 8.1 の視点およびデータベース・オブジェクトの詳細なリストについては、393 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。

フォールバック

フォールバックとは、ユーザーを QMF 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、z/OS 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 65 ページの『インストール後の終結処理』で説明しているため、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。

フォールバックとは何か

フォールバックとは、ユーザーを QMF バージョン 8.1 からそれより前のリリースに移行する過程です。QMF バージョン 8.1 互換モードから QMF バージョン 8.1 新機能モードへのフォールバック・モードはありません。終結処置については 65 ページの『インストール後の終結処理』で説明しているため、ここでは説明しません。

前のプロファイルの再確立

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。

以前のリリースでの DB2 QMF バージョン 8.1 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は、409 ページの『QMF オブジェクトの移行』を参照してください。

QMF 前のリリースにフォールバックする場合、QMF バージョン 8.1 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は、この点を考慮してください。以下に、一部のバージョン 8.1 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。

- データ・オブジェクト

QMF データ・フォーマットを使用してエクスポートされた QMF データ・オブジェクトは、以前の QMF のリリースにインポートできません。IXF データ・フォーマットを使用してエクスポートされた IXF データ・オブジェクトは、18 文字より長い列名を使用しない、提供された QMF の以前のリリースにインポートすることができます。

- 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 8.1 の照会に、いくつかの制約事項が適用されます。

- 指示照会: バージョン 8.1 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。新機能モードでバージョン 8.1 を実行中、エクスポートまたは保管されている指示照会は、以前の QMF のバージョンでは使用できません。

- プロシージャ

バージョン 8.1 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 8.1 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 8.1 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。

- 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、『前のリリースでの QMF バージョン 8.1 のコマンドの使用』を参照してください。

以前の QMF リリースと DB2 QMF バージョン 8.1 との相違点の詳細については、「DB2 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 QMF バージョン 8.1 のコマンドの使用

QMF バージョン 8.1 のプロシージャおよびアプリケーションは、以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 の以前のリリースでは正しく実行されることがあります。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。

付録 F. 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものであり、本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-0032

東京都港区六本木 3-2-31

IBM World Trade Asia Corporation

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せん。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
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。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

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が、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 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。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。

商標

以下は、IBM Corporation の商標です。

AIX	iSeries
C/370	MVS
CICS	OS/390
COBOL/370	Parallel Sysplex
DataJoiner	PL/I
DB2	QMF
DB2 Information Integrator	RACF
DB2 Universal Database	S/390
Distributed Relational Database Architecture	SQL/DS
DRDA	VM/ESA
GDDM	VSE/ESA
IBM	VTAM
IBMLink	WebSphere
IMS	z/OS
	zSeries

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。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

UNIX は、The Open Group の米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名などはそれぞれ各社の商標または登録商標です。

付録 G. 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、本書の索引を参照してください。

[ア行]

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。

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、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。

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、所要のデータを持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 (OS/390 版) では、アプリケーション・サーバーは、完全な DB2 UDB (OS/390 版) 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。

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 (OS/390 版) 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 (OS/390 版) 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 (OS/390 版) 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、「ローカル DB2 UDB (OS/390 版)」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。

用語集

異常終了 (abend). タスクの異常終了。

一時記憶域 (temporary storage). 現在使用している照会、書式、プロシージャ、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。

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持つ記憶域。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。

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。

[力行]

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。

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。

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。

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 エレメント。

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。

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。

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。

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。

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。

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。

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、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。

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持つ 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。

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。

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。

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・リクエスト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 (OS/390 版) サブシステム)

構造化照会言語 (Structured Query Language (SQL)). DB2 UDB (OS/390 版) 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。記述句で照会を書くために使用される。

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。

コマンド同義語 (command synonym). インストール先定義コマンドの verb または verb / オブジェクトの部分。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。

[サ行]

作業単位 (unit of work). (1) アプリケーション・プロセス内のリカバリー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

用語集

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ーとして扱う一連の 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。

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。

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。

システム・ネットワーク体系 (Systems Network Architecture (SNA)). 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。

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、異常発生時の記述、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。

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。

集約関数 (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。

集約変数 (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集約関数。その値は、作成された報告書に、切れ目脚注、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れる。

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')。

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。

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。

図形データ表示管理プログラム (Graphical Data Display Manager (GDDM)). 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。

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 (OS/390 版) 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 (OS/390 版) の構造。ほとんどの DB2 UDB (OS/390 版) 機能は、スレッド構造のもとで実行される。

線形構文 (linear syntax). あるプログラムまたはプロシージャの 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。

線形手順 (linear procedure). REXX コメントで始まっていない プロシージャ。線形手順には、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 (*procedure with logic*) も参照。

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。

[タ行]

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、行を組み合わせた、報告書に定数値を組み込んだりする空の表。

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。

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)。

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、プロシージャ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。

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持つ。

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・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。

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。

用語集

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 エLEMENT。

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。

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。

[ナ行]

ネットワーク制御プログラム (Network Control Program (NCP)). シングル・ドメイン、マルチドメイン、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。

[ハ行]

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。

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 (OS/390 版) では、出力がアプリケーション・プランとなることがある。)

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。

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。

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 エLEMENT。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。

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。

引き数 (argument). 独立変数。

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。

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。

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。

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。

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。

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 という名前を持つ。ユーザーは、それぞれプロファイルを 1 つしか持つことができない。

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 (OS/390 版) 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。

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。

分散リレーショナル・データベース体系 (Distributed Relational Database Architecture (DRDA)). 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。

別名 (alias). DB2 UDB (OS/390 版) で、同じ、またはリモートの DB2 UDB (OS/390 版) 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。

用語集

QMF で、ローカルまたはリモート DB2 UDB (OS/390 版) 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。

[マ行]

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、行を組み合わせた、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。

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。

明細見出しテキスト (detail heading text). 報告書の見出しのテキスト。見出しを印刷するかどうかは、FORM.DETAIL で指定する。

[ヤ行]

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

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。

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。

[ラ行]

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。

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。

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。

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。

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。

列折り返し (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。長さが幅幅を超える値を列に入れるときに、しばしば使用される。

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。

ローカル DB2 UDB (OS/390 版) (local DB2 UDB (OS/390 版)). DB2 UDB (OS/390 版) 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 (OS/390 版) サブシステムの一部である。したがって、DB2 UDB (OS/390 版) 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 (OS/390 版) は、CICS 領域で開始された DB2 UDB (OS/390 版) のサブシステム ID である。

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れると、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。

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 (OS/390 版) サブシステムは 1 つのロケーションと見なされる。

ロジックを持つプロシージャ (procedure with logic). REXX のコメントで始まる QMF プロシージャ。ロジックを持つプロシージャでは、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。線形手順 (*linear procedure*) も参照。

論理装置 (LU) (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。

[ワ行]

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 Universal Database (UDB)、DB2 Common Server、DB2 パラレル・エディション、および DataJoiner)。

[数字]

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。

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持つ言語には、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。

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。

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 (OS/390 版)) 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。

A

ABENDx. 異常終了問題のキーワード。

APAR. プログラム診断依頼書 (Authorized Program Analysis Report (APAR))。

C

CICS. 顧客情報管理システム (Customer Information Control System)。

CMS. 会話型モニター・システム (Conversational Monitor System)。

CP. VM 用制御プログラム (Control Program)。

CSECT. 制御セクション (Control section)。

D

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))。

E

EBCDIC. 拡張 2 進化 10 進コード (Extended Binary-Coded Decimal Interchange Code)。

H

HTML. ハイパーテキスト・マークアップ言語 (Hypertext Markup Language)。インターネットで表示される文書用の標準化されたマークアップ言語。

I

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。

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。

J

JCL. OS/390 用のジョブ制御言語。

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

M

MSGx. メッセージ問題のキーワード。

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)。

用語集

N

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)。

P

PERFM. パフォーマンス問題のキーワード。

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)。

Q

QMF 管理権限 (QMF administrative authority). 最低でも、Q.PROFILES コントロール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。

R

RDBMS. リレーショナル・データベース管理システム (Relational database management system)。

REXX. 再構造化拡張実行プログラム (Restructured extended executor)。

S

SBCS. 1 バイト文字セット (single-byte character set)。

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。

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。

T

- 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)。
- TSO.** タイム・シェアリング・オプション (Time Sharing Option)。

U

- 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

V

- 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。

W

- WAIT.** 無限待ち状態問題のキーワード。

付録 H. 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。

CICS の資料

CICS Transaction Server for OS390

CICS User's Handbook

CICS Transaction Server for z/OS CICS アプリケーション・プログラミング・リファレンス

CICS Transaction Server for z/OS CICS アプリケーション・プログラミング・ガイド

CICS Transaction Server for z/OS CICS DB2 ガイド

CICS Transaction Server for z/OS CICS リソース定義ガイド

CICS 問題判別の手引き

CICS システム定義の手引き

CICS Transaction Server for z/OS CICS 相互通信ガイド

CICS Performance Guide

CICS Transaction Server (VSE/ESA 版)

User's Handbook

アプリケーション・プログラミング解説書

アプリケーション・プログラミングの手引き

リソース定義ガイド

問題判別の手引き

システム定義の手引き

相互通信の手引き

パフォーマンスの手引き

COBOL の資料

COBOL VSE/ESA 版 言語解説書

COBOL VSE/ESA 版 プログラミングの手引き

DB2 Universal Database (z/OS 版) の資料

DB2 Universal Database (z/OS 版)

インストール・ガイド
管理ガイド
SQL 解説書
コマンド解説書
アプリケーション・プログラミングおよび SQL の手引き
メッセージとコード
ユーティリティー・ガイドおよび解説書
リモート DRDA リクエスターおよびサーバー解説書

IBM DB2 Server (VSE および VM 版)

診断の手引きおよび解説書
DB2 Server (VSE 版) メッセージおよびコード
DB2 Server (VM 版) メッセージおよびコード
DB2 Server (VSE 版) システム管理
DB2 Server (VM 版) システム管理
DB2 Server (VSE および VM 版) オペレーション
DB2 Server (VSE および VM 版) SQL 解説書
DB2 Server (VSE および VM 版) アプリケーション・プログラミング
DB2 Server (VSE および VM 版) ISQL ガイドおよびリファレンス
DB2 Server (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 Server (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 Universal Database for iSeries

SQL 解説書
SQL Programming with Host Languages

DB2 Universal Database

コマンド解説書
SQL 解説書
メッセージ解説書

DB2 DataJoiner

DataJoiner Application Programming and SQL Reference Supplement

DB2 Intelligent Miner

DB2 インテリジェント・マイナーの使用

Document Composition Facility (DCF) の資料

DCF and DLF General Information

Distributed Relational Database Architecture (DRDA) の資料

Every Manager's Guide

接続の手引き

Graphical Data Display Manager (GDDM) の資料

GDDM 概説書

GDDM 基本適用業務プログラミングの手引き

GDDM 使用者の手引き

GDDM/VSE Program Directory

GDDM メッセージ

GDDM システムのカスタマイズおよび管理

High Level Assembler (HLASM) の資料

High-Level Assembler for MVS, VM and VSE Programming Guide

High-Level Assembler for MVS, VM and VSE Language Reference

Interactive System Productivity Facility (ISPF) の資料

OS/390

対話式システム生産性向上機能(ISPF) 計画とカスタマイズ

対話式システム生産性向上機能(ISPF) ダイアログ開発者 ガイドとリファレンス

VM

ISPF for VM Dialog Management Guide and Reference

OS/390 の資料

JCL

OS/390 MVS JCL 解説書

OS/390 MVS JCL 使用者の手引き

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/E

TSO/E 入門

TSO/E ユーザーズ・ガイド

参考文献

SMP/E

OS/390 SMP/E メッセージおよびコード

OS/390 SMP/E 解説書

OS/390 SMP/E 使用者の手引き

OS PL/I の資料

OS PL/I Programming Language Reference

OS PL/I Programming Guide

REXX の資料

OS/390 環境

TSO/E REXX/MVS User's Guide

TSO/E REXX/MVS Reference

VM 環境

System Product Interpreter Reference

REXX/VM 使用者の手引き

VM/ESA の資料

VM/ESA 計画および管理

VM/ESA コマンド解説書

VSE/ESA の資料

計画

システム・ユーティリティー

問題解決の手引き

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

アクセス

データ 4

QMF アプリケーション・プランのパッケージに対する 126

アプリケーション・サーバー 8

アプリケーション・リクエスター 8

アンパーサンド (&)

コマンド同義語内 224

暗黙作成表スペース 149

移行, QMF バージョン 7 への

オブジェクト

異なる DB2 サブシステム
408

オブジェクトの互換性

QMF バージョン 7 のもとでの前のオブジェクト 409

プロファイルの考慮事項, 異なる

DB2 サブシステム 407

一時記憶域キュー

QMF サービスを使用する印刷
211

一時データ・キュー

出力の経路指定 200

QMF サービスを使用する印刷
211

印刷

エラー 361

ユーザーに使用可能にする 199

GDDM サービスの使用 200

QMF と GEM 199

インストール

パラメーター

値 21

NLF 180

インストール (続き)

分散作業単位を使用するための設定 9

リモート DB2 UDB (OS/390 版)

サブシステムへのアクセス 8

ワークシート

NLF 180

NLF 177

インストール後の終結処置 66

NLF 195

インストール・パラメーター

ワークシート, NLF 180

インターフェース制御ブロック

DXEGOVA 310

DXEXCBA 310

ウィンドウ・パネル

カスタマイズした機能キーの例
239

ID 241

上書き, デフォルト値の 33

エラー

初期化 359

エラー・メッセージ 387

オブジェクト

インストール 6

共用 157

コントロール表 153

削除 158

作成の標準 139

内部表現 154

名前, コマンド同義語 220

表示 157

保守 153

リスト 156

ウィンドウ ID 242

カスタマイズ 144

オブジェクトおよびオブジェクトと

分散データとの関係 7

オブジェクトの所有権の転送 157

オブジェクト・グループおよびイン

ストール 6

オブジェクト・リスト

グローバル変数を使用したカスタマイズ 145

[カ行]

カーソル固定 138

外部参照 390

概要

QMF の 3

概要, インストール・プロセスの 6

拡張浮動小数点, 編集ルーチン 248

カスタマイズ 231

QMF セッションの動作

ユーザー・プロファイルの使用
113

仮想記憶域

見積もり

CICS 18

MVS/ESA 16

仮想記憶域要件 16, 18

カタログ表, DB2 164

環境

カスタマイズ 113

デフォルトの設定 401

管理

表, 作成 147

ユーザー・プロファイルおよびオブジェクト 123

管理プログラム出口ルーチン

入り口点 302

機能呼び出し 309

機能呼び出しのタイプ 302

コマンド処理 305, 307

スクラッチパッド域 323

制御情報, 保管 323

制御の流れ 298

説明 287

出口ルーチン情報 316

取り消しサービス 324

パフォーマンス 309

管理プログラム出口ルーチン (続き)
プログラム構造 298
リソース制御情報を渡す 310
リソース・グループの指定 296
リソース・コントロール表 287
CICS 制御ブロック・インターフェース 298
キーワード、問題の報告 381
記憶域
表スペース、サイズの増加 159
編集ルーチンからのデータ 252
CICS/ESA 領域 16
記憶域要件
NLF SMP/E の 178
規則
機能キーのカスタマイズ 235
コマンド同義語 220
機能キー
カスタマイズ
アクティブ化の問題 235
機能キー表の更新 235
新規の定義のアクティブ化 243
表 234
作成 234
定義の入力 235
表の索引 235
機能呼び出し
タイプ 302
ブランチ・アドレス 309
基本 QMF コマンド、同義語としての 221
行、検索数の制御 287
許可
インストール検査プロシージャ
CICS 57
エラー 388
コマンド同義語 229
DBA、ユーザー Q 113
ID Q 8
IVP の実行 57
QMF へのアクセス 113
許可 ID 73
1 次 / 2 次
SAVE および IMPORT コマンド 132

許可、DB2
サンプル照会のインストール 194
サンプル照会の削除 194
クラス ID、機能キーのカスタマイズ 236
グラフィックス・プリンター、ニックネームの定義 200
グローバル変数
印刷 212
ウィンドウ ID 241
NLF の英語サポート 175
計画、インストールのオペレーティング・システム 15
仮想記憶域 16
ハードウェア要件 15
DB2 UDB (OS/390 版) の要件 6
SMP/E データ・セット 19
警告メッセージ 360, 387
計算、予備ファイル・サイズの 91
権限、DB2
配布、概要 161
権限のカスケード 134
検査プロシージャ 59
高性能オプション / マネージャー 287
互換モード 11, 27, 46
互換モードへの移行 28, 50
顧客情報管理システム (CICS)
QMF プロファイルの提供、移行 407
コマンド
CMS、同義語定義 222
RUN
同義語定義 222
SET PROFILE 121
コマンド同義語の省略 226
コマンド同義語表
作成 218
視点 228
保守 227
コメント
機能キー表に関する 234
コントロール表
再編成する時期 163
所有権 113

コントロール表 (続き)
バッファ・プールの切り替え 164
保守
環境 162
ユーザー管理のデータ・セットの VSAM クラスタ 162
Q.ERROR_LOG 380
Q.RESOURCE_VIEW 292

[サ行]

サーバー・データベース・インストール
ワークシート、QMF 181
QMF の開始 33
作業単位、定義 4
索引
再作成 159
Q.PROFILES 表 116
削除
旧リリースからのライブラリーの 66
QMF NLF 189
QMF NLF サンプル表 189
作成
QMF NLF
コントロール表 189
サンプル表 189
TSO ログオン・プロシージャ 29
サポート製品
設定 401
GDDM 402
参照、外部 390
サンプル
照会、インストール 62
表
削除 183
作成 184
TSO プロシージャ 29
サンプル QMF 照会
インストール、NLF 194
削除 193
サンプル QMF プロシージャ
インストール、NLF 194

- サンプル QMF プロシージャ (続き)
 - プロシージャ
 - インストール 194
 - サンプル表 61
 - 削除
 - NLF 183
 - 作成
 - NLF 184
 - NLF のインストール 183
 - 指示照会
 - 印刷 200, 214
 - ウィンドウ ID 242
 - DB2 特権 128
 - SQL 特権 136
 - システム
 - エラー・メッセージ 368
 - システム・カタログ表
 - 2 進データの警告 164
 - 視点
 - アクセスの制御 135
 - 基本オブジェクト 130
 - 削除 165
 - 作成するための特権 131
 - 照会に関する特権 136
 - スクリーニング・ツール 130
 - 読み取り専用 130
 - リスト 164
 - Q.RESOURCE_VIEW、管理プログラム
 - ラム出口 292
 - 視点特権
 - 視点の所有者の 131
 - 付与 130
 - 自動経路指定、出力の印刷 200
 - シフト文字 284
 - 終了メッセージ 377
 - 終了呼び出し、編集ルーチン 256
 - 出力域
 - 終了の制御 256
 - フォーマット設定の制御 251
 - 照会
 - 削除 158
 - 内部記憶フォーマット 155
 - 表示 157
 - リスト 156
 - 障害追及
 - 記憶域要件 366
 - 診断援助機能 366
 - パフォーマンス上の問題 365
 - 常駐の QMF プログラム 401
 - 初期化
 - 障害追及 359
 - パフォーマンス 401
 - メッセージ番号 368
 - 書式 247
 - 印刷 214
 - ウィンドウ ID 241
 - 新規編集コードの作成 248
 - 内部記憶フォーマット 155
 - 表示 157
 - リスト 156
 - NLF サポート 175
 - 所有権
 - 転送 157
 - QMF の追跡方法 153
 - 新機能モード 28, 53
 - 新機能モード (NFM) 11
 - 新機能モードへの移行 54
 - 診断
 - 援助 366
 - 症状 366
 - ダンプ 377
 - トレース機能 369
 - 数値データ変換、編集ルーチン 248
 - スクラッチパッド域
 - 管理プログラム出口ルーチン
 - 323
 - スタンプ、QMF プログラム 376
 - 図表
 - 印刷 199, 214
 - 特定のオブジェクト 214
 - フォーマット 152
 - 制御セクション (CSECT)、診断 384
 - 整数データ、編集ルーチン 248
 - 静的照会 128
 - セッション 113
 - 接続機能、CICS/DB2 UDB (OS/390 版) 用の 39
 - 線形プロシージャ、コマンド同義語内の 223
 - 選択されたグローバル変数の初期デフォルト値のオーバーライド 111
 - 選択メニュー、ISPF の 34
 - 全データベース・インストール
 - QMF の開始 33
 - 前提知識、DB2 UDB (OS/390 版) に関する 7
 - 操作環境 15
 - ソフトウェア要件 15
 - ソフトウェア・サポート機能 (SSF) 381
- ## [夕行]
- ターゲット・ライブラリー
 - 内容 19
 - 必要な DASD スペース、
 - NLF 178
 - タイムアウト、QMF トランザクション
 - メッセージ表示の定義 377
 - CICS 領域サイズ 366
 - タイム・シェアリング・オプション (TSO)
 - パフォーマンス 88
 - 対話モード
 - IVP 193
 - 短整数データ、編集ルーチン 248
 - ダンプ、診断のための 377
 - ダンプ・データ・セット 32
 - 端末
 - GDDM ニックネーム 201
 - 端末モニター・プログラム TMP 74
 - 端末モニター・プログラム (TMP) 29
 - 調整、QMF および CICS のための GDDM の 20
 - 調整、QMF の Workstation Database Server 用
 - QMF NLF サンプル表の削除 189
 - QMF NLF の削除 189
 - QMF NLF の作成
 - コントロール表 189
 - サンプル表 189
 - 通知メッセージ 387

データベース
接続
権限 113
リモート 73
専用のインストール
NLF の 180, 181
低いパフォーマンス 365
データベース権限、保守 131
データ・オブジェクト
SAVE DATA に関する特権 136
データ・セット
管理
概要 162
テスト、QMF の 60
デフォルト
GDDM モジュール
ADMADFT 207
デフォルトの QMF プロファイル
115
デフォルトのシステム初期化プロシ
ージャ 109
動的照会 128
トグル・スイッチ、管理プログラム
出口ルーチン 291
特記事項 415
特権 121
データベース・オブジェクト
135
特権、DB2
静的照会 128
他のユーザーの付与の取り消し
134
動的照会 128
配布
参照：照会の GRANT
参照：照会の REVOKE
表エディター 128
QMF コマンド 128
SAVE/IMPORT コマンド 132
STATS および REORG 161
参照：表特権
トランザクション
MRO および ISC による経路指定
要求 403
取り消しサービス、管理プログラム
324

トレース
機能
停止 376
詳細レベル 371
データ
表示 373
メッセージ・ログ 360
トレース機能 391
トレース・データ・セット
割り振り 369

[ナ行]

名前
ADMMNICK 指定 202
ニックネーム
印刷時のエラー 361
定義済み 201
複数のプリンターの定義 205
入力域
終了の制御 256
フォーマット設定の制御 251

[ハ行]

ハードウェアとプログラムの要件
NLF の 177
ハードウェア要件 15
配布ライブラリー
内容 19
必要な DASD スペース、
NLF 178
必要なディスク記憶域 178
パスワード 36
バッチ
インストール検査プロシージャ
NLF 193, 195
TSO の実行 63
TSO のための設定 36
照会またはプロシージャの実行
215
予備ファイルの使用 94
CSD の更新 42
IVP の実行 63
IVP の実行、NLF 195
IVP のための設定 36

バッチ (続き)
NLF 193
バッファ・プール、コントロール
表の割り当て 164
パネル
印刷および表示サポート 402
管理プログラム・プロンプト
287
クラス ID 236
ID 240, 241
パフォーマンス
常駐プログラム 401
表索引 148
予備ファイルの使用 95
CICS (顧客情報管理システ
ム) 88
DSQSIROW、大きい値 97
DSQSIROW、小さい値 96
TSO (タイム・シェアリング・オ
プション) 88
パフォーマンス強化 17
パラメーター
受け渡し 87
編集ルーチンへ渡す 251
非コミット読み取り 138
ビット編集コード 365
非同期処理の印刷
199
表
アクセスの制御 135
印刷 214
機能キー 231
索引 148
削除 165
保守 164
リスト 164
リソース制御、管理プログラム出
口 291
ロック 138
表エディター
必要な SQL 特権 136
ADD および CHANGE 128
表スペース 149
暗黙作成表スペース 149
拡大 159
削除 123
タイプ 150

- 表スペース (続き)
 - 表の作成 147
 - 明示 / 暗黙オプション
 - CREATE TABLE 照会 133
 - SAVE および IMPORT コマンド 149
 - 明示作成表スペース 149, 150
 - ユーザー・プロファイル内の指定 116
 - 割り当て 148, 149
 - 表特権
 - 概要 129
 - ファミリー 1 プリンター 202, 203
 - ファミリー 2 プリンター 202, 203
 - ファミリー 3 プリンター 202, 204, 205
 - ファミリー 4 プリンター 205
 - フォールバック
 - 定義 412
 - 複数のリリース 30
 - 複数領域オプション (MRO) 19
 - 浮動小数点データ、編集ルーチン 248
 - プラン ID 30, 33
 - プランチ・アドレス、管理プログラム 309
 - プリンター
 - 制御キーワード (PRINTCTL) 206
 - ダンプ用の設定値 32
 - ニックネーム 201
 - ニックネーム (GDDM) 390
 - マルチアドレス 202
 - ANSI サポート
 - グラフィック装置 200
 - プリンターを定義するマクロ 208
 - フルスクリーン・パネル
 - カスタマイズした機能キーの例 238
 - パネル ID 240
 - プログラム製品 15
 - 必須、NLF 177
 - プログラム・パラメーター
 - 要約 102
 - DSQSDBUG 97
 - DSQSIROW 95
 - プロシージャ
 - 印刷 214
 - オブジェクトの保守 155
 - コマンド同義語で使用 223
 - 内部記憶フォーマット 155
 - 表示 157
 - リスト 156
 - プロセッサ時間
 - 限界の設定 287
 - 制御使用 289
 - プロファイル
 - 印刷 214
 - 更新 121, 122
 - コマンド同義語 226
 - 削除 115, 123
 - 作成 113
 - 初期化の検索順序 121
 - デフォルト値 116
 - 複数の (NLF) 115
 - 保守 153
 - CASE の設定、カスタマイズした機能キー 238
 - SET PROFILE コマンド 121
 - プロンプト・パネル
 - カスタマイズした機能キーの例 240
 - パネル ID 242
 - 分散作業単位
 - サポート 4, 6
 - QMF の設定 9
 - 分散データ 4, 7
 - 分離レベル
 - カーソル固定 138
 - 非コミット読み取り 138
 - ヘルプ
 - パネル機能キーのカスタマイズ 242
 - ヘルプ・パネル
 - カスタマイズした機能キーの例 240
 - ヘルプ・パネル・テスト、IVP 実行時の 61
 - ヘルプ・パネル・テスト、IVP の実行 59
- 変換
 - 管理プログラム出口ルーチン 329
 - 編集
 - コード
 - 数値データ処理 248
 - プロファイルの CASE フィールド 249
 - 出口インターフェース 247
 - 終了呼び出し 256
 - 出力域 254, 256
 - 制御ブロック・フィールド 252
 - 入力域 254
 - 出口ルーチン 252
 - ルーチン 247
 - DBCS データ 283
 - 編集ルーチン
 - さまざまなコードの処理 254
 - 変数
 - 同義語定義内 224
 - &ALL の使用 224
 - ホーム・パネル 60
 - IVP 実行時 60
 - 報告書
 - 印刷 214
 - データ・フォーマット 247
 - 低いパフォーマンス 365
 - 2 進データ 364
 - Q.ERROR_LOG 表 380
 - 報告書の表示 (DPRE) 216
 - 保守
 - オブジェクトの表示 157
 - オブジェクトのリスト 156
 - オブジェクト用の表スペースの拡大 159
 - コマンド同義語表 227
- [マ行]**
- マスター・アプリケーション・メニュー 34
 - 見積もり
 - SMP/E 記憶域 19
 - 無効なサブシステム ID 388
 - 無効な名前 proflex 388

明示作成表スペース 150
メッセージ
行限界の超過 288
ユーザー・アクティビティの取り
消し、管理プログラム 325
QMF メッセージ・サービス 367
問題報告 381

[ヤ行]

ユーザー
オブジェクト 156
オブジェクトに関する許可 135
新規追加 114
ユーザー編集ルーチン
さまざまなコードの処理 254
DATE データ 250
TIME データ 250
呼び出し可能インターフェース
一般エラー 388
プログラム・パラメーターの変更
34
QMF の開始 33
予備ファイル
計算例 93
サイズの見積もり 91
パフォーマンスの問題 90

[ラ行]

リクエスト・データベース・イン
ストール
ワークシート、QMF 181
QMF の開始 33
リストの視点
作成 145
リソース
グループ 115
所有権 113
プロファイル管理 115
リソース限定表 332
リテラル、コマンド同義語の 225
リモート作業単位
オブジェクトへのアクセス 143
コマンド同義語表の作成 75
サポート 4

リモート作業単位 (続き)
図 5
リモート・データベース接続のカ
スタマイズ 73
QMF の設定 8
SQL デフォルトの視点 142
リモート・サーバーでの互換モード
45
領域サイズ、TSO 用 29
リリース番号 381
リンク・エディットのステートメン
ト
管理プログラム出口ルーチン
328, 330
リンク・エディットのメッセージ
389
リンク・バック域 17
ロード・モジュールの配置 17, 30
ロケーション・ウィンドウ ID 242
ロング・ネーム 11
論理トランザクション、定義 4

[ワ行]

ワークシート、インストール用の
21
割り込み機能
使用 378

[数字]

10 進数データ、編集ルーチン 248
16 進編集コード 365
2 か国語使用のサポート
書式 175
2 バイト文字セット (DBCS) サポート
編集コード 283
ラテン文字 283
3 部分名 5
A
ABENDASRA 387
ABEND、QMF インストール時の
387

ADMADFC デフォルト・モジュール
207
ADMADFC、GDDM デフォルト・モ
ジュール 20
ADMCFORM DD 名 152
ADMMNICK 指定 203
AEY9 ABEND 388
APAR 19
APAR (プログラム診断依頼書) 376,
381
APPLDATA 列 155
ASCPUT サービス、印刷 200
AUTHID 42
AZTS ABEND 388

C

CASE 列 (Q.PROFILES) 116
CDS 司環境の作成 196
CEBR トランザクション 356
CICS
一般エラー 387
インストール検査プロシージャ
57, 59
仮想記憶域要件 18
管理テーブル
各国語機能 188
DCT (宛先管理テーブル) 41
管理プログラムへのインターフェ
ース 300
始動ジョブ 43
診断機能 377
調整 39, 45
各国語機能 187, 189
管理テーブル 41
始動ジョブ・ストリーム 43
図表およびトレース・ファイ
ルのインストール 39
データ・セットの定義および
ロード 39
複数領域オプション
(MRO) 19
マップのインストール 39
AUTHID を共用する TSO お
よび CICS 42

- CICS (続き)
 - 調整 (続き)
 - DB2 UDB (OS/390 版) から CICS への接続 39
 - GDDM 20
 - QMF プロファイル表 42
 - QMF/CICS テーブル項目の作成 41
 - トランザクション ID 388
 - トレース機能の使用 391
 - CICS への DB2 UDB (OS/390 版) のインストール 39
 - CSD 変更 42
 - ESA 188
 - IVP の実行
 - QMF の初期化 58
 - QMFE トランザクション 60
 - TYPETERM 項目、QMF 表示 377
 - V3 管理テーブル
 - DCT (宛先管理テーブル) 41
 - CICS 管理テーブル、NLF 188
 - CICS (顧客情報管理システム)
 - 一時記憶域キュー 211
 - 一時データ・キュー 211
 - パフォーマンス 88
 - ENVIRONMENT 値、QMF プロファイル 116
 - HANDLE CONDITION 305
 - IMPORT コマンド 132
 - QMF CMS コマンド
 - コマンド同義語 222
 - TYPETERM 項目、QMF 表示 377
 - CICS 用の NLF の調整
 - ジョブ DSQ1nCSD の調整および実行 189
 - DB2 RCT への NLF/QMF トランザクション ID の追加 187
 - IVP の実行 189
 - CICS/DB2 接続機構 3
 - CLIST
 - ログオン・プロシーチャーの代替 30
 - CONFIRM 列 (Q.PROFILES) 116
 - CONNECT コマンド 4
 - CONNECT コマンド (続き)
 - エラー 360
 - CREATE TABLE ステートメント
 - コマンド同義語 218
 - ユーザーの表 146
 - SAVE DATA に関する特権 136
 - CREATE ステートメント、SQL
 - CREATETS/CREATETAB 特権 132
 - CREATETAB 権限 151
 - CREATETS/CREATETAB 特権
 - 定義 132
 - CREATE TABLE 照会を実行するための特権 133
 - CREATOR 列 (Q.PROFILES)
 - 定義済み 116
 - プロファイル初期化の役割 121
 - CSD データ・セット 41
 - CSECT (制御セクション)、診断 384
- D**
- DATE 列
 - 参照： ユーザー編集ルーチン
- DB2
 - 管理プログラム 287
 - リソース限定表 332
- DB2 UDB (OS/390 版) (DB2 Universal Database (OS/390 版))
 - 許可、IVP の実行 57
 - 終結処置 66
- DB2 UDB (OS/390 版) 接続機能、CICS 用の 388
- DB2 許可
 - 作成
 - コマンド同義語表、NLF 183
- DB2 特権の除去
 - カスケード効果 134
 - 不完全な取り消し 128
 - PUBLIC に対する付与の取り消し 134
- DBADM 権限
 - データベース保守 131
- DBCS (2 バイト文字セット) サポート
 - 編集コード 283
- DBCS (2 バイト文字セット) サポート (続き)
 - ラテン文字 283
- DCT (宛先管理テーブル)
 - CICS V3 41
- DD 名 31
- DECOPT 列 (Q.PROFILES) 116
- DEVTOK キーワード、ADMMNICK 指定 203
- DISPLAY コマンドに必要な SQL 特権 135
- DOS プリンター 207
- DRAW コマンドに必要な SQL 特権 135
- DRDA AS 45
- DRDA (分散リレーショナル・データベース体系) 4
- DSNT302I 388
- DSQ0BINS 18
- DSQ0BSQL 18
- DSQ10297 388
- DSQ10493 388
- DSQ1EGLK 387
- DSQ1EINV 29, 36
- DSQ1ELNK 387
- DSQ1VSTP 389
- DSQ36805 389
- DSQCBST 17
- DSQCCI 17
- DSQCCISW 17
- DSQCEBLT 17
- DSQCELTT 17
- DSQCFR80 18
- DSQCI 18
- DSQCIA 17
- DSQCIB 17
- DSQCICX 17
- DSQCIF 17
- DSQCIFE 17
- DSQCIPL 17
- DSQCIPX 17
- DSQCIR 17
- DSQCIX 17
- DSQCMAPB 18
- DSQCSUB 17
- DSQCT080 18

DSQCTOPX 17
DSQDEBUG
調整 32
CICS で 391
DSQI0026 389
DSQI004I 389
DSQIRDBR 8
DSQPNLE
FCT 定義 387
VSAM CI サイズ 42
DSQQMF 17
DSQQMFE 17, 387
DSQSBSTG 87
DSQSCMDE 34
DSQSDBNM プログラム・パラメータ 5, 8
DSQSDBQN 99, 101
DSQSDBQT 98, 101
DSQSDBG 97, 100
DSQSIROW 95
DSQSPILL 90
DSQSPRID (プロファイル・キー)
QMF へのアクセスの制御 127
DSQSRSTG 88
DSQUCFRM ddname 152
DSQUDUMP 32
DSQUECIC 編集プログラム 250
DSQUEDIT 17
DSQUEGV1 18
DSQUEGV1 モジュール、管理プログラム出口 304
DSQUEGV3 18
DSQUEGV3 モジュール、管理プログラム出口 304
DSQUOPTS 111
DSQUXIA 17
DSQUXIC 17
DSQUXIP 17
DXEGOVA 制御ブロック 310
DXEXCBA 制御ブロック 316

E

EDIT TABLE コマンド
必要な SQL 特権 135
並行編集 138

ENQ コマンド
印刷キュー 211
ENTRY_TYPE 列 (機能キー表) 237
ENVIRONMENT 列 42
ENVIRONMENT 列 (Q.PROFILES)
プロファイル初期化の役割 121
EXECUTE 特権
アクセス、QMF アプリケーション・プランおよびパッケージ 126
EXPORT TABLE、SQL 特権 135

F

FCT (ファイル管理テーブル)
パネル・ファイルに合わせた調整 387
FMID
NLF 179
FSFRCE サービス、印刷 200

G

G050 ABEND 389
GDDM (図形データ表示管理プログラム) 402
印刷 200
エラー・メッセージ、印刷時 361
の使用 3
プリンターのニックネーム 201, 390
ADMMNICK 指定 202
ADMADFT デフォルト・モジュール 207
ADMF データ・セットへのマップの追加 188
QMF 始動時の一般エラー 387
GDDM-PGF 402
GRANT
オプション 129
要件 129
例 129
照会 128

H

HANDLE CONDITION
CICS 305
HEX 関数 364
HPO/ マネージャー 287

I

IBM ソフトウェア配布 (ISD) テープ
NLF
の内容 179
IDC055II 389
IDC3009I 389
IDC3012I 389
IEW0342 389
IEW0461 390
IMPORT TABLE コマンド
必要な SQL 特権 135
表の作成 147
ISC (システム間連絡) 403
iSeries での互換モード 48
ISPF (対話式システム生産性向上機能)
一般エラー 388
選択メニューのカスタマイズ 34
マスター・アプリケーション・メニュー 34
更新 186
ライブラリーの調整 31
ログオン・プロシーチャーの調整 30
ISPSTART コマンドの使用 33
IVP (インストール検査プロシーチャー) 59
CICS のもとの QMF のための 57
NLF のための実行 189
NLF のテスト 189
QMF 対話モード、基本 193
QMF 対話モードの
QMF 193
QMF バッチ・モードの 64
テストのオブジェクト 64

L

LENGTH 列 (Q.PROFILES) 116
LIST コマンド
ALL キーワード 157

M

MODEL 列 116, 155
MRO (複数領域操作) 403
MVS/ESA 3

N

NLF
英語サポート 175
コマンド同義語 223
リリース番号、ServiceLink 381
NLF (各国語機能)
記憶域要件 178
NLF のインストール 177
NLF の計画
ターゲット・ライブラリー 178
ハードウェア要件 177
配布ライブラリー 178
SMP/E 要件 177
NUMBER 列 (機能キー表) 238

O

OBJECT 列 (同義語表) 222
OBJECTLEVEL 列、QMF コントロール表 155
OWNER 列、QMF コントロール表 155

P

PANEL 列 (機能キー表) 236
PC プリンター 207
PFKEYS 列 (Q.PROFILES) 116
PF_SETTING 列 (機能キー表) 238
PLPA (ページング可能リンク・パッチ域) 16
PPT (処理プログラム・テーブル)
GDDM のカスタマイズ 387

PRINT TABLE コマンドに必要な
SQL 特権 135
PRINT コマンド
指名宛先への経路指定 199, 211
PRINTER 列 (Q.PROFILES) 116
PROCOPT パラメーター、印刷 206
PROFILE PREFIX ステートメント
339
PROG 759 390
PSP バケット 19
PTF 19
PUBLIC AT ALL LOCATIONS に対する付与 128
PUBLIC キーワード 125, 137
PUBLIC に対する付与
参照: 照会の GRANT

Q

Q ユーザー・プロファイル 113
QBE 照会
印刷 214
DB2 特権 128
SQL 特権 136
QMF
コマンド
QMF リリース間の互換性
409
コマンド接頭部としての 409
ユーザー・サポートの確立 113
QMF インストール・ユーザー出口
111
QMF 管理者 111
QMF コマンドの同義語 215, 220
アクティブ化の問題 221
引用符 225
オブジェクト名 220
構文 225
初期化メッセージ 360
同義語定義 222
同義語表の作成 218
表の保守 227
変数の使用 224
verb 220
QMF サンプル表の削除 47

QMF (照会報告書作成プログラム)
アプリケーション照会 62
概要 11
必要なオブジェクト 7
IVP
CICS 57
QMF タスクに必要な特権 135
QMF の TSO (タイム・シェアリング・オプション) 用調整
バッチ IVP を実行するためのバッチ・ジョブの設定 193
NLF 193
ISPF を使用した QMF の開始
186
NLF 186
ISPF を使用しない QMF の開始
187
NLF 187
NLF 用のログオン PROC に対する変更 186
QMF のアプリケーション・プラン
アクセス・タイプ 126
QMF の開始
一般エラー 387
表ロックの障害 138
ISPF を使用して 33, 186
NLF 186
ISPF を使用しない 35, 187
NLF 187
QMF プロファイルの初期化 121
QMF の概要 3
QMF の削除 47
QMF の並行バージョン 30
QMF バージョン 7 への移行 405
オブジェクト
同じ DB2 サブシステム 406
異なる DB2 サブシステム
406
オブジェクトの互換性
前のリリースでのバージョン
7 のオブジェクト 412
ステップの概要 405
前のオブジェクトの保護 412
前のリリースとのコマンドの互換性 413

QMF バージョン 7 への移行 (続き)
 CICS での呼び出し可能インター
 フェース 410
 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 410
 QMF へのアクセスの制御 126, 127
 QMF へのログオン
 イネープリング 113
 制限 115
 QMF 前のリリースへのフォールバッ
 ク 412
 QMFЕ トランザクション 60
 QUEUENAME、QUEUETYPE キーワ
 ード 211
 Q.COMMAND_SYNONYMS_n 表
 作成するためのジョブ 182
 Q.ERROR_LOG コントロール表
 380
 Q.PROFILES コントロール表 42
 更新 121
 表の構造 116
 ユーザー変更 121
 ユーザー・プロファイルの削除
 115
 ユーザー・プロファイルの追加
 114
 NLF に合わせた更新 182
 RESOURCE_GROUP フィールド
 の更新 289
 Q.RESOURCE_VIEW、管理プログラ
 ム 292

R

RACF
 バッチのセキュリティ 336
 RACF の考慮事項 36
 RCT (リソース管理テーブル) 388
 REMARKS 列 156
 RESTRICTED 列
 値を NO に変更 157
 定義済み 155
 REVOKE 照会
 PUBLIC に対する付与 134

REVOKE、照会の
 例 128
 REXX 34
 RLST 332
 RUN コマンド
 コマンド同義語 222
 必要な SQL 特権 135

S

SAVE および IMPORT コマンドの
 拡張
 明示表スペース 149
 DB2 特権
 何が必要かの判別 132
 付与 133
 SAVE コマンド
 拡張 149
 必要な SQL 特権 135
 DATA キーワード 135
 TABLE キーワード 135
 SCOPE リソース・オプション 291
 SDSQEXCE 31
 SDSQLOAD 17
 ロード・モジュールの配置 30
 SEQ 列 155
 ServiceLink 381
 SET PROFILE コマンド 121
 SMP/E (拡張システム修正変更プログ
 ラム)
 記憶域要件 19
 のフォーマット 179
 必要な DASD スペース 19
 NLF の記憶域要件 178
 SPACE 列 (Q.PROFILES) 116
 SQL
 ステートメント
 GRANT 137
 INSERT (新規のユーザー・プ
 ロファイル) 114
 UPDATE 122
 特権 114
 指示照会、QBE 照会に関する
 136
 QMF コマンドのための 135
 戻りコード 369

SQL (続き)
 HEX 関数 364
 ID 114
 コマンド同義語表 229
 Q 113
 QMF の保管方法 155
 SQLCODE 389
 SREL 179
 SUBTYPE 列、QMF コントロール表
 155
 SUSPEND キーワード 211
 SYNONYMS 列 (Q.PROFILES) 116
 SYNONYM_DEFINITION 列 222
 SYSADM 権限
 アプリケーション・プランへのア
 クセスの取り消し 127
 データベースの保守 132
 REVOKE、照会の 134
 SYSOUT
 印刷 211
 SYSTEM プロファイル
 削除 115
 デフォルト値の変更 122
 SYSUDUMP 32

T

TCT (端末管理テーブル)、プリンタ
 ーの定義 208
 TERMINAL フィールド、CICS
 TCT 203
 TIME データ
 参照：ユーザー編集ルーチン
 TIMESTAMP データ
 参照：ユーザー編集ルーチン
 TOFAM キーワード、ADMMNICK
 指定 203
 TONAME キーワード、ADMMNICK
 指定 203
 TRACE 列 (Q.PROFILES) 116
 TRANSLATION 列
 (Q.PROFILES) 116
 TRMIDNT フィールド、CICS
 TCT 203

TSO (タイム・シェアリング・オペレーション)

仮想記憶域 87

管理プログラムへのインターフェース 300

パフォーマンス 88

割り込み機能 378

TSO (タイム・シェアリング・オペレーション) 用の調整

バッチ IVP を実行するためのバッチ・ジョブの設定 36

必要な領域サイズ 29

ISPF マスター・アプリケーション・メニュー 35

ISPF を使用した QMF の開始 33

ISPF を使用しない QMF の開始 35

QMF が TSO を使用する方法 3

QMF 用ログオン・プロシージャ

要件 29

TYPE 列、QMF コントロール表 155

TYPETERM 指定 378

TYPETERM、QMF 表示用 377

U

U 編集コード、形式

定義済み 248

入力域 254

UDN 編集コード 251

V

V 編集コード、書式

定義済み 248

入力域 254

VERB 列 (同義語表) 220

VSAM データ・セット

索引と表スペース用に使用される 162

VSS 編集コード 251

W

WIDTH 列 (Q.PROFILES) 116



プログラム番号: 5625-DB2

Printed in Japan

GC88-9832-00



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