

DB2 照会報告書作成プログラム
バージョン 12 リリース 1

TSO/CICS 用 DB2 QMF
インストールおよび
管理の手引き

IBM

DB2 照会報告書作成プログラム
バージョン 12 リリース 1

TSO/CICS 用 DB2 QMF
インストールおよび
管理の手引き

IBM

お願い

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

本書は、IBM DB2 12 for z/OS (5650-DB2) および IBM DB2 11 for z/OS (5615-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) Classic Edition および Enterprise Edition バージョン 12 リリース 1 に適用されます。また、スタンドアロン IBM DB2 for z/OS ツールである IBM DB2 QMF for z/OS (5697-QM2) バージョン 12 リリース 1 にも適用されます。この情報は、新しい版で明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

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

原典： GC27-8877-00
DB2 Query Management Facility
Version 12 Release 1
Installing and Managing DB2 QMF for
TSO and CICS

発行： 日本アイ・ビー・エム株式会社

担当： トランスレーション・サービス・センター

© Copyright IBM Corporation 1982, 2016.

© Rocket Software, Inc. 2013, 2016. All rights reserved.

目次

| | |
|-----------------|-----|
| 本書について | x1 |
| 前提知識 | xi |
| サービス更新およびサポート情報 | xii |
| 強調表示規則 | xii |
| 構文図の読み方 | xii |
| ご意見の送付方法 | xiv |

第 1 部 QMF for TSO/CICS のインストール 1

第 1 章 構成の計画 3

| | |
|------------------------------|----|
| サポートされる構成 | 3 |
| QMF バージョン 12.1 をサポートするデータベース | 3 |
| インストール・モード | 4 |
| リリースの共存 | 5 |
| 分散データ・ネットワーク内の QMF | 5 |
| リモート作業単位 | 6 |
| 分散作業単位 | 10 |

第 2 章 QMF インストールの概要およびロードマップ 15

| | |
|--|----|
| QMF インストール・プロセスの概要 | 15 |
| 一般的なインストールの概念 | 16 |
| QMF のインストールおよび管理に必要な権限 | 17 |
| QMF のインストールに必要な権限 | 17 |
| QMF 管理に必要な権限 | 18 |
| ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ) | 20 |
| インストール・パス A: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースへの QMF V12.1 のインストール | 23 |
| インストール・パス B: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースでの QMF V11.2、V11.1、V10 または QMF NFM リリースから QMF V12.1 への移行 | 24 |
| ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール | 26 |
| インストール・パス C: DB2 for z/OS、iSeries、または LUW サーバーへの QMF V12.1 のインストール | 30 |
| インストール・パス D: DB2 for z/OS、iSeries、または LUW サーバーでの QMF V11.2、QMF V11.1、QMF V10、または QMF NFM リリースから QMF V12.1 への移行 | 33 |
| インストール・パス E: DB2 (VSE および VM 版) サーバーでの QMF V7.2 以前から V12.1 への移行 | 35 |
| インストール・パス F: DB2 (VSE および VM 版) サーバーでの QMF V8.1、V9.1、V10.1、V11.1、または V11.2 から QMF V12.1 への移行 | 37 |
| ロードマップ 3: 3 部構成の名前によってアクセスされるサーバー・データベースのインストール | 39 |
| インストール・パス G: 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーの準備 | 40 |
| ロードマップ 4: 英語以外の言語でアクセスされるデータベース | 41 |
| QMF を除く DB2 for z/OS のアップグレード時に実行する作業 | 43 |

第 3 章 リクエスター (DB2 for z/OS) のデータベースへの QMF のインストールまたは移行 45

| | |
|---|----|
| リクエスター (DB2 for z/OS) データベースへの QMF のインストールの準備 | 45 |
| リクエスター (DB2 for z/OS) データベースのインストール前提条件 | 45 |
| 配布メディアからの QMF バージョン 12.1 ライブラリーのコピー | 47 |
| VSAM パネル・ライブラリーのデータの設定 | 50 |
| 記憶域のアドレッシング要件 | 50 |

| | |
|---|------------|
| パフォーマンスを向上させるためのモジュールの移動 | 52 |
| リクエスターのインストール用にインストール・ジョブ・パラメーターを設定する方法 | 54 |
| リクエスター (DB2 for z/OS) データベース用インストール・ジョブの実行 | 59 |
| 旧リリースがインストールされていない場合に QMF V12.1 をインストールするジョブ | 60 |
| QMF V8 NFM、QMF V9 NFM、QMF V10、QMF V11.1、または QMF V11.2 を QMF V12.1 に移行するジョブ | 61 |
| TSO および CICS に対するプログラム、リソースおよびプリファレンスの定義 | 64 |
| TSO でのリクエスター・インストールのカスタマイズ | 64 |
| CICS でのリクエスター・インストールのカスタマイズ | 73 |
| 第 4 章 サーバー・データベースでの QMF のインストールまたは移行 | 81 |
| サーバー・インストールのためのインストール・ジョブ・パラメーターの設定 | 81 |
| QMF CONNECT コマンドによりアクセスされるサーバーを準備するインストール・ジョブの実行 | 83 |
| 旧リリースがインストールされていないサーバー・データベースに QMF V12.1 をインストールするジョブ | 83 |
| z/OS、iSeries、または LUW サーバーで QMF V8 NFM、QMF V9 NFM、QMF V10、QMF V11.1、または QMF V11.2 を QMF V12.1 に移行するジョブ | 85 |
| QMF V7.2 以前のバージョンを VM または VSE サーバーの QMF V12.1 に移行するジョブ | 87 |
| VM または VSE サーバーで QMF V8、V9、V10、V11.1、または V11.2 を QMF V12.1 に移行するジョブ | 89 |
| サーバー・データベースでの旧リリースへのフォールバック | 90 |
| 3 部構成の名前を含む QMF コマンドによりアクセスされるサーバーを準備するインストール・ジョブの実行 | 91 |
| 第 5 章 QMF の正しいインストールの検査 | 93 |
| リクエスター・データベースへの QMF インストールのテスト | 93 |
| QMF の開始 | 93 |
| QMF パネル・ライブラリーの正しいバージョンの確認 | 98 |
| QMF ライブラリーが適切に割り振られていることの確認 | 100 |
| 必要な QMF 視点がすべてインストールされていることの確認 | 101 |
| インストール検査手順に必要なサンプル照会およびサンプル・プロシージャのインストール | 102 |
| インストール検査手順の実行 | 104 |
| サーバー・データベースでの QMF インストールのテスト | 109 |
| 第 6 章 移行およびフォールバックの考慮事項 | 113 |
| QMF の旧リリースのプロファイルおよびオブジェクトへのアクセス | 113 |
| 別のデータベースにある QMF プロファイルへのアクセス | 113 |
| 表、視点、および QMF オブジェクトの移行 | 114 |
| リリース間のオブジェクトの互換性 | 115 |
| QMF バージョン 12 リリース 1 と旧リリースとの前方互換性 | 115 |
| QMF バージョン 12.1 と旧リリースとの互換性 | 116 |
| 2 GB 境界より上の記憶域の要件に関する変更 | 117 |
| 第 7 章 QMF の旧リリースの削除 | 119 |
| DB2 for z/OS データベースからの QMF の削除 | 119 |
| 新旧のリリースが同じサブシステムにある場合 | 119 |
| 新旧のリリースが異なるサブシステムにある場合 | 123 |
| DB2 for iSeries または DB2 for Linux, UNIX, and Windows データベースからの QMF の削除 | 124 |
| 新旧のリリースが同じリモート・サーバーにある場合 | 125 |
| 新旧のリリースが異なるサーバーにある場合 | 125 |
| DB2 (VSE および VM 版) データベースからの QMF の削除 | 127 |
| 新旧のリリースが同じ DB2 (VSE および VM 版) サーバーにある場合 | 127 |
| 新旧のリリースが異なるサーバーにある場合 | 127 |
| 第 8 章 オプション機能のインストール | 129 |
| QMF Analytics for TSO の構成 | 129 |
| QMF Analytics for TSO のロード・ライブラリーに関する考慮事項 | 129 |
| QMF Analytics for TSO で使用されるファイルの割り振り | 129 |
| QMF Analytics for TSO 用のファイル・システムのカスタマイズ | 130 |

| | |
|---|-----|
| QMF Analytics for TSO サンプル表のインストール | 131 |
| QMF Analytics for TSO が正しくインストールされているかを確認 | 132 |
| QMF Data Service (QDS) の構成 | 134 |
| QMF 各国語機能のインストール | 134 |
| QMF 各国語機能の移行 | 142 |
| QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ) | 143 |
| 拡張 LIST コマンド機能のインストール (z/OS のみ) | 147 |
| I QMF Data Service インターフェースの構成 | 150 |

第 2 部 QMF for TSO/CICS の管理 153

第 9 章 QMF の開始 157

| | |
|---|-----|
| TSO での QMF の開始 | 157 |
| 必要なファイルとリソースの割り振り | 157 |
| QMF の CLIST および EXEC ライブラリーを TSO へ追加 | 157 |
| TSO CALL コマンドを使用した QMF の開始 | 158 |
| DSQQMFE モジュールを使用した QMF の直接開始 | 158 |
| バッチ環境での QMF の開始 | 159 |
| QMF の DB2 for z/OS ストアード・プロシージャとしての開始 | 160 |
| ISPF 上での QMF の開始 | 167 |
| ISPF メニューからの QMF の開始 | 167 |
| QMF プログラム・ライブラリーを割り振る LIBDEF ステートメントの使用 | 168 |
| ISPF でのバッチ・モードによる QMF の開始 | 171 |
| ISPF のもとでの QMF の開始の例 | 172 |
| ネイティブ z/OS バッチでの QMF の開始 | 173 |
| CICS での QMF の開始 | 174 |
| QMF 開始時の大域変数および QMF セッション動作の初期化 | 175 |
| DSQUOPTS ルーチンによる大域変数の設定 | 176 |
| 大域変数表を使用した大域変数の設定 | 179 |
| デフォルト・システム初期化プロシージャによる初期化 | 182 |
| 独自の初期化プロシージャの作成 | 184 |

第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定 187

| | |
|---|-----|
| プログラム・パラメーターの要約 | 187 |
| データベースおよび環境のパラメーターの設定 | 195 |
| TSO で QMF を開始する DB2 (z/OS 版) サブシステムの名前の指定 | 195 |
| 初期データベース接続の指定 | 196 |
| QMF アプリケーション・プラン名の指定 | 197 |
| TSO で QMF プロファイル・キーとして使用する ID の指定 | 197 |
| 報告書用記憶域の定義 | 198 |
| 固定量の報告書用仮想記憶域の定義 | 199 |
| 可変量の報告書用仮想記憶域の定義 | 199 |
| 仮想記憶域で不要になったデータ用の追加記憶域の取得 | 200 |
| フェッチおよび挿入操作のパフォーマンスの制御 | 209 |
| 報告書待ち時間の制御 | 209 |
| 複数行のフェッチと挿入のサポートの使用可能化 | 211 |
| 複数のデータベース・スレッドの使用 | 212 |
| QMF アクティビティの自動化 | 213 |
| 操作モードの指定 (対話式またはバッチ) | 214 |
| QMF の開始時に実行する初期プロシージャの指定 | 215 |
| トレース・オプションの設定 | 221 |
| TSO のトレースの設定 | 221 |
| CICS のトレースの設定 | 222 |
| 非 DBCS ディスプレイ装置からの DBCS データの印刷 | 223 |

| | |
|---|------------|
| 第 11 章 ユーザーの登録と特権の設定 | 225 |
| アプリケーション・プランおよびパッケージに対するアクセスの制御 | 225 |
| 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 | 225 |
| 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取り消し | 225 |
| QMF ユーザー・プロファイルの作成 | 226 |
| Q.PROFILES 表について | 226 |
| サイトのためのプロファイル構造の確立 | 233 |
| ユーザー・プロファイルの追加 | 234 |
| ユーザー・プロファイルの更新 | 236 |
| ユーザー・プロファイルの削除 | 237 |
| 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付与 | 238 |
| QMF コマンドおよび機能に必要な特権 | 238 |
| 特権の付与と取り消し | 240 |
| オブジェクトの標準の設定および非コミット読み取りの許可 | 243 |
| 他のユーザーとの QMF オブジェクトの共用 | 243 |
| 非コミット読み取りの許可 | 244 |
| ユーザーのオブジェクト・リスト | 245 |
| ユーザーのオブジェクト・リストのカスタマイズ | 245 |
| QMF LIST および DESCRIBE コマンドのデフォルトの動作 | 247 |
| オブジェクト・リストの記憶域要件 | 251 |
| | |
| 第 12 章 データベース内のオブジェクトの作成と保守 | 253 |
| 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 | 253 |
| 表を作成するための手順 | 253 |
| SAVE DATA および IMPORT コマンド用の表スペースの割り当て | 254 |
| ユーザーへの表の作成権限の付与 | 256 |
| 機密データをフィルターするための視点の使用 | 257 |
| 視点の作成 | 257 |
| 視点の特権の付与 | 258 |
| QMF オブジェクト・カタログの保守 | 259 |
| Q.OBJECT_DIRECTORY 表の構造 | 259 |
| Q.OBJECT_DATA 表の構造 | 263 |
| Q.OBJECT_REMARKS 表の構造 | 264 |
| QMF オブジェクト・カタログ用の表スペースの拡大 | 264 |
| QMF オブジェクトのリスト | 267 |
| QMF オブジェクトの表示 | 267 |
| QMF オブジェクトの所有権の移譲 | 268 |
| 廃止された QMF オブジェクトの削除 | 269 |
| z/OS データ・セットからの照会、書式、およびプロシージャのインポート | 269 |
| DB2 for z/OS サブシステムの保守 | 269 |
| データ・セットの管理 | 270 |
| QMF コントロール表の保守 | 271 |
| | |
| 第 13 章 印刷および図表作成機能のセットアップ | 273 |
| 印刷に QMF または GDDM のいずれのサービスを使用するか | 273 |
| GDDM サービスを使用した印刷の処理 | 274 |
| QMF が GDDM ニックネームとインターフェースをとる方法 | 274 |
| GDDM がニックネームを検索する場所 | 275 |
| さまざまなプリンター・ファミリーのニックネームの例 | 275 |
| 特定プリンターのニックネーム定義の例 | 277 |
| 印刷を処理するための GDDM サービスのセットアップ | 278 |
| 印刷を処理するための QMF サービスの使用 | 283 |
| ネイティブ z/OS バッチ、TSO、および ISPF で印刷するための QMF サービス | 283 |
| CICS の印刷での QMF サービスの使用 | 284 |
| ユーザーが QMF を終了することなく印刷できるようにする | 285 |
| オブジェクト・タイプごとの印刷の要件 | 287 |

| | |
|---|------------|
| 図表作成機能の有効化 | 288 |
| TSO および ISPF での図表サポートの有効化 | 288 |
| CICS での図表サポートの有効化 | 289 |
| 第 14 章 コマンド同義語 | 291 |
| QMF に用意されているデフォルト同義語の使用 | 291 |
| デフォルト同義語のリスト | 291 |
| DPRE 同義語: ISPF を使用した印刷報告書のプレビュー | 292 |
| 同義語のガイドライン | 294 |
| 同義語 verb | 294 |
| 同義語オブジェクト名 | 295 |
| 同義語定義 | 295 |
| コマンド同義語のカスタマイズ | 299 |
| コマンド同義語表の作成 | 299 |
| コマンド同義語定義を表を入力 | 301 |
| 同義語のアクティブ化 | 301 |
| コマンド同義語表の保守の最小化 | 302 |
| 1 つの同義語表を全ユーザーに割り当てる | 303 |
| 同義語表の視点を個々のユーザーに割り当てる | 303 |
| 第 15 章 カスタム・ファンクション・キー | 307 |
| QMF ファンクション・キーのカスタマイズ | 307 |
| パネル ID の表示 | 307 |
| カスタマイズしたいキーの選択 | 311 |
| ファンクション・キー表の作成 | 313 |
| ユーザーのファンクション・キー定義を表を入力 | 314 |
| 新規ファンクション・キー定義のアクティブ化 | 316 |
| ファンクション・キー表に関する問題の検査と診断 | 317 |
| キー定義の例 | 318 |
| 第 16 章 QMF 書式のためのカスタム編集出口ルーチン | 321 |
| 編集出口ルーチンおよび QMF | 321 |
| 出口ルーチンとの間で受け渡されるフィールド | 323 |
| インターフェース制御ブロックのフィールド | 323 |
| 入出力域を特徴付けるフィールド | 325 |
| 編集コードの選択 | 326 |
| 2 バイト文字セット・データおよび編集ルーチン | 327 |
| DBCS データ、および編集ルーチンが受け取るもの | 328 |
| 編集ルーチンは必ず正しい結果を戻すこと | 329 |
| 日付、時刻、タイム・スタンプ・データ、および編集ルーチン | 329 |
| 日付、時間、およびタイム・スタンプ情報で必要な形式 | 330 |
| TSO での日時データ用 DB2 出口 | 331 |
| プログラミング言語の編集ルーチン | 332 |
| 編集ルーチンをハイレベル・アセンブラで作成 | 332 |
| 編集ルーチンを PL/I で作成 | 339 |
| 編集ルーチンを COBOL で作成 | 346 |
| 第 17 章 QMF リソースの制御 | 357 |
| QMF for TSO/CICS に付属のデフォルト管理プログラム出口ルーチン | 357 |
| 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 | 357 |
| デフォルト管理プログラム出口によるリソース制限 | 361 |
| 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 | 368 |
| QMF と管理プログラムとの対話方法 | 369 |
| 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 | 370 |
| デフォルトの管理プログラム出口ルーチンの変更または独自のルーチンの作成 | 379 |
| リソース制御情報を管理プログラム出口に渡す | 380 |

| | |
|---|------------|
| QMF セッション中のリソース制御情報の保管 | 394 |
| 取り消されたアクティビティーに対するメッセージ | 394 |
| 管理プログラム出口ルーチンの変換、アSEMBル、リンク・エディット | 396 |
| ユーザー・アクティビティーの取り消し | 399 |
| DB2 リソース限定機能の使用 | 400 |
| 管理プログラム間の相違点 | 401 |
| DB2 管理プログラムでモニター可能な QMF コマンド | 401 |
| DB2 管理プログラムにより照会が取り消された場合の QMF の対応方法 | 403 |
| QMF コマンドの DB2 管理の構成 | 403 |
| 第 18 章 バッチ・モードでの QMF の実行 | 407 |
| z/OS での QMF のバッチ・プログラムとしての実行 | 407 |
| バッチ・モードで操作する権限 | 407 |
| QMF バッチ・ジョブを実行するための JCL | 408 |
| QMF バッチのネイティブ z/OS での実行 | 411 |
| TSO での QMF バッチの実行 | 413 |
| QMF BATCH コマンドを使用した ISPF での QMF バッチの実行 | 414 |
| ISPF または TSO のフォアグラウンドでの QMF バッチ・ジョブの開始 | 424 |
| バッチ・モードのプロシージャーまたはアプリケーションのデバッグ | 424 |
| リモート DB2 クライアントからの QMF バッチ・ジョブの開始 | 425 |
| QMF の CICS でのバッチ・トランザクションとしての実行 | 426 |
| CICS での端末からのバッチ・アクティビティーの実行 | 427 |
| CICS での端末を使用しないバッチ・アクティビティーの実行 | 427 |
| CICS でのプロシージャーのデバッグ | 427 |
| 第 19 章 問題のトラブルシューティングと診断 | 429 |
| QMF サービスの適用 | 429 |
| 一般的な問題の訂正 | 430 |
| 初期化時に発生する可能性があるエラー | 430 |
| QMF の開始後の警告メッセージ | 441 |
| 正しくない出力 | 442 |
| 印刷の問題 | 442 |
| 表示エラー | 445 |
| I 記憶域関連問題の解決 | 445 |
| QMF パフォーマンスの管理 | 445 |
| 動的ステートメントの EXPLAIN 情報の取り込み | 446 |
| QMF 照会を照会加速に適格にする | 446 |
| 記憶域の問題の解決 | 447 |
| リソース競合の問題の解決 | 449 |
| I DSQC_BUFFER_SIZE 大域変数設定による QMF パフォーマンスの向上 | 449 |
| 診断援助機能の使用 | 449 |
| メッセージ・サポートを使用した問題の診断 | 450 |
| 診断情報を取り込むための割り込みの作成 | 452 |
| トレース機能 | 454 |
| エラー・ログ報告書の検査 | 463 |
| 環境に固有の診断の使用 | 464 |
| IBM への問題報告 | 466 |
| 既に報告されている問題の検索 | 467 |
| IBM ソフトウェア・サポートでの作業 | 467 |
| 第 3 部 付録 | 469 |
| 付録 A. 旧リリースでの変更の要約 | 471 |
| バージョン 11.2 での変更点 | 471 |
| バージョン 11.1 での変更点 | 474 |
| バージョン 10.1 での変更点 | 481 |

| | |
|---|------------|
| バージョン 9.1 での変更点 | 485 |
| バージョン 8.1 での変更点 | 487 |
| バージョン 7.2 での変更点 | 488 |
| バージョン 7.1 での変更点 | 490 |
| バージョン 6.1 での変更点 | 491 |
| 付録 B. 特定のサポートを必要とする QMF 機能 | 493 |
| データベース・タイプによって異なる機能 | 493 |
| CICS で使用できない機能 | 494 |
| 付録 C. DB2 に常駐している QMF オブジェクト | 497 |
| QMF プラン | 497 |
| QMF パッケージ | 497 |
| TSO および CICS での QMF コントロール表および表スペース | 497 |
| QMF 視点 | 499 |
| DB2 for z/OS 記憶域グループ | 500 |
| LOB データを保管するためのスペース | 500 |
| TSO/CICS 用の VSAM クラスタ | 501 |
| QMF サンプル表 | 501 |
| 付録 D. 外部エディター | 503 |
| エディターを使用した QMF 照会およびプロシージャの変更 | 503 |
| QMF 報告書のドキュメントへの挿入 | 505 |
| アプリケーションの変更 | 505 |
| ドキュメント・インターフェース・マクロ (DSQAED1P) の名前変更 | 505 |
| Q.DSQAED1S プロシージャをデータベースに入れる | 506 |
| データ・コンポーネントの変更 | 506 |
| CLIST およびマクロの変更 | 508 |
| 付録 E. QMF ユーザー定義関数 | 513 |
| QMF ユーザー定義関数およびプロシージャ | 513 |
| APPL_AUTHNAMES | 514 |
| CALL DSQABA1E | 515 |
| DSQABA1E | 516 |
| 付録 F. QMF および GDDM プログラムの CICS への定義の方法 | 517 |
| QMF プログラムの CICS への定義の方法 | 517 |
| QMF のインストール時における GDDM 定義のロード方法 | 518 |
| CICS でのリソース使用を制御するためのトランザクション経路指定 | 519 |
| 特記事項 | 521 |
| プログラミング・インターフェース情報 | 522 |
| 商標 | 523 |
| 用語集 | 525 |
| 索引 | 541 |

本書について

IBM® DB2® 照会報告書作成プログラム (QMF) for TSO/CICS® は、堅固に統合された強力で信頼性の高いツールであり、照会および報告書作成の機能を提供し、以下のリレーショナル・データベースのデータへのアクセスおよびデータの提示を支援します。

- DB2 for z/OS®
- DB2 for Linux, UNIX, and Windows
- DB2 for iSeries
- DB2 Server (VSE および VM 版)

これらのトピックは、次の作業においてデータベース管理者およびシステム・プログラマーに役立つように作成されています。

- 新規 QMF™ のインストールの計画と実行、または以前のリリースから現行の QMF リリースへのマイグレーション
- DB2 for z/OS アプリケーション・リクエストでの QMF の開始
- ユーザー用 QMF 環境のカスタマイズと管理
- リソース管理機能を確立し、ユーザー定義のデータ形式設定コードをサポートする出口ルーチンの作成
- バッチ・モードでの QMF の実行
- 問題のトラブルシューティングと診断

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。これらのコンポーネントのリストと説明を以下に示します。

- z/OS オペレーティング・システム。
- タイム・シェアリング・オプション (TSO)。QMF とその関連製品をサポートする環境です。
- 対話式システム生産性向上機能 (ISPF)。QMF 用のダイアログ管理機能です。
- 顧客情報管理システム (CICS)。汎用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 により、QMF と z/OS 間のインターフェースが提供されます。
- 基本の図形データ表示管理プログラム (GDDM) 製品。QMF でパネルの表示を可能にします。基本の GDDM 製品は、報告書やその他のオブジェクトを印刷するためにも使用できます。GDDM-PGF は図表の作成に必要です。
- DB2 (z/OS 版)。QMF 用のデータベース・マネージャーです。
- SMP/E (システム修正変更プログラム拡張版)。最初にインストールされた QMF プログラムを実行する DB2 (z/OS 版) ターゲット・システムに、配布メディアから QMF のコンテンツをロードするために使用するツールです。

- ハイレベル・アセンブラー (HLASM)。デフォルト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出口ルーチンの作成に必要です。HLASM は、QMF 書式用に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。
- COBOL および PL/I。QMF 書式用にユーザー独自の編集コードを作成する場合に使用できます。
- REXX。QMF をインストールする EXEC を作成する場合に使用します。
- WLM。単一の z/OS イメージ内、または複数のイメージ間で同時に複数のワークロードを実行することを可能にする、z/OS のコンポーネント。

これらのトピックのタスクを開始する前に、QMF の基本機能に習熟しておく必要があります。QMF の概念と機能についての詳細は、以下の情報を参照してください。

- DB2 QMF 入門, GC27-8876-00
- DB2 QMF 使用の手引き, SC27-8879-00
- DB2 QMF 解説書, SC27-8880-00

サービス更新およびサポート情報

サービス更新およびサポート情報 (ソフトウェア・フィックスパック、PTF、よくある質問 (FAQ)、技術情報、トラブルシューティング情報、およびダウンロードなど) を検索するには、次の Web ページを参照してください。

IBM ソフトウェア・サポート Web サイト

強調表示規則

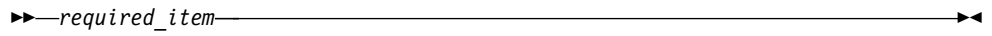
本書では、以下のような強調表示の規則を使用しています。

- 太字体タイプは、コマンドまたはユーザー・インターフェース・コントロール (フィールド名、フォルダー名、アイコン名、メニュー選択名など) を示します。
- モノスペース・タイプは、ユーザーが示されたとおり正確に入力するテキストの例を示します。
- イタリック は、他の資料の表題、または重要な用語の強調を示します。ユーザーが値を置き換える必要のある変数を示すためにも使用されます。

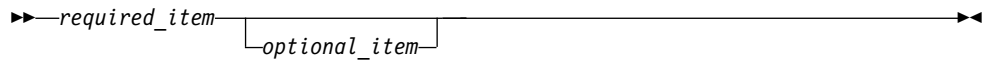
構文図の読み方

本書で使用されている構文図には、以下の規則が適用されます。

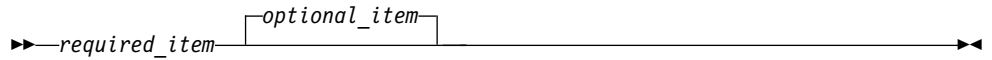
- 構文図は線の経路に沿って、左から右、上から下へ読んでいきます。以下の規則が使用されます。
 - >>--- 記号は、構文図の始まりを示します。
 - ---> 記号は、構文図が次の行に続くことを示します。
 - >--- 記号は、構文図が前の行から続いていることを示します。
 - ---<< 記号は、構文図の終わりを示します。
- 必須項目は、水平線 (メインパス) 上に表示されます。



- オプション項目は、メインパスの下に表示されます。

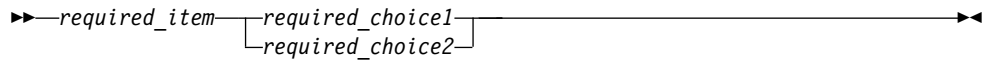


オプション項目がメインパスより上に表示されている場合、その項目は構文エレメントの実行には影響を与えず、読みやすさのためにのみ使用されています。

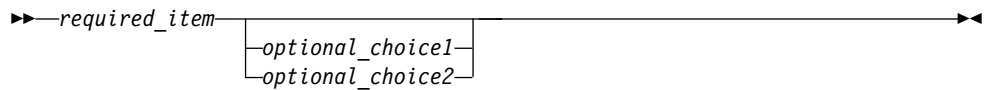


- 複数の項目から選択できる場合、項目は縦に積み重ねて表示されます。

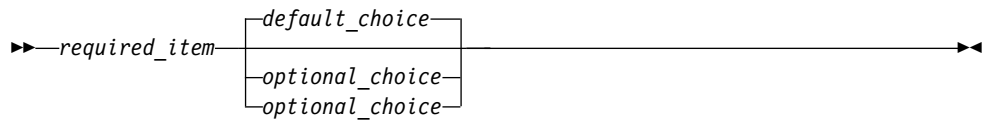
項目の 1 つを選択することが必須である場合は、積み重ねられた項目の 1 つがメインパス上に表示されます。



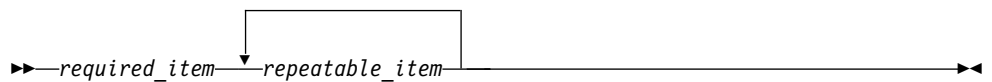
項目の 1 つを選択することが任意である場合は、積み重ねられた項目全体がメインパスより下に表示されます。



項目の 1 つがデフォルトである場合、その項目はメインパスより上に表示され、残りの項目は下に表示されます。



- 主線より上の、左側に戻る矢印は、繰り返し可能な項目を示します。



繰り返し矢印にコンマが含まれている場合は、繰り返される項目をコンマで区切る必要があります。



積み重ねられた項目より上にある繰り返し矢印は、積み重ねの中の項目が繰り返し可能であることを示します。

- キーワードとその最短の省略形 (該当する場合) は、大文字で表示されます。これらは、示されているとおり正確にスペルを入力する必要があります。変数はすべて小文字のイタリック体 (例えば *column-name*) で表示されます。これらは、ユーザー指定の名前または値を表します。
- 構文図でキーワードとパラメーターの間に句読点がない場合は、キーワードとパラメーターを 1 つ以上のスペースで区切ってください。
- 句読記号、括弧、算術演算子、およびその他の記号は、構文図に示されているとおり正確に入力してください。
- 脚注は、例えば (1) のように、括弧内の番号で示されます。

ご意見の送付方法

IBM お客様のご意見をお寄せください。本マニュアルまたは他の資料に関するご意見やご感想は、次の方法でお送りください。

- 次の場所にあるオンラインの読者ご意見フォームを使用する。

<http://www.ibm.com/software/data/rcf>

- ご意見を E メールで comments@us.ibm.com に送信する。本書の表題、本書のパーツ・ナンバー、製品のバージョン、およびご意見の対象となるテキストの位置 (該当する場合。例えばページ番号や表番号など) を記入してください。

第 1 部 QMF for TSO/CICS のインストール

| | | | |
|--|-----------|--|-----------|
| 第 1 章 構成の計画 | 3 | インストール・パス E: DB2 (VSE および VM 版) サーバーでの QMF V7.2 以前から V12.1 への移行 | 35 |
| サポートされる構成 | 3 | インストール・パス F: DB2 (VSE および VM 版) サーバーでの QMF V8.1、V9.1、V10.1、V11.1、または V11.2 から QMF V12.1 への移行 | 37 |
| QMF バージョン 12.1 をサポートするデータベース | 3 | ロードマップ 3: 3 部構成の名前によってアクセスされるサーバー・データベースのインストール | 39 |
| インストール・モード | 4 | インストール・パス G: 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーの準備 | 40 |
| リリースの共存 | 5 | ロードマップ 4: 英語以外の言語でアクセスされるデータベース | 41 |
| 分散データ・ネットワーク内の QMF | 5 | QMF を除く DB2 for z/OS のアップグレード時に実行する作業 | 43 |
| リモート作業単位 | 6 | 第 3 章 リクエスター (DB2 for z/OS) のデータベースへの QMF のインストールまたは移行 | 45 |
| リモート作業単位構成での接続方法 | 6 | リクエスター (DB2 for z/OS) データベースへの QMF のインストールの準備 | 45 |
| QMF CONNECT コマンドを使用した場合のアクセス対象 | 8 | リクエスター (DB2 for z/OS) データベースのインストール前提条件 | 45 |
| リモート作業単位アクセス用に QMF をインストールする必要がある場所 | 8 | 配布メディアからの QMF バージョン 12.1 ライブラリーのコピー | 47 |
| リモート作業単位構成のロードマップ | 8 | VSAM パネル・ライブラリーのデータの設定 | 50 |
| リモート作業単位の構成例 | 9 | 記憶域のアドレッシング要件 | 50 |
| 分散作業単位 | 10 | 初期化中にモジュールをロードするために必要な記憶域 | 51 |
| 分散作業単位構成での接続方法 | 10 | 報告書操作のための仮想記憶域要件 | 51 |
| 3 部構成の名前を使用した場合のアクセス対象 | 11 | パフォーマンスを向上させるためのモジュールの移動 | 52 |
| 分散作業単位アクセス用に QMF をインストールする必要がある場所 | 11 | リクエスターのインストール用にインストール・ジョブ・パラメーターを設定する方法 | 54 |
| 分散作業単位構成のロードマップ | 12 | 共通パラメーターのインストール・デフォルト | 54 |
| 分散作業単位の構成例 | 12 | リクエスター・インストール用のサイト固有のインストール・デフォルトを EXEC で設定 | 56 |
| 第 2 章 QMF インストールの概要およびロードマップ | 15 | インストール・ジョブのデフォルトのオーバーライド | 58 |
| QMF インストール・プロセスの概要 | 15 | リクエスター (DB2 for z/OS) データベース用インストール・ジョブの実行 | 59 |
| 一般的なインストールの概念 | 16 | 旧リリースがインストールされていない場合に QMF V12.1 をインストールするジョブ | 60 |
| QMF のインストールおよび管理に必要な権限 | 17 | QMF V8 NFM、QMF V9 NFM、QMF V10、QMF V11.1、または QMF V11.2 を QMF V12.1 に移行するジョブ | 61 |
| QMF のインストールに必要な権限 | 17 | TSO および CICS に対するプログラム、リソースおよびプリファレンスの定義 | 64 |
| QMF 管理に必要な権限 | 18 | TSO でのリクエスター・インストールのカスタマイズ | 64 |
| ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ) | 20 | GDDM 外部デフォルトのカスタマイズ | 64 |
| インストール・パス A: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースへの QMF V12.1 のインストール | 23 | | |
| インストール・パス B: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースでの QMF V11.2、V11.1、V10 または QMF NFM リリースから QMF V12.1 への移行 | 24 | | |
| ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール | 26 | | |
| インストール・パス C: DB2 for z/OS、iSeries、または LUW サーバーへの QMF V12.1 のインストール | 30 | | |
| インストール・パス D: DB2 for z/OS、iSeries、または LUW サーバーでの QMF V11.2、QMF V11.1、QMF V10、または QMF NFM リリースから QMF V12.1 への移行 | 33 | | |

| | | | |
|---|-----|--|-----|
| QMF デバイスが照会可能であることの確認 | 66 | サーバー・データベースでの QMF インストールの | |
| TSO ログオン・プロシーチャーの準備 | 66 | テスト | 109 |
| CICS でのリクエスト・インストールのカスタマイズ | 73 | 第 6 章 移行およびフォールバックの考慮事項 | 113 |
| DB2 に対して QMF を記述 | 73 | QMF の旧リリースのプロファイルおよびオブジェクトへのアクセス | 113 |
| CICS に対して QMF を記述 | 74 | 別のデータベースにある QMF プロファイルへのアクセス | 113 |
| CICS 用に管理プログラム出口を準備する方法 | 74 | 表、視点、および QMF オブジェクトの移行 | 114 |
| CICS での GDDM サポートのカスタマイズ | 75 | リリース間のオブジェクトの互換性 | 115 |
| CICS 始動ジョブ・ストリームの更新 | 78 | QMF バージョン 12 リリース 1 と旧リリースとの前方互換性 | 115 |
| EXPORT コマンドおよび IMPORT コマンドで使用する記憶域タイプの判別 | 78 | QMF バージョン 12.1 と旧リリースとの互換性 | 116 |
| 最大 2 MB の SQL 照会を保管できるように EDSA の制限サイズを修正する | 79 | 2 GB 境界より上の記憶域の要件に関する変更 | 117 |
| 第 4 章 サーバー・データベースでの QMF のインストールまたは移行 | 81 | 第 7 章 QMF の旧リリースの削除 | 119 |
| サーバー・インストールのためのインストール・ジョブ・パラメーターの設定 | 81 | DB2 for z/OS データベースからの QMF の削除 | 119 |
| QMF CONNECT コマンドによりアクセスされるサーバーを準備するインストール・ジョブの実行 | 83 | 新旧のリリースが同じサブシステムにある場合 | 119 |
| 旧リリースがインストールされていないサーバー・データベースに QMF V12.1 をインストールするジョブ | 83 | 旧アプリケーション・プランの解放 | 120 |
| z/OS、iSeries、または LUW サーバーで QMF V8 NFM、QMF V9 NFM、QMF V10、QMF V11.1、または QMF V11.2 を QMF V12.1 に移行するジョブ | 85 | SMP/E を使用して旧リリースのライブラリーを削除する | 121 |
| QMF V7.2 以前のバージョンを VM または VSE サーバーの QMF V12.1 に移行するジョブ | 87 | 旧リリースからのパッケージの除去 | 122 |
| VM または VSE サーバーで QMF V8、V9、V10、V11.1、または V11.2 を QMF V12.1 に移行するジョブ | 89 | 新旧のリリースが異なるサブシステムにある場合 | 123 |
| サーバー・データベースでの旧リリースへのフォールバック | 90 | DB2 for iSeries または DB2 for Linux, UNIX, and Windows データベースからの QMF の削除 | 124 |
| 3 部構成の名前を含む QMF コマンドによりアクセスされるサーバーを準備するインストール・ジョブの実行 | 91 | 新旧のリリースが同じリモート・サーバーにある場合 | 125 |
| 第 5 章 QMF の正しいインストールの検査 | 93 | 新旧のリリースが異なるサーバーにある場合 | 125 |
| リクエスト・データベースへの QMF インストールのテスト | 93 | QMF コントロール表およびパッケージの削除 | 126 |
| QMF の開始 | 93 | QMF サンプル表の削除 | 126 |
| TSO での QMF の開始 | 94 | DB2 (VSE および VM 版) データベースからの QMF の削除 | 127 |
| CICS での QMF の開始 | 98 | 新旧のリリースが同じ DB2 (VSE および VM 版) サーバーにある場合 | 127 |
| QMF パネル・ライブラリーの正しいバージョンの確認 | 98 | 新旧のリリースが異なるサーバーにある場合 | 127 |
| QMF ライブラリーが適切に割り振られていることの確認 | 100 | 第 8 章 オプション機能のインストール | 129 |
| 必要な QMF 視点がすべてインストールされていることの確認 | 101 | QMF Analytics for TSO の構成 | 129 |
| インストール検査手順に必要なサンプル照会およびサンプル・プロシーチャーのインストール | 102 | QMF Analytics for TSO のロード・ライブラリーに関する考慮事項 | 129 |
| インストール検査手順の実行 | 104 | QMF Analytics for TSO で使用されるファイルの割り振り | 129 |
| TSO 用 IVP の実行 | 105 | QMF Analytics for TSO 用のファイル・システムのカスタマイズ | 130 |
| CICS 用 IVP の実行 | 108 | QMF Analytics for TSO サンプル表のインストール | 131 |
| | | QMF Analytics for TSO が正しくインストールされているかを検査 | 132 |
| | | QMF Data Service (QDS) の構成 | 134 |
| | | QMF 各国語機能のインストール | 134 |
| | | QMF 各国語機能の移行 | 142 |
| | | QMF ストアード・プロシーチャー・インターフェースのインストール (TSO のみ) | 143 |
| | | 拡張 LIST コマンド機能のインストール (z/OS のみ) | 147 |
| | | QMF Data Service インターフェースの構成 | 150 |

第 1 章 構成の計画

QMF のインストール方法は、旧リリースの QMF がデータベースにインストールされているかどうかによって異なります。また、インストールは、QMF のターゲット・データベースが分散ネットワーク内でどのように機能するかによって異なります。

QMF が既に存在する場合、既存のリリースからの移行がサポートされているかどうかを調べる必要があります。QMF が存在していない場合、このデータベース・タイプで新規インストールがサポートされているかどうかを調べる必要があります。

また、この製品のインストール先となるデータベースが分散データ構成内でどのように機能するかについて考慮する必要があります。このデータベースはスタンドアロン (他のデータベースとの接続は不要)、リクエスターとして機能、サーバーとして機能、あるいは両方の役割で機能します。

これらのトピックを読んでから、インストール・ロードマップに進みます。各ロードマップでは、QMF 構成、インストールまたは移行のタイプ、および環境に適したインストール・パスへ導くための、一連の質問が出されます。

関連概念:

15 ページの『第 2 章 QMF インストールの概要およびロードマップ』
QMF はさまざまなデータベースにインストールでき、またさまざまなバージョンから QMF バージョン 12.1 に移行できるため、インストール・プロセスおよび移行プロセスをガイドするためのロードマップが用意されています。

17 ページの『QMF のインストールおよび管理に必要な権限』
QMF のインストールおよび一般的な QMF 管理には、特定の権限が必要です。

サポートされる構成

このトピックでは、QMF バージョン 12.1 のインストール、または QMF バージョン 12.1 への移行をサポートするデータベースをリストしてあります。また、旧インストール・モードからの移行についての情報、および同一 DB2 サブシステム内のバージョン 12.1 と共存可能な QMF のバージョンについての情報が記載されています。

QMF バージョン 12.1 をサポートするデータベース

各 DB2 リリースでは、QMF バージョン 12.1 の特定のインストールがサポートされています。

下の表は、データベース・タイプに応じて、QMF バージョン 12.1 でサポートされるインストールを示しています。

表に示されていない QMF バージョン 12.1 と DB2 for z/OS リリースとの互換性については、<http://www.ibm.com/support/docview.wss?uid=swg21409518> を参照してください。場合によっては、QMF が適切に動作するために QMF PTF の適用が必要になることがあります。

表 1. QMF バージョン 12.1 のサポートされるインストール

| 最低限のデータベース・リリース | 新規インストール (QMF の旧リリースは存在しない) | QMF 8.1 NFM、QMF 9.1 NFM、および QMF 10.1 以降 からの移行 |
|--|--------------------------------|---|
| 以下のいずれかが該当します。 <ul style="list-style-type: none"> • APAR PM45482 が適用された、新機能モードで稼働する DB2 for z/OS バージョン 9.1 (VUE および MLC) • APAR PM50434 および PM72274 が適用された、すべてのモード (CM8、CM8*、ENFM8、および ENFM8* を除く) で稼働する DB2 for z/OS バージョン 10.1 (VUE および MLC) • DB2 for z/OS バージョン 11.1 (VUE および MLC) • DB2 for z/OS V12.1 (VUE および MLC) | X | X |
| DB2 for iSeries バージョン 5.4 | X | X |
| DB2 for Linux, UNIX, and Windows バージョン 9.5 | X | X |
| DB2 Server (VSE および VM 版) バージョン 7.3 | | X |

インストール・モード

QMF バージョン 10 より前には、互換モードおよび新機能モードの 2 つのインストール・モードがサポートされていました。

バージョン 10 より前に QMF でサポートされていた 2 つのインストール・モードには以下の特性がありました。

- QMF 互換モードの場合、QMF オブジェクト・カタログは、最大 8 文字の許可 ID と最大 18 文字の QMF オブジェクト名をサポートしていました。
- QMF 新機能モードの場合、QMF オブジェクト・カタログは、最大 128 文字の許可 ID と最大 128 文字の QMF オブジェクト名をサポートしていました。

QMF オブジェクト・カタログは、Q.OBJECT_DIRECTORY、Q.OBJECT_DATA、および Q.OBJECT_REMARKS のコントロール表で構成されています。これら 3 つの表には、QMF の照会、書式、プロシージャ、分析、およびフォルダー・オブジェクトに関する情報が保管されています。

QMF バージョン 10 から、2 つの異なるインストール・モードはサポートされなくなりました。QMF バージョン 12.1 のオブジェクト・カタログでは、QMF バージョン 10、QMF バージョン 9 新機能モード、および QMF バージョン 8 新機能モードでサポートされていたものと同じ長さの許可 ID およびオブジェクト名がサポートされます。カタログの列の長さにおけるこの差異に従って、同一データベースで共存可能な QMF のリリースが決まります。

QMF バージョン 11 リリース 2 は、互換モードの QMF リリースからの移行がサポートされる最後のリリースです。QMF バージョン 12 リリース 1 では、QMF

バージョン 9.1 互換モード、QMF バージョン 8.1 互換モード、QMF バージョン 7.2、またはそれより前のリリースからの移行はサポートされていません。QMF のこれらのバージョンのいずれかを使用している場合は、QMF バージョン 12 リリース 1 に移行する前に、新機能モードに移行する必要があります。

関連資料:

『リリースの共存』

QMF バージョン 12.1 は、QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モードや QMF バージョン 10.1、QMF バージョン 11.1、または QMF バージョン 11.2 とのみ、同じデータベース内に共存できます。

リリースの共存

QMF バージョン 12.1 は、QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モードや QMF バージョン 10.1、QMF バージョン 11.1、または QMF バージョン 11.2 とのみ、同じデータベース内に共存できます。

QMF バージョン 12.1 の共存の規則は、QMF カタログでサポートされる許可 ID と QMF オブジェクト名の長さの違いによるものです。下の表に、サポートされる長さの違いを示します。

表 2. QMF バージョン 12.1 と前のリリースにおいて QMF オブジェクト・カタログでサポートされる許可 ID と QMF オブジェクト名の長さ

| QMF のリリース | QMF カタログでサポートされる許可 ID の最大の長さ | QMF カタログでサポートされるオブジェクト名の最大の長さ |
|--|------------------------------|-------------------------------|
| <ul style="list-style-type: none"> • QMF バージョン 12.1 • QMF バージョン 11.2 • QMF バージョン 11.1 • QMF バージョン 10.1 • QMF バージョン 9.1 新機能モード • QMF バージョン 8.1 新機能モード | 128 | 128 |
| <ul style="list-style-type: none"> • QMF バージョン 9.1 互換モード • QMF バージョン 8.1 互換モード • QMF バージョン 7.2 以前 | 8 | 18 |

分散データ・ネットワーク内の QMF

QMF の最初のインストールは、DB2 for z/OS データベースに対して行う必要があります。このインストールが完了すると、DB2 for z/OS データベースは、スタンドアロンとして使用するか、リクエスターとして機能するか、あるいは他の QMF バージョン 12.1 インストール済み環境のリクエスターとサーバーの両方として機能します。

DB2 for z/OS のもとで QMF をインストールしてリクエスターとして機能させる場合は、これを使用して以下のリモート・データベース・サーバー・タイプのいずれからでもデータにアクセスできます。

- 別の DB2 for z/OS サブシステム

- DB2 for Linux, UNIX, and Windows
- DB2 for iSeries
- DB2 (VSE および VM 版)

QMF では、リモート作業単位 および分散作業単位 の 2 つのリモート・データ・アクセス方式がサポートされます。どちらのタイプのアクセス方式も、作業単位 の定義に基づいています。この作業単位が 1 つの論理的なトランザクションとなります。論理的な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います。このシーケンスでは、操作のすべてが正常に実行されるか、そのシーケンス全体が失敗とみなされるかのいずれかになります。例えば、COMMIT ステートメントを含まない複数ステートメント QMF SQL 照会内のすべての SQL ステートメントは、単一の作業単位のもとで実行されます。

重要: リモート・データ・アクセスでは両方のタイプとも、リクエスターとサーバー・データベース間に DRDA[®] 通信を定義して、運用する必要があります。

制約事項: データベースによって、データ・タイプおよび各種機能のサポート・レベルは異なります。リモート・データベースで表または視点を表示する際に、オブジェクトがそのデータベースでサポートされないデータ・タイプを含んでいるか参照している場合、SAVE や EXPORT などの一部の操作はサポートされないことがあります。

関連情報:

 [DB2 資料](#)

DRDA のセットアップおよびテストに関する情報を検索します。

リモート作業単位

QMF では、リモート作業単位 のリモート・データ・アクセスがサポートされています。リモート作業単位は、分散リレーショナル・データベース処理の 1 つの形式であり、そこではアプリケーション・プログラムは、1 つの作業単位内でリモート・データベース上のデータにアクセスできます。

リモート作業単位には、複数のリレーショナル・データベース要求を含めることができますが、すべての要求は同一のリモート・データベースを対象にする必要があります。データベースに対するすべての要求は、要求が他のデータベースに送信される前に、完了させる (コミットまたはロールバックさせる) 必要があります。

QMF がリモート作業単位構成に関係する場合、DB2 for z/OS データベース内の各インストール環境は、リクエスターとしてのみ、またはリクエスターとサーバーの両方として、機能することができます。リモート作業単位構成では、DB2 for z/OS 以外のデータベース・タイプでの QMF インストール済み環境は、リクエスターとしてではなく、サーバーとしてのみ機能することができます。

リモート作業単位構成での接続方法

リモート作業単位データ・アクセス方式を使用して、2 つの方法のいずれかでサーバーに接続することができます。

以下のいずれかの方法で、リクエスターからサーバーに接続します。

- 確立された QMF セッション、プロシージャー、あるいはプログラム内から QMF CONNECT コマンドを発行します。
- QMF の始動時に、DSQSDBNM プログラム・パラメーターを指定します。このパラメーターでは、QMF の接続先となる最初のサーバーの名前を指定します。

DSQSDBNM パラメーターと DSQSSUBS パラメーターは、同時に指定する必要があります。QMF は、DSQSSUBS パラメーターで指定されたローカル・データベース・リクエスターに初めて接続します。その後、内部的に CONNECT コマンドを発行して、DSQSDBNM で指定されたサーバー・ロケーションに接続すると、QMF ホーム・パネルに表示されるようになります。

制約事項: QMF for TSO が、DB2 for z/OS リクエスター・データベースでストアード・プロシージャーとして開始された場合、リモート・サーバーとの接続はサポートされません。

下の図は、リモート作業単位構成内の QMF を示しています。

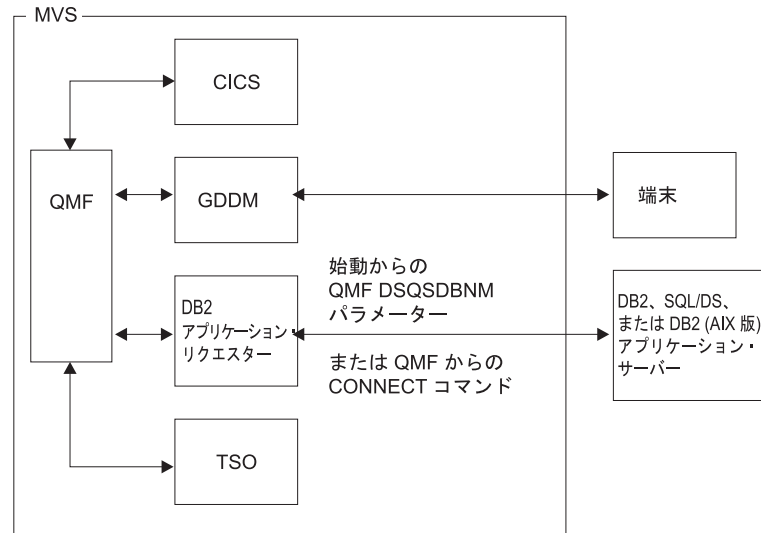


図 1. QMF リモート作業単位構成

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

QMF CONNECT コマンドを使用した場合のアクセス対象

QMF CONNECT コマンド、あるいは DSQSDBNM プログラム・パラメーターを使用してリモート・サーバーに接続すると、リモート・サーバー内の QMF オブジェクトとデータベース・オブジェクトの両方に、ローカルへのアクセス時と同じ方法でアクセス可能です。

DB2 (VSE および VM 版) データベース上の QMF カタログでは、許可 ID は最大 8 文字、QMF オブジェクト名は最大 18 文字です。その他すべてのデータベース・タイプ上での QMF バージョン 12.1 カタログでは、最大 128 文字の許可 ID と QMF オブジェクト名がサポートされます。CONNECT コマンドが DB2 (VSE および VM 版) データベースを参照する場合は、2 つのシステム間での QMF カタログの構造上の差異は許されます。ただし、DB2 (VSE および VM 版) システムでサポートされる長さに注意する必要があるのは、QMF の照会とプロシーチャーの実行、および QMF コマンドの発行を行う場合です。

リモート作業単位アクセス用に QMF をインストールする必要がある場所

QMF が QMF CONNECT コマンドまたは DSQSDBNM パラメーターを使用してリモート・サーバーに接続することができるようにするには、リクエスターとサーバーの両方に同じ QMF のリリース・レベルが存在している必要があります。

両方の場所に以下のオブジェクトを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 アプリケーション・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 (オプション)

QMF アプリケーション計画が必要なのは、リクエスター側のみです。

QMF コントロール表およびパッケージがリモート・サーバー上にあり、リクエスター上のもと同じリリース・レベルである場合は、接続処理が正常に行われ、接続の確立後もリモート・サーバー上の処理は続行されます。

関連資料:

497 ページの『付録 C. DB2 に常駐している QMF オブジェクト』
これらの QMF オブジェクトは、DB2 for z/OS サブシステム内で QMF バージョン 12.1 を実行する必要があります。必要に応じてリカバリー操作時にこの情報をガイドとして使用することができます。

リモート作業単位構成のロードマップ

QMF CONNECT コマンドを使用してリモート・サーバーに接続するよう QMF をセットアップするには、2 つの手順が必要です。

QMF CONNECT コマンドを使用して 1 つ以上のリモート・サーバーのデータにアクセスするために QMF をセットアップするには、以下の操作を行います。

1. 20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』に従って、リクエスター・データベースに QMF をインストールします。

DB2 for z/OS データベースのみリクエスターとして機能できます。リモート・サーバー上の QMF バージョン 12 リリース 1 にアクセスする必要があるリクエスターごとに、ロードマップ 1 を使用します。

2. 26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』に従って、各リモート・サーバー・データベース内に QMF をインストールします。これにより、リクエスター・データベースから発行される QMF CONNECT コマンドによる各サーバーへのアクセスが可能になります。

リモート・サーバーが、サーバーとリクエスターの両方の役割として機能する DB2 for z/OS データベースの場合、ロードマップ 1 のインストール・パスでは両方の役割で機能するデータベースが準備されます。

ロードマップ 2 に関連するインストール・パスをすべて完了すると、そのサーバーは QMF コマンド内の 3 部構成の名前によるアクセスにも対応可能となります。

制約事項: 3 部構成の名前を含む QMF コマンドは、DB2 (VSE および VM 版) サーバーには送信できません。

関連概念:

10 ページの『分散作業単位』

QMF では、リモート・データ・アクセスの分散作業単位 がサポートされています。分散作業単位は、分散リレーショナル・データベース処理の 1 つの形式であり、そこではユーザーまたはアプリケーション・プログラムは、1 つの作業単位内で複数のロケーションのデータを読み取ったり、更新することができます。

リモート作業単位の構成例

この例は、サーバーとリクエスターの両方が DB2 for z/OS データベースであるリモート作業単位構成を示しています。

この構成では、以下のアーキテクチャーが使用されます。

- z/OS オペレーティング・システム z/OS1 には、2 つの DB2 12 for z/OS 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 z/OS2 には、1 つの DB2 12 for z/OS サブシステム DB2C があります。このシステムはバッチ・システムであり、DB2C は、z/OS1 上の TSO ユーザーがアクセスできるアプリケーション・サーバーです。
- QMF は、アプリケーション・リクエスターとして DB2A にインストールする必要がある、かつアプリケーション・サーバーとして DB2B および DB2C にインストールする必要があります。DB2A への権限を持つユーザーは、異なる z/OS オペレーティング・システムにログインしなくても DB2B と DB2C に格納されたデータにアクセスできます。

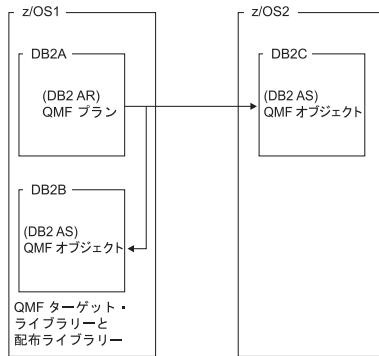


図 2. リモート作業単位構成の QMF。この場合、サーバーとリクエスターの両方が DB2 for z/OS です。

分散作業単位

QMF では、リモート・データ・アクセスの分散作業単位 がサポートされています。分散作業単位は、分散リレーショナル・データベース処理の 1 つの形式であり、ここではユーザーまたはアプリケーション・プログラムは、1 つの作業単位内で複数のロケーションのデータを読み取ったり、更新することができます。

1 つの作業単位の中では、1 つのシステムで稼働しているアプリケーションは、複数のリモート・データベース管理システムに対して、それらのシステムでサポートされる SQL を使用して SQL 要求を送信できます。例えば、小売店の在庫プログラムは、あるシステム上の在庫表の更新および別システム上の売掛金勘定表の更新を、1 つの作業単位内で行えます。

QMF が分散作業単位の構成の一員である場合、データ要求を開始するデータベースは常に DB2 for z/OS となります。理由は、そのデータベース上で QMF が稼働しているからです。DB2 for z/OS 以外のデータベース・タイプは、リクエスターとしてではなく、サーバーとしてのみ機能することができます。

関連概念:

5 ページの『分散データ・ネットワーク内の QMF』

QMF の最初のインストールは、DB2 for z/OS データベースに対して行う必要があります。このインストールが完了すると、DB2 for z/OS データベースは、スタンドアロンとして使用するか、リクエスターとして機能するか、あるいは他の QMF バージョン 12.1 インストール済み環境のリクエスターとサーバーの両方として機能します。

分散作業単位構成での接続方法

QMF は、QMF コマンドでの別名または 3 部構成の名前の使用を通じて、分散作業単位をサポートします。

3 部構成の名前は以下の書式です。

location_name.auth_ID.object_name

この構文では、

- *location_name* は、要求のターゲットとなるデータベースの名前を指定します。
- *auth_ID* は、オブジェクトを作成したユーザーの許可 ID を指定します。
- *object_name* は、アクセスする表または視点の名前を指定します。

制約事項:

- QMF for TSO が、DB2 for z/OS リクエスター・データベースでストアード・プロシージャーとして開始された場合、リモート・サーバーとの接続はサポートされません。
- 3 部構成の名前を含む QMF コマンドは、DB2 (VSE および VM 版) サーバーに送信することはできません。
- デフォルトでは、3 部構成の名前を含む QMF コマンドを使用して、ラージ・オブジェクト (LOB) データを含むリモート表にアクセスすることはできません。3 部構成の名前を持つリモート表内の LOB データへのアクセスを可能にするには、DSQEC_LOB_RETRV 大域変数を 3 に設定する必要があります。また、DSQEC_LOB_RETRV を 2 に設定すると、LOB メタデータのみを検索が可能になります。
- 3 部構成の名前を含むコマンドを DB2 for z/OS に送信する場合を除き、QMF は複数行フェッチをオフにして開始する必要があります。

関連概念:

211 ページの『複数行のフェッチと挿入のサポートの使用可能化』
DSQSMRFI パラメーター。データベースがフェッチおよび挿入の際に処理する行が複数行か、単一行かを制御します。

関連情報:

 [DB2 資料](#)

3 部構成の名前に関する情報を検索します。

3 部構成の名前を使用した場合のアクセス対象

QMF コマンドで 3 部構成の名前を使用すると、データベースの表および視点に対してのみアクセスすることができます。QMF オブジェクトにアクセスするには、リモート作業単位構成を使用します。

リモート・サーバー上の QMF オブジェクト (照会、書式、プロシージャー、分析、およびフォルダー・オブジェクト) にアクセスするには、最初に QMF CONNECT コマンドを使用してサーバー・データベースに接続し、次に要求を出す必要があります。QMF CONNECT コマンドには、リモート作業単位構成が必要です。

関連概念:

6 ページの『リモート作業単位』

QMF では、リモート作業単位 のリモート・データ・アクセスがサポートされています。リモート作業単位は、分散リレーショナル・データベース処理の 1 つの形式であり、そこではアプリケーション・プログラムは、1 つの作業単位内でリモート・データベース上のデータにアクセスできます。

分散作業単位アクセス用に **QMF** をインストールする必要がある場所
ユーザーが QMF コマンドで別名または 3 部構成の名前を使用できるようにするために、QMF を 2 つの方法でインストールできます。

ユーザーが QMF コマンドで別名や 3 部構成の名前を使用できるようにするには、以下のタスクのいずれか 1 つを実行する必要があります。

- リクエスターとサーバーの両方に、QMF を完全にインストールします。
- リクエスターには QMF を完全にインストールし、サーバーでは 3 部構成の名前のインストールを行います。

関連タスク:

91 ページの『3 部構成の名前を含む QMF コマンドによりアクセスされるサーバーを準備するインストール・ジョブの実行』

このジョブ・シーケンスは、QMF コマンドで 3 部構成の名前を使用してリモート・サーバーのデータにアクセスするように、リクエスターとサーバーの両方を準備します。

分散作業単位構成のロードマップ

分散作業単位方式用に QMF をセットアップする方法は、3 部構成の名前を含む QMF CONNECT コマンドを使用するかどうかによって異なります。

リモート・データ・アクセスに、3 部構成の名前を使用するが、QMF CONNECT コマンドを使用する予定はない場合は、以下の手順に従ってください。

1. 20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』に従って、リクエスター・データベースに QMF をインストールします。
2. 39 ページの『ロードマップ 3: 3 部構成の名前によってアクセスされるサーバー・データベースのインストール』に従って、QMF の 3 部構成の名前によるアクセス用にリモート・サーバーを準備します。

リモート・データ・アクセスに、3 部構成の名前に加えて QMF CONNECT コマンドも使用する場合は、以下の手順に従ってください。

1. 20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』に従って、リクエスター・データベースに QMF をインストールします。
2. 26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』に従って、リモート・サーバーに QMF をインストールします。ロードマップ 2 に関連付けられたインストール・パスでは、QMF CONNECT コマンドと 3 部構成の名前の両方でアクセスされるサーバーが準備されます。

制約事項: 3 部構成の名前を含む QMF コマンドは、DB2 (VSE および VM 版) サーバーに送信することはできません。

分散作業単位の構成例

この構成例は、DB2 for z/OS から DB2 for z/OS への分散作業単位接続を示しています。

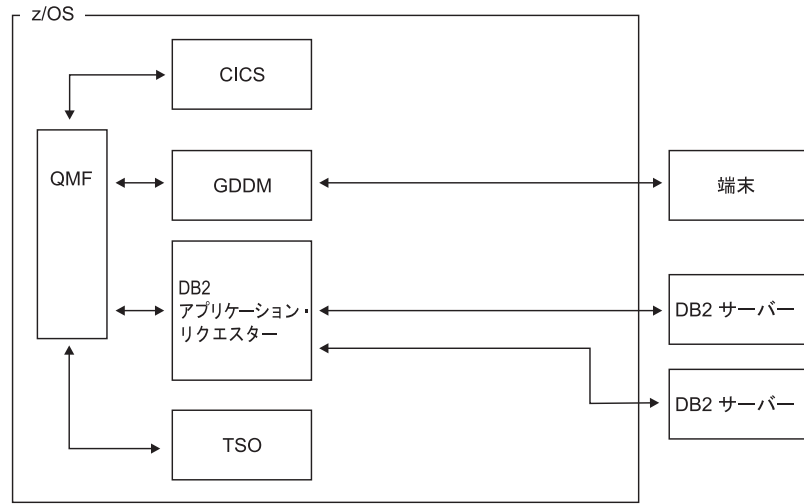


図 3. QMF 分散作業単位構成 (この図では、DB2 for z/OS から DB2 for z/OS への構成を示します)

第 2 章 QMF インストールの概要およびロードマップ

QMF はさまざまなデータベースにインストールでき、またさまざまなバージョンから QMF バージョン 12.1 に移行できるため、インストール・プロセスおよび移行プロセスをガイドするためのロードマップが用意されています。

例えば、手順、要件、および考慮事項は、QMF のインストール先がスタンドアロン・データベースであるか、QMF CONNECT コマンドによってアクセスされるサーバー・データベースであるか、または 3 部構成の名前によってアクセスされるサーバー・データベースであるかによって異なります。また、QMF が英語以外の言語のデータにアクセスできるようにすることも可能です。これらのオプションのそれぞれについて、ロードマップが用意されています。

重要: QMF バージョン 12.1 のインストール後にデータベースをアップグレードする場合は、いくつかのジョブを実行して、QMF の適切な運用を継続できるようにする必要があります。

QMF インストール・プロセスの概要

QMF のインストール・プロセスは、選択したデータベースへの QMF のインストールまたは移行、および使用するオプション機能のインストールから構成されます。

表 3. QMF インストール・プロセスの概要

| インストール・プロセスの手順 | この手順の詳細 |
|--|---|
| <p>1. 最初のデータベースでの QMF のインストールまたは移行</p> <p>データベースは DB2 for z/OS データベースである必要があります。</p> <p>QMF バージョン 12.1 は、以下のいずれかの構成にインストールできます。</p> <ul style="list-style-type: none">• スタンドアロン・データベース。 <p>この構成では、このデータベース内の QMF インストールは、他のデータベース内の QMF インストール済み環境にアクセスすることも、アクセスされることもありません。</p> <ul style="list-style-type: none">• 分散データ構成内のアプリケーション・リクエスター。 <p>この構成では、このデータベース内の QMF インストール済み環境は、リモート・サーバー・データベース内の他の QMF インストールからデータを要求します。</p> <ul style="list-style-type: none">• 分散データ構成内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・サーバーの両方。 <p>この構成では、このデータベース内の QMF インストールは、リモート・データベース内の他の QMF インストールにアクセスし、アクセスされます。</p> | <p>20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』のロードマップ 1 を参照してください。</p> |

表 3. QMF インストール・プロセスの概要 (続き)

| インストール・プロセスの手順 | この手順の詳細 |
|--|---|
| <p>2. 分散データ構成内の各追加データベースの準備</p> <p>分散データ構成での QMF のインストール場所とインストール方法は、データ・ネットワークの構成と、データへのアクセス計画によって異なります。</p> <ul style="list-style-type: none"> • QMF CONNECT コマンドを使用してリモート・サーバーのデータにアクセスする計画の場合は、ロードマップ 2 に従ってください。 • QMF CONNECT コマンドを使用してこのサーバーに接続する計画ではなく、3 部構成の名前を含む QMF コマンドをこのサーバーに送信して表および視点にアクセスする予定の場合は、ロードマップ 3 に従ってください。 | <p>以下の情報を参照してください。</p> <ul style="list-style-type: none"> • 26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』のロードマップ 2。 • 39 ページの『ロードマップ 3: 3 部構成の名前によってアクセスされるサーバー・データベースのインストール』のロードマップ 3。 |
| <p>3. オプション: QMF Analytics for TSO の構成</p> <p>QMF Analytics for TSO には、TSO 環境のユーザー用に追加の図表および統計分析機能が用意されています。</p> | <p>詳しくは、129 ページの『QMF Analytics for TSO の構成』を参照してください。</p> |
| <p>4. オプション: QMF 各国語機能のインストール</p> <p>各国語機能 (NLF) により、英語以外の言語で QMF を使用できるようになります。</p> | <p>41 ページの『ロードマップ 4: 英語以外の言語でアクセスされるデータベース』のロードマップ 4 を参照してください。</p> |
| <p>5. 必要に応じたオプション機能のインストール</p> <ul style="list-style-type: none"> • QMF for TSO は、DB2 for z/OS のストアード・プロシージャーとして開始できます。 • また、LIST コマンドの機能拡張をインストールすると、1 次および 2 次許可 ID の両方に許可された表および視点をリストできるようになります。この場合、これらのオブジェクトについて PUBLIC に特権を付与する必要がなくなります。 | <p>以下の情報を参照してください。</p> <ul style="list-style-type: none"> • 143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』 • 147 ページの『拡張 LIST コマンド機能のインストール (z/OS のみ)』 |

関連情報:



DB2 資料

3 部構成の名前に関する情報を検索します。

一般的なインストールの概念

QMF for TSO/CICS は DB2 for z/OS アプリケーションの 1 つであるため、DB2 for z/OS のインストール時と同じ概念の多くを理解する必要があります。

インストール・プロセスを開始する前に、以下の知識があることを確認してください。

- 情報に基づく意思決定を行うことができるように、ご使用の環境に合わせて QMF を構成するさまざまな方法についての理解が必要です。3 ページの『第 1 章 構成の計画』をお読みください。
- QMF のインストールおよび管理の許可要件についての理解が必要です。17 ページの『QMF のインストールおよび管理に必要な権限』をお読みください。
- リクエスター・データベースの要件についての理解が必要です。45 ページの『リクエスター (DB2 for z/OS) データベースのインストール前提条件』をお読みください。

- 分散データ構成の一部である各種データベースに QMF をインストールする場合は、リモート作業単位および分散作業単位に関する概念についての理解が必要です。5 ページの『分散データ・ネットワーク内の QMF』をお読みください。
- 以下の概念についての理解が必要です。
 - CREATE、INSERT、および GRANT の各 SQL ステートメント。特権または権限を PUBLIC に付与する意味についても理解する必要があります。
 - 用語アプリケーション計画、パッケージ、およびバインド。
 - データベース、表スペース、表、および視点。
 - データベース・セキュリティー。

関連概念:

5 ページの『分散データ・ネットワーク内の QMF』

QMF の最初のインストールは、DB2 for z/OS データベースに対して行う必要があります。このインストールが完了すると、DB2 for z/OS データベースは、スタンドアロンとして使用するか、リクエスターとして機能するか、あるいは他の QMF バージョン 12.1 インストール済み環境のリクエスターとサーバーの両方として機能します。

関連情報:

 DB2 資料

DB2 for z/OS のインストールに関連する用語および概念に関する情報を検索します。

QMF のインストールおよび管理に必要な権限

QMF のインストールおよび一般的な QMF 管理には、特定の権限が必要です。

関連概念:

45 ページの『リクエスター (DB2 for z/OS) データベースのインストール前提条件』

スタンドアロン・データベースまたはリクエスター・データベースとして機能する DB2 for z/OS データベースに QMF をインストールするには、ハードウェアおよびソフトウェアの要件を満たす必要があります。

3 ページの『第 1 章 構成の計画』

QMF のインストール方法は、旧リリースの QMF がデータベースにインストールされているかどうかによって異なります。また、インストールは、QMF のターゲット・データベースが分散ネットワーク内でどのように機能するかによって異なります。

QMF のインストールに必要な権限

QMF をインストールするデータベースに対して必要な権限を持っていることを確認してください。

以下の権限が必要です。

- DB2 for z/OS データベース上でのインストールには、SYSADM 権限が必要です。

お客様設置場所で RACF® セキュリティー・グループを使用し、そのサイトで 2 次許可 ID を使用して QMF をインストールする必要がある場合、その許可 ID は SYSADM 権限も保有している必要があります。DSQ1DEFS EXEC の SECAUTH 変数をこの ID で更新する必要があります。

- DB2 (Linux 版、UNIX 版、または Windows版) データベースへインストールする場合は、SYSADM 権限が必要です。
- iSeries データベースへインストールする場合は、*ALLOBJ 権限が必要です。
- VM または VSE データベースへインストールする場合は、DBA 権限が必要です。

QMF のインストールに使用されるユーザー ID とは無関係に、以下のリソースは「Q」許可 ID によって所有されます。

- すべての QMF コントロール表
- サンプルのプロシージャおよび照会
- サンプル表
- データベース・オブジェクト・リストのデフォルト視点

関連概念:

54 ページの『共通パラメーターのインストール・デフォルト』

QMF1210.SDSQEXCE(DSQ1DEFS) EXEC のデフォルト値。

関連タスク:

245 ページの『ユーザーのオブジェクト・リストのカスタマイズ』

DB2 は許可情報を SYSIBM.SYSTABAUTH 表から収集するため、表リストおよび列情報に関して QMF から提供されるデフォルト視点を使用すると、処理時間が長くな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追加のセキュリティ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を作成することを検討してください。

関連資料:

102 ページの『インストール検査手順に必要なサンプル照会およびサンプル・プロシージャのインストール』

TSO にのみ、サンプルの QMF 照会およびプロシージャをインストールします。

497 ページの『TSO および CICS での QMF コントロール表および表スペース』
QMF に付属のコントロール表は以下のとおりです。

QMF 管理に必要な権限

特定のデータベース・オブジェクトに対して複数の GRANT ステートメントを発行することを避けるには、大部分の QMF 管理作業およびカスタマイズ作業に、DBADM または同等の権限を持つ許可 ID を使用する必要があります。ユーザーがデータを保管するすべてのデータベースで、このレベルの権限が必要です。

DB2 for z/OS システムでは、以下のデータベースに対して DBADM または同等の権限を持つ許可 ID が必要です。

DSQDBCTL

QMF コントロール表はこのデータベースに保管されます。

DSQDBDEF

ユーザーが SAVE DATA コマンドを発行すると、結果の表は、表スペース DSQTSDEF 内のこのデータベースにデフォルトで保管されます。

ユーザーが DB2 for Linux, UNIX, and Windows データベース内でデータベースおよび QMF オブジェクトを取り扱う場合は、DSQTSCTL および DSQTSOBJ データベース・パーティション・グループに対する DBADM 権限が必要になります。

QMF を製品をインストールしたときの ID とは異なる ID で管理する場合は、新しい ID に DBADM 権限を付与してください。

QMF は、初期化時に QMF を始動するユーザーの許可 ID を確認して、その ID が Q.PROFILES コントロール表に対する INSERT または DELETE のいずれかの特権を持っているか調べます。許可 ID がこれらいずれかの特権を持っている場合、QMF はユーザーを QMF 管理者 と見なします。QMF 管理者権限を使用すると、QMF の管理および保守が容易になります。QMF 管理者は、他のユーザーが所有する QMF 照会、書式、およびプロシージャーに対して、これらのオブジェクトを全ユーザーと共有するようその所有者に要求することなく、以下のコマンドを実行できます。

- SAVE
- ERASE
- IMPORT
- EXPORT
- DISPLAY

QMF 管理者は、他のユーザーが所有する ANALYTIC オブジェクトに対して DISPLAY コマンドを、また ANALYTIC オブジェクトおよび FOLDER オブジェクトに対して ERASE コマンドを発行することもできます。

QMF コントロール表が保管されるデータベース (DSQDBCTL) に対して DBADM あるいはそれに相当する権限を持つユーザーには、自動的に QMF 管理者権限が付与されます。

DSQUOPTS 出口ルーチンは、QMF の初期化時に実行され、QMF 管理者権限チェックを無効にする手段を提供します。この無効化には、DSQEC_DISABLEADM 大域変数を使用します。ただし、この機能が無効にする場合は、ユーザーの QMF オブジェクトに対する管理者アクセス権限を明示的にセットアップする必要があります。例えば、SHARE=YES パラメーターを指定してオブジェクトをユーザーが保存することを要求できます。あるいは DSQEC_SHARE 大域変数を使用して SHARE パラメーターのデフォルト値を YES に変更することができます。DSQUOPTS 出口ルーチンでは、DSQEC_SHARE 大域変数に新しいデフォルト値を設定することもできます。

関連概念:

175 ページの『QMF 開始時の大域変数および QMF セッション動作の初期化』いくつかの方法を使用して、QMF の開始時に QMF 大域変数を設定できます。この方法には、DSQUOPTS ルーチン、Q.GLOBAL_VARS 大域変数表、またはデフォルト・システム初期化プロシージャの変更があります。このプロシージャは、ホーム・パネルが表示される前にユーザーの QMF セッションの他の側面を設定するためにも使用できます。

関連資料:

497 ページの『TSO および CICS での QMF コントロール表および表スペース』QMF に付属のコントロール表は以下のとおりです。

ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)

この情報は、QMF を初めてインストールする場合、およびデータベースがスタンドアロン・データベース、リクエスターとしてのみ機能するデータベース、またはリクエスターとサーバーの両方として機能するデータベースである場合に使用します。DB2 (z/OS 版) データベースのみが、スタンドアロンになるか、またはリクエスターとして機能できます。

このタスクについて

スタンドアロン・データベースまたはリクエスター・データベースのロードマップおよびインストール・パスは、以下のインストール・タイプに適用されます。

- QMF バージョン 12 リリース 1 の初回インストール。最初のインストールは、必ず DB2 for z/OS データベース内である必要があります。これは、DB2 for z/OS のみが、QMF を開始できるデータベース・タイプであるためです。
- リモート・サーバーにアクセスしない、またはリモート・リクエスターからアクセスされないデータベースへの QMF のインストール。これらのデータベースは、スタンドアロン・データベースと呼ばれます。
- 分散データ構成内でリクエスターとしてのみ機能するデータベースへの QMF のインストール。
- 分散データ構成内でサーバーおよびリクエスターの両方として機能するデータベースへの QMF のインストール。

手順

次のロードマップを使用して、環境に適したインストール・パスを識別します。

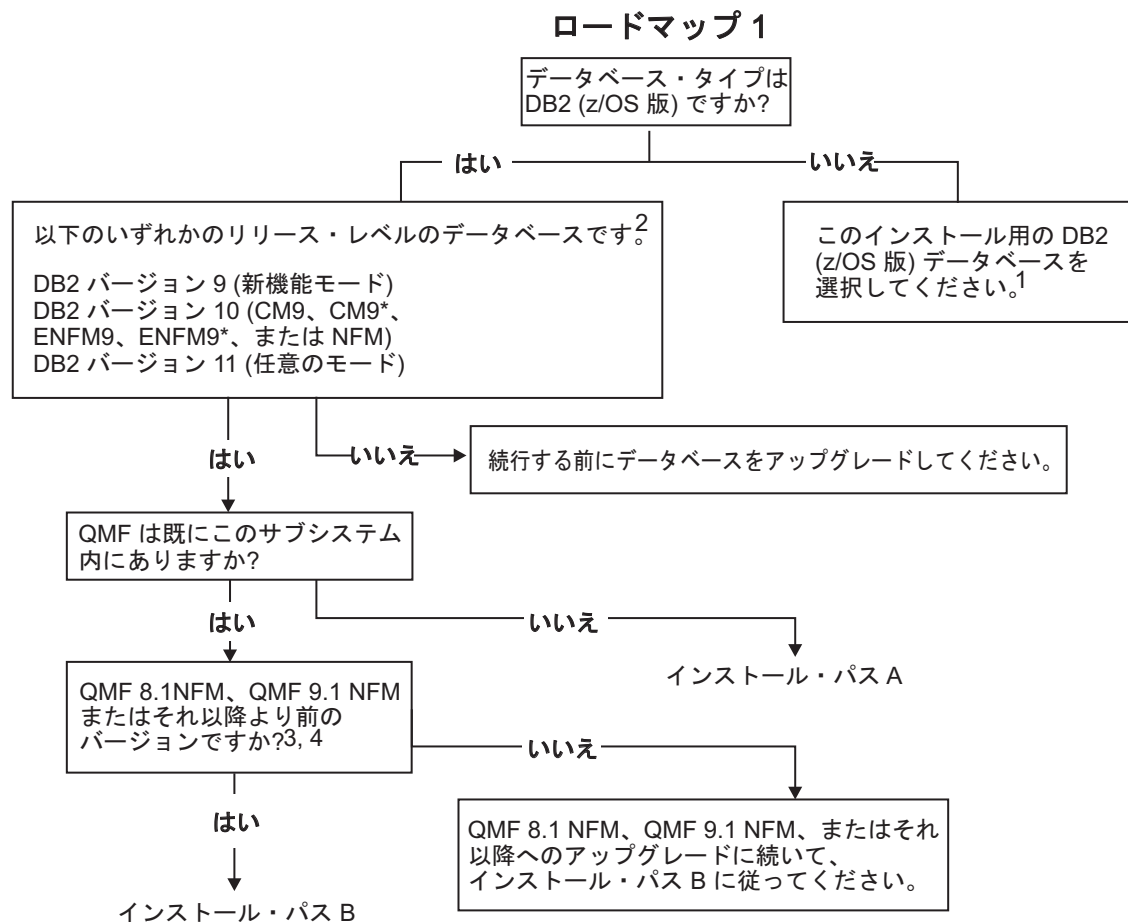


図 4. QMF バージョン 12 リリース 1 を初めてインストールする場合、あるいはデータベースがスタンドアロン・データベース、リクエスターとしてのみ機能するデータベース、またはリクエスターとサーバーの両方として機能するデータベースである場合のインストール・ロードマップ

ロードマップの注:

- 最初のインストールは、必ず DB2 for z/OS データベースに対して行う必要があります。これは、DB2 for z/OS のみが、QMF を開始できるデータベース・タイプであるためです。
- インストールされている DB2 for z/OS のリリースを判断するには、次の照会を実行してください。

```
SELECT GETVARIABLE('SYSIBM.VERSION')
FROM SYSIBM.SYSDUMMY1
```

結果は DSNvvrrm で返されます。DSN はデータベースが DB2 for z/OS であることを示し、vv はバージョン、rr はリリース・レベル、m は修正レベルを示します。例えば、DB2 10 for z/OS データベースの場合は、次の結果が返されます。

```
DSN10015
```

DB2 インストールが DB2 Value Unit Edition であるかどうかを判別するには、SYSPROC.ADMIN_INFO_SYSPARM ストアド・プロシージャを実行して、OTC_LICENSE パラメーターの値を確認します。DB2 VUE インストー

ルの場合、このパラメーターの値は TERMS_ACCEPTED となり、それ以外の場合、この値は NOT_USED となります。

3. インストールされている QMF リリースを判別するには、QMF を開始します。バージョンおよびリリースが、QMF ホーム・パネルに表示されます。
4. 既存の QMF インストールのモードを判断するには、次の照会を実行してください。この照会は、QMF、SPUFI、またはそれ以外の使用可能な SQL ツールから実行できます。

```
SELECT LENGTH  
FROM SYSIBM.SYSCOLUMNS  
WHERE NAME = 'USERID'  
AND TBNAME = 'ERROR_LOG'  
AND TBCREATOR = 'Q'
```

- この照会で値 8 が戻された場合、既存の QMF インストールは互換モードです。
- この照会で値 128 が戻された場合、既存の QMF インストールは新機能モードです。
- この照会で値が戻されなかった場合、確認対象のサーバーには QMF がインストールされていません。

次のタスク

このロードマップを通じて得られた結果に基づいて、インストール・パス A または B に進みます。

関連概念:

5 ページの『分散データ・ネットワーク内の QMF』

QMF の最初のインストールは、DB2 for z/OS データベースに対して行う必要があります。このインストールが完了すると、DB2 for z/OS データベースは、スタンドアロンとして使用するか、リクエスターとして機能するか、あるいは他の QMF バージョン 12.1 インストール済み環境のリクエスターとサーバーの両方として機能します。

関連タスク:

23 ページの『インストール・パス A: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースへの QMF V12.1 のインストール』

この手順は、DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースへの QMF の初回インストールに使用します。このインストール・パスでは、旧リリースの QMF が含まれていない DB2 for z/OS データベースに QMF バージョン 12.1 をインストールするために必要な手順について説明します。

24 ページの『インストール・パス B: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースでの QMF V11.2、V11.1、V10 または QMF NFM リリースから QMF V12.1 への移行』

この手順は、QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モード、QMF バージョン 10 や QMF バージョン 11.1、または QMF バージョン 11.2 を QMF バージョン 12.1 に移行するために使用します。

関連情報:

 DB2 資料

SYSPROC.ADMIN_INFO_SYSPARM ストアード・プロシージャのインストール

方法および実行方法に関する情報を検索します。

インストール・パス **A: DB2 for z/OS** スタンドアロン・データベースまたはリクエスター・データベースへの **QMF V12.1** のインストール

この手順は、DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースへの QMF の初回インストールに使用します。このインストール・パスでは、旧リリースの QMF が含まれていない DB2 for z/OS データベースに QMF バージョン 12.1 をインストールするために必要な手順について説明します。

始める前に

このトピックの手順では、スタンドアロン・データベースおよびリクエスター・データベースのロードマップを使用して、これが正しいインストール・パスであることを既に判別していると想定しています。

インストール・プロセスを開始する前に、以下の手順を実行します。

- 製品をインストールするデータベースに対する適切な権限があることを確認してください。詳しくは、17 ページの『QMF のインストールに必要な権限』を参照してください。
- DB2 QMF for z/OS をインストールしている場合は、データベースが DB2 for z/OS バージョン 9.1 新機能モード 以降であることを確認してください。データベース・リリースがバージョン 9.1 新機能モード よりも前の場合は、作業を続行する前にデータベースをアップグレードしてください。DB2 10 は、バージョン 8 からの変換モード、バージョン 8 からの変換モード*、バージョン 8 からの新機能有効化モード、およびバージョン 8 からの新機能有効化モード* (CM8、CM8*、ENFM8、または ENFM8*) で実行することはできません。

手順

旧リリースが存在しない DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースに QMF バージョン 12.1 をインストールするには、以下の手順に従ってください。

1. QMF バージョン 12.1 のインストール先であるリクエスター・データベースを開始します。
2. 45 ページの『リクエスター (DB2 for z/OS) データベースへの QMF のインストールの準備』のステップを完了します。

46 ページの『前提条件のソフトウェア』にリストされている必須製品すべてのインストールを十分にテストしてから、QMF のインストールを開始してください。製品 (例えば、GDDM) にインストール検査手順 (IVP) が関連付けられている場合は、IVP がエラーなしで終了したことを確認してから、QMF のインストールを開始してください。

3. 60 ページの『旧リリースがインストールされていない場合に QMF V12.1 をインストールするジョブ』のジョブを実行して、QMF をインストールします。
4. QMF を実行している z/OS 環境用に QMF をカスタマイズします。

- TSO で QMF を実行している場合は、64 ページの『TSO でのリクエスター・インストールのカスタマイズ』のステップを完了します。
 - CICS で QMF を実行している場合は、73 ページの『CICS でのリクエスター・インストールのカスタマイズ』のステップを完了します。
5. 93 ページの『リクエスター・データベースへの QMF インストールのテスト』の手順に従って、インストール済み環境をテストします。
 6. 該当する環境用のインストール検査手順を実行します。
 - TSO で QMF を実行している場合は、105 ページの『TSO 用 IVP の実行』のステップを完了します。
 - CICS で QMF を実行している場合は、108 ページの『CICS 用 IVP の実行』のステップを完了します。
 7. 新規リリースを実稼働環境で使用する前に、サイト固有のワークロード、ユーザー、アプリケーション、およびオブジェクトを使用して、新規リリースを十分にテストしてください。新規リリースのテストに使用するオブジェクトが異なるデータベースにある場合、113 ページの『第 6 章 移行およびフォールバックの考慮事項』の説明に従って、QMF バージョン 12.1 がインストールされているデータベースにオブジェクトを移行することができます。

次のタスク

必要に応じて、以下のロードマップを使用してインストール・プロセスを続行します。

- 他のスタンドアロン・データベースまたはリクエスター・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 1 に進みます。
- QMF CONNECT コマンドによってアクセスされるサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 2 に進みます。
- 3 部構成の名前を使用してアクセスされるサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 3 に進みます。
- 各国語機能を有効にしたい場合は、ロードマップ 4 に進みます。

関連概念:

175 ページの『QMF 開始時の大域変数および QMF セッション動作の初期化』いくつかの方法を使用して、QMF の開始時に QMF 大域変数を設定できます。この方法には、DSQUOPTS ルーチン、Q.GLOBAL_VARS 大域変数表、またはデフォルト・システム初期化プロシージャの変更があります。このプロシージャは、ホーム・パネルが表示される前にユーザーの QMF セッションの他の側面を設定するためにも使用できます。

インストール・パス B: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースでの QMF V11.2、V11.1、V10 または QMF NFM リリースから QMF V12.1 への移行

| この手順は、QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モード、QMF バージョン 10 や QMF バージョン 11.1、または QMF バージョン 11.2 を QMF バージョン 12.1 に移行するために使用します。

始める前に

このトピックの手順では、スタンドアロン・データベースおよびリクエスター・データベースのロードマップを使用して、これが正しいインストール・パスであることを既に判別していると想定しています。

インストール・プロセスを開始する前に、以下の手順を実行します。

- 製品をインストールするデータベースに対する適切な権限があることを確認してください。詳しくは、17 ページの『QMF のインストールに必要な権限』を参照してください。
- DB2 QMF for z/OS をインストールしている場合は、データベースが DB2 for z/OS バージョン 9.1 新機能モード 以降であることを確認してください。データベース・リリースがバージョン 9.1 新機能モード よりも前の場合は、作業を続行する前にデータベースをアップグレードしてください。DB2 10 は、バージョン 8 からの変換モード、バージョン 8 からの変換モード*、バージョン 8 からの新機能有効化モード、およびバージョン 8 からの新機能有効化モード* (CM8、CM8*、ENFM8、または ENFM8*) で実行することはできません。

手順

DB2 for z/OS リクエスターで、QMF 新機能モード・リリースから、QMF バージョン 10.1、QMF バージョン 11.1 から、または QMF バージョン 11.2 から QMF バージョン 12.1 に移行するには、以下の手順に従ってください。

1. QMF の使用に不可欠なユーザー作成ルーチンまたはプログラムのソース・コードおよび関連するすべてのロード・モジュールをバックアップします。 そのようなルーチンには、以下のような項目が含まれます。
 - 管理プログラム出口ルーチン
 - QMF 書式の編集コード
 - 書式計算または条件付き形式設定を行う REXX EXEC
 - QMF 呼び出し可能インターフェースまたはコマンド・インターフェースを使用するアプリケーション
 - コマンド同義語定義、およびこれらの定義をサポートするすべてのルーチン
 - カスタマイズされたファンクション・キー定義
2. QMF バージョン 12.1 のインストール先であるリクエスター・データベースを開始します。
3. 45 ページの『リクエスター (DB2 for z/OS) データベースへの QMF のインストールの準備』のステップを完了します。

46 ページの『前提条件のソフトウェア』にリストされている必須製品すべてのインストールを十分にテストしてから、QMF のインストールを開始してください。製品 (例えば、GDDM) にインストール検査手順 (IVP) が関連付けられている場合は、IVP がエラーなしで終了したことを確認してから、QMF のインストールを開始してください。
4. 61 ページの『QMF V8 NFM、QMF V9 NFM、QMF V10、QMF V11.1、または QMF V11.2 を QMF V12.1 に移行するジョブ』のジョブを実行します。
5. QMF を実行している z/OS 環境用に QMF をカスタマイズします。

- TSO で QMF を実行している場合は、64 ページの『TSO でのリクエスター・インストールのカスタマイズ』のステップを完了します。
 - CICS で QMF を実行している場合は、73 ページの『CICS でのリクエスター・インストールのカスタマイズ』のステップを完了します。
6. 93 ページの『リクエスター・データベースへの QMF インストールのテスト』の手順に従って、インストール済み環境をテストします。
 7. 該当する環境用のインストール検査手順を実行します。
 - TSO で QMF を実行している場合は、105 ページの『TSO 用 IVP の実行』のステップを完了します。
 - CICS で QMF を実行している場合は、108 ページの『CICS 用 IVP の実行』のステップを完了します。
 8. 113 ページの『第 6 章 移行およびフォールバックの考慮事項』の移行に関する考慮事項を参照し、ご使用のインストール済み環境または機能に、本リリース特有の移行に関する考慮事項が適用されるかどうかを判別してください。
 9. 新規リリースを実稼働環境で使用する前に、サイト固有のワークロード、ユーザー、アプリケーション、およびオブジェクトを使用して、新規リリースを十分にテストしてください。

次のタスク

必要に応じて、以下のロードマップを使用してインストール・プロセスを続行します。

- 他のスタンドアロン・データベースまたはリクエスター・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 1 に進みます。
- QMF CONNECT コマンドによってアクセスされるサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 2 に進みます。
- 3 部構成の名前を使用してアクセスされるサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 3 に進みます。
- 各国語機能を有効にしたい場合は、ロードマップ 4 に進みます。

関連概念:

175 ページの『QMF 開始時の大域変数および QMF セッション動作の初期化』いくつかの方法を使用して、QMF の開始時に QMF 大域変数を設定できます。この方法には、DSQUOPTS ルーチン、Q.GLOBAL_VARS 大域変数表、またはデフォルト・システム初期化プロシージャの変更があります。このプロシージャは、ホーム・パネルが表示される前にユーザーの QMF セッションの他の側面を設定するためにも使用できます。

ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール

次のロードマップを使用して、QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによってアクセスされるサーバー専用のデータベースのためのインストール・パスを識別します。このロードマップ内のパスを使用すると、サーバーは QMF コマンド内の 3 部構成の名前によるアクセスも使用可能になるように設定されます。

制約事項:

- QMF for TSO が、DB2 for z/OS リクエスター・データベースでストアード・プロシージャーとして開始された場合、リモート・サーバーとの接続はサポートされません。
- 3 部構成の名前を含む QMF コマンドは、DB2 (VSE および VM 版) サーバーに送信することはできません。
- デフォルトでは、3 部構成の名前を含む QMF コマンドを使用して、ラージ・オブジェクト (LOB) データを含むリモート表にアクセスすることはできません。3 部構成の名前を持つリモート表内の LOB データへのアクセスを可能にするには、DSQEC_LOB_RETRV 大域変数を 3 に設定する必要があります。また、DSQEC_LOB_RETRV を 2 に設定すると、LOB メタデータのみを検索が可能になります。
- 3 部構成の名前を含むコマンドを DB2 for z/OS に送信する場合を除き、QMF は複数行フェッチをオフにして開始する必要があります。

始める前に

この情報を使用する前に、5 ページの『分散データ・ネットワーク内の QMF』の概念を理解しておいてください。

手順

次のロードマップを使用して、環境に適したインストール・パスを識別します。

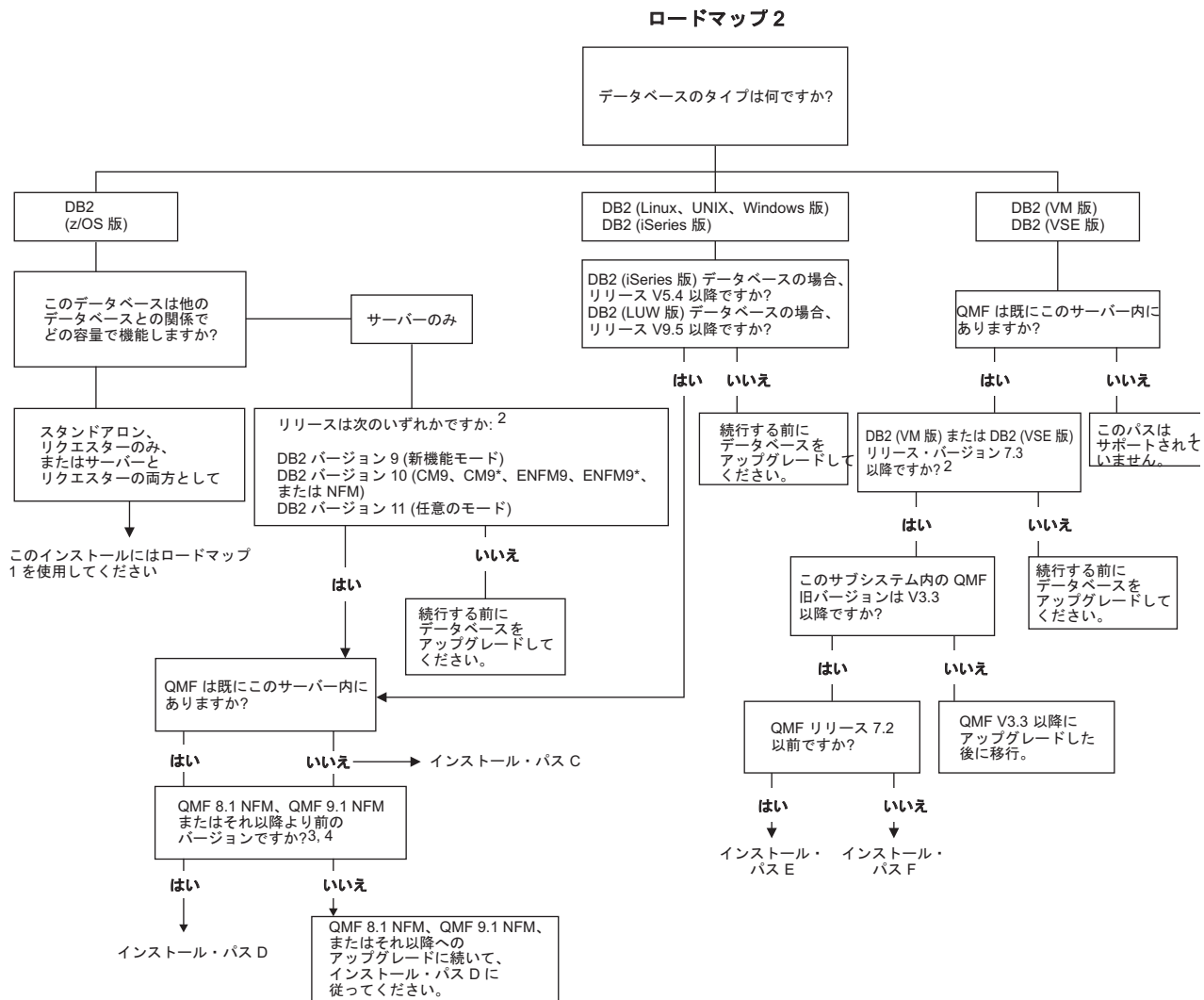


図 5. QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによってアクセスされるデータベースのインストール・ロードマップ

ロードマップの注:

1. QMF バージョン 12.1 は、DB2 Server (VSE および VM 版) での新規インストールには対応していません。QMF バージョン 3.3 以降をインストールしないと、このデータベースは、QMF バージョン 12.1 がインストールされたリクエストのサーバーとして機能することができません。
2. インストールされているデータベース・リリースを判別するには、以下の該当するステップを実行します。
 - DB2 for z/OS データベースの場合は、次の照会を実行します。

```
SELECT GETVARIABLE('SYSIBM.VERSION')
FROM SYSIBM.SYSDUMMY1
```

結果は DSN*vvrrm* で返されます。DSN はこのデータベースが DB2 for z/OS であることを示し、*vv* はバージョン、*rr* はリリース・レベル、*m* は修正レベルを示します。例えば、DB2 for z/OS バージョン 8.1.5 データベースの場合は、次の結果が返されます。

DSN08015

- DB2 for iSeries データベースの場合は、DSPPTF コマンドを発行します。リリースは、「基本オプション」フィールドの「リリース」に記録されます。
- DB2 for Linux, UNIX, and Windows データベースの場合は、次の照会を実行します。

```
SELECT SERVICE_LEVEL
FROM SYSIBMADM.ENV_INST_INFO
```

最初の 3 つの数字は、バージョン、リリース、および修正レベルを示します。例えば、「DB2 v9.1.600.647」は、DB2 for Linux, UNIX, and Windows バージョン 9.1.6 を示します。

- DB2 (VSE および VM 版) データベースの場合は、次の照会を実行します。

```
SELECT *
FROM SYSTEM.SYSOPTIONS
WHERE SQLOPTION='RELEASE'
```

結果は *v.r.m* で返されます。ここで、*v* はバージョン、*r* はリリース・レベル、*m* は修正レベルを示します。

3. インストールされている QMF リリースを判別するには、QMF をローカルの DB2 for z/OS サブシステムで開始します。QMF ホーム・パネルが表示されたら、CONNECT コマンドを発行して、リモート・サーバーに接続します。または、QMF の開始時に DSQSDBNM パラメーターを使用して、初期データベース接続を指定することもできます。リモート・サーバーに接続されたら、QMF ホーム・パネルに示されているバージョンおよびリリースを確認してください。

CONNECT コマンドについて詳しくは、『CONNECT in CICS』または『CONNECT in TSO』を参照してください。

4. 既存の QMF インストールのモードを判断するには、次の照会を実行してください。次の照会は、QMF、SPUFI、またはそれ以外の使用可能な SQL ツールから実行できます。

```
SELECT LENGTH
FROM SYSIBM.SYSCOLUMNS
WHERE NAME = 'USERID'
AND TBNAME = 'ERROR_LOG'
AND TBCREATOR = 'Q'
```

この照会で値 8 が戻された場合、既存の QMF インストールは互換モードです。この照会で値 128 が戻された場合、既存の QMF インストールは新機能モードです。この照会で値が戻されなかった場合、確認対象のサーバーには QMF がインストールされていません。

次のタスク

このロードマップを通じて得られた結果に基づいて、インストール・パス C、D、E、または F に進みます。

関連タスク:

30 ページの『インストール・パス C: DB2 for z/OS, iSeries、または LUW サーバーへの QMF V12.1 のインストール』

この手順は、旧リリースの QMF がインストールされていない DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバー

に QMF バージョン 12.1 をインストールするために使用します。

33 ページの『インストール・パス D: DB2 for z/OS、iSeries、または LUW サーバーでの QMF V11.2、QMF V11.1、QMF V10、または QMF NFM リリースから QMF V12.1 への移行』

この手順は、DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバーで QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モード、QMF バージョン 10、QMF バージョン 11.1、または QMF バージョン 11.2 を QMF バージョン 12.1 に移行するために使用します。

35 ページの『インストール・パス E: DB2 (VSE および VM 版) サーバーでの QMF V7.2 以前から V12.1 への移行』

この手順は、DB2 (VSE および VM 版) サーバーでバージョン 7.2 以前である既存のリリースの QMF を QMF バージョン 12.1 に移行するために使用します。このインストール・パスでは、DB2 (VSE および VM 版) サーバーが QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターからアクセスできるように準備します。

37 ページの『インストール・パス F: DB2 (VSE および VM 版) サーバーでの QMF V8.1、V9.1、V10.1、V11.1、または V11.2 から QMF V12.1 への移行』

この手順は、DB2 (VSE および VM 版) サーバーで既存の QMF バージョン 8.1、バージョン 9.1、バージョン 10.1、バージョン 11.1、またはバージョン 11.2 インストール済み環境を QMF バージョン 12.1 に移行するために使用します。このインストール・パスでは、DB2 (VSE および VM 版) サーバーが QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターからアクセスできるように準備します。

39 ページの『ロードマップ 3: 3 部構成の名前によってアクセスされるサーバー・データベースのインストール』

次のロードマップを使用して、QMF コマンドで 3 部構成の名前を使用した、リモート・サーバー上のデータへのアクセスをサポートするインストール・パスを識別します。

196 ページの『初期データベース接続の指定』

ローカルの DB2 for z/OS サブシステム内で QMF を開始して、QMF ホーム・パネルが表示される前に、すぐに別のデータベースに接続することができます。

インストール・パス C: DB2 for z/OS、iSeries、または LUW サーバーへの QMF V12.1 のインストール

この手順は、旧リリースの QMF がインストールされていない DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバーに QMF バージョン 12.1 をインストールするために使用します。

インストールの完了後、QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターを使用して、QMF バージョン 12.1 がインストールされたリクエスト・データベースからこのサーバーに接続できます。3 部構成の表名および視点名を含む QMF コマンドもこのサーバーに送信できます。

始める前に

このトピックの手順では、QMF CONNECT コマンドによってアクセスされるサーバー・データベースのロードマップを使用して、これが正しいインストール・パスであることを既に判別していると想定しています。

インストール・プロセスを開始する前に、以下の手順を実行します。

- まだ実行済みでない場合は、QMF バージョン 12.1 を、このサーバーにアクセスする少なくとも 1 つの DB2 for z/OS リクエスター・データベースにインストールし、準備およびテストしてください。DB2 for z/OS データベースのみリクエスターとして機能できます。
リクエスターのインストール・パスについては、20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』を参照してください。
- データベースが以下のいずれかのリリースであることを確認します。必要であれば、続行する前にデータベースをアップグレードしてください。
 - DB2 for z/OS サーバーは、バージョン 9.1 新機能モード 以降である必要があります。DB2 10 は、バージョン 8 からの変換モード、バージョン 8 からの変換モード*、バージョン 8 からの新機能有効化モード、およびバージョン 8 からの新機能有効化モード* (CM8、CM8*、ENFM8、または ENFM8*) で実行することはできません。

DB2 for z/OS データベースがリクエスターおよびサーバーの両方として機能する場合は、20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』を参照して、インストール・パスを決定してください。このロードマップのインストール・パスの場合、データベースは両方の役割で機能できます。

- DB2 for iSeries サーバーは、バージョン 5.4 以降である必要があります。
 - DB2 for Linux, UNIX, and Windows サーバーは、バージョン 9.5 以降である必要があります。
- 製品をインストールするデータベースに対する適切な権限があることを確認してください。詳しくは、17 ページの『QMF のインストールに必要な権限』を参照してください。

手順

旧リリースの QMF がインストールされていない DB2 for z/OS、iSeries、または Linux、UNIX、および Windows サーバーに QMF バージョン 12.1 をインストールするには、以下の手順に従ってください。

1. QMF バージョン 12.1 をインストールするサーバーと QMF バージョン 12.1 を既にインストール済みのリクエスターとの間の DRDA 通信を定義およびテストします。

サーバーとリクエスター間の DRDA 通信が、サーバーでの QMF バージョン 12.1 のインストールを行うために使用されます。DRDA の定義およびテスト方法の詳細については、「DB2 資料」を参照してください。

2. リクエスター・データベースおよびサーバー・データベースの両方を開始します。 リクエスター・データベース内の DB2 for z/OS サブシステムは、分散データ機能 (DDF) を開始した状態で開始する必要があります。
3. DB2 for iSeries サーバーの場合のみ: DB2 for iSeries 照会マネージャーから、サーバーに対して次の SQL ステートメントを発行します。 使用するユーザー ID は、セキュリティー担当者のものであるか、または *ALLOBJ 権限を持つユーザー ID である必要があります。
CREATE COLLECTION Q
4. 81 ページの『サーバー・インストールのためのインストール・ジョブ・パラメーターの設定』で説明されているとおり、インストール・パラメーターを設定します。
5. 83 ページの『旧リリースがインストールされていないサーバー・データベースに QMF V12.1 をインストールするジョブ』のジョブを実行します。これらのジョブは、ローカル・データベースから実行する必要があります。
6. リモート・サーバーでインストール済み環境をテストします。
 - a. 109 ページの『サーバー・データベースでの QMF インストールのテスト』のステップを完了します。
 - b. 新規リリースを実稼働環境で使用する前に、サイト固有のワークロード、ユーザー、アプリケーション、およびオブジェクトを使用して、新規リリースを十分にテストしてください。 新規リリースのテストに使用するオブジェクトが異なるデータベースにある場合、113 ページの『第 6 章 移行およびフォールバックの考慮事項』の説明に従って、QMF バージョン 12.1 がインストールされているデータベースにオブジェクトを移行することができます。
7. DB2 for iSeries サーバーのみ: QMF ユーザーが Q *LIB に対して *USE 権限を持っていることを確認してください。

次のタスク

必要に応じて、以下のロードマップを使用してインストール・プロセスを続行します。

- 他のスタンドアロン・データベースまたはリクエスター・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 1 に進みます。
- QMF CONNECT コマンドによってアクセスされる他のサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 2 に進みます。
- 3 部構成の名前を使用してアクセスされるサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 3 に進みます。
- 各国語機能を有効にしたい場合は、ロードマップ 4 に進みます。

関連タスク:

26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』

次のロードマップを使用して、QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによってアクセスされるサーバー専用のデータベースのためのインストール・パスを識別します。このロードマップ内のパスを使用すると、サーバーは QMF コマンド内の 3 部構成の名前によるアクセスも使用可能になるよ

うに設定されます。

インストール・パス D: DB2 for z/OS、iSeries、または LUW サーバーでの QMF V11.2、QMF V11.1、QMF V10、または QMF NFM リリースから QMF V12.1 への移行

この手順は、DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバーで QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モード、QMF バージョン 10、QMF バージョン 11.1、または QMF バージョン 11.2 を QMF バージョン 12.1 に移行するために使用します。

移行が完了すると、QMF バージョン 12.1 がインストールされたリクエスター・データベースからこのサーバーへ QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターで接続できます。また、3 部構成の表名および視点名を含む QMF コマンドもこのサーバーへ送信できるようになります。

始める前に

このトピックの手順では、QMF CONNECT コマンドによってアクセスされるサーバー・データベースのロードマップを使用して、これが正しいインストール・パスであることを既に判別していると想定しています。

インストール・プロセスを開始する前に、以下の手順を実行します。

- まだ実行済みでない場合は、QMF バージョン 12.1 を、このサーバーにアクセスする少なくとも 1 つの DB2 for z/OS リクエスター・データベースにインストールし、準備およびテストしてください。DB2 for z/OS データベースのみリクエスターとして機能できます。
リクエスターのインストール・パスについては、20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』を参照してください。
- データベースが以下のいずれかのリリースであることを確認します。必要であれば、続行する前にデータベースをアップグレードしてください。
 - DB2 for z/OS サーバーは、バージョン 9.1 新機能モード 以降である必要があります。DB2 10 は、バージョン 8 からの変換モード、バージョン 8 からの変換モード*、バージョン 8 からの新機能有効化モード、およびバージョン 8 からの新機能有効化モード* (CM8、CM8*、ENFM8、または ENFM8*) で実行することはできません。
 - DB2 for iSeries サーバーは、バージョン 5.4 以降である必要があります。
 - DB2 for Linux, UNIX, and Windows サーバーは、バージョン 9.5 以降である必要があります。
- 製品をインストールするデータベースに対する適切な権限があることを確認してください。詳しくは、17 ページの『QMF のインストールに必要な権限』を参照してください。

手順

DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバーで QMF 新機能モード・リリースから QMF バージョン 12.1 に移行するには、以下の手順に従ってください。

1. QMF バージョン 12.1 をインストールするサーバーと QMF バージョン 12.1 を既にインストール済みのリクエスターとの間の DRDA 通信を定義およびテストします。

サーバーとリクエスター間の DRDA 通信が、サーバーでの QMF バージョン 12.1 のインストールを行うために使用されます。DRDA の定義およびテスト方法の詳細については、「DB2 資料」を参照してください。

2. リクエスター・データベースおよびサーバー・データベースの両方を開始します。リクエスター・データベース内の DB2 for z/OS サブシステムは、分散データ機能 (DDF) を開始した状態で開始する必要があります。
3. 81 ページの『サーバー・インストールのためのインストール・ジョブ・パラメーターの設定』で説明されているとおり、インストール・パラメーターを設定します。
4. 85 ページの『z/OS、iSeries、または LUW サーバーで QMF V8 NFM、QMF V9 NFM、QMF V10、QMF V11.1、または QMF V11.2 を QMF V12.1 に移行するジョブ』のジョブを実行します。これらのジョブは、ローカル・データベースから実行する必要があります。
5. リモート・サーバーでインストール済み環境をテストします。
 - a. 109 ページの『サーバー・データベースでの QMF インストールのテスト』のステップを完了します。
 - b. 新規リリースを実稼働環境で使用する前に、サイト固有のワークロード、ユーザー、アプリケーション、およびオブジェクトを使用して、新規リリースを十分にテストしてください。新規リリースのテストに使用するオブジェクトが異なるデータベースにある場合、113 ページの『第 6 章 移行およびフォールバックの考慮事項』の説明に従って、QMF バージョン 12.1 がインストールされているデータベースにオブジェクトを移行することができます。
6. 113 ページの『第 6 章 移行およびフォールバックの考慮事項』の移行に関する考慮事項を参照し、ご使用のインストール済み環境または機能に、本リリース特有の移行に関する考慮事項が適用されるかどうかを判別してください。

次のタスク

必要に応じて、以下のロードマップを使用してインストール・プロセスを続行します。

- 他のスタンドアロン・データベースまたはリクエスター・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 1 に進みます。
- QMF CONNECT コマンドによってアクセスされる他のサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 2 に進みます。
- 3 部構成の名前を使用してアクセスされるサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 3 に進みます。
- 各国語機能を有効にしたい場合は、ロードマップ 4 に進みます。

関連タスク:

26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』

次のロードマップを使用して、QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによってアクセスされるサーバー専用のデータベースのためのインストール・パスを識別します。このロードマップ内のパスを使用すると、サーバーは QMF コマンド内の 3 部構成の名前によるアクセスも使用可能になるように設定されます。

インストール・パス E: DB2 (VSE および VM 版) サーバーでの QMF V7.2 以前から V12.1 への移行

この手順は、DB2 (VSE および VM 版) サーバーでバージョン 7.2 以前である既存のリリースの QMF を QMF バージョン 12.1 に移行するために使用します。このインストール・パスでは、DB2 (VSE および VM 版) サーバーが QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターからアクセスできるように準備します。

始める前に

このトピックの手順では、QMF CONNECT コマンドによってアクセスされるサーバー・データベースのロードマップを使用して、これが正しいインストール・パスであることを既に判別していると想定しています。

インストール・プロセスを開始する前に、以下の手順を実行します。

- まだ実行済みでない場合は、QMF バージョン 12.1 を、このサーバーにアクセスする少なくとも 1 つの DB2 for z/OS リクエスター・データベースにインストールし、準備およびテストしてください。DB2 for z/OS データベースのみリクエスターとして機能できます。
リクエスターのインストール・パスについては、20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』を参照してください。
- DB2 (VSE および VM 版) サーバーがバージョン 7.3 以降であることを確認します。
- 製品をインストールするデータベースに対する適切な権限があることを確認してください。詳しくは、17 ページの『QMF のインストールに必要な権限』を参照してください。

手順

DB2 (VSE および VM 版) サーバーで QMF バージョン 7.2 以前から QMF バージョン 12.1 に移行するには、以下の手順に従ってください。

1. QMF バージョン 12.1 をインストールするサーバーと QMF バージョン 12.1 を既にインストール済みのリクエスターとの間の DRDA 通信を定義およびテストします。

サーバーとリクエスター間の DRDA 通信が、サーバーでの QMF バージョン 12.1 のインストールを行うために使用されます。DRDA の定義およびテスト方法の詳細については、「DB2 資料」を参照してください。

2. リクエスター・データベースおよびサーバー・データベースの両方を開始します。 リクエスター・データベース内の DB2 for z/OS サブシステムは、分散データ機能 (DDF) を開始した状態で開始する必要があります。
3. 81 ページの『サーバー・インストールのためのインストール・ジョブ・パラメーターの設定』で説明されているとおり、インストール・パラメーターを設定します。
4. 87 ページの『QMF V7.2 以前のバージョンを VM または VSE サーバーの QMF V12.1 に移行するジョブ』のジョブを実行します。これらのジョブは、ローカル・データベースから実行する必要があります。
5. リモート・サーバーでインストール済み環境をテストします。
 - a. 109 ページの『サーバー・データベースでの QMF インストールのテスト』のステップを完了します。
 - b. 新規リリースを実稼働環境で使用する前に、サイト固有のワークロード、ユーザー、アプリケーション、およびオブジェクトを使用して、新規リリースを十分にテストしてください。新規リリースのテストに使用するオブジェクトが異なるデータベースにある場合、113 ページの『第 6 章 移行およびフォールバックの考慮事項』の説明に従って、QMF バージョン 12.1 がインストールされているデータベースにオブジェクトを移行することができます。
6. 113 ページの『第 6 章 移行およびフォールバックの考慮事項』の移行に関する考慮事項を参照し、ご使用のインストール済み環境または機能に、本リリース特有の移行に関する考慮事項が適用されるかどうかを判別してください。

次のタスク

必要に応じて、以下のロードマップを使用してインストール・プロセスを続行します。

- 他のスタンドアロン・データベースまたはリクエスター・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 1 に進みます。
- QMF CONNECT コマンドによってアクセスされる他のサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 2 に進みます。
- 3 部構成の名前を使用してアクセスされるサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 3 に進みます。
- 各国語機能を有効にしたい場合は、ロードマップ 4 に進みます。

関連タスク:

26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』

次のロードマップを使用して、QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによってアクセスされるサーバー専用のデータベースのためのインストール・パスを識別します。このロードマップ内のパスを使用すると、サーバーは QMF コマンド内の 3 部構成の名前によるアクセスも使用可能になるように設定されます。

インストール・パス F: DB2 (VSE および VM 版) サーバーでの QMF V8.1、V9.1、V10.1、V11.1、またはV11.2 から QMF V12.1 への移行

この手順は、DB2 (VSE および VM 版) サーバーで既存の QMF バージョン 8.1、バージョン 9.1、バージョン 10.1、バージョン 11.1、またはバージョン 11.2 インストール済み環境を QMF バージョン 12.1 に移行するために使用します。このインストール・パスでは、DB2 (VSE および VM 版) サーバーが QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターからアクセスできるように準備します。

始める前に

このトピックの手順では、QMF CONNECT コマンドによってアクセスされるサーバー・データベースのロードマップを使用して、これが正しいインストール・パスであることを既に判別していると想定しています。

インストール・プロセスを開始する前に、以下の手順を実行します。

- まだ実行済みでない場合は、QMF バージョン 12.1 を、このサーバーにアクセスする少なくとも 1 つの DB2 for z/OS リクエスター・データベースにインストールし、準備およびテストしてください。DB2 for z/OS データベースのみリクエスターとして機能できます。
リクエスターのインストール・パスについては、20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』を参照してください。
- DB2 (VSE および VM 版) サーバーがバージョン 7.3 以降であることを確認します。
- 製品をインストールするデータベースに対する適切な権限があることを確認してください。詳しくは、17 ページの『QMF のインストールに必要な権限』を参照してください。

手順

DB2 (VSE および VM 版) サーバーで QMF バージョン 8.1、バージョン 9.1、バージョン 10.1、バージョン 11.1、またはバージョン 11.2 から QMF バージョン 12.1 に移行するには、以下の手順に従ってください。

1. QMF バージョン 12.1 をインストールするサーバーと QMF バージョン 12.1 を既にインストール済みのリクエスターとの間の DRDA 通信を定義およびテストします。

サーバーとリクエスター間の DRDA 通信が、サーバーでの QMF バージョン 12.1 のインストールを行うために使用されます。DRDA の定義およびテスト方法の詳細については、「DB2 資料」を参照してください。

2. リクエスター・データベースおよびサーバー・データベースの両方を開始します。リクエスター・データベース内の DB2 for z/OS サブシステムは、分散データ機能 (DDF) を開始した状態で開始する必要があります。

3. 81 ページの『サーバー・インストールのためのインストール・ジョブ・パラメーターの設定』で説明されているとおり、インストール・パラメーターを設定します。
4. 89 ページの『VM または VSE サーバーで QMF V8、V9、V10、V11.1、または V11.2 を QMF V12.1 に移行するジョブ』のジョブを実行します。これらのジョブは、ローカル・データベースから実行する必要があります。
5. リモート・サーバーでインストール済み環境をテストします。
 - a. 109 ページの『サーバー・データベースでの QMF インストールのテスト』のステップを完了します。
 - b. 新規リリースを実稼働環境で使用する前に、サイト固有のワークロード、ユーザー、アプリケーション、およびオブジェクトを使用して、新規リリースを十分にテストしてください。新規リリースのテストに使用するオブジェクトが異なるデータベースにある場合、113 ページの『第 6 章 移行およびフォールバックの考慮事項』の説明に従って、QMF バージョン 12.1 がインストールされているデータベースにオブジェクトを移行することができます。
6. 113 ページの『第 6 章 移行およびフォールバックの考慮事項』の移行に関する考慮事項を参照し、ご使用のインストール済み環境または機能に、本リリース特有の移行に関する考慮事項が適用されるかどうかを判別してください。

次のタスク

必要に応じて、以下のロードマップを使用してインストール・プロセスを続行します。

- 他のスタンドアロン・データベースまたはリクエスター・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 1 に進みます。
- QMF CONNECT コマンドによってアクセスされる他のサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 2 に進みます。
- 3 部構成の名前を使用してアクセスされるサーバー・データベースで QMF をインストールまたは移行したい場合は、ロードマップ 3 に進みます。
- 各国語機能を有効にしたい場合は、ロードマップ 4 に進みます。

関連タスク:

26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』

次のロードマップを使用して、QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによってアクセスされるサーバー専用のデータベースのためのインストール・パスを識別します。このロードマップ内のパスを使用すると、サーバーは QMF コマンド内の 3 部構成の名前によるアクセスも使用可能になるように設定されます。

ロードマップ 3: 3 部構成の名前によってアクセスされるサーバー・データベースのインストール

次のロードマップを使用して、QMF コマンドで 3 部構成の名前を使用した、リモート・サーバー上のデータへのアクセスをサポートするインストール・パスを識別します。

手順

次のロードマップを使用して、環境に適したインストール・パスを識別します。

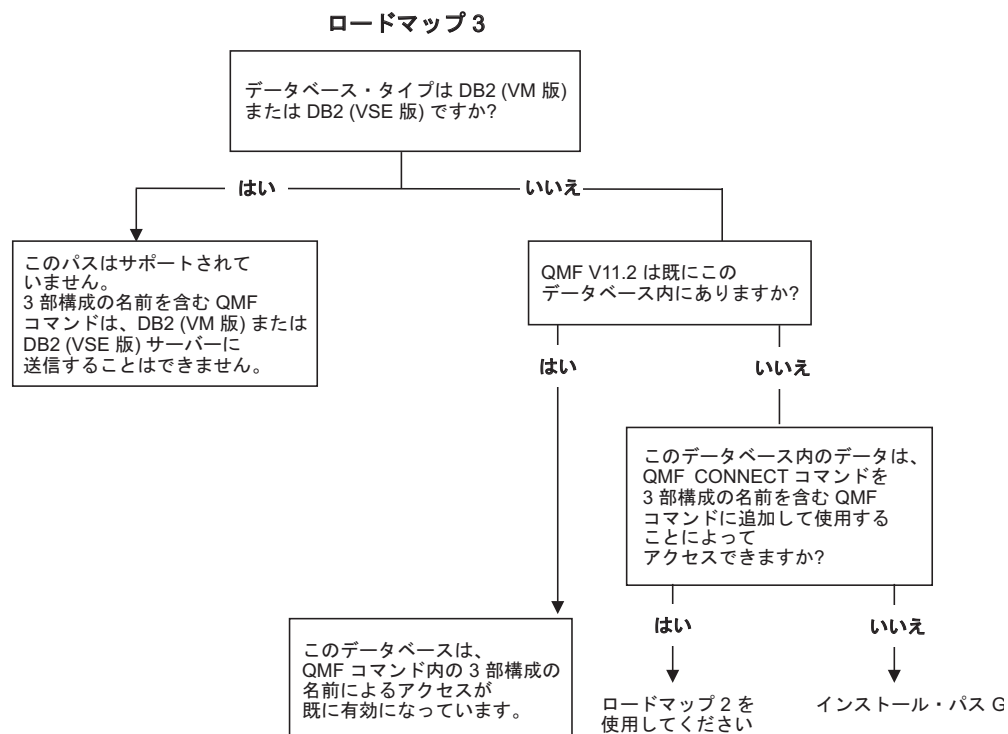


図 6. QMF コマンドで 3 部構成の名前を使用してリモート・サーバーのデータにアクセスするインストールのインストール・ロードマップ

次のタスク

このロードマップを通じて得られた結果に基づいて、ロードマップ 2 またはインストール・パス G に進みます。

関連タスク:

40 ページの『インストール・パス G: 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーの準備』

この手順は、3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーを準備するために使用します。

26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』

次のロードマップを使用して、QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによってアクセスされるサーバー専用のデータベースのためのインストール・パスを識別します。このロードマップ内のパスを使用すると、サーバーは QMF コマンド内の 3 部構成の名前によるアクセスも使用可能になるように設定されます。

インストール・パス G: 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーの準備

この手順は、3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーを準備するために使用します。

3 部構成の表名および視点名を含む QMF コマンドには、以下の規則が適用されます。

- このようなコマンドは、DB2 for z/OS データベースのみから開始できます。
- DB2 (VSE および VM 版) を除くすべてのタイプのサーバーに送信できます。
- このようなコマンドは、QMF for TSO が DB2 for z/OS のストアード・プロシージャーとして開始されている場合は使用できません。

QMF for TSO へのストアード・プロシージャー・インターフェースは、オプションのインターフェースであり、DB2 for z/OS のストアード・プロシージャーを実行できる任意の製品で QMF for TSO を開始できます。

- このようなコマンドは、複数行フェッチがアクティブになっていて、コマンドが DB2 for z/OS 以外のサーバーに送信されると、エラーを発生させます。

ユーザーが 3 部構成の名前を含む QMF コマンドを DB2 for z/OS 以外のサーバーに対して発行する場合は、複数行フェッチをオフにして DB2 for z/OS リクエスターの QMF を開始してください。

デフォルトでは、3 部構成の名前を含む QMF コマンドを使用して、ラージ・オブジェクト (LOB) データを含むリモート表にアクセスすることはできません。3 部構成の名前を持つリモート表内の LOB データへのアクセスを可能にするには、DSQEC_LOB_RETRV 大域変数を 3 に設定する必要があります。また、DSQEC_LOB_RETRV を 2 に設定すると、LOB メタデータのみを検索が可能になります。

3 部構成の表名および視点名について詳しくは、DB2 資料 を参照してください。

始める前に

このトピックの手順では、3 部構成の名前によってアクセスされるサーバー・データベースのロードマップを使用して、これが正しいインストール・パスであることを既に判別していると想定しています。

インストール・プロセスを開始する前に、以下の手順を実行します。

- まだ実行済みでない場合は、QMF バージョン 12.1 を、このサーバーにアクセスする少なくとも 1 つの DB2 for z/OS リクエスター・データベースにインストールし、準備およびテストしてください。DB2 for z/OS データベースのみ

クエスターとして機能できます。

リクエスターのインストール・パスについては、20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』を参照してください。

- 製品をインストールするデータベースに対する適切な権限があることを確認してください。詳しくは、17 ページの『QMF のインストールに必要な権限』を参照してください。

手順

3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーを準備するには、以下の手順に従ってください。

1. QMF バージョン 12.1 をインストールするサーバーと QMF バージョン 12.1 を既にインストール済みのリクエスターとの間の DRDA 通信を定義およびテストします。

サーバーとリクエスター間の DRDA 通信が、サーバーでの QMF バージョン 12.1 のインストールを行うために使用されます。DRDA の定義およびテスト方法の詳細については、「DB2 資料」を参照してください。

2. リクエスター・データベースおよびサーバー・データベースの両方を開始します。リクエスター・データベース内の DB2 for z/OS サブシステムは、分散データ機能 (DDF) を開始した状態で開始する必要があります。
3. 91 ページの『3 部構成の名前を含む QMF コマンドによりアクセスされるサーバーを準備するインストール・ジョブの実行』のジョブを実行します。これらのジョブは、ローカル・データベースから実行する必要があります。

関連概念:

211 ページの『複数行のフェッチと挿入のサポートの使用可能化』
DSQSMRFI パラメーター。データベースがフェッチおよび挿入の際に処理する行が複数行か、単一行かを制御します。

関連タスク:

39 ページの『ロードマップ 3: 3 部構成の名前によってアクセスされるサーバー・データベースのインストール』

次のロードマップを使用して、QMF コマンドで 3 部構成の名前を使用した、リモート・サーバー上のデータへのアクセスをサポートするインストール・パスを識別します。

143 ページの『QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

ロードマップ 4: 英語以外の言語でアクセスされるデータベース

英語以外の言語で QMF を使用する必要がある場合は、下のロードマップを使用します。

手順

次のロードマップを使用して、環境に適したインストール・パスを識別します。

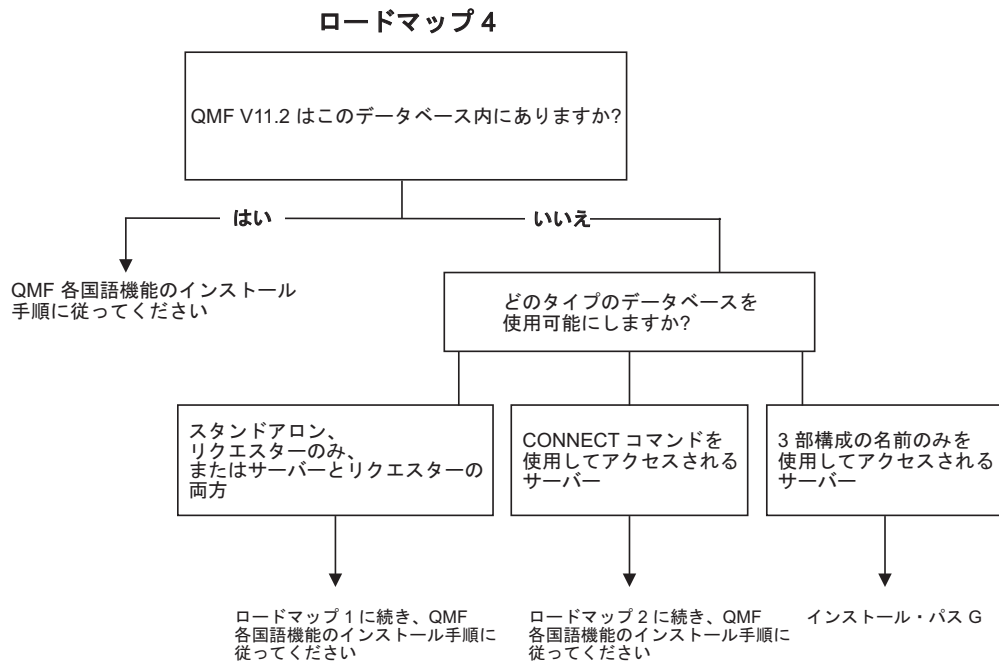


図 7. QMF が英語以外の言語のデータにアクセスできるようにするためのインストール・ロードマップ

次のタスク

このロードマップを通じて得られた結果に基づいて、ロードマップ 1、ロードマップ 2、インストール・パス G、または QMF 各国語機能をインストールするための手順に進みます。

関連タスク:

40 ページの『インストール・パス G: 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーの準備』

この手順は、3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーを準備するために使用します。

20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』

この情報は、QMF を初めてインストールする場合、およびデータベースがスタンドアロン・データベース、リクエスターとしてのみ機能するデータベース、またはリクエスターとサーバーの両方として機能するデータベースである場合に使用します。DB2 (z/OS 版) データベースのみが、スタンドアロンになるか、またはリクエスターとして機能できます。

26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』

次のロードマップを使用して、QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによってアクセスされるサーバー専用のデータベースのためのインストール・パスを識別します。このロードマップ内のパスを使用すると、サーバーは QMF コマンド内の 3 部構成の名前によるアクセスも使用可能になるよ

うに設定されます。

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

QMF を除く DB2 for z/OS のアップグレード時に実行する作業

QMF がインストールされている DB2 for z/OS データベースのモードまたはリリースのいずれかをアップグレードしても、QMF はアップグレードしない場合は、2 つの QMF バインド・ジョブを実行する必要があります。

QMF の既存のインストール済み環境に対して変更を行わない場合でも、これらのジョブは実行する必要があります。また、分散ネットワーク内のデータベースの機能 (スタンドアロン、リクエスター、サーバーのみ、またはリクエスターおよびサーバーの両方) に関わらず、以下のジョブを実行する必要もあります。

重要: 本書、ジョブ自体、または IBM ソフトウェア・サポートによる指示がない限り、プランおよびインストール・パッケージの QMF バインド・ジョブ内で BIND PACKAGE または BIND PLAN オプションを変更または追加することはできません。

表 4. DB2 for z/OS データベースのリリースまたはモードをアップグレードする場合に実行するジョブ・シーケンス。既存の QMF に対する変更が不要な場合も実行。

| ジョブ名 | 説明 |
|----------|--|
| DSQ1BSQL | <p>このジョブは、DB2 for z/OS リクエスター・データベースの QMF インストール・プログラムをサーバー・データベースにバインドします。</p> <p>2 次許可 ID で QMF をインストールした場合は、このジョブを編集して、OWNER(secauth) パラメーターまたは SET CURRENT SQLID='secauth' 指定を BIND PACKAGE および BIND PLAN ステートメントに追加して、QMF が 2 次許可 ID を解決できるようにします。DSQ1DEFS exec の SECAUTH パラメーターは、このジョブで指定する SECAUTH の値と一致する必要があります。</p> <p>最初にこのジョブを実行する必要があります。</p> |
| DSQ1BPKG | <p>このジョブは、QMF パッケージをバインドします。</p> <p>このジョブは、DSQ1BSQL ジョブの実行後に実行する必要があります。</p> |

関連タスク:

147 ページの『拡張 LIST コマンド機能のインストール (z/OS のみ)』

拡張 LIST コマンド機能がインストールされている場合、表と視点に対する特権は PUBLIC の代わりに、ユーザーの 1 次または 2 次許可 ID にのみ付与する必要があります。この許可 ID は、OWNER=ALL オプションを LIST TABLES または LIST ALL コマンドに指定した場合にこのコマンドで生成されるオブジェクト・リ

ストに含まれています。

第 3 章 リクエスター (DB2 for z/OS) のデータベースへの QMF のインストールまたは移行

DB2 for z/OS データベースへの QMF のインストールまたは移行を開始する前に、ご使用の環境およびシステムを準備します。

リクエスター (DB2 for z/OS) データベースへの QMF のインストールの準備

DB2 for z/OS データベースへの QMF のインストールまたは移行を開始する前に、ご使用の環境およびシステムを準備します。

このタスクについて

このトピックの手順は、スタンドアロン・データベースまたはリクエスター・データベース用のインストール・パスに関連付けられています。これらの作業を行う前に、正しいインストール・パスに従っていることを確認してください。

次のタスク

これらの作業が完了した後で、選択したインストール・パスに関連付けられているインストール手順に戻って、インストールの次のステップを判別してください。

関連タスク:

20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』

この情報は、QMF を初めてインストールする場合、およびデータベースがスタンドアロン・データベース、リクエスターとしてのみ機能するデータベース、またはリクエスターとサーバーの両方として機能するデータベースである場合に使用します。DB2 (z/OS 版) データベースのみが、スタンドアロンになるか、またはリクエスターとして機能できます。

リクエスター (DB2 for z/OS) データベースのインストール前提条件

スタンドアロン・データベースまたはリクエスター・データベースとして機能する DB2 for z/OS データベースに QMF をインストールするには、ハードウェアおよびソフトウェアの要件を満たす必要があります。

前提条件のハードウェア

QMF は、当該オペレーティング・システムでサポートされている任意のプロセッサ上で実行されます。ただし、QMF を稼働させようとするプロセッサが 10 進浮動小数点命令をサポートしていない場合、10 進浮動小数点データを含む列では、一部の操作は機能しません。

QMF がアクセス可能な DASD デバイスは、z/OS および DB2 for z/OS がサポートするすべての DASD です。また、QMF がアクセス可能なディスプレイ・デバイスは、GDDM がサポートするすべてのディスプレイです。

2 バイト文字セット (DBCS) を使用する各国語機能を実装するためには、DBCS が使用可能なワークステーションが必要です。このデバイスは、GDDM によりサポートされていることを確認してください。

前提条件のソフトウェア

このトピックにリストされたソフトウェアは、QMF バージョン 12.1 のインストールとその使用に必要とされるソフトウェアです。この他に、特別な機能に必要なオプションのソフトウェアもリストしてあります。これらの製品にはそれぞれ、インストールと構成に必要な前提条件となるハードウェアまたはソフトウェアがある可能性があります。

• 必須のソフトウェア

下の表にリストされたプログラム製品が、QMF バージョン 12.1 をサポートするために必要です。

表 5. DB2 QMF for TSO/CICS バージョン 12.1 の最小限の前提ソフトウェア

| 機能 | 必要最小限の製品 |
|-------------------|---|
| データベース機能 | 以下のいずれかが該当します。 <ul style="list-style-type: none"> • APAR PM45482 が適用された、新機能モードで稼働する DB2 for z/OS バージョン 9.1 (VUE および MLC) • APAR PM50434 および PM72274 が適用された、すべてのモード (CM8、CM8*、ENFM8、および ENFM8* を除く) で稼働する DB2 for z/OS バージョン 10.1 (VUE および MLC) • DB2 for z/OS バージョン 11.1 (VUE および MLC) • DB2 for z/OS V12.1 (VUE および MLC) |
| パネル表示 | GDDM GDDM は、z/OS の基本エレメントとして提供されます。 |
| QMF for CICS 操作のみ | CICS Transaction Server for z/OS、バージョン 3 リリース 1 またはそれ以降、5655-M15 |

• オプションのソフトウェア

QMF で図表を作成するには、GDDM-PGF バージョン 2.1.3 (5668-812) が必要です。

関連概念:

17 ページの『QMF のインストールおよび管理に必要な権限』

QMF のインストールおよび一般的な QMF 管理には、特定の権限が必要です。

関連タスク:

288 ページの『図表作成機能の有効化』


QMF では、GDDM-PGF 製品に付属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することができます。

関連情報:

 [IBM Publications Center](#)

QMF バージョン 12.1 に必要な製品に関する資料を参照してください。

 <http://www.ibm.com/software/data/support/lifecycle/>

QMF バージョン 12.1 に必要な製品のサポートされるバージョンに関する情報を検索します。

 <http://www.ibm.com/support/docview.wss?uid=swg21409518>

QMF バージョン 12.1 と他の DB2 リリース間の互換性に関する情報を参照してください。

 <http://www.ibm.com/support/docview.wss?uid=swg21570476>

QMF と z/OS との間の互換性に関する情報を参照してください。

配布メディアからの QMF バージョン 12.1 ライブラリーのコピー

配布メディアから DB2 for z/OS ターゲット・システムに QMF ライブラリーをロードするには、システム修正変更プログラム拡張版 (SMP/E) を使用します。最初の QMF バージョン 12.1 インストールを実行するシステムにライブラリーをコピーします。

DB2 照会報告書作成プログラム・ライブラリーで使用可能な QMF プログラム・ディレクトリーに、システム修正変更プログラム拡張版 (SMP/E) の使用方法の説明があります。ご使用のリリースおよび各国語用の正しいプログラム・ディレクトリーを読み、プログラム・ディレクトリーに記載されたすべての SMP/E 作業を必ず実施してください。

SMP/E を使用して、次の 2 つのタイプのライブラリーへ QMF モジュールをロードします。

- 稼働システムの実行可能コードを含む、ターゲット・ライブラリー。

下の表に、各ターゲット・ライブラリーの目的が示されています。ライブラリー名の末尾が *n* の場合、その「*n*」はサポートされている各国語ごとに使用されるライブラリーのバージョンを表します。*n* は各 QMF NLF に関連付けられた 1 文字の言語 ID を表します。

デフォルトでは、すべてのターゲット・ライブラリーには、QMF バージョン 12.1 のデフォルトの高位修飾子である QMF1210 が接頭部として指定されています。この接頭部を変更する場合は、新たな命名規則で、QMF のバージョンおよびリリースを識別できるようにしてください。この情報が必要になるのは、問題の追跡時、または IBM ソフトウェア・サポートとやりとりする時点です。

表 6. QMF バージョン 12.1 ターゲット・ライブラリーとその説明

| ライブラリー名 | 説明 |
|----------|--|
| SDSQBASE | サンプルの基本 QMF for TSO/CICS SMP/E インストール・ジョブ。 NLF でこのライブラリーに相当するものではありません。 |

表 6. QMF バージョン 12.1 ターゲット・ライブラリーとその説明 (続き)

| ライブラリー名 | 説明 |
|-------------|--|
| SDSQSAM n | サンプルの NLF SMP/E インストール・ジョブ。 英語 (E) でこのライブラリーに相当するものはありません。 |
| SDSQCHRT | QMF に付属の図表形式 |
| SDSQCLT n | CLIST ライブラリー |
| SDSQDBRM | QMF パッケージを含みます このライブラリーは、QMF ランタイム環境には必要ではありません。インストールと構成の目的でのみ使用されます。 |
| SDSQEXC n | 実行ライブラリー |
| SDSQEXIT | 以下に対して作成することができるユーザー出口用のライブラリー <ul style="list-style-type: none"> ユーザーが作成した編集コード QMF 環境の初期化 (DSQUOPTS および Q.SYSTEM_INI 出口ルーチンはこの目的で使用されます) リソースの管理 <p>QMF ロード・ライブラリーを保管するために、カスタマイズしたライブラリーを使用することもできます。</p> |
| SDSQLOAD | QMF ロード・ライブラリー |
| SDSQMAP n | デフォルト GDDM マップ |
| SDSQMLB n | ISPF メッセージ・ライブラリー。 ISPF 下で QMF を実行している場合にのみ必要です。 |
| SDSQPLB n | ISPF パネル・ライブラリー。 ISPF 下で QMF を実行している場合にのみ必要です。 |
| SDSQPVR n | ISPF 以外の QMF ユーザー・インターフェース・パネル、オンライン・ヘルプ・パネル、およびメッセージ・ヘルプ・パネルを含みます。インストール処理中に DSQ1 n PNL ジョブを実行する時点で、このライブラリーからのパネルを使用して、VSAM データ・セット DSQPNL n にデータを追加します。DSQPNL n が DD 名 DSQPNLE に割り振られるのは、QMF の開始時です。 このライブラリーは、QMF ランタイム環境には必要ではありません。インストールと構成の目的でのみ使用されます。 |
| SDSQSAP n | QMF のインストール・ジョブが入っています。その他に、呼び出し可能インターフェースと、ユーザー作成の編集コードをサポートするプログラムの両方のサンプル・ソース・コードも入っています。 このライブラリーは、QMF ランタイム環境には必要ではありません。インストールと構成の目的でのみ使用されます。 |
| SDSQSLBE | ISPF スケルトン・ライブラリー。 ISPF 下で QMF を実行している場合にのみ必要です。このライブラリーは英語のみであり、NLF で相当するものはありません。 |
| SDSQTLB n | ISPF テーブル・ライブラリー。 ISPF 下で QMF を実行している場合にのみ必要です。 |

表 6. QMF バージョン 12.1 ターゲット・ライブラリーとその説明 (続き)

| ライブラリー名 | 説明 |
|-------------|---|
| SDSQUSR n | ユーザー作成管理プログラムまたはユーザー編集出口ルーチンのユーザー・ライブラリー このライブラリーは、QMF ランタイム環境には必要ではありません。インストールと構成の目的でのみ使用されます。 |

- すべてのシステム・エレメントのマスター・コピーを含む、配布ライブラリー。

下の表に、各配布ライブラリーの目的が示されています。ライブラリー名の末尾が n の場合、その「 n 」はサポートされている各国語ごとに使用されるライブラリーのバージョンを表します。 n は各 QMF NLF に関連付けられた 1 文字の言語 ID を表します。

デフォルトでは、すべての配布ライブラリーには、QMF バージョン 12.1 のデフォルトの高位修飾子である QMF1210 が接頭部として指定されています。この接頭部を変更する場合は、新たな命名規則で、QMF のバージョンおよびリリースを識別できるようにしてください。この情報が必要になるのは、問題の追跡時、または IBM ソフトウェア・サポートとやりとりする時点です。

表 7. QMF バージョン 12.1 配布ライブラリーとその説明

| ライブラリー名 | 説明 |
|-------------|---|
| ADSQBASE | インストールに使用されるサンプル SMP/E ジョブ用の配布ライブラリー |
| ADSQSAM n | NLF インストールに使用されるサンプル SMP/E ジョブ用の配布ライブラリー |
| ADSQCHRT | GDDM 図表用に QMF で提供される配布ライブラリー (SDSQCHRT ライブラリーにあります) |
| ADSQCLT n | CLIST 用の配布ライブラリー (SDSQCLT n ライブラリーにあります) |
| ADSQDBRM | QMF データベース・パッケージ用の配布ライブラリー (SDSQDBRM ライブラリーにあります) |
| ADSQEXC n | Exec 用の配布ライブラリー (SDSQEXC n ライブラリーにあります) |
| ADSQMAP n | GDDM マップ用に QMF で提供される配布ライブラリー (SDSQMAP n ライブラリーにあります) |
| ADSQMLB n | ISPF メッセージ用の配布ライブラリー (SDSQMLB n ライブラリーにあります) |
| ADSQOBJ | ロード・モジュール用の配布ライブラリー (SDSQLOAD ライブラリーにあります) |
| ADSQPLB n | ISPF パネル用の配布ライブラリー (SDSQPLB n ライブラリーにあります) |
| ADSQPVR n | QMF パネル・ライブラリー SDSQPVR n 用の配布ライブラリー |
| ADSQSAP n | サンプル・コード用の配布ライブラリー (SDSQSAP n ライブラリーにあります) |
| ADSQSLBE | ISPF スケルトン用の配布ライブラリー (SDSQSLBE ライブラリーにあります) |

表 7. QMF バージョン 12.1 配布ライブラリーとその説明 (続き)

| ライブラリー名 | 説明 |
|-------------|--|
| ADSQTLB n | ISPF テーブル用の配布ライブラリー (SDSQTLB n ライブラリーにあります) |
| ADSQUSR n | ユーザーが変更可能なサンプル・コード用の配布ライブラリー (SDSQUSR n ライブラリーにあります) |


プログラム・ディレクトリーには、SMP/E および QMF のすべてのターゲットと配布ライブラリー・データ・セットのサイズの見積もりが提示されています。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

関連情報:

 DB2 照会報告書作成プログラム・ライブラリー

QMF バージョン 12.1 プログラム・ディレクトリーに関する情報を参照してください。


VSAM パネル・ライブラリーのデータの設定

SMP/E のインストールが完了したら、QMF VSAM パネル・ライブラリー DSQPNLE にデータを追加します。

QMF VSAM パネル・ライブラリーにデータを追加するには、ジョブ SDSQSAPE(DSQ1EPNL) を編集して実行します。このジョブは、メンバー DSQPNLE を SDSQPVRE ターゲット・ライブラリーから VSAM パネル・データ・セット (DSQPNLE) にコピーします。このジョブは、SMP/E のインストールごとに 1 回実行してください。

SMP/E のインストールについて詳しくは、ご使用の QMF リリースおよび各国語に該当する QMF プログラム・ディレクトリーを参照してください。

関連情報:

 DB2 照会報告書作成プログラム・ライブラリー

QMF バージョン 12.1 プログラム・ディレクトリーに関する情報を参照してください。

記憶域のアドレッシング要件

SMP/E を使用して配布メディアから QMF バージョン 12.1 ライブラリーをコピーするために必要な記憶域の容量については、ご使用の QMF バージョンおよび各国語に該当する QMF プログラム・ディレクトリーを参照してください。

領域サイズを計画する場合は、初期化中にモジュールをロードするために必要な記憶域、および報告書操作のための仮想記憶域要件を考慮してください。

制約事項: TSO の場合にのみ、QMF 以外のアプリケーションを実行するために必要なスペースの容量を考慮してください。

関連情報:

 DB2 照会報告書作成プログラム・ライブラリー

QMF バージョン 12.1 プログラム・ディレクトリーに関する情報を参照してください。

初期化中にモジュールをロードするために必要な記憶域

これらの記憶域容量は、QMF の開始時に QMF モジュールをロードするために必要です。これら記憶域が必要になるのは、QMF モジュールがリンク・パック域に配置されない場合です。

リンク・パック域に特定のモジュールを配置することにより、QMF 操作のパフォーマンスが向上する場合があります。

- 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。モジュール DSQCTOPX (TSO と CICS の両方のインストール済み環境で使用される) および DSQCCI (TSO インストール済み環境のみで使用される) はこのカテゴリに分類されます。これらは 16 MB 未満のアドレスで実行する必要があります。
- 31 ビット・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。

DSQCTOPX および DSQCCI を除くすべてのモジュールは、31 ビット・アドレッシング・モードで実行できます。

ほとんどのロード・モジュールは再入可能です。

関連概念:

52 ページの『パフォーマンスを向上させるためのモジュールの移動』

リンク・パック域に特定のモジュールを配置することにより、QMF のパフォーマンスが向上する場合があります。ただし、リンク・パック域に配置できないモジュールもあります。

報告書操作のための仮想記憶域要件

お客様の具体的な要件は、ユーザーが通常作成する報告書のサイズや使用する書式設定オプションの複雑さに応じて、以下の TSO または CICS の要件よりも大きくなる可能性があります。

- TSO の報告書記憶域

QMF for TSO では、QMF照会の実行、および QMF 報告書データの保持に必要な仮想記憶域の最小限の大きさは、QMF ユーザーごとに 2 MB です。QMF for TSO では、報告書記憶域を次の 2 つの方法で割り振ることができます。

- DSQSBSTG プログラム・パラメーターを使用して、記憶域の最大容量を固定容量として指定するか、または TSO 領域の使用可能な記憶域のパーセンテージを指定します。

- DSQSRSTG プログラム・パラメーターを使用して、QMF 以外のアプリケーションのために TSO 領域に予約される記憶域の大きさを指定します。その後、QMF は、予約されていない容量の記憶域を必要に応じて動的に割り振ります。これは報告書操作のためです。

QMF for TSO においても、報告書で表示する必要がない限り、LOB データおよび XML データは 2 GB 境界より上の記憶域に保管されます。QMF コマンド、または LOB データおよび XML データを含む SQL ステートメントを処理するには、2 GB 境界より上の記憶域へのアクセス権限がユーザーに必要です。

- CICS の報告書記憶域

QMF は、CICS 環境で会話型トランザクションとして稼働します。この環境では、1 つの CICS アドレス・スペース内で QMF の複数ユーザーが存在しています。拡張 CICS 動的記憶域 (ECSDA) から 16 MB の仮想記憶域を準備し、アクティブな QMF ユーザーごとにこの領域から 2 MB の仮想記憶域を追加します。QMF の開始時に DSQSBSTG パラメーターを指定して、各ユーザーごとに報告書記憶域を設定します。

QMF トランザクション専用の CICS 領域に、QMF トランザクション処理を分離して稼働することを考慮してください。その理由は、QMF for CICS は大規模な会話型トランザクションだからです。使用可能な記憶域量によっては、1 つの CICS 領域で QMF を実行できるユーザー数に上限ができてしまいます。より多くの QMF ユーザーをサポートするには、CICS 複数領域操作により、複数の CICS 領域を使用してください。QMF トランザクションを 1 つの CICS システム (例えば、端末所有領域) から QMF トランザクションを処理するために指定された CICS システム (例えば、アプリケーション所有領域) に、経路指定をすることが必要になる場合があります。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。

関連概念:

199 ページの『固定量の報告書用仮想記憶域の定義』

DSQSBSTG パラメーターを使用して、QMF が報告書操作で使用する仮想記憶域の固定バイト数を指定します。

関連タスク:

199 ページの『QMF 以外のアプリケーションに使用するための TSO 領域からの仮想記憶域の予約』

QMF for TSO の DSQSRSTG パラメーターを使用して、TSO コマンド、REXX、または ISPF などの QMF 以外のアプリケーションで使用するために TSO 領域に予約する仮想記憶域の最大容量を指定します。この予約された記憶域は、QMF 報告書操作に使用されなくなります。

パフォーマンスを向上させるためのモジュールの移動

リンク・パック域に特定のモジュールを配置することにより、QMF のパフォーマンスが向上する場合があります。ただし、リンク・パック域に配置できないモジュールもあります。

SMP/E を使用して QMF ライブラリーを配布メディアからアンロードすると、ライブラリー QMF1210.SDSQLOAD に QMF のロード・モジュールが入ります。

下表には、パフォーマンス向上のためにリンク・パック域に移動可能なモジュールを記載してあります。

表 8. 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 DSQCCISW DSQCBST DSQCELTT DSQCEBLT DSQCIX DSQQMRRS DSQUOPTS DSQABA1E | これらのモジュールは、QMF 実行時に使用されます。 モジュール DSQCCISW およびモジュール DSQCIX が使用されるのは、QMF for TSO だけです。 |
| DSQUEDIT DSQUECIC | これらのモジュールは、カスタム出口ルーチンに関連しており、データをフォーマット設定するためのユーザー定義編集コードを呼び出します。頻繁に使用することが予想される場合、それらをリンク・パック域に移動してください。 DSQUEDIT は TSO 用のユーザー編集ルーチン・モジュールです。一方、DSQUECIC は CICS 用のユーザー編集ルーチン・モジュールです。 ユーザー定義編集コードをサポートするカスタム書式設定ルーチンの作成について詳しくは、321 ページの『第 16 章 QMF 書式のためのカスタム編集出口ルーチン』を参照してください。 |
| DSQUEGV3 | このモジュールは、CICS の英語管理プログラム・モジュールです。 管理プログラムについて詳しくは、357 ページの『第 17 章 QMF リソースの制御』を参照してください。 |

下表には PLPA に置くことができるモジュールをリストしてあります。これらのモジュールは、24 ビットのアドレッシング・モードで実行する必要があります。

表 9. PLPA に置くことができるモジュール

| モジュール | 説明 |
|----------|--|
| DSQCTOPX | QMF は、QMF for TSO/CICS の両方の初期化中にこのモジュールを使用します。 |
| DSQCCI | QMF は、ISPF コマンド・インターフェース (QMF for TSO のみ) にこのモジュールを使用します。 このインターフェースについて詳しくは、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。 |

下表には、リンク・パック域に配置してはならないモジュールを記載してあります。この表の最初の 3 行にリストされたモジュールは、再入可能ではありません。

表 10. リンク・パック域に配置してはならないモジュール

| モジュール | 説明 |
|---|---|
| DSQCI | このモジュールは、QMF の呼び出し時に使用されます。TSO にのみ使用されます。 |
| DSQUEGV1 | このモジュールは、TSO の英語管理プログラム・モジュールです。 管理プログラムについて詳しくは、357 ページの『第 17 章 QMF リソースの制御』を参照してください。 |
| DSQ0BINS DSQ0BSQL DSQ0BTDC | これらのモジュールは、インストールとサービス・アップデート中に使用される QMF ユーティリティーです。QMF サービスの適用について詳しくは、429 ページの『QMF サービスの適用』を参照してください。これらのモジュールは、TSO にのみ使用されます。 |
| DSQUXIA DSQUXIC DSQUXILE DSQUXIP | これらのモジュールは、ユーザー編集ルーチン・プログラム DSQUEDIT (TSO) または DSQUECIC (CICS) のリンク・エディット中に使用されます。 ユーザー編集出口ルーチンについて詳しくは、321 ページの『第 16 章 QMF 書式のためのカスタム編集出口ルーチン』を参照してください。) |
| DSQCIB (COBOL) DSQCICX (C/370™) DSQCIA (アセンブラー) DSQCIFE (FORTRAN) DSQCIF (FORTRAN) DSQCIPX (PL/I) DSQCIPL (PL/I) DSQCIX (REXX) | QMF 呼び出し可能インターフェースは、これらのモジュールを使用します。これらは再入可能で、EPLPA に配置できます。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・モジュールと一緒にリンク・エディットされます。 |

リクエスターのインストール用にインストール・ジョブ・パラメーターを設定する方法

各 QMF インストールで、一連のバッチ・インストール・ジョブを実行します。ジョブを実行する前に、ジョブで使用するパラメーターおよび変数値をカスタマイズする必要があります。

このタスクについて

インストール・デフォルト EXEC である DSQ1DEFS のデフォルト値を変更するか、各インストール・ジョブの実行前に値をオーバーライドすることができます。バッチ・インストール・ジョブはすべて、QMF1210.SDSQSAPE データ・セットのメンバーです。

共通パラメーターのインストール・デフォルト

QMF1210.SDSQEXCE(DSQ1DEFS) EXEC のデフォルト値。

下の表には、QMF 提供のインストール・デフォルトがリストされています。これらのデフォルト値は、リクエスター・インストールで最も一般的に使用されるパラメーターに指定される値です。それ以外のパラメーターの説明に関しては、DSQ1DEFS EXEC のジョブ・コメントを見てください。

表 11. DSQ1DEFS EXEC で指定されるデフォルトのインストール値

| DSQ1DEFS EXEC 内の変数名 | QMF 提供の デフォルト | 説明 |
|--|--------------------------|--|
| SSID | DSN | QMF をインストールしているローカルの DB2 for z/OS リクエスター・データベースの 4 文字の名前。 |
| VCATNAME | QMFDSN | 新しいインストールでのみ (移行ではない) QMF STOGROUP の作成に使用される QMF 表スペース・カタログの別名。 |
| VOLUMES | '**' | 新しいインストールでのみ (移行ではない) STOGROUP の作成に使用される VOLUMES パラメーターの値。 |
| QMFPLNAME | QMF1210 | QMF アプリケーション・プランの名前。 |
| DBPTGROUP | 0 | DB2 UDB サーバーにインストールするための CREATE DATABASE PARTITION GROUP ステートメントの DBPARTITIONNUM 値。 |
| DBCCSID | UNICODE | DB2 for z/OS サーバーでの CREATE DATABASE ステートメントのデフォルト CCSID 値。 |
| DSQWLMNM | | QMF ストアード・プロシージャおよびユーザー定義関数を定義するために使用されるデフォルト WLM 環境。 |
| (以下の変数については、注 1 を参照してください。) | | |
| TSCT1PRI TSCT1SEC | 200 20 | DSQTSCT1 表スペースは、 Q.OBJECT_DIRECTORY 表を保持します。 |
| TSCT2PRI TSCT2SEC | 200 20 | DSQTSCT2 表スペースは、 Q.OBJECT_REMARKS 表を保持します。 |
| TSCT3PRI TSCT3SEC | 5000 200 | DSQTSCT3 表スペースは、 Q.OBJECT_DATA 表および Q.OBJECT_DATA2 表を保持します。 |
| TSPROPRI TSPROSEC | 100 20 | DSQTSPRO 表スペースは、 Q.PROFILES 表を保持します。 |
| TSLOGPRI TSLOGSEC | 100 20 | DSQTSLOG 表スペースは、 Q.ERROR_LOG 表を保持します。 |
| TSSYNPRI TSSYNSEC | 100 20 | DSQTSSYN 表スペースは、 Q.COMMAND_SYNONYMS 表を保持します。 |
| TSGOVPRI TSGOVSEC | 100 20 | DSQTSGOV 表スペースは、 Q.RESOURCE_TABLE 表を保持します。 |
| TSRDOPRI TSRDOSEC | 12 4 | DSQTSRDO 表スペースは、 Q.DSQ_RESERVED 表を保持します。 |
| TSDEFPRI TSDEFSEC | 100 20 | DSQTSDEF 表スペースは、 QMF SAVE DATA コマンドの結果、得られる表を保持します。オプションのジョブ DSQ1STGJ がこの表スペースを作成し、これらのデフォルト値を使用します。 |
| TSGLVPRI TSGLVSEC TSGLVBP TBGLVVARVALUE | 100 20 BP0 2000 | DSQTSGLV 表スペースは、 Q.GLOBAL_VARS 表を保持します。 TSGLVPRI パラメーターおよび TSGLVSEC パラメーターは、 DSQTSGLV のデフォルトの 1 次および 2 次スペース割り振りを指定します。 TSGLVBP は、 DSQTSGLV のデフォルト・バッファ・プールを指定します。 TBGLVVARVALUE は、 Q.GLOBAL_VARS 内の VARVALUE 列のデフォルト VARCHAR 長を指定します。 |
| (以下の変数については、注 2 を参照してください。) | | |
| IXODRPRI IXODRSEC | 200 20 | これらの変数は Q.OBJECT_DIRECTORY のインデックスである Q.OBJECT_DIRECTORYX のスペース数を指定します。 |
| IXORMPRI IXORMSEC | 200 20 | これらの変数は Q.OBJECT_REMARKS のインデックスである Q.OBJECT_REMARKSX のスペース数を指定します。 |
| IXODTPRI IXODTSEC | 200 20 | これらの変数は Q.OBJECT_DATA のインデックスである Q.OBJECT_OBJDATA のスペース数を指定します。 |

表 11. DSQ1DEFS EXEC で指定されるデフォルトのインストール値 (続き)

| DSQ1DEFS EXEC 内の変数名 | QMF 提供の デフォルト | 説明 |
|------------------------|---------------------|---|
| IXOPROPRI IXOPROSEC | 200 20 | これらの変数は Q.PROFILES 表のインデックスである Q.PROFILEX のスペース数量を指定します。 |
| IXCOMPRI IXCOMSEC | 100 20 | これらの変数は Q.COMMAND_SYNONYMS 表のインデックスである Q.COMMAND_SYNONYMSX のスペース数量を指定します。 |
| IXGLVPRI IXGLVSEC | 100 20 | これらの変数は Q.GLOBAL_VARS 表のインデックスである Q.GLOBAL_VARSX のスペース数量を指定します。 |
| SECAUTH | デフォルトなし | お客様サイトで、RACF セキュリティー・グループを使用し、そのために 2 次許可 ID の下で QMF を DB2 for z/OS データベース (ローカルまたはリモートのいずれか) にインストールすることが必要な場合は、この変数を使用してその ID を指定します。指定する 2 次許可 ID は、QMF のインストール対象となるターゲット・データベースに対して、SYSADM または同等の権限が必要です。 この変数の値を指定しない場合、QMF はパッケージとプランをバインドするインストール・ステップで、2 次許可 ID を処理しようとしません。 |

注:

- このセクションの変数は、QMF コントロール表に対してデフォルトの 1 次スペースと 2 次スペースの割り振り (1 KB 単位で表示) を指定します。QMF コントロール表がインストールされるのは、QMF の以前のリリースがデータベースに存在していなかった場合だけです。名前に PRI がある変数は 1 次スペース割り振りを示します。名前に SEC がある変数は 2 次スペース割り振りを示します。これらの変数の値は、一般的に変更されません。
- このセクションの変数は、QMF コントロール表のインデックスに対してデフォルトの 1 次スペースと 2 次スペースの割り振り (1 KB 単位で表示) を指定します。インデックスが作成されるのは、QMF の以前のリリースがデータベースに存在していなかった場合だけです。

すべての QMF インストール・ジョブが DSQ1DEFS EXEC のすべての値を使用するわけではありません。このインストール・ジョブには、参照される各 DSQ1DEFS 値が明確に記述されています。DSQ1DEFS 値がジョブで参照されない場合、その値は無視されます。

関連資料:

497 ページの『TSO および CICS での QMF コントロール表および表スペース』QMF に付属のコントロール表は以下のとおりです。

リクエストター・インストール用のサイト固有のインストール・デフォルトを **EXEC** で設定

インストール・デフォルト EXEC を変更して、デフォルト・インストール値を変更します。

このタスクについて

各インストール・ジョブでの大半のデータベース・インストール値のデフォルト値は、QMF1210.SDSQEXCE(DSQ1DEFS) EXEC にあります。インストール・ジョブはこの EXEC からデフォルト値を継承するため、これらの値をお客様サイトおよびインストールに適切な値に変更します。すべてのインストール・パラメーターが

DSQ1DEFS EXEC があるわけではないため、インストール・ジョブによってすべての値が継承されるわけではありません。デフォルト値を使用しない場合は、各ジョブでこれらのパラメーターの値をオーバーライドします。

重要: DSQ1DEFS から変数を削除しないでください。

手順

DSQ1DEFS EXEC の値を変更するためには、以下の手順を実行します。

ヒント: 同様のステップを実行することにより、各データベースのインストールに使用する DSQ1DEFS のコピーを変更し、その後、コピーされ、名前を変更されたファイルを使用して、そのデータベースのすべての QMF インストール・ジョブのデフォルトを指定することができます。

1. QMF1210.SDSQEXCE(DSQ1DEFS) をコピーして任意のメンバー名を付けます。
2. この新しいメンバーを編集し、55 ページの表 11 に示されるデフォルト値をサイト固有の値に変更します。例えば、以下のサンプル行では、新しいデフォルト (SSID に対して DB2L、LOCATION に対して ZOS1DB2L、VCATNAME に対して DB2LDSN) を指定しています。

```
SSID = "DB2L"  
LOCATION = "ZOS1DB2L"  
VCATNAME = "DB2LDSN"
```

3. ジョブの DSQDEFS DD ステートメントを変更して、変更済みの DSQ1DEFS のコピーをポイントするようにします。
4. ジョブの SYSTSIN ステートメントから変数を削除します。DSQ1INST 呼び出しの後の QMFBSQL 値はそのままにします。

例

次の例は、ジョブ DSQ1BLNM からの行であり、DB2 for z/OS バージョン 11.1 (DSN1110) を使用した場合の QMF バージョン 12.1 (QMF1210) のインストール・デフォルトを示しています。DB2 for z/OS バージョン 11.1 以外のバージョンを使用して QMF バージョン 12.1 をインストールしている場合は、そのバージョンの修飾子を使用してください。例えば、デフォルトの DB2 for z/OS バージョン 11 の高位修飾子は DSN1110 です。

```
//DSQ1BLNM JOBcard  
//DSQEXSQL PROC RGN='2048K',  
//          QMFTPRE='QMF1210',  
//          DB2EXIT='DSN1110.SDSNEXIT',  
//          DB2LOAD=DSN1110.SDSNLOAD'  
//STEP1 EXEC PGM=IKJEFT01,REGION=&RGN  
//STEPLIB DD DSN=&QMFTPRE..SDSQLLOAD,DISP=SHR  
//          DD DSN=&DB2EXIT.,DISP=SHR  
//          DD DSN=&DB2LOAD.,DISP=SHR  
//SYSTSPRT DD SYSOUT=*,DCB=BLKSIZE=132  
//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(DSQ1BNFM),DISP=SHR  
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLND),DISP=SHR  
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLNA),DISP=SHR  
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLNR),DISP=SHR  
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLNO),DISP=SHR
```

```
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLNE),DISP=SHR
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLC),DISP=SHR
//          PEND
//DSQCTBL EXEC DSQEXSQL
//* =====
/* Tailor SSID and LOCATION values below:
/* =====
//STEP1.SYSTSIN DD *
%DSQ1INST QMFBSQL <-- Variables have been removed after this line
```

インストール・ジョブのデフォルトのオーバーライド

以下の JCL 例は、SYSTSIN ステートメントでパラメーター値を変更することにより、継承されたデフォルト値をインストール・ジョブ内でオーバーライドする方法を示しています。

QMF インストール・ジョブは、DSQ1DEFS EXEC から値を継承します。各ジョブには、カスタマイズする必要がある値と、ジョブの実行依頼方法について、数多くの指示コメントが付けられています。

この例では、以下のデフォルト値が変更されています。

- ssid のデフォルトは、DB2L に変更されました。
- location のデフォルトは、ZOS1DB2L に変更されました。
- vcatname のデフォルトは、DB2LDSN に変更されました。

```
//DSQ1BLNM JOBcard
//DSQEXSQL PROC RGN='2048K',
//          QMFTPRE='QMF1210',
//          DB2EXIT=DSN1110.SDSNEXIT',
//          DB2LOAD=DSN1110.SDSNLOAD'
//STEP1 EXEC PGM=IKJEFT01,REGION=&RGN
//STEPLIB DD DSN=&QMFTPRE..SDSQLOAD,DISP=SHR
//          DD DSN=&DB2EXIT.,DISP=SHR
//          DD DSN=&DB2LOAD.,DISP=SHR
//SYSTSPRT DD SYSOUT=*,DCB=BLKSIZE=132
//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(DSQ1BNFM),DISP=SHR
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLND),DISP=SHR
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLNA),DISP=SHR
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLNR),DISP=SHR
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLNO),DISP=SHR
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLNE),DISP=SHR
//          DD DSN=&QMFTPRE..SDSQSAPE(DSQ1BLC),DISP=SHR
//          PEND
//DSQCTBL EXEC DSQEXSQL
//* =====
/* Tailor SSID and LOCATION values below:
/* =====
//STEP1.SYSTSIN DD*
%DSQ1INST QMFBSQL SSID(DB2L) LOCATION(ZOS1DB2L) +
          VCATNAME(DB2LDSN) VOLUMES('*')
```

すべてのインストール・パラメーターが DSQ1DEFS EXEC の中にあるわけではありません。このため、各インストール・ジョブがすべての値を継承できるわけではありません。継承されない以下のパラメーターが QMF インストール・ジョブにより使用されます。デフォルト値を使用したくない場合、実行前に各ジョブの中で指定する必要があります。

表 12. インストール・デフォルト・ファイル *DSQ1DEFS* 内がないインストール・ジョブのパラメーター

| インストール変数 | デフォルト値 | 説明 |
|----------|------------------|---|
| QMFTPRE | QMF1210 | QMF ターゲット・ライブラリーの接頭部 |
| DB2EXIT | DSN1210.SDSNEXIT | ローカル DB2 for z/OS の出口ライブラリー デフォルト値は、DB2 for z/OS バージョン 12 出口ライブラリーを指します。DB2 for z/OS バージョン 12 以外のバージョンを使用して QMF バージョン 12.1 をインストールしている場合は、そのバージョンの修飾子を使用してください。例えば、デフォルトの DB2 for z/OS バージョン 9 の高位修飾子は DSN910 です。 |
| DB2LOAD | DSN1210.SDSNLOAD | ローカル DB2 for z/OS ロード・ライブラリー デフォルト値は、DB2 for z/OS バージョン 12 出口ライブラリーを指します。DB2 for z/OS バージョン 12 以外のバージョンを使用して QMF バージョン 12.1 をインストールしている場合は、そのバージョンの修飾子を使用してください。例えば、デフォルトの DB2 for z/OS バージョン 9 の高位修飾子は DSN910 です。 |

インストールの準備に関連する作業の完了後、選択したインストール・パスに関連付けられているインストール手順に戻って、インストールの次のステップを判別してください。

リクエスター (DB2 for z/OS) データベース用インストール・ジョブの実行

インストール・ジョブを実行して、DB2 for z/OS リクエスター・データベースに QMF バージョン 12.1 をインストールするか、移行する必要があります。

旧リリースがインストールされていない場合に QMF V12.1 をインストールするジョブ

このインストール・パスは、旧リリースの QMF がインストールされていない場合に、QMF バージョン 12.1 を DB2 for z/OS リクエスター・データベースにインストールします。

始める前に

- これらの作業を行う前に、正しいインストール・パスに従っていることを確認してください。このトピックの手順は、『インストール・パス A: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースへの QMF V12.1 のインストール』に関連付けられています。
- データベースが DB2 for z/OS バージョン 9.1 新機能モード 以降であることを確認してください。データベース・リリースがバージョン 9.1 新機能モード よりも前の場合は、作業を続行する前にデータベースをアップグレードしてください。DB2 10 は、バージョン 8 からの変換モード、バージョン 8 からの変換モード*、バージョン 8 からの新機能有効化モード、およびバージョン 8 からの新機能有効化モード* (CM8、CM8*、ENFM8、または ENFM8*) で実行することはできません。

このタスクについて

重要: 本書、ジョブ自体、または IBM ソフトウェア・サポートによる指示がない限り、プランおよびインストール・パッケージの QMF バインド・ジョブ内で BIND PACKAGE または BIND PLAN オプションを変更または追加することはできません。

手順

- 各ジョブの JCL 先頭部分の指示に従うか、サイト固有のインストール・デフォルトを設定するか、またはインストール・デフォルトをオーバーライドすることにより、必要に応じてジョブのパラメーター値をカスタマイズします。その後、指定された順番でジョブを実行してください。下表にリストされたジョブは、QMF1210.SDSQSAPE データ・セットのメンバーとして存在します。

表 13. 旧リリースの QMF が存在しない場合に QMF バージョン 12.1 を DB2 for z/OS バージョン 9 新機能モード以降のデータベースにインストールするためのジョブ・シーケンス

| ジョブ名 | 説明 |
|----------|---|
| DSQ1TBAJ | オプション: QMF VCAT 名を作成します。 |
| DSQ1BSQL | QMF インストール・プログラムを DB2 for z/OS リクエスター・データベースにバインドします。 2 次許可 ID で QMF をインストールする場合は、このジョブを編集して OWNER(secauth) パラメーターまたは SET CURRENT SQLID='secauth' 指定のいずれかを BIND PACKAGE および BIND PLAN ステートメントに追加することで、QMF がその ID を解決できるようにします。 DSQ1DEFS exec の SECAUTH パラメーターは、このジョブで指定する SECAUTH の値と一致する必要があります。 |
| DSQ1BLNI | QMF バージョン 12.1 コントロール表を作成します。 |

表 13. 旧リリースの QMF が存在しない場合に QMF バージョン 12.1 を DB2 for z/OS バージョン 9 新機能モード以降のデータベースにインストールするためのジョブ・シーケンス (続き)

| ジョブ名 | 説明 |
|----------|---|
| DSQ1BVW | QMF LIST コマンドをサポートするデフォルト視点を作成します。 |
| DSQ1BPKG | QMF アプリケーション・パッケージをバインドします。 |
| DSQ1BINR | QMF アプリケーション・プランをバインドします。 2 次許可 ID で QMF をインストールする場合は、このジョブを編集して OWNER(secauth) パラメーターまたは SET CURRENT SQLID='secauth' 指定のいずれかを BIND PACKAGE および BIND PLAN ステートメントに追加することで、QMF がその ID を解決できるようにします。 DSQ1DEFS exec の SECAUTH パラメーターは、このジョブで指定する SECAUTH の値と一致する必要があります。 |
| DSQ1STGJ | SAVE DATA コマンドに必要な記憶域グループ、データベース、および表スペースを作成します。このコマンドは、QMF インストール検査手順 (IVP) で使用されます。 このジョブを実行する前に、TSDEFPRI および TSDEFSEC 変数をこのインストール用に正しく設定してください。詳細については、54 ページの『リクエスターのインストール用にインストール・ジョブ・パラメーターを設定する方法』を参照してください。 |
| DSQ1EIVS | オプション: QMF サンプル表を作成します。 |

- 各ジョブを実行した後で、戻りコード 0 または 4 を検査してください。両方のコードは正常に実行が完了したことを示します。

次のタスク

すべてのジョブが正常に実行された後で、23 ページの『インストール・パス A: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースへの QMF V12.1 のインストール』に戻り、インストール・プロセスの次ステップを実行してください。

関連タスク:

147 ページの『拡張 LIST コマンド機能のインストール (z/OS のみ)』

拡張 LIST コマンド機能がインストールされている場合、表と視点に対する特権は PUBLIC の代わりに、ユーザーの 1 次または 2 次許可 ID にのみ付与する必要があります。この許可 ID は、OWNER=ALL オプションを LIST TABLES または LIST ALL コマンドに指定した場合にこのコマンドで生成されるオブジェクト・リストに含まれています。

QMF V8 NFM、QMF V9 NFM、QMF V10、QMF V11.1、または QMF V11.2 を QMF V12.1 に移行するジョブ

このインストール・パスは、リクエスターとして、またはリクエスターとサーバーの両方として機能する DB2 for z/OS データベースで、QMF 新機能モード、QMF

バージョン 10、QMF バージョン 11.1、または QMF バージョン 11.2 のリリースを QMF バージョン 12.1 に移行します。

始める前に

- これらの作業を行う前に、正しいインストール・パスに従っていることを確認してください。このトピックの手順は、『インストール・パス B: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースでの QMF V11.2、QMF V11.1、QMF V10、または QMF NFM リリースから QMF V12.1 への移行』に関連付けられています。
- 既存のリリースの QMF が、バージョン 8.1 新機能モード、バージョン 9.1 新機能モード、バージョン 10、バージョン 11.1、またはバージョン 11.2 であることを確認してください。
- データベースが DB2 for z/OS バージョン 9.1 新機能モード 以降であることを確認してください。データベース・リリースがバージョン 9.1 新機能モード よりも前の場合は、作業を続行する前にデータベースをアップグレードしてください。DB2 10 は、バージョン 8 からの変換モード、バージョン 8 からの変換モード*、バージョン 8 からの新機能有効化モード、およびバージョン 8 からの新機能有効化モード* (CM8、CM8*、ENFM8、または ENFM8*) で実行することはできません。

このタスクについて

重要: 本書、ジョブ自体、または IBM ソフトウェア・サポートによる指示がない限り、プランおよびインストール・パッケージの QMF バインド・ジョブ内で BIND PACKAGE または BIND PLAN オプションを変更または追加することはできません。

手順

- 各ジョブの JCL 先頭部分の指示に従うか、サイト固有のインストール・デフォルトを設定するか、またはインストール・デフォルトをオーバーライドすることにより、必要に応じてジョブのパラメーター値をカスタマイズします。その後、指定された順番でジョブを実行してください。下表にリストされたジョブは、QMF1210.SDSQSAPE データ・セットのメンバーとして存在します。

表 14. QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モード、QMF バージョン 10、QMF バージョン 11.1、または QMF バージョン 11.2 から QMF バージョン 12.1 に移行するためのジョブ・シーケンス

| ジョブ名 | 説明 |
|----------|--|
| DSQ1BSQL | QMF インストール・プログラムを DB2 for z/OS リクエスター・データベースにバインドします。 2 次許可 ID で QMF をインストールする場合は、このジョブを編集して OWNER(<i>secauth</i>) パラメーターまたは SET CURRENT SQLID='secauth' 指定のいずれかを BIND PACKAGE および BIND PLAN ステートメントに追加することで、QMF がその ID を解決できるようにします。 |

表 14. QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モード、QMF バージョン 10、QMF バージョン 11.1、または QMF バージョン 11.2 から QMF バージョン 12.1 に移行するためのジョブ・シーケンス (続き)

| ジョブ名 | 説明 |
|----------|---|
| DSQ1BGLV | 新しい QMF バージョン 12.1 Q.GLOBAL_VARS コントロール表を作成します。 例外: QMF 11.1 以降から移行する場合、DSQ1BGLV を実行する必要はありません。 |
| DSQ1ERTS | 新しいコマンド同義語 RUNTSO および RU を Q.COMMAND_SYNONYMS に挿入し、DPRE 定義を更新します。 |
| DSQ1BVW | QMF LIST コマンドをサポートするデフォルト QMF 視点を作成します。拡張 LIST コマンド機能をインストールした場合は、DSQ1BVW の実行後に DSQ1BUDV インストール・ジョブを実行します。 |
| DSQ1BPKG | QMF パッケージにバインドします。このジョブでは、DSQDEFS DD ステートメントはオプションです。 |
| DSQ1BINR | QMF アプリケーション・プランをリクエスター・データベースでバインドします。 2 次許可 ID で QMF をインストールする場合は、このジョブを編集して OWNER(secauth) パラメーターまたは SET CURRENT SQLID='secauth' 指定のいずれかを BIND PACKAGE および BIND PLAN ステートメントに追加することで、QMF がその ID を解決できるようにします。DSQ1DEFS exec の SECAUTH パラメーターは、このジョブで指定する SECAUTH の値と一致する必要があります。 |

- 各ジョブを実行した後で、戻りコード 0 または 4 をチェックしてください。両方のコードは正常に実行が完了したことを示します。
- 旧リリースの QMF のサンプル表が必要な場合は、これらのジョブを実行します。旧リリースの QMF サンプル表は、新バージョンの QMF でも有効です。
 - 旧サンプル表が以前にインストールされておらず、新バージョンでそれらをインストールしたい場合は、ジョブ DSQ1EIVS を実行します。
 - 旧サンプル表が以前にインストールされており、それらを再インストールしたい場合は、まずジョブ DSQ1EDSJ を実行して、既存のサンプルを削除します。次にジョブ DSQ1EIVS を実行します。

次のタスク

すべてのジョブが正常に実行された後で、24 ページの『インストール・パス B: DB2 for z/OS スタンドアロン・データベースまたはリクエスター・データベースでの QMF V11.2、V11.1、V10 または QMF NFM リリースから QMF V12.1 への移行』に戻り、インストール・プロセスの次ステップを実行してください。

関連タスク:

54 ページの『リクエスターのインストール用にインストール・ジョブ・パラメーターを設定する方法』

各 QMF インストールで、一連のバッチ・インストール・ジョブを実行します。ジョブを実行する前に、ジョブで使用するパラメーターおよび変数値をカスタマイズ

する必要があります。

147 ページの『拡張 LIST コマンド機能のインストール (z/OS のみ)』
拡張 LIST コマンド機能がインストールされている場合、表と視点に対する特権は PUBLIC の代わりに、ユーザーの 1 次または 2 次許可 ID にのみ付与する必要があります。この許可 ID は、OWNER=ALL オプションを LIST TABLES または LIST ALL コマンドに指定した場合にこのコマンドで生成されるオブジェクト・リストに含まれています。

TSO および CICS に対するプログラム、リソースおよびプリファレンスの定義

インストール後に、サポートしている製品やコンポーネント (ISPF および GDDM など) の中で、特定のリソースの定義と割り振り、さらにプリファレンスのカスタマイズを行う必要があります。必要なカスタマイズは、QMF が実行される環境によって異なります。

このタスクについて

重要: QMF ライブラリーのいずれも、QMF が TSO または CICS のどちらかで実行する機能に対して、APF 許可を必要としません。

TSO でのリクエスター・インストールのカスタマイズ

QMF for TSO を実行している場合は、インストール後にカスタマイズを行う必要があります。GDDM を構成して、ログオン・プロシーチャーを準備します。

GDDM 外部デフォルトのカスタマイズ

QMF for TSO を実行している場合、データベースからの情報を GDDM で正しく表示するには、いくつかの外部デフォルトをカスタマイズする必要があります。

このタスクについて

QMF は、GDDM で提供されるサービスを通じて、画面から情報を受け取り、画面に情報を表示します。

DB2 が送信したデータが、GDDM により正しく表示されるには、GDDM コード・ページが現行データベース・マネージャーのコード化文字セット ID (CCSID) と一致することが必要です。CCSID には、文字の意味と表現を割り当てるのに必要な情報、およびそれらを保持するのに必要な情報がすべて含まれています。これは、この ID の処理と変換を行うさまざまな段階を経由して行われます。この情報には常に、少なくとも 1 つのコード・ページが含まれています。また、CCSID には、さまざまなコード・ポイントを処理する方法を管理する関連コード化スキームも含まれています。

QMF は、GDDM アプリケーション・コード・ページ (APPCPG) を次の 2 つの異なる CCSID 値と比較します。

- CURRENT APPLICATION ENCODING SCHEME 特殊レジスター

データベースのコード化スキームは、DB2 for z/OS の特殊レジスターで指定されます。1 バイト文字セット (SBCS) 言語の場合、GDDM アプリケーション・コード・ページはこの特殊レジスターの値と一致する必要があります。

- SYSIBM.SYSTEM_EBCDIC_CCSID セッション変数

このセッション変数は、現行システムの EBCDIC データに対して使用中の CCSID を指定します。2 バイト文字セット (DBCS) 言語の場合、GDDM アプリケーション・コード・ページはこのセッション変数の値と一致する必要があります。

CCSID が一致しない場合、表示用に DB2 に送信されるものと、GDDM で表示されるもの間で差異が発生する可能性があります。

必要な場合、GDDM APPCPG パラメーター値を以下のいずれかの方法で修正してください。

- GDDM 外部デフォルト・ファイル ADMADFT を変更してから、ファイルのアセンブルとリンク・エディットを行います。このファイルが、初期化中に QMF にアクセス可能であることを確認してください。
- 以下に示すように、GDDM デフォルトを含むデータ・セットを作成し、割り振ります。

手順

次の例は、APPCPG パラメーターを値 037 に変更する方法を示しています。この値は、DB2 for z/OS で使用している CCSID と互換性があります。

1. レコード長 80 で、QMF1210.ADMDEFS という名の固定長ブロック化順次ファイルを作成します。

QMF1210 は、QMF バージョン 12.1 ライブラリーのデフォルトの接頭部です。QMF ライブラリーのデフォルトの接頭部を変更した場合、QMF1210 の代わりに、その接頭部を置換してください。

2. 編集用に ADMDEFS ファイルを開き、次の ADMMDFT ステートメントを挿入して、APPCPG パラメーターに値 037 を指定します。2 桁目から始まるステートメントを挿入してください。

```
ADMMDFT APPCPG=037
```

1 桁目から始まる行を挿入すると、その行は自動的にコメントとみなされます。

3. ファイルを保存し、次の DD ステートメントを QMF を開始するために使用する下記 JCL に追加してください。

```
//ADMDEFS DD DSN=QMF1210.ADMDEFS,DISP=SHR
```

QMF ライブラリーのデフォルトの接頭部を変更した場合、QMF1210 の代わりに、その接頭部を置換してください。TSO のデータ・セットの割り振りについて詳しくは、66 ページの『TSO ログオン・プロシーチャーの準備』を参照してください。

また、次の TSO コマンドを使用してファイルを割り振ることもできます。

```
ALLOC FI(ADMDEFS) DS(QMF1210.ADMDEFS) SHR
```

次のタスク

APPCPG パラメーターの値を変更するとともに、使用している端末エミュレーターのコード・ページが、使用されている他の CCSID と整合していることも確認してください。

ユーザーがユーロ通貨記号の表示または印刷を必要としている場合、APPCPG パラメーターに割り当てる値はこの記号のコード・ページに関連付けられていることを確認してください。例えば、英語出力でユーロ通貨記号を表示する場合、適切な APPCPG 値として 1140 を使用します。QMF 稼働用の言語として使用する適切な APPCPG 値を選択して、その値を APPCPG パラメーターに割り当てます。

関連情報:

 [IBM Publications Center](#)

外部デフォルト・ファイルの変更方法に関する情報を GDDM 資料で検索します。

QMF デバイスが照会可能であることの確認

GDDM は、QMF オペレーションで使用されるすべてのデバイスの装置特性を照会できる必要があります。

QMF は、GDDM サービスを使用して、パネルおよびデータの表示を管理します。デバイス固有形式にして各データ・ストリームの作成と変換を行うために、GDDM は QMF 操作に使用する全デバイスの装置特性を照会できる必要があります。

各デバイスを照会可能なことを確認するには、VTAM[®] MODEENT マクロの PSERVIC オペランドの各バイトを検査します。これらのバイトは、表示デバイスまたはプリンター・タイプを定義します。その場合、照会可能デバイスと照会不能デバイスでは設定が異なります。

関連情報:

 [IBM Publications Center](#)

PSERVIC オペランドの有効な値に関する情報を GDDM 資料で検索します。

TSO ログオン・プロシージャーの準備

ユーザーの記憶域やその他の要件をサポートするために TSO ログオン・プロシージャーを変更する必要があります。この手順では、ロード・ライブラリーおよびデータ・セットの割り振りを行います。

このタスクについて

端末モニター・プログラム (TMP) は、ユーザーとディスプレイ・デバイス間の基本的なインターフェースであり、ユーザーの TSO セッション中に使用されます。ご使用のサイトは、それ独自の TMP を使用している場合と IBM 提供の標準の TMP を使用している場合があります。TMP が標準でない場合は、このトピックの説明は適用できない場合もあります。

TSO にログオンするたびに、TMP は TSO ログオン・プロシージャーを呼び出します。TSO ログオン・プロシージャーは、TSO セッション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、QMF を使用しない TSO ユーザー

より多くのリソースを必要とします。ログオン・プロシージャーを使用して、適切な TSO 環境の確立に必要なこの追加リソースを確実に提供してください。

IBM は、必要に応じて使用と変更が可能なサンプル・ログオン・プロシージャー (名前は DSQ1EINV) を提供します。このサンプル・ログオン・プロシージャーは、単に QMF にアクセスする手段として 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 QMF へのアクセス以上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・プロシージャーで割り振られるリソースの一部は、QMF を呼び出す CLIST または REXX EXEC でも割り振ることができます。

TSO 下の QMF オペレーション用のリソースを設定するには、次のようにしてサンプルの TSO ログオン・プロシージャーを変更してください。

手順

1. QMF1210.SDSQSAPE(DSQ1EINV) を編集します。
2. REGION パラメーターを探し出し、それが 50 ページの『記憶域のアドレッシング要件』の説明のように最小記憶域要件を満たしていることを確認します。以下に例を示します。

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

3. プログラム・ロード・ライブラリーを割り振ります。ISPF、ISPF/PDF、QMF、DB2 for z/OS、および GDDM のプログラム・ロード・ライブラリーは、QMF の始動前に、STEPLIB ステートメントで指定するか、または CLIST を使用して使用可能でなければなりません。このステップで、各種製品のロード・ライブラリーがリストされ、割り振りステップの例が示されます。
 - a. STEPLIB ステートメントまたは CLIST のどちらかを使用してプログラム・モジュールを割り振りたいかを定める。必要に応じて、QMF ユーザー出口ライブラリー (QMF1210.SDSQEXIT) を STEPLIB 連結に追加します。このステップが必要となるのは、QMF1210.SDSQEXIT に何らかの出口ルーチンがある場合だけです。サンプルのログオン・プロシージャーには、ISPF、ISPF/PDF、QMF、DB2 for z/OS、および GDDM のロード・ライブラリーが含まれます。これらのすべてのライブラリーを STEPLIB ステートメントに含める必要があるわけではありません。一部のライブラリーは、あとで CLIST を使用して割り振ることができます。QMF の開始前に、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。次の図は、DSQ1EINV ログオン・プロシージャーの中のロード・ライブラリー用 STEPLIB ステートメントを示します。

```

//*****
//*          PROGRAM LOAD LIBRARIES          *
//*****
//STEPLIB DD DSN=QMF1210.SDSQEXIT,DISP=SHR      * QMF MODULES *
//          DD DSN=QMF1210.SDSQLOAD,DISP=SHR    * QMF MODULES *
//          DD DSN=ISP.SISPLOAD,DISP=SHR        * ISPF MODULES *
//          DD DSN=DSN1110.SDSNEXIT,DISP=SHR    * DB2 MODULES *
//          DD DSN=DSN1110.SDSNLOAD,DISP=SHR    * DB2 MODULES *
//          DD DSN=GDDM.SADMMOD,DISP=SHR        * GDDM MODULES *

```

図 8. DSQ1EINV ログオン・プロシーチャーのロード・ライブラリー

STEPLIB ステートメントではなく、ISPLLIB DD ステートメントを使用しても、ISPF ロード・ライブラリーを割り振ることができます。

- b. 異なるバージョンが共存する QMF が、同じ DB2 for z/OS サブシステム内で実行させたいかを決めます。2 つの異なるバージョンの QMF が共存する場合、各リリースごとに個別のログオン・プロシーチャーを使用して、QMF ロード・ライブラリーを割り振る必要があります。

下の表に、QMF バージョン 12.1、および移行がサポートされる旧リリースの QMF のロード・モジュール・ライブラリー名を示します。

| 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 12 リリース 1.0 | QMF1210.SDSQLOAD |
| バージョン 11 リリース 2.0 | QMF1120.SDSQLOAD |
| バージョン 11 リリース 1.0 | QMF1110.SDSQLOAD |
| バージョン 10 リリース 1.0 | QMF1010.SDSQLOAD |
| バージョン 9 リリース 1.0 | QMF910.SDSQLOAD |
| 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 リリース 1.0 | QMF610.SDSQLOAD |
| バージョン 3 リリース 3.0 | QMF330.DSQLOAD |

4. SYSEXEC または SYSPROC に SDSQEXCE を割り振ります。SYSEXEC への割り振りを下図に示します。SYSPROC、SYSEXEC、および SYSHELP 割り振りステートメントは、デフォルトのデータ・セット名を示します。このデータ・セット名は、お客様のシステム環境では別の名前になっている可能性があります。

各プログラムに対する TSO 検索順序として、お客様のシステム環境で設定済みの DD ステートメントを使用します。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。REXX プログラムに対してお客様システム環境での検索順位がわからない場合は、SYSEXEC および SYSPROC の両方に対して SDSQEXCE を割り当ててください。

```

//*****
//*          DATA SETS USED BY TSO          *
//*****
//SYSPROC DD DSN=SYS2.CLIST,DISP=SHR
//          DD DSN=QMF1210.SDSQCLTE,DISP=SHR
//SYSEXEC DD DSN=SYS2.EXEC,DISP=SHR
//          DD DSN=QMF1210.SDSQEXCE,DISP=SHR
//SYSHELP DD DSN=SYS1.HELP,DISP=SHR
//EDT     DD DSN=&EDIT,UNIT=SYSDA,SPACE=(1688,(40,12))
//UTL     DD DSN=&SYSUT1,UNIT=SYSDA,SPACE=(TRK,(10,5))
//SYSPRINT DD TERM=TS,SYSOUT=A
//SYSTEM DD TERM=TS,SYSOUT=A
//SYSIN   DD TERM=TS

```

図 9. ログオン・プロシージャ内での SDSQEXCE の割り振り

- 適宜、ISPF ライブラリーをカスタマイズします。ISPF ライブラリーはオプションです。ISPF で QMF を開始した場合、または ISPF に関連した機能を使用した場合、次の図に示すように、これらのライブラリーを割り振ります。図に示された名前は ISPF ライブラリーのデフォルト名です。これらの名前はお客様固有の ISPF システム環境とは異なっている場合があります。

```

//*****
//*          DATASETS USED BY ISPF          *
//*****
//ISPPLIB DD DSN=QMF1210.SDSQPLBE,DISP=SHR
//          DD DSN=ISP.SISPPENU,DISP=SHR
//ISPMLIB DD DSN=QMF1210.SDSQMLBE,DISP=SHR
//          DD DSN=ISP.SISPMENU,DISP=SHR
//ISPPLIB DD DSN=QMF1210.SDSQSLBE,DISP=SHR
//          DD DSN=ISP.SISPSENU,DISP=SHR
//          DD DSN=ISP.SISPLIB,DISP=SHR
//ISPTLIB DD DSN=QMF1210.SDSQTLBE,DISP=SHR
//          DD DSN=ISP.SISPTENU,DISP=SHR
//ISPPROF DD UNIT=SYSDA,SPACE=(TRK,(9,1,4)),
//          DCB=(LRECL=80,BLKSIZE=8800,RECFM=FB,DSORG=PO)

```

図 10. ログオン・プロシージャでの ISPF ライブラリーの割り振り

- GDDM データ・セットを検査します。下表には、各 GDDM データ・セットの名前とその目的がリストされています。

表 15. GDDM データ・セット

| データ・セット名 | 説明 |
|----------|--|
| ADMGGMAP | QMF によってマップされるパネルの GDDM マップ・グループ デフォルトのマップ・ライブラリーは、QMF1210.SDSQMAPE です。 |
| ADMCFORM | QMF 提供の図表形式。図表機能が必要な場合にのみ使用します。 デフォルトの図表ライブラリーは、QMF1210.SDSQCHRT です。 |

表 15. GDDM データ・セット (続き)

| データ・セット名 | 説明 |
|----------|---|
| DSQUCFRM | ユーザー定義の図表オブジェクト QMF for TSO 下で GDDM-PGF を使用時にユーザー独自の図表形式を作成済みだったユーザーは、このデータ・セットから図表オブジェクトを検索したり、そこに図表オブジェクトを保存することができます。 |
| ADMSYMBL | GDDM 提供のシンボル・セット。図表機能が必要な場合にのみ使用します。 |
| ADMGDF | 図形データ形式ファイル。図表機能が必要な場合にのみ使用します。 |
| ADMCDATA | 図表データ・ファイル。図表機能が必要な場合にのみ使用します。 |
| ADMDEFS | GDDM 外部デフォルト (APPCPG パラメーターのデフォルトなど) の設定を含めるようにこのファイルを変更した場合は、DD 名 ADMDEFS を割り振ります。印刷用に GDDM サービスを使用する場合、ADMDEFS ファイルは GDDM ニックネームも保管します。 |

下図には、DSQ1EINV プロシージャーに指定された、これらのデータ・セットの割り振りステートメントが記載してあります。

```

//*****
//*          QMF/GDDM DATA SETS          *
//*****
//ADMGGMAP DD DSN=QMF1210.SDSQMAPE,DISP=SHR * GDDM Map Group
//ADMCFORM DD DSN=QMF1210.SDSQCHRT,DISP=SHR * QMF-Supplied Chart Formats
//DSQUCFRM DD DSN=aaaaaaa,DISP=SHR          * Saves User-defined ICUFORMS
//ADMCDATA DD DSN=xxxx,DISP=SHR
//ADMGDF DD DSN=xxxx,DISP=SHR
//ADMSYMBL DD DSN=xxxx,DISP=SHR

```

図 11. GDDM データ・セット用の割り振りステートメント

独自のプロシージャー内でこれらのデータ・セットを割り振るには、以下の手順に従ってください。

- a. 上に示すように、ADMGGMAP DD ステートメントが QMF ADMGGMAP ライブラリー QMF1210.SDSQMAPE を指していることを確認します。
- b. QMF for TSO 下で GDDM-PGF の実行時に作成済みのユーザー固有の図表形式を保存するユーザーごとに、個別のライブラリーを割り振ります。次の例のような DD ステートメントを使用して、新しいライブラリーを作成します。DSN、UNIT、VOL、および SPACE パラメーターの値は指定しますが、DCB パラメーターは変更しないでください。

```

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

```

- 1) DSQ1EINV で DSQUCFRM の項目を見つけます。
 - 2) DSQUCFRM に割り振られているデータ・セットごとにこの項目を複写およびカスタマイズします。
 - 3) 複写される各項目では、DSQUCFRM に割り振られている各データ・セットの名前で *aaaaaaaa* を置換します。
- 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ます。これらのデータ・セットが存在しない場合は、次のようなステートメントを使用してそれらを定義します。

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

7. QMF データ・セットを割り振ります。下表は、TSO の中で QMF が使用したデータ・セットをリストします。これらのファイルは、DSQ で始まる DD 名に割り振られています。これらのファイルを別に割り振りたい場合は、ログオン・プロシージャ CLIST でそれらを変更するか、QMF の起動に使用する exec で変更してください。

表 16. TSO の QMF で使用されるデータ・セット

| データ・セット | 説明 |
|----------|---|
| DSQPNLE | QMF VSAM パネル・ファイル このファイルは、あるリリースから次のリリースへ移るときに変更され、インストール・ジョブ QMF1210.SDSQSAPE(DSQ1EPNL) の実行時に作成されます。 |
| DSQDUMP | QMF スナップ・ダンプ出力 |
| DSQDEBUG | QMF トレース・ダンプ出力 このデータ・セットについて詳しくは、454 ページの『トレース機能』を参照してください。 |
| DSQPRINT | 印刷データ出力 |
| DSQSPILL | 予備データ・ファイル 予備データのファイルを割り振る代わりに、拡張記憶域にデータを書き出すことができます。詳細については、201 ページの『拡張仮想記憶域への報告書データの書き込み (TSO のみ)』を参照してください。 |
| DSQEDIT | 編集転送ファイル このファイルは、ISPF が編集サービスのために呼び出されている間に、EDIT コマンドで参照されるクエリーまたはプロシージャを一時的に保持します。 |

データ・セット DSQDEBUG および DSQDUMP はプリンターに対してデフォルト設定を行います。この定義をカスタマイズして、デフォルトを使用する代わりに、データ・セットにカスタマイズされた情報を送信できます。

DSQDUMP、DSQDEBUG、および DSQPRINT のすべての DD ステートメントには、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 値に 1 を追加します。

次の図は、データ・セットの DSQ グループのデフォルトの割り振りステートメントを示します。

```
//*****  
//*          DATA SETS USED BY QMF          *  
//*****  
//DSQPNLE DD DSN=QMF1210.DSQPNLE,DISP=SHR  
//DSQPRINT DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330)  
//DSQDEBUG DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210)  
//DSQEDIT DD UNIT=SYSVIO,DCB=(RECFM=FBA,LRECL=79,BLKSIZE=4029),  
// DISP=NEW,SPACE=(CYL,(1,1))  
//DSQDUMP DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=125,BLKSIZE=1632)  
//SYSUDUMP DD SYSOUT=A  
//DSQSPILL DD DSN=&&SPILL,DISP=(NEW,DELETE),  
// UNIT=SYSVIO,SPACE=(CYL,(10,20),RLSE),  
// DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)
```

図 12. QMF DSQ データ・セット用のデフォルトの割り振りステートメント

次のタスク

これらの作業が完了した後で、選択したインストール・パスに関連付けられているインストール手順に戻って、インストールの次のステップを判別してください。

関連概念:

226 ページの『QMF ユーザー・プロファイルの作成』

すべての QMF ユーザーは、ユーザー・プロファイルにアクセスする必要があります。このユーザー・プロファイルは、特定のユーザーからの個々の入力を 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 (例えば、プリンター出力の宛先、または入力の大文字変換の有無など) を制御します。

187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターについて説明します。

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

160 ページの『CALL ステートメントの形式』

このトピックでは、必要な CALL ステートメントの形式について説明します。

関連タスク:

274 ページの『GDDM サービスを使用した印刷の処理』
GDDM サービス (QMF 以外) を使用して、ネイティブ z/OS バッチ、
TSO、ISPF、および CICS で印刷を処理することができます。

関連資料:

5 ページの『リリースの共存』

QMF バージョン 12.1 は、QMF バージョン 8.1 新機能モード、QMF バージョン
9.1 新機能モードや QMF バージョン 10.1、QMF バージョン 11.1、または QMF
バージョン 11.2 とのみ、同じデータベース内に共存できます。

102 ページの『インストール検査手順に必要なサンプル照会およびサンプル・プロ
シーチャーのインストール』

TSO にのみ、サンプルの QMF 照会およびプロシーチャーをインストールします。

関連情報:

 IBM Publications Center

GDDM データ・セットおよび外部デフォルトに関する情報を GDDM 資料で検索
します。

CICS でのリクエスター・インストールのカスタマイズ

QMF for CICS のカスタマイズを行う前に、DB2 for z/OS および GDDM の両方
をインストールしてカスタマイズし、CICS 環境で実行できるようにする必要があります。

関連情報:

 DB2 資料

DB2 のカスタマイズに関する情報を検索します。

 IBM Publications Center

GDDM 資料を参照してください。

DB2 に対して QMF を記述

CICS 下で QMF を実行する前に、QMF を DB2 に対して記述する必要があります。

手順

1. DB2 for z/OS-to-CICS connection、および CICS 用の DB2 (z/OS 版) 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 for z/OS のデー
タにアクセスします。

2. QMF がインストールされているデータベースに対して、DB2CONN 定義が存
在することを確認します。 CICS オンライン・リソース定義 (RDO) 機能を使
用して、この定義を作成することができます。QMF に固有の DB2ENTRY 定
義が、DSQ1ECSD ジョブの実行時に作成されます。

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。

CICS に対して QMF を記述

CICS で QMF を実行する前に、QMF を CICS に記述する必要があります。これを行うには、CICS システム定義 (CSD) を更新するジョブを変更し、実行します。

このタスクについて

ジョブ DSQ1ECSD は QMF と呼ばれる新しいリストを作成します。これは、CICS システム定義ファイル (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 コントロール表を更新してください。また、DSQ1ECSD は DB2ENTRY も定義します。この DB2ENTRY は、正しいプラン名に関連付けられ、QMF トランザクション別にグループ化します。

手順

1. RDO VIEW Lsrpool *name* コマンドを使用して、LSRPOOL の現行定義を検査します。QMF パネル・データ・セット (DSQPNLE) では、32 K の VSAM CI サイズが必要です。QMF は明示的に LSRPOOL 項目を定義しません。代わりに、QMF は 1 という値の 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 32 KB に満たない場合は、DFHCSDUP を使用して、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. お客様サイトのニーズに合わせて、QMF1210.SDSQSAPE(DSQ1ECSD) を編集し、ジョブのインストリーム・プロシージャ内のインストール・パラメーターを検査または変更します。以下に例を示します。

```
//DSQ1ECSD PROC REG=2048K,           Job step region
//      QMFTPRE='QMF1210',           DSN prefix for QMF
//      CLOAD='CICS.SDFHLOAD',       Name of CICS program library
//      CCSD='CICS.DFHCSO',         Name of CICS CSD file
//      OUTC='*'                     Print sysout class
```

3. ジョブをサブミットし、戻りコード 0 でジョブの実行が完了したかをチェックします。ゼロ以外の戻りコードが戻された場合は、ジョブ出力を検査し、エラーを訂正してください。

CICS 用に管理プログラム出口を準備する方法

CICS で QMF を開始する前に、QMF に付属の管理プログラムを変換、アセンブル、およびリンク・エディットする必要があります。

手順

1. ジョブ QMF1210.SDSQSAPE(DSQ1EGLK) のコメントに従ってジョブを編集してください。
2. ジョブ QMF1210.SDSQSAPE(DSQ1EGLK) をサブミットします。
3. すべてのステップについて、戻りコード 0 を検査します (ただし、戻りコードが 4 となる可能性がある LKEDPROG は除く)。戻りコードが 0 または 4 ではない場合は、問題を訂正し、ジョブを再実行します。

関連概念:

357 ページの『第 17 章 QMF リソースの制御』

管理プログラム出口ルーチンにより、ユーザーのサイトでエンド・ユーザーのアクティビティーを制限し、システム・リソースの使用を制御できます。

CICS での GDDM サポートのカスタマイズ

CICS 下で GDDM をサポートするには、追加のカスタマイズが必要です。

GDDM 外部デフォルトのカスタマイズ:

QMF との対話が正しく行われるようにするには、いくつかの GDDM 外部デフォルトを ADMADFC ファイルに設定する必要があります。

以下の外部デフォルトを設定する必要があります。

IOSYNCH

IOSYNCH 外部デフォルトが必ず「YES」に設定されるようにします。

APPCPG

QMF は、GDDM で提供されるサービスを通じて、画面から情報を受け取り、画面に情報を表示します。DB2 が送信したデータが、GDDM により正しく表示されるには、GDDM コード・ページが現行データベース・マネージャーのコード化文字セット ID (CCSID) と一致することが必要です。CCSID には、文字の意味と表現を割り当てるのに必要な情報、およびそれらを保持するのに必要な情報がすべて含まれています。これは、この ID の処理と変換を行うさまざまな段階を経由して行われます。この情報には常に、少なくとも 1 つのコード・ページが含まれています。また、CCSID には、さまざまなコード・ポイントを処理する方法を管理する関連コード化スキームも含まれています。

QMF は、GDDM アプリケーション・コード・ページ (APPCPG) を次の 2 つの異なる CCSID 値と比較します。

- CURRENT APPLICATION ENCODING SCHEME 特殊レジスター

データベースのコード化スキームは、DB2 for z/OS の特殊レジスターで指定されます。1 バイト文字セット (SBCS) 言語の場合、GDDM アプリケーション・コード・ページはこの特殊レジスターの値と一致する必要があります。

- SYSIBM.SYSTEM_EBCDIC_CCSID セッション変数

このセッション変数は、現行システムの EBCDIC データに対して使用中の CCSID を指定します。2 バイト文字セット (DBCS) 言語の場合、GDDM アプリケーション・コード・ページはこのセッション変数の値と一致する必要があります。

CCSID が一致しない場合、表示用に DB2 に送信されるものと、GDDM で実際に表示されるもの間で差異が発生する可能性があります。必要な場合、GDDM 外部デフォルト・ファイル (ADMADFC) にある GDDM APPCPG パラメーター値を修正します。APPCPG パラメーターが 037 の場合は、DB2 for z/OS 上で使用中の CCSID と互換性があります。このファイルの変更が終了したら、ファイルのアSEMBルとリンク・エディットを行って、初期化時に QMF にアクセス可能であることを確認します。外部デフォルトの変更方法について詳しくは、IBM Publications Centerにある GDDM 資料を参照してください。

APPCPG パラメーターの値を変更するとともに、使用している端末エミュレーターのコード・ページが、使用されている他の CCSID と整合していることも確認してください。

ユーザーがユーロ通貨記号の表示または印刷を必要としている場合、APPCPG パラメーターに割り当てる値はこの記号のコード・ページに関連付けられていることを確認してください。例えば、英語出力でユーロ通貨記号を表示する場合、適切な APPCPG 値として 1140 を使用します。QMF 稼働用の言語として使用する適切な APPCPG 値を選択して、その値を APPCPG パラメーターに割り当てます。

QMF デバイスが照会可能であることの確認:

GDDM は、QMF オペレーションで使用されるすべてのデバイスの装置特性を照会できる必要があります。

QMF は、GDDM サービスを使用して、パネルおよびデータの表示を管理します。デバイス固有形式にして各データ・ストリームの作成と変換を行うために、GDDM は QMF 操作に使用する全デバイスの装置特性を照会できる必要があります。

各デバイスを照会可能なことを確認するには、VTAM MODEENT マクロの PSERVIC オペランドの各バイトを検査します。これらのバイトは、表示デバイスまたはプリンター・タイプを定義します。その場合、照会可能デバイスと照会不能デバイスでは設定が異なります。

関連情報:

 IBM Publications Center

PSERVIC オペランドの有効な値に関する情報を GDDM 資料で検索します。

GDDM ADMF データ・セットへの QMF GDDM マップのロード:

この手順では、データ・セットの既存のマップを置換します。

このタスクについて

ADMF データ・セットは、一度に 1 つだけの QMF リリースからマップを含むことができます。

手順

QMF バージョン 12.1 で定義されている GDDM マップを GDDM ADMF データ・セットにロードするには、以下の手順に従ってください。

1. QMF1210.SDSQSAPE(DSQ1EADM) を編集して、インストリーム・プロシージャのインストール・パラメーターが、このジョブ (ジョブ・ステップも同様) のニーズを満たしていることを確認してください。以下に例を示します。

```
//DSQ1EADM PROC RGN='2048K',      Job-step region size
//          QMFTPRE='QMF1210',    QMF prefix name for target libraries
//          GDDMADM='GDDM.ADMF'  GDDM ADMF data set name
```

2. QMF1210.SDSQSAPE(DSQ1EADM) をサブミットします。何らかの理由で、移行元の QMF のリリースからのマップに戻す必要がある場合は、ジョブ

QMF`Form.SDSQSAPE(DSQ1EADM)` を実行します。この場合、`orm` は、移行元のリリースのバージョン、リリース・レベル、およびモディフィケーション・レベルです。

3. 0 という戻りコードを検査する。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。

サンプル図表と **QMF** トレース・データ・セットの作成:

このジョブは、デフォルトの QMF 図表形式と QMF トレース・データ・セットを作成するステートメント (`DSQ1CFRM` 内) を実行します。

始める前に

QMF の旧リリースから QMF のバージョン 12.1 に移行する場合は、このステップをスキップしてください。

手順

1. `QMF1210.SDSQSAPE(DSQ1BFRM)` を編集します。
2. ジョブのインストリーム・プロシージャでインストール・パラメーターを見つけ、その値がお客様の要件に合致していることを確認します。以下に例を示します。

```
//DSQ1BFRM PROC QMFTPRE='QMF1210',   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` を編集します。これは、`DSQ1BFRM` ジョブの `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カスタマイズします。以下に例を示します。

```
DEFINE CLUSTER (NAME(QMF1210.DSQCFRM) -
                VOLUMES(QMFVOL) -
                UNIQUE -
                RECSZ(400 400) -
                CONTROLINTERVALSIZE(2048) -
                KEYS(20 0)) -
                DATA -
                (RECORDS(1000 300)) -
                CATALOG(VSAMUSERCAT)
```

5. `QMF1210.SDSQSAPE(DSQ1BFRM)` をサブミットします。
6. 0 という戻りコードを検査します。戻りコードが 0 でない場合は、次を実行します。
 - a. `DSQ1CFRM` を編集して正常に実行されたステップを削除します。そうしないと、オブジェクトが既に存在している旨のエラー・メッセージを受け取ります。
 - b. トレース・データ・セット (`DSQDEBUG`) にエラーがないかどうか検査します。このデータ・セットについて詳しくは、454 ページの『トレース機能』を参照してください。
 - c. 問題が見つかったら訂正し、再実行します。

CICS 始動ジョブ・ストリームの更新

CICS 始動ジョブ・ストリームの DD ステートメントを更新して、QMF の初期化中に正しいデータ・セットがアクセスされるようにします。

手順

1. QMF、GDDM、および DB2 for z/OS 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・リストである DFHRPL に配置します。以下に例を示します。

```
//DFHRPL DD ...  
//      DD DSN=QMF1210.SDSQEXIT,DISP=SHR  
//      DD DSN=QMF1210.SDSQLOAD,DISP=SHR  
//      DD DSN=GDDM.SADMMOD,DISP=SHR  
//      DD DSN=DSN1110.SDSNEXIT,DISP=SHR  
//      DD DSN=DSN1110.SDSNLOAD,DISP=SHR
```

DB2 出口とロード・ライブラリーを参照するステートメント上で、DB2 for z/OS のリリース・レベルが正しく指定されていることを確認してください。

2. 下記のデータ・セットにアクセスできるようにします。このデータ・セットは GDDM および QMF が要求とします。

```
//*      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=QMF1210.DSQPNLE,DISP=SHR  QMF panel file  
//DSQDEBUG DD DSN=QMF1210.DSQDEBUG,DISP=SHR Trace and error messages  
//DSQUCFRM DD DSN=QMF1210.DSQUCFRM,DISP=SHR User-defined ICU forms
```

3. CICS のシャットダウンと再始動を行って、変更を CICS テーブルと CICS 始動ジョブに組み込みます。

EXPORT コマンドおよび IMPORT コマンドで使用する記憶域タイプの判別

QMF バージョン 12.1 のインストール後は、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がアクティブになります。ただし、互換性の理由から、QMF バージョン 12.1 は引き続き、TSO データ・セットのサポートをアクティブにすることを許可します。

このタスクについて

QMF の旧リリースでは、QMF トランザクションから TSO データ・セットを直接使用できました。ただし、TSO データ・セットの使用により、QMF トランザクションを実行する CICS アドレス・スペースで予測不能な結果が生じる可能性があります。したがって、QMF EXPORT コマンドおよび IMPORT コマンドでは、CICS 一時記憶域または一時データ・キューを使用する必要があります。

手順

QMF IMPORT コマンドおよび EXPORT コマンドのために、TSO データ・セットのサポートをアクティブにするには、以下の手順に従ってください。

1. QMF エクスポート/インポート制御モジュール (DSQCTLXI) を使用不可にします。このモジュールを使用不可にするには、CICS に付属の CEMT トランザクションを使用します。

```
CEMT SET PROGRAM(DSQCTLXI) DISABLE
```

また、CICS CSD ファイルから削除することにより、DSQCTLXI を使用不可にできます。DSQCTLXI を使用不可にすると、CICS で実行中のすべての QMF セッションにおいて、EXPORT コマンドと IMPORT コマンドに TSO データ・セットを使用できるようになります。

2. CICS 記憶域保護を使用している場合 (SIT STGPROT=YES)、QMF モジュール DSQCBST の実行キーを CICS に設定します。記憶域保護が使用されているかどうかを判別するには、次のコマンドを発行してください。

```
CEMT INQUIRE SYSTEM
```

STOREPROTECT オプションが ACTIVE に設定されている場合、記憶域保護は使用されています。

DSQCBST モジュールの EXECKey オプションが USER に設定されている場合、次の手順を実行して、この値を CICS に変更します。

- a. 次のコマンドを実行します。

```
CEDA ALTER PROGRAM(DSQCBST)
```

- b. CICS 領域をリサイクルします。

- c. 次のコマンドを実行します。

```
CEMT INQUIRE PROGRAM(DSQCBST)
```

Prog オプションの隣にカーソルを置いて、Enter キーを押し、すべてのオプションを表示します。EXECKey オプションが Cexeckey に設定されていることを確認します。これは、CICS の値を示します。

次のタスク

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 コマンドを使用することによって、または CICS CSD ファイル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
```

CICS では、IMPORT コマンドの使用を控えめにします。これは、同じアドレス・スペース内のほかのユーザーの QMF パフォーマンスに影響を与える可能性があるからです。QMF は、QSAM で操作をするときに GET/PUT サービスを使用します。このサービスでは入出力操作中に、同じ CICS 領域内の他の QMF ユーザーをロックアウトしてしまう可能性があります。

最大 2 MB の SQL 照会を保管できるように EDSA の制限サイズを修正する

QMF バージョン 12.1 では、DB2 for z/OS を対象とした SQL 照会のサイズを、最大 2 MB までサポートします。

このタスクについて

DSQEC_SQLQRYSZ_2M パラメーターを使用して、記憶域の全容量の上限を定義します。この範囲内で、CICS が 16 MB 境界より上にある個々の拡張動的記憶域 (EDSA) を割り振ることが可能です。QMF for CICS で最大 2 MB の SQL 照会を実行するには、最小 62914560 の EDSALIMIT 値が必要です。

手順

このサイズの照会を実行する予定がある場合は、DSQEC_SQLQRYSZ_2M 大域変数を 1 に設定し、CICS EDSALIM パラメーターの値を変更してください。

次のタスク

これらの作業が完了した後で、選択したインストール・パスに関連付けられているインストール手順に戻って、インストールの次のステップを判別してください。

第 4 章 サーバー・データベースでの QMF のインストールまたは移行

このトピックは、リモート・サーバーに QMF バージョン 12.1 をインストールまたは移行する方法を説明します。リモート・サーバーは、DRDA 通信を通じて、ローカルの DB2 (z/OS 版) サブシステムからアクセス可能なサーバーです。

始める前に

リモート・サーバーに QMF バージョン 12.1 をインストールするか、移行する前に、必要な構成のための正しいインストール・パスに従っていることを確認するためには、26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』を参照してください。

このタスクについて

リモート作業単位、または分散作業単位のいずれかを使用してリモート・サーバーにアクセスするかどうかによって、リモート・サーバーのインストール・タスクは異なります。

関連概念:

5 ページの『分散データ・ネットワーク内の QMF』

QMF の最初のインストールは、DB2 for z/OS データベースに対して行う必要があります。このインストールが完了すると、DB2 for z/OS データベースは、スタンドアロンとして使用するか、リクエスターとして機能するか、あるいは他の QMF バージョン 12.1 インストール済み環境のリクエスターとサーバーの両方として機能します。

サーバー・インストールのためのインストール・ジョブ・パラメーターの設定

QMF の各インストールで、一連のバッチ・インストール・ジョブを実行します。ジョブを実行する前に、ジョブで使用するパラメーターおよび変数値をカスタマイズする必要があります。

このタスクについて

インストール・デフォルト EXEC である DSQ1DEFS のデフォルト値を変更するか、各インストール・ジョブの実行前に値をオーバーライドすることができます。バッチ・インストール・ジョブはすべて、QMF1210.SDSQSAPE データ・セットのメンバーです。

制約事項:

- 3 部構成の名前を含む QMF コマンドは、DB2 (VSE および VM 版) サーバーに送信することはできません。
- デフォルトでは、3 部構成の名前を含む QMF コマンドを使用して、ラージ・オブジェクト (LOB) データを含むリモート表にアクセスすることはできません。

3 部構成の名前を持つリモート表内の LOB データへのアクセスを可能にするには、DSQEC_LOB_RETRV 大域変数を 3 に設定する必要があります。また、DSQEC_LOB_RETRV を 2 に設定すると、LOB メタデータのみを検索が可能になります。

- QMF for TSO が、DB2 for z/OS リクエスター・データベースでストアード・プロシージャとして開始された場合、リモート・サーバーとの接続はサポートされません。

手順

インストールで使用されるパラメーターおよび変数値をカスタマイズするには、以下のいずれかの手順に従ってください。

- インストールのデフォルト EXEC である DSQ1DEFS のデフォルト値を変更します。

QMF は、共通パラメーター用にリストされているインストール・デフォルトに加えて、サーバー・インストール中に以下のパラメーターを使用します。

LOCATION

このパラメーターは、CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによりアクセスされるサーバーでのインストールに使用されます。デフォルト値 THISLOCN を QMF をインストールしているリモート・サーバーの名前と置換します。

T3PARTNM

このパラメーターは、『インストール・パス I: 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーの準備』に関連付けられています。

データが QMF コマンドで 3 部構成の名前のみを使用してリモート・サーバーでアクセスされる場合、このパラメーターに対して YES を指定します。QMF CONNECT コマンドと 3 部構成の名前を含むコマンドの両方を使用してリモート・サーバーのデータにアクセスすることにした場合、このパラメーターをブランクにしてください。CONNECT コマンドによりアクセスされるサーバーのインストール・パスは自動的に 3 部構成の名前のアクセスを提供するからです。

インストール・パラメーターについて詳しくは、DSQ1DEFS EXEC のジョブ・コメントを検討してください。

- ジョブを実行する前に、各ジョブ内の値をカスタマイズすることにより、デフォルト値をオーバーライドします。

重要: リモート・サーバーに送信されるすべてのインストール・ジョブ内の SSID パラメーターが、リモート・インストールの実行元となるローカル DB2 for z/OS サブシステムの ID に設定されていることを確認してください。

次のタスク

これらの作業が完了した後で、選択したインストール・パスに関連付けられているインストール手順に戻って、インストールの次のステップを判別してください。

関連概念:

54 ページの『共通パラメーターのインストール・デフォルト』
QMF1210.SDSQEXCE(DSQ1DEFS) EXEC のデフォルト値。

58 ページの『インストール・ジョブのデフォルトのオーバーライド』
以下の JCL 例は、SYSTSIN ステートメントでパラメーター値を変更することにより、継承されたデフォルト値をインストール・ジョブ内でオーバーライドする方法を示しています。

QMF CONNECT コマンドによりアクセスされるサーバーを準備するインストール・ジョブの実行

インストール・ジョブを実行して、サーバー・データベースに QMF バージョン 12.1 をインストールするか、移行する必要があります。

QMF for TSO が、DB2 for z/OS リクエスター・データベースでストアード・プロシージャーとして開始された場合、リモート・サーバーとの接続はサポートされません。

実行しているインストールのタイプに適切なトピックを選択してください。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

旧リリースがインストールされていないサーバー・データベースに QMF V12.1 をインストールするジョブ

このジョブ・シーケンスは、旧リリースの QMF がインストールされていない DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバーに QMF バージョン 12.1 をインストールします。

始める前に

- これらの作業を行う前に、正しいインストール・パスに従っていることを確認してください。この手順は、『インストール・パス C: DB2 for z/OS、iSeries、または LUW サーバーへの QMF V12.1 のインストール』に関連付けられています。
- データベースが以下のいずれかのリリースであることを確認します。必要であれば、続行する前にデータベースをアップグレードしてください。
 - DB2 for z/OS サーバーは、バージョン 9.1 新機能モード 以降である必要があります。DB2 10 は、バージョン 8 からの変換モード、バージョン 8 からの変換モード*、バージョン 8 からの新機能有効化モード、およびバージョン 8 からの新機能有効化モード* (CM8、CM8*、ENFM8、または ENFM8*) で実行することはできません。
 - DB2 for iSeries サーバーは、バージョン 5.4 以降である必要があります。
 - DB2 for Linux, UNIX, and Windows サーバーは、バージョン 9.5 以降である必要があります。

このタスクについて

制約事項: DB2 (VSE および VM 版) では、どのタイプの QMF バージョン 12.1 の新規インストールもサポートされません。DB2 (VSE および VM 版) サーバーを QMF CONNECT コマンドによるアクセスのために準備するには、以下のインストール・パスの 1 つを参照してください。

- 35 ページの『インストール・パス E: DB2 (VSE および VM 版) サーバーでの QMF V7.2 以前から V12.1 への移行』
- 37 ページの『インストール・パス F: DB2 (VSE および VM 版) サーバーでの QMF V8.1、V9.1、V10.1、V11.1、またはV11.2 から QMF V12.1 への移行』

重要: 本書、ジョブ自体、または IBM ソフトウェア・サポートによる指示がない限り、プランおよびインストール・パッケージの QMF バインド・ジョブ内で BIND PACKAGE または BIND PLAN オプションを変更または追加することはできません。

手順

- 各ジョブの JCL 先頭部分の指示に従うか、サイト固有のインストール・デフォルトを設定するか、またはインストール・デフォルトをオーバーライドすることにより、必要に応じてジョブのパラメーター値をカスタマイズします。その後、指定された順番でジョブを実行してください。下表にリストされたジョブは、QMF1210.SDSQSAPE データ・セットのメンバーとして存在します。

表 17. 旧リリースの QMF がインストールされていないリモート DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバーに QMF バージョン 12.1 をインストールするためのジョブ・シーケンス

| ジョブ名 | 説明 |
|----------|--|
| DSQ1TBAJ | オプション: QMF VCAT (DB2 for z/OS サーバーのみ) を作成する |
| DSQ1BSQL | DB2 for z/OS リクエスター・データベースからの QMF インストール・プログラムをサーバー・データベースにバインドします 2 次許可 ID で QMF をインストールする場合は、このジョブを編集して OWNER(secauth) パラメーターまたは SET CURRENT SQLID='secauth' 指定のいずれかを BIND PACKAGE および BIND PLAN ステートメントに追加することで、QMF がその ID を解決できるようにします。DSQ1DEFS exec の SECAUTH パラメーターは、このジョブで指定する SECAUTH の値と一致する必要があります。 |
| DSQ1BLNI | QMF バージョン 12.1 コントロール表を作成します |
| DSQ1BVW | QMF LIST コマンドをサポートするデフォルト視点を作成します |
| DSQ1BPKG | QMF パッケージをバインドします。 |

表 17. 旧リリースの QMF がインストールされていないリモート DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバーに QMF バージョン 12.1 をインストールするためのジョブ・シーケンス (続き)

| ジョブ名 | 説明 |
|----------|--|
| DSQ1STGJ | DB2 for z/OS サーバーのみ: QMF SAVE DATA コマンドの記憶域グループ、データベース、および表スペースを作成します SAVE DATA コマンドは、インストール検査手順中に使用されます。 |
| DSQ1EIVS | オプション: DB2 for z/OSサーバーのみ: QMF サンプル表を作成します |
| DSQ1EDJ4 | オプション: DB2 for Linux, UNIX, and Windowsサーバーのみ: QMF サンプル表を作成します |
| DSQ1EAS4 | オプション: DB2 for iSeries のみ: QMF サンプル表を作成します |

- 各ジョブを実行した後で、戻りコード 0 または 4 を検査してください。両方のコードは正常に実行が完了したことを示します。

次のタスク

すべてのジョブが正常に実行された後で、30 ページの『インストール・パス C: DB2 for z/OS、iSeries、または LUW サーバーへの QMF V12.1 のインストール』に戻り、インストール・プロセスの次ステップを実行してください。

関連タスク:

147 ページの『拡張 LIST コマンド機能のインストール (z/OS のみ)』

拡張 LIST コマンド機能がインストールされている場合、表と視点に対する特権は PUBLIC の代わりに、ユーザーの 1 次または 2 次許可 ID にのみ付与する必要があります。この許可 ID は、OWNER=ALL オプションを LIST TABLES または LIST ALL コマンドに指定した場合にこのコマンドで生成されるオブジェクト・リストに含まれています。

z/OS、iSeries、または LUW サーバーで QMF V8 NFM、QMF V9 NFM、QMF V10、QMF V11.1、または QMF V11.2 を QMF V12.1 に移行するジョブ

このジョブ・シーケンスは、QMF 新機能モード、QMF バージョン 10、QMF バージョン 11.1、または QMF バージョン 11.2 のリリースを QMF バージョン 12.1 に移行します。

始める前に

- これらの作業を行う前に、正しいインストール・パスに従っていることを確認してください。この手順は、『インストール・パス E: DB2 for z/OS、iSeries、または LUW サーバーでの QMF V11.2、V11.1、V10、または QMF NFM リリースから QMF V12.1 への移行』に関連付けられています。
- データベースが以下のいずれかのリリースであることを確認します。必要であれば、続行する前にデータベースをアップグレードしてください。

- DB2 for z/OS サーバーは、バージョン 9.1 新機能モード 以降である必要があります。DB2 10 は、バージョン 8 からの変換モード、バージョン 8 からの変換モード*、バージョン 8 からの新機能有効化モード、およびバージョン 8 からの新機能有効化モード* (CM8、CM8*、ENFM8、または ENFM8*) で実行することはできません。
- DB2 for iSeries サーバーは、バージョン 5.4 以降である必要があります。
- DB2 for Linux, UNIX, and Windows サーバーは、バージョン 9.5 以降である必要があります。

このタスクについて

重要: 本書、ジョブ自体、または IBM ソフトウェア・サポートによる指示がない限り、プランおよびインストール・パッケージの QMF バインド・ジョブ内で BIND PACKAGE または BIND PLAN オプションを変更または追加することはできません。

手順

- 各ジョブの JCL 先頭部分の指示に従うか、サイト固有のインストール・デフォルトを設定するか、またはインストール・デフォルトをオーバーライドすることにより、必要に応じてジョブのパラメーター値をカスタマイズします。その後、指定された順番でジョブを実行してください。下表にリストされたジョブは、QMF1210.SDSQSAPE データ・セットのメンバーとして存在します。

表 18. DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバーで QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モード、QMF バージョン 10、QMF バージョン 11.1、または QMF バージョン 11.2 から QMF バージョン 12.1 に移行するためのジョブ・シーケンス

| ジョブ名 | 説明 |
|----------|--|
| DSQ1BSQL | DB2 for z/OS リクエスター・データベースからの QMF インストール・プログラムをサーバー・データベースにバインドします 2 次許可 ID で QMF をインストールする場合は、このジョブを編集して OWNER(secauth) パラメーターまたは SET CURRENT SQLID='secauth' 指定のいずれかを BIND PACKAGE および BIND PLAN ステートメントに追加することで、QMF がその ID を解決できるようにします。DSQ1DEFS exec の SECAUTH パラメーターは、このジョブで指定する SECAUTH の値と一致する必要があります。 |
| DSQ1BGLV | 新しい QMF バージョン 12.1 Q.GLOBAL_VARS コントロール表を作成します。 例外: QMF 11.1 以降から移行する場合、DSQ1BGLV を実行する必要はありません。 |
| DSQ1ERTS | 新しいコマンド同義語 RUNTSO および RU を Q.COMMAND_SYNONYMS に挿入し、DPRE 定義を更新します。 |

表 18. DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows サーバーで QMF バージョン 8.1 新機能モード、QMF バージョン 9.1 新機能モード、QMF バージョン 10、QMF バージョン 11.1、または QMF バージョン 11.2 から QMF バージョン 12.1 に移行するためのジョブ・シーケンス (続き)

| ジョブ名 | 説明 |
|----------|--|
| DSQ1BVW | QMF LIST コマンドをサポートするデフォルト QMF 視点を作成します。拡張 LIST コマンド機能をインストールした場合は、DSQ1BVW の実行後に DSQ1BUDV インストール・ジョブを実行します。 |
| DSQ1BPKG | QMF パッケージをバインドします。 |

- 各ジョブを実行した後で、戻りコード 0 または 4 をチェックしてください。両方のコードは正常に実行が完了したことを示します。
- 旧リリースの QMF のサンプル表が必要な場合は、これらのジョブを実行します。旧リリースの QMF サンプル表は、新バージョンでも有効です。
 - 旧サンプル表がインストールされておらず、新バージョンでそれらをインストールしたい場合は、次の表の該当するジョブを実行します。

表 19. サンプル表をインストールするジョブ

| ジョブ名 | 説明 |
|----------|--|
| DSQ1EIVS | DB2 for z/OS サーバーにサンプル表をインストールします。 |
| DSQ1EAS4 | DB2 for iSeries サーバーにサンプル表をインストールします。 |
| DSQ1EDJ4 | DB2 for Linux, UNIX, and Windows サーバーにサンプル表をインストールします。 |

- 旧サンプル表が以前にインストールされており、それらを再インストールしたい場合は、まず既存のサンプルを削除します。
- ジョブ DSQ1EDSJ は、DB2 for z/OS データベースからこれらのサンプルを削除します。
- ジョブ DSQ1EDX2 は、DB2 for iSeries データベースおよび DB2 for Linux, UNIX, and Windows データベースからこれらを削除します。

次のタスク

すべてのジョブが正常に実行された後で、33 ページの『インストール・パス D: DB2 for z/OS、iSeries、または LUW サーバーでの QMF V11.2、QMF V11.1、QMF V10、または QMF NFM リリースから QMF V12.1 への移行』に戻り、インストール・プロセスの次ステップを実行してください。

QMF V7.2 以前のバージョンを VM または VSE サーバーの QMF V12.1 に移行するジョブ

このジョブ・シーケンスは、DB2 (VSE および VM 版) サーバーで QMF バージョン 7.2 以前を QMF バージョン 12.1 に移行します。

始める前に

- これらの作業を行う前に、正しいインストール・パスに従っていることを確認してください。この手順は、『インストール・パス G: DB2 (VSE および VM 版) サーバーでの V7.2 以前から QMF V12.1 への移行』に関連付けられています。
- DB2 (VSE および VM 版) がバージョン 7.3 以降であることを確認します。

手順

- 各ジョブの JCL 先頭部分の指示に従うか、サイト固有のインストール・デフォルトを設定するか、またはインストール・デフォルトをオーバーライドすることにより、必要に応じてジョブのパラメーター値をカスタマイズします。その後、指定された順番でジョブを実行してください。下表にリストされたジョブは、QMF1210.SDSQSAPE データ・セットのメンバーとして存在します。

表 20. DB2 (VSE および VM 版) サーバーで QMF バージョン 7.2 以前を QMF バージョン 12.1 に移行するためのジョブ・シーケンス

| ジョブ名 | 説明 |
|----------|--|
| DSQ1BSQL | DB2 for z/OS リクエスター・データベースからの QMF インストール・プログラムをサーバー・データベースにバインドします 2 次許可 ID で QMF をインストールする場合は、このジョブを編集して OWNER(secauth) パラメーターまたは SET CURRENT SQLID='secauth' 指定のいずれかを BIND PACKAGE および BIND PLAN ステートメントに追加することで、QMF がその ID を解決できるようにします。DSQ1DEFS exec の SECAUTH パラメーターは、このジョブで指定する SECAUTH の値と一致する必要があります。 |
| DSQ1BVW | QMF LIST コマンドをサポートするデフォルト視点を作成します |
| DSQ1BPKG | QMF パッケージをバインドします。 |

- 各ジョブを実行した後で、戻りコード 0 または 4 を検査してください。両方のコードは正常に実行が完了したことを示します。
- 必要であれば、QMF for VSE/VM の資料に従って QMF サンプル表をインストールします。旧リリースの QMF サンプル表は、新バージョンの QMF でも有効です。

次のタスク

すべてのジョブが正常に実行された後で、35 ページの『インストール・パス E: DB2 (VSE および VM 版) サーバーでの QMF V7.2 以前から V12.1 への移行』に戻り、インストール・プロセスの次ステップを実行してください。

関連情報:

 IBM Publications Center

QMF サンプル表のインストールに関する情報を QMF (VM または VSE 版) の資料で検索します。

VM または VSE サーバーで QMF V8、V9、V10、V11.1、または V11.2 を QMF V12.1 に移行するジョブ

このジョブ・シーケンスは、DB2 (VSE および VM 版) サーバーで既存の QMF バージョン 8.1、バージョン 9.1、バージョン 10、バージョン 11.1、またはバージョン 11.2 インストール済み環境を QMF バージョン 12.1 に移行します。

始める前に

- これらの作業を行う前に、正しいインストール・パスに従っていることを確認してください。この手順は、『インストール・パス H: DB2 (VSE および VM 版) サーバーでの QMF V8.1、V9.1、V10、V11.1、または V11.2 から QMF V12.1 への移行』に関連付けられています。
- DB2 (VSE および VM 版) がバージョン 7.3 以降であることを確認します。

このタスクについて

重要: 本書、ジョブ自体、または IBM ソフトウェア・サポートによる指示がない限り、プランおよびインストール・パッケージの QMF バインド・ジョブ内で BIND PACKAGE または BIND PLAN オプションを変更または追加することはできません。

手順

- 各ジョブの JCL 先頭部分の指示に従うか、サイト固有のインストール・デフォルトを設定するか、またはインストール・デフォルトをオーバーライドすることにより、必要に応じてジョブのパラメーター値をカスタマイズします。その後、指定された順番でジョブを実行してください。下表にリストされたジョブは、QMF1210.SDSQSAPE データ・セットのメンバーとして存在します。

表 21. DB2 (VSE および VM 版) サーバーで既存の QMF バージョン 8.1、バージョン 9.1、バージョン 10、バージョン 11.1、またはバージョン 11.2 インストール済み環境を QMF バージョン 12.1 に移行するためのジョブ・シーケンス

| ジョブ名 | 説明 |
|----------|---|
| DSQ1BSQL | DB2 for z/OS リクエスター・データベースからの QMF インストール・プログラムをサーバー・データベースにバインドします。 2 次許可 ID で QMF をインストールする場合は、このジョブを編集して OWNER(secauth) パラメーターまたは SET CURRENT SQLID='secauth' 指定のいずれかを BIND PACKAGE および BIND PLAN ステートメントに追加することで、QMF がその ID を解決できるようにします。DSQ1DEFS exec の SECAUTH パラメーターは、このジョブで指定する SECAUTH の値と一致する必要があります。 |
| DSQ1BPKG | QMF パッケージにバインドします。 |

- 各ジョブを実行した後で、戻りコード 0 または 4 をチェックしてください。両方のコードは正常に実行が完了したことを示します。

- 必要であれば、QMF for VSE/VM の資料に従って QMF サンプル表をインストールします。旧リリースの QMF サンプル表は、新バージョンの QMF でも有効です。

関連情報:

 IBM Publications Center

QMF サンプル表のインストールに関する情報を QMF (VM または VSE 版) の資料で検索します。

サーバー・データベースでの旧リリースへのフォールバック

QMF 互換モード・リリース (QMF バージョン 3.3、6.1、7.1、7.2、バージョン 8.1 互換モード、またはバージョン 9.1 互換モード) から QMF バージョン 12.1 に移行中にエラーが検出された場合は、旧リリースに戻すことができます。

手順

QMF コントロール表とその表スペースを元の移行前の状態に戻すには、以下の手順に従ってください。

1. QMF コントロール表を元の構造で再作成します。以下のいずれかの方法を使用できます。
 - 移行されたオブジェクトを除去し、それらの元の構造とデータを使用して再作成します。移行時に作成したバックアップを使用してください。
 - ALTER ステートメントを使用して、移行前に存在したオリジナルの指定と一致するようにオブジェクトの定義を変更します。
2. 元のイメージ・コピーをリカバリーします。QMF バージョン 12.1 の移行プロセスは元のコントロール表の名前を変更して、名前変更された表を除去するため、元のイメージ・コピーをリカバリーする必要があります。DB2 for z/OS サーバーの QMF を移行している場合、元のイメージ・コピーをリカバリーするために、RECOVER ユーティリティではなく、DSN1COPY ユーティリティを使用します。元のイメージ・コピーの OBID は新しく作成された表の OBID と一致しないため、OBIDLAT パラメーターが必要です。

DSN1COPY ユーティリティについて詳しくは、DB2 資料を参照してください。

注: DB2 バージョン 9 以降では SIMPLE 表スペースがサポートされますが、新しい表はセグメント表スペースでのみ作成されます。DB2 バージョン 9 より前に QMF 表が作成された場合は、DSN1COPY の代わりに、APAR PK60612 の修正が適用された DB2 UNLOAD ユーティリティを使用してください。新しい表がセグメントとして定義されていても、UNLOAD ユーティリティはこの修正を使用して、非セグメントだった表スペースのイメージ・コピーを処理することができます。詳しくは、「DB2 9 for z/OS: Using the Utilities Suite」の『Unloading from an image copy』セクションを参照してください。

3. オブジェクトのインデックスをリカバリーします。
4. 旧リリースから再作成されたオブジェクトのフルイメージ・コピーを取得します。

3 部構成の名前を含む QMF コマンドによりアクセスされるサーバーを準備するインストール・ジョブの実行

このジョブ・シーケンスは、QMF コマンドで 3 部構成の名前を使用してリモート・サーバーのデータにアクセスするように、リクエスターとサーバーの両方を準備します。

始める前に

- これらの作業を行う前に、正しいインストール・パスに従っていることを確認してください。この手順は、『インストール・パス I: 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーの準備』に関連付けられています。
- 3 部構成の名前を含む QMF コマンドに加えて、QMF CONNECT コマンドを使用することにより、リモート・サーバーでデータにアクセスする場合、26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』を参照して、インストール・パスを判別してください。

このタスクについて

制約事項:

- QMF for TSO が、DB2 for z/OS リクエスター・データベースでストアード・プロシージャとして開始された場合、リモート・サーバーとの接続はサポートされません。
- 3 部構成の名前を含む QMF コマンドは、DB2 (VSE および VM 版) サーバーに送信することはできません。
- デフォルトでは、3 部構成の名前を含む QMF コマンドを使用して、ラージ・オブジェクト (LOB) データを含むリモート表にアクセスすることはできません。3 部構成の名前を持つリモート表内の LOB データへのアクセスを可能にするには、DSQEC_LOB_RETRV 大域変数を 3 に設定する必要があります。また、DSQEC_LOB_RETRV を 2 に設定すると、LOB メタデータのみを検索が可能になります。
- 3 部構成の名前を含むコマンドを DB2 for z/OS に送信する場合を除き、QMF は複数行フェッチをオフにして開始する必要があります。

手順

QMF コマンドで 3 部構成の名前のみを使用して、QMF がインストールされていないリモート・サーバーのデータにアクセスするには、以下の手順を実行して、リクエスターとサーバーの両方を準備します。

1. リモート・サーバーで QMF1210.SDSQSAPE(DSQ1BSQL) を実行します。ローカル・データベースからこのジョブを実行します。
2. QMF1210.SDSQEXCE(DSQ1DEFS) を編集して、T3PARTNM パラメーターを YES に設定します。以下に例を示します。

```
T3PARTNM="YES"
```
3. ジョブ DSQ1BPKG の DSQDEFS DD ステートメントが、2 の手順で変更した DSQ1DEFS メンバーをポイントするようにしてください。

4. リモート・サーバーで QMF1210.SDSQSAPE(DSQ1BPKG) を実行します。ローカル・データベースからこのジョブを実行します。戻りコードが 0 または 4 について検査します。両方のコードは、実行が正常に行われたことを示します。

関連概念:

211 ページの『複数行のフェッチと挿入のサポートの使用可能化』
DSQSMRFI パラメーター。データベースがフェッチおよび挿入の際に処理する行が複数行か、単一行かを制御します。

関連タスク:

143 ページの『QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

第 5 章 QMF の正しいインストールの検査

QMF が正しくインストールされていることを検査する方法は、データベースのタイプによって異なります。

リクエスター・データベースへの QMF インストールのテスト

リクエスター・データベースでの QMF のインストールをテストするには、以下の手順に従ってください。

始める前に

- 選択したインストール・パスに関連するインストール手順の各ステップをすべて完了したことを確認します。
- テストの実行に必要な正しい権限を持っていることを確認します。

テストの実行に使用する許可 ID は、次の権限のいずれかを持っている必要があります。

- SYSADM (または同等の権限)
- 以下のデータベースに対する DBADM (または同等の権限)

DSQDBCTL

このデータベースには、QMF オブジェクト・カタログの QMF コントロール表が含まれています。

DSQDBDEF

このデータベースには、DSQTSDEF 表スペースが含まれています。この表スペースは、SAVE DATA コマンドが発行されたときに使用されるデフォルトのスペースです。

このタスクについて

20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』のインストール・パスには、次のテスト手順が適用されます。

CICS で QMF を使用する予定の場合は、TSO での手順をすべて実行してから、CICS 用の手順を再度実行してください。

QMF の開始

QMF の開始に使用する方法は、QMF を TSO または CICS のどちらで実行するかによって異なります。

始める前に

QMF を開始する前に、リクエスター・データベースが開始済みであることを確認します。リクエスターのインストールのテスト後にサーバー・データベースへの QMF

インストールを進める場合は、リクエスター・データベースを DDF で開始します。

このタスクについて

TSO 下での QMF の開始と CICS 下での QMF の開始には、別個の手順が用意されています。

タスクの結果

QMF の開始に失敗した場合、または QMF の開始時にエラーまたは警告が発生した場合は、考えられる原因について次のいずれかのトピックを参照してください。

- 430 ページの『初期化時に発生する可能性があるエラー』
- 441 ページの『QMF の開始後の警告メッセージ』

開始時に QMF が 1 つ以上前のリリースのロード・モジュールにアクセスしている場合は、初期化問題が発生する可能性があります。これが問題となっているかどうかを調査する方法については、431 ページの『QMF が現行のロード・モジュールを使用していないときに発生する問題』に関する説明を参照してください。

TSO での QMF の開始

ISPF で QMF を使用する場合は、ISPF SELECT サービスおよび ISPSTART コマンドを使用して QMF を開始できます。ISPF を使用しないで QMF を開始するには、DSQQMF n モジュールを使用できます (n は、1 文字の言語 ID)。

このタスクについて

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

関連概念:

187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターについて説明します。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

66 ページの『TSO ログオン・プロシージャの準備』

ユーザーの記憶域やその他の要件をサポートするために TSO ログオン・プロシージャを変更する必要があります。この手順では、ロード・ライブラリーおよびデータ・セットの割り振りを行います。

関連情報:

 IBM Publications Center

ISPF ダイアログに関する情報を ISPF 資料で検索します。

ISPF を使用した QMF の開始:

QMF の開始は、ISPSTART コマンドを使用して行うか、あるいは、QMF がメニュー選択の 1 つに組み込まれるように ISPF 選択メニューをカスタマイズして行うことができます。

手順

以下のいずれかの方法で、ISPF を使用して QMF を開始します。

- ISPSTART コマンドを使用するには、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 コマンドを実行します。

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 (QMF12) のデフォルト値を上書きする方法を示します。

```
ISPSTART PGM(DSQMFn) NEWAPPL(DSQn) PARM(DSQSSUBS=ssid,  
DSQSPLAN=planid,...)
```

このコマンドには、次の変数が含まれています。

- n** 適切な 1 文字の言語 ID を使用して、ユーザーの NLF 環境で QMF を開始します。例えば、英語環境で QMF を開始する場合、PGM パラメーターの値は DSQQMFE であり、NEWAPPL パラメーターの値は DSQE です。

ssid

QMF バージョン 12.1 がインストールされている DB2 for z/OS サブシステムの 4 文字の名前。

planid

リクエスターにバインドされている QMF アプリケーション・プランの名前。デフォルトは QMF12 です。

上記の ISPSTART コマンドを発行すると、98 ページの『QMF パネル・ライブラリーの正しいバージョンの確認』に示すように、QMF ホーム・パネルが表示されます。

- ISPF で提供されている ISPF マスター・アプリケーション・メニューをコーディングして、QMF を ISPF 選択メニュー項目の 1 つとして組み込むには、以下の例に従ってください。

図の下半分で強調表示されている呼び出しは、DB2 for z/OS サブシステムの名前 DB2P とプラン名 QRY10 を受け渡します。

ISPF を使用しない QMF の開始:

ISPF を使用しないで QMF を開始するには、アプリケーション・プログラムと QMF 呼び出し可能インターフェースを使用するか、あるいは TSO CALL コマンドを使用します。

手順

- アプリケーション・プログラムと呼び出し可能インターフェースを使用するには、次のステートメントを入力します。DSQQMFn
DSQSSUBS=*ssid*,DSQSPLAN=*planid*, ...

このコマンドには、次の変数が含まれています。

- n 適切な 1 文字の言語 ID を使用して、ユーザーの NLF 環境で QMF を開始します。例えば、英語環境で QMF を開始するには、コマンドは DSQQMFE です。

ssid

QMF バージョン 12.1 がインストールされている DB2 for z/OS サブシステムの 4 文字の名前。

planid

リクエスターにバインドされている QMF アプリケーション・プランの名前。QMF バージョン 12.1 のアプリケーション・プラン ID は QMF12 です。

示されている省略符号を追加の値に置き換えることもできます。

- TSO CALL コマンドを使用するために、TSO CALL コマンドを使用して、ローカルの DB2 for z/OS リクエスター・データベース内の QMF を初期化することができます。以下に例を示します。

```
CALL 'QMF1210.SDSQLOAD(DSQQMFn)' 'DSQSSUBS=ssid,DSQSPLAN=planid,...'
```

このコマンドには、次の変数が含まれています。

QMF1210

QMF バージョン 12 リリース 1 ライブラリーのデフォルト接頭部。QMF ライブラリーのデフォルトの接頭部を変更した場合、QMF1210 の代わりに、その接頭部を置換してください。

- n 適切な 1 文字の言語 ID を使用して、ユーザーの NLF 環境で QMF を開始します。例えば、英語環境で QMF を開始するには、DSQQMFE を使用します。

ssid

QMF バージョン 12.1 がインストールされている DB2 for z/OS サブシステムの 4 文字の名前。

planid

リクエスターにバインドされている QMF アプリケーション・プランの名前。デフォルトは QMF12 です。

関連概念:

187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターについて説明します。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

CICS での QMF の開始

QMF が CICS 下で正しくインストールされていることを検証する一環として、QMF が正しく開始することを確認します。

手順

QMF for CICS を開始するには、以下の手順に従います。

1. QMF がインストールされている CICS システムにサインオンします。
2. Esc (エスケープ) キーを押して、ネイティブ CICS セッションを開始する。
3. 以下のように、QMF を開始します。
 - a. CICS トランザクション QMF *n* を発行します。 *n* を、ご使用の NLF の 1 文字の各国語 ID に置き換えてください。例えば、英語の場合、トランザクション名は QMFE です。
 - a. プログラム・パラメーター DSQSDBQT および DSQSDBQN を指定して、トレース・データを保持するキューのタイプと名前を構成します。例えば、一時記憶域キュー名 DSQD で QMF を開始するには、次のコマンドを発行します。

```
QMFn DSQSDBQT=TS,DSQSDBQN=DSQD
```

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

QMF パネル・ライブラリーの正しいバージョンの確認

ホーム・パネルとヘルプ・パネルの両方が正しく表示されている必要があります。

手順

正しいバージョンの QMF パネル・ライブラリーにアクセスしていることを確認するには、以下の手順に従ってください。

1. 下図に示すように、正しいバージョンの QMF ホーム・パネルが表示されていることを確認します。ホーム・パネルで、以下の項目を確認します。
 - a. パネル名の下に、「バージョン 12 リリース 1」と表示されていることを確認します (**A**)。このバージョンおよびリリース情報が間違っていて、QMF を TSO で開始した場合は、DD ステートメント ADMGGMAP の割

り振りを検査して、このステートメントが QMF1210.SDSQMAPn を指していることを確認します。ユーザーの NLF に関連付けられた 1 文字の各国語 ID に置き換えます。例えば、英語版 QMF の場合、DD ステートメントは QMF1210.SDSQMAPE を指している必要があります。このバージョンおよびリリース情報が間違っていて、QMF を CICS で開始した場合は、QMF が存在する CICS 領域の始動 JCL 内の ADMF DD ステートメントが GDDM ADMF データ・セットを正しく指していることを確認します。DSQ1nADM インストール・ジョブ (英語版 QMF では DSQ1EADM) は、QMF バージョン 12.1 の GDDM マップをこのデータ・セットに書き込みます。

- b. QMF をインストールしたばかりの DB2 for z/OS データベースに接続していることを確認します (**B**)。次の例では、データベースの名前は DB2P です。

```

QMF HOME PANEL
Version 12 Release 1 A
Query      Management  Facility
*****   **   **   *****
Authorization ID
KMRAMEY   **   **   ***   ***   **
**   **   ****   ****   *****
Connected to
DB2P B  **   **   **   **   **
*****   **   ***   **   **
**
http://www.ibm.com/qmf
Enter a command on the command line or press a function key.
For help, press the Help function key or enter the command HELP.
1=Help      2=List      3=End      4=Show      5=Chart      6=Query
7=Retrieve   8=Edit Table  9=Form     10=Proc     11=Profile   12=Report
OK, you may enter a command.
COMMAND ==>

```

図 14. QMF バージョン 12.1 ホーム・パネル

2. バージョン 12.1 オンライン・ヘルプの存在を検査する。
 - a. ヘルプファンクション・キーを押します。下の図に示すように、QMF バージョン 12.1 のメイン・ヘルプ・メニューが表示されます。トピック 1 に正しい QMF のバージョンが反映されていることを確認します。

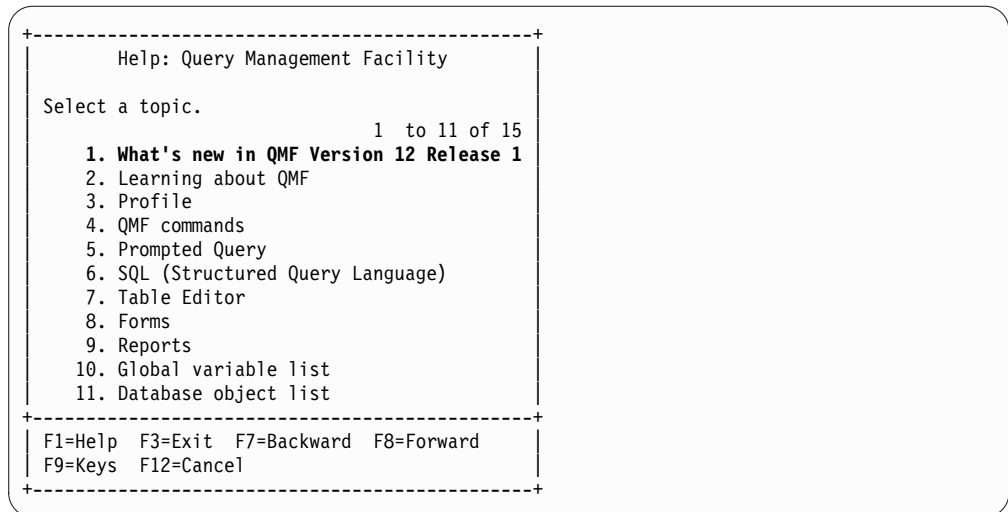


図 15. QMF バージョン 12.1 のメイン・ヘルプ・メニュー

- b. ホーム・パネルに戻るには、「キャンセル」(デフォルトで F12) を押します。

次のタスク

バージョンおよびリリース情報が正しくない場合、またはヘルプが全く表示されない場合は、DSQ1nPNL インストール・ジョブが正しく実行されたこと、および DSQPNLn ファイルの DD ステートメントが正しいバージョン 12 リリース 1 データ・セットを指していることを確認してください。*n* をユーザーの NLF に関連付けられた 1 文字の各国語 ID に置き換えます。

QMF を CICS で開始した場合は、さらに CICS 始動ジョブ・ストリームを検査して、DSQPNLn データ・セットが適切に割り振られていることを確認します。

関連概念:

50 ページの『VSAM パネル・ライブラリーのデータの設定』
SMP/E のインストールが完了したら、QMF VSAM パネル・ライブラリー DSQPNLE にデータを追加します。

関連タスク:

134 ページの『QMF 各国語機能のインストール』
QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

QMF ライブラリーが適切に割り振られていることの確認

QMF バージョン 12.1 のライブラリーが適切に割り振られていることを確認するには、QMF 内から SHOW GLOBALS コマンドを発行します。

大域変数をスクロールして、以下の変数の値を確認します。

- DSQAO_QMF_VER_RLS

この変数の値は、QMFV12R1.0 である必要があります。別の値が表示されている場合は、64 ページの『TSO および CICS に対するプログラム、リソースおよびプリファレンスの定義』で説明されている QMF ライブラリーの割り振りに関する情報を再確認してください。

- DSQAO_OTC_LICENSE

この変数の値は通常は 0 になります。この設定は、DSQUOPTS 初期化ルーチン内の OTC_LICENSE オプションの値が NOT_USED であることを示します。

変数値が正しくない場合は、DSQUOPTS 初期化ルーチンを編集し、変数を適切な値に設定してください。その後で、QMF を再始動してください。

関連概念:

175 ページの『QMF 開始時の大域変数および QMF セッション動作の初期化』いくつかの方法を使用して、QMF の開始時に QMF 大域変数を設定できます。この方法には、DSQUOPTS ルーチン、Q.GLOBAL_VARS 大域変数表、またはデフォルト・システム初期化プロシージャの変更があります。このプロシージャは、ホーム・パネルが表示される前にユーザーの QMF セッションの他の側面を設定するためにも使用できます。

関連タスク:

93 ページの『QMF の開始』

QMF の開始に使用する方法は、QMF を TSO または CICS のどちらで実行するかによって異なります。

必要な QMF 視点がすべてインストールされていることの確認

接続しているデータベースに対して、視点が適切であることを確認します。

手順

適切な視点がインストールされていることを確認するには、以下のようになります。

1. 次のコマンドを実行します。

```
LIST TABLES (OWNER=Q NAME=DSQ%)
```

2. QMF に付属する視点に対して、コマンドの結果を確認します。

3. リスト内に適切な視点がない場合は、DSQ1BVW インストール・ジョブを再実行します。拡張 LIST コマンド機能をインストールした場合は、DSQ1BVW の実行後に DSQ1BUDV インストール・ジョブを実行します。

関連タスク:

93 ページの『QMF の開始』

QMF の開始に使用する方法は、QMF を TSO または CICS のどちらで実行するかによって異なります。

関連資料:

499 ページの『QMF 視点』

以下の視点が QMF に付属しています。QMF はこれらの視点を、表に示されているプラットフォーム上で使用し、QMF LIST コマンドが発行されたときにオブジェクト・リストを作成します。

インストール検査手順に必要なサンプル照会およびサンプル・プロシージャのインストール

TSO にのみ、サンプルの QMF 照会およびプロシージャをインストールします。

これらのオブジェクトの一部は、TSO および CICS の両方でインストール検査手順に必要です。TSO でオブジェクトをインストールすると、CICS で使用可能になります。その他のオブジェクトは、QMF の使用方法を学ぶ際の参照用として提供されています。

サンプルの照会およびプロシージャは QMF バージョン 12.1 で変更されたため、旧バージョンを削除してから新バージョンをインストールする必要があります。

次の表に、サンプルの照会について説明します。

表 22. QMF バージョン 12.1 サンプル照会

| 照会のインポート元となるデータ・セットの名前 | QMF がオブジェクトのインポート後にそのオブジェクトの保存に使用する名前 | 説明 |
|------------------------|---------------------------------------|---|
| DSQ0EQ1 | Q.SAMPLE_SELECT1 | 単純な SELECT 照会 |
| DSQ0EQ2 | Q.SAMPLE_SELECT2 | IN キーワードおよび ORDER BY 文節を持つ SELECT 照会 |
| DSQ0EQ3 | Q.SAMPLE_SELECT3 | BETWEEN および NULL キーワードと AND および OR 演算子を持つ SELECT 照会 |
| DSQ0EQ4 | Q.SAMPLE_SELECT4 | LIKE キーワードを使用する SELECT 照会 |
| DSQ0EQ5 | Q.SAMPLE_SELECT5 | 2 つの表の結合を行う SELECT 照会 |
| DSQ0EQ6 | Q.SAMPLE_SELECT6 | MAX、MIN、および AVG 関数を使用する SELECT 照会 |
| DSQ0EQ7 | Q.SAMPLE_SELECT7 | GROUP BY および HAVING 文節、COUNT 関数を持つ SELECT 照会 |
| DSQ0EQ8 | Q.SAMPLE_SELECT8 | UNION 演算子を使用する SELECT 照会 |
| DSQ0EQ9 | Q.SAMPLE_SELECT9 | 副照会を使用する SELECT 照会 |
| DSQ0EQ10 | Q.SAMPLE_SELECT10 | 相関副選択を持つ SELECT 照会 |
| DSQ0EQ11 | Q.SAMPLE_SELECT11 | EXISTS キーワードを持つ SELECT 照会 |
| DSQ0EQ12 | Q.SAMPLE_INSERT1 | 単一行を表に追加する INSERT 照会 |
| DSQ0EQ13 | Q.SAMPLE_INSERT2 | 1 つ以上の行を表にコピーする INSERT 照会 |
| DSQ0EQ14 | Q.SAMPLE_UPDATE | UPDATE 照会 |
| DSQ0EQ15 | Q.SAMPLE_DELETE | DELETE 照会 |
| DSQ1EQ16 | Q.SAMPLE_DATABASE | CREATE DATABASE ステートメントを使用する照会 |
| DSQ1EQ17 | Q.SAMPLE_TABLESPACE | 表スペースを作成する照会 |
| DSQ1EQ18 | Q.SAMPLE_CREATETABLE | 表を作成する照会 |
| DSQ0EQ19 | Q.SAMPLE_CREATEINDEX | CREATE INDEX 照会 |
| DSQ0EQ20 | Q.SAMPLE_CREATEVIEW | 視点を作成する照会 |
| DSQ0EQ21 | Q.SAMPLE_GRANT | 更新特権を付与する照会 |
| DSQ0EQ22 | Q.SAMPLE_REVOKE | 更新特権を取り消す照会 |

表 22. QMF バージョン 12.1 サンプル照会 (続き)

| 照会のインポート元となるデータ・セットの名前 | QMF がオブジェクトのインポート後にそのオブジェクトの保存に使用する名前 | 説明 |
|------------------------|---------------------------------------|------------------|
| DSQ0EQ23 | Q.SAMPLE_DROP | データベースから表を除去する照会 |
| DSQ0EQ25 | Q.SAMPLE_COMMENT | 表にコメントを保存する照会 |

次の表に、サンプルのプロシージャーについて説明します。

表 23. QMF バージョン 12.1 サンプル・プロシージャー

| プロシージャーのインポート元となるデータ・セットの名前 | QMF がオブジェクトのインポート後にそのオブジェクトの保存に使用する名前 | 説明 |
|----------------------------------|--|---|
| DSQ0BIQ1 | Q.IVP_QUERY_COUNT | 特定の入力表について行カウントを行う |
| DSQ0BIQ2 | Q.IVP_QUERY_DELETE | カタログから QMF_IVP_QUERY を削除する |
| DSQ1EIVP | Q.DSQ1EIVP | TSO での対話式操作を検査する IVP プロシージャー |
| DSQ1EBAT | Q.DSQ1EBAT | TSO でのバッチ操作を検査するバッチ IVP プロシージャー |
| DSQ1EIVC | Q.DSQ1EIVC | CICS IVP プロシージャー |
| DSQAED1S | Q.DSQAED1S | 文書編集インターフェース機能をインストールする |
| DSQABL01 DSQAEL0D DSQABL0D | Q.DSQABL01 Q.DSQAEL0D Q.DSQABL0D | LAYOUT コマンド同義語をサポートする LAYOUT アプリケーション用のプロシージャーをインストールする |

以下のステップに従って、QMF の旧バージョンのサンプル照会およびサンプル・プロシージャーを削除し、QMF バージョン 12.1 のオブジェクトをインストールしてください。

1. QMF セッションを開始する。
2. QMF をインストールしたばかりの DB2 for z/OS サブシステムに接続していることを確認します。
3. 旧バージョンのサンプル照会およびサンプル・プロシージャーを削除します。
QMF バージョン 12.1 の旧リリースが存在しないシステムに QMF をインストールした場合は、このステップをスキップします。
 - a. 次のコマンドを入力して、照会とプロシージャーを削除するプロシージャーを表示します。

```
IMPORT PROC FROM QMF1210.SDSQSAPE(DSQ1ESQD)'
```

上記のコマンド、および DSQ1ESQD プロシージャー全体を通して、QMF1210 が QMF バージョン 12.1 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、コマンド内、およびプロシージャー内の QMF1210 がある場所すべてで、QMF1210 を、ユーザーが選択したサイト固有の接頭部に変更してください。TSO で QMF を使用している場合は、EDIT PROC コ

マンドを使用し、次に CHANGE などの ISPF コマンドを使用して、接頭部情報を変更できます。例えば、次の ISPF コマンドは、接頭部を QMF1210 から DB2TOOLS に変更します。

```
CHANGE 'QMF1210' 'DB2TOOLS' ALL
```

b. RUN PROC コマンドを発行してプロシージャを実行します。

- バージョン 12.1 のサンプル照会およびサンプル・プロシージャをインストールします。

次のコマンドを入力して、バージョン 12.1 の照会とプロシージャをインストールするプロシージャを表示します。

```
IMPORT PROC FROM 'QMF1210.SDSQSAPE(DSQ1ESQI)'
```

上記のコマンド、および DSQ1ESQI プロシージャ全体を通して、QMF1210 が QMF バージョン 12.1 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、コマンド内、およびプロシージャ内の QMF1210 がある場所すべてで、QMF1210 を、ユーザーが選択したサイト固有の接頭部に変更してください。TSO で QMF を使用している場合は、EDIT PROC コマンドを使用し、次に CHANGE などの ISPF コマンドを使用して、接頭部情報を変更できます。例えば、次の ISPF コマンドは、接頭部を QMF1210 から QMF12 に変更します。

```
CHANGE 'QMF1210' 'QMF12' ALL
```

- RUN PROC コマンドを発行してプロシージャを実行します。

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を受け取りません。エラーが発生した場合は、DSQ1ESQD プロシージャを使用して既に作成済みのオブジェクトを削除し、再開できます。

メッセージ DSQ21662 は単に通知メッセージであり、エラーを示しているわけではありません。このメッセージを無視して、IVP の実行を続行できます。

関連タスク:

93 ページの『QMF の開始』

QMF の開始に使用する方法は、QMF を TSO または CICS のどちらで実行するかによって異なります。

インストール検査手順の実行

リクエスター・データベースでの QMF インストールのインストール検査手順 (IVP) は、リクエスター・データベースでのインストールを TSO または CICS のどちらで行うかによって、わずかに異なります。

このタスクについて

QMF バージョン 12.1 が正しくインストールされていることをテストするには、102 ページの『インストール検査手順に必要なサンプル照会およびサンプル・プロシージャのインストール』でインストールした IVP を実行する必要があります。

関連タスク:

93 ページの『QMF の開始』

QMF の開始に使用する方法は、QMF を TSO または CICS のどちらで実行するかによって異なります。

TSO 用 IVP の実行

TSO で実行される QMF インストールには、2 つのインストール検査手順があります。1 つは対話式操作をテストし、もう 1 つはバッチ操作をテストします。

このタスクについて

QMF をバッチ・モードで使用する予定の場合は、両方の検査手順を完了してください。

TSO での対話式操作のテスト:

TSO での QMF の対話式操作をテストするには、以下の手順に従ってください。

始める前に

まだ開始していない場合は、QMF を開始します。

手順

1. コマンド `DISPLAY PROC Q.DSQ1EIVP` を発行して、TSO 用の QMF インストール検査手順を表示します。このプロシージャは、102 ページの『インストール検査手順に必要なサンプル照会およびサンプル・プロシージャのインストール』で説明したプロセスでデータベースにインポートおよび保存されたプロシージャです。プロシージャの最初にあるコメントには、プロシージャのテスト内容が説明されています。
2. PROC パネルで、必要場合はプロシージャをカスタマイズします。QMF1210 が QMF バージョン 12.1 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、DSQ1EIVP 全体の QMF1210 を、ユーザーが選択したサイト固有の接頭部に変更してください。

プロシージャを変更する場合は、次のコマンドを発行してプロシージャを保存してください。SAVE PROC AS Q.DSQ1EIVP (SHARE=YES

3. コマンド `RUN PROC` を発行して、プロシージャを実行します。各プロンプトへの応答で値 1 を入力します。

タスクの結果

IVP が正しく完了すると、次のメッセージが表示されます。The QMF IVP procedure has successfully completed.

EXIT コマンドを発行して QMF セッションを終了します。

次のタスク

IVP が正常に完了しない場合は、すべてのジョブが正常に完了したことを確認するために、実行した QMF インストール・ジョブの出力を検討してください。また、QMF メッセージおよびメッセージ・ヘルプ・パネル (ヘルプ・キーを押すと表示されます) を使用して、問題を診断および修正できます。

TSO でのバッチ操作のテスト:

TSO での QMF のバッチ操作をテストするには、以下の手順に従ってください。
QMF をバッチ・モードで実行する予定がない場合は、この手順をスキップしてください。

このタスクについて

QMF には、バッチ操作をテストするためのサンプル・プロシージャー DSQ1nBAT (n は、1 文字の各国語 ID) が付属しています。このプロシージャーは、QMF コントロール表が存在することと、基本的な QMF 操作が正しく完了することを検査します。

別の方式を使用して QMF をバッチ・モードで開始し、Q.DSQ1nBAT プロシージャーを実行することもできます。以下の方法がお客様のサイトの要件に合わない場合は、他のオプションについて 157 ページの『第 9 章 QMF の開始』を参照してください。

手順

TSO で DSQ1nBAT プロシージャーを使用して QMF のバッチ操作をテストするには、以下の手順に従ってください。

1. まだ開始していない場合は、QMF を開始します。
2. QMF ホーム・パネルが表示されたら、Q.DSQ1nBAT プロシージャーを表示します。例えば、英語版のバッチ IVP を表示するには、次のコマンドを発行します。

```
DISPLAY Q.DSQ1EBAT
```

プロシージャーが見つからない場合は、102 ページの『インストール検査手順に必要なサンプル照会およびサンプル・プロシージャーのインストール』でインストール手順を参照してください。QMF NLF のバッチ・モードをテストする場合は、バッチ・モード IVP のインストール方法について 134 ページの『QMF 各国語機能のインストール』のステップ 10 (140 ページ) を参照してください。

3. プロシージャーを検討して、必要な場合はカスタマイズします。

QMF1210 が QMF バージョン 12.1 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、プロシージャー全体の QMF1210 を、ユーザーが選択したサイト固有の接頭部に変更してください。

プロシージャーを変更する場合は、次のようなコマンドを発行してプロシージャーを保存してください。

```
SAVE PROC AS Q.DSQ1EBAT (SHARE=YES
```

4. TSO ログオン・プロシージャーをカスタマイズして QMF をバッチ・モードで開始し、Q.DSQ1nBAT プロシージャーを実行するようにします。
 - a. サンプル・ログオン・プロシージャー (DSQ1EINV) のコピーを作成します。
 - b. サイトの必要性に応じた JOB ステートメントを追加します。
 - c. SYSTERM および SYSIN の各 DD ステートメントを削除します。

- d. 印刷機能をセットアップします。

バッチ・モードで実行する QMF プロシージャは、QMF PRINT コマンドをテストします。このコマンドが成功するように、DSQPRINT ファイルを割り振る DD ステートメントを追加します。QMF PRINT コマンドからの出力は、このファイルに書き込まれます。次に例を示します。

```
//DSQPRINT DD SYSOUT=A
```

- e. ログオン・プロシージャの最後に次のようなステートメントを追加して、バッチ・モードからの出力の制御方法を指定します。

```
//SYSTSPRT DD SYSOUT=A
```

- f. SYSTSPRT の後ろに次のステートメントを追加します。 *n* は、テスト中の言語に対応する 1 文字の各国語 ID で置き換えます。

```
//SYSTSIN DD *  
    PROFILE PREFIX(username)  
    ISPSTART PGM(DSQMFn) NEWAPPL(DSQn) PARM(M=B,I=Q.DSQ1nBAT,S=ssid)  
/*
```

2 番目の JCL ステートメント内の最初の制御カードが必要となるのは、お客様のシステム環境で RACF を使用しない場合に限りです。上記の *username* を置き換えて、このステップを実行中のユーザーのログオン ID にします。このステートメントについて詳しくは、408 ページの『QMF バッチ・ジョブを実行するための JCL』を参照してください。

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、QMF がバッチ・モード (M=B) で呼び出されます。上記の *ssid* を置き換えて、QMF のインストール先の DB2 for z/OS サブシステムの 4 文字のサブシステム ID にします。サブシステム ID を指定しない場合は、デフォルトの DSN が使用されます。このように呼び出されると、QMF は Q.DSQ1*n*BAT プロシージャを実行します。

ISPSTART コマンドで受け渡せるその他の QMF プログラム・パラメーターについて詳しくは、187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』を参照してください。

5. 修正した TSO ログオン・プロシージャを保存します。
6. ジョブをサブミットします。これにより、QMF が開始され、バッチ・ジョブとして Q.DSQ1*n*BAT プロシージャがサブミットされます。

プロシージャの実行後、制御が TSO に戻ります。この結果、SYSTSIN に TSO ステートメントがもうないため、ジョブが終了します。ジョブ出力を調査して、エラーがないことを確認します。ジョブが正しく完了すると、トレース出力に次のメッセージが表示されます。

```
OK, TSO command executed successfully.  
OK, your procedure was run.
```

IVP が正常に完了しない場合は、すべてのジョブが正常に完了したことを確認するために、実行した QMF インストール・ジョブの出力を検討してください。また、QMF メッセージおよびメッセージ・ヘルプ・パネル (ヘルプ・キーを押すと表示されます) を使用して、問題を診断および修正できます。

関連概念:

407 ページの『第 18 章 バッチ・モードでの 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。したがって、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。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

CICS 用 IVP の実行

CICS での QMF の正しい操作をテストするには、以下の手順に従ってください。

手順

1. トレース・データを保持するデフォルトの一時データ・キュー (DSQD) を確認して、QMF トレース機能がインストールされていることを検査します。CICS のクリア画面から、コマンド CEMT INQUIRE QUEUE(DSQD) を入力します。次のような出力が表示されます。

```
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 が使用可能でオープンされていることが表示されていない場合は、CICS システム定義ファイルに対して行った変更が正しかったかどうかチェックします。詳細については、74 ページの『CICS に対して QMF を記述』を参照してください。

2. QMF をまだ開始していない場合は、93 ページの『QMF の開始』での説明に従って QMF を開始します。
3. コマンド DISPLAY PROC Q.DSQ1EIVC を入力して、CICS 用の QMF インストール検査手順を表示します。このプロシージャーは、102 ページの『インストール検査手順に必要なサンプル照会およびサンプル・プロシージャーのインストール』で説明したプロセスでデータベースにインポートおよび保存されたプロシージャーです。プロシージャーの最初にあるコメントには、プロシージャーのテスト内容が説明されています。
4. PROC パネルで、必要な場合はプロシージャーをカスタマイズします。プロシージャーを変更する場合は、次のコマンドを発行してプロシージャーを保存してください。SAVE PROC AS Q.DSQ1EIVC (SHARE=YES
5. コマンド RUN PROC を入力して、プロシージャーを実行します。各プロンプトへの応答で値 1 を入力します。

タスクの結果

IVP が正しく完了すると、次のメッセージが表示されます。The QMF IVP procedure has successfully completed.

EXIT コマンドを発行して QMF セッションを終了します。

プロシージャが正常に実行されない場合は、すべてのジョブが正常に完了したことを確認するために、実行した QMF インストール・ジョブ出力にエラーがないか確認します。ステップ 1 (108 ページ) でチェックした一時記憶キューをブラウズして、QMF 警告メッセージがあるかどうかを確認するには、次の CICS トランザクションを発行します。CEBR DSQD

サーバー・データベースでの QMF インストールのテスト

QMF CONNECT コマンドまたは DSQSDBNM プログラム・パラメーターによってアクセスされるサーバー・データベース用のインストール・パスの一部として、サーバー・データベースでの QMF のインストールをテストします。

始める前に

サーバー・データベースでの QMF のインストールをテストする前に、リクエスター・データベースでのインストールをテスト済みであることを以下のように確認してください。

- サーバーにアクセスするすべてのリクエスター・データベースでの QMF のインストールをテストします。
- また、リモート・サーバー・データベースが DB2 for z/OS である場合は、リクエスター用のテスト手順も実行します。手順については、93 ページの『リクエスター・データベースへの QMF インストールのテスト』を参照してください。

このタスクについて

26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』のインストール・パスには、次のテスト手順が適用されます。

手順

z/OS 以外のサーバー・データベースでの QMF インストールをテストするには、以下の手順に従ってください。

1. DB2 for z/OS リクエスター・データベース内の QMF を、分散データ機能 (DDF) を使用して開始します。
 - TSO で QMF を実行している場合は、94 ページの『TSO での QMF の開始』を参照してください。
 - CICS で QMF を実行している場合は、98 ページの『CICS での QMF の開始』を参照してください。

DSQSSUBS プログラム・パラメーターには、QMF を開始する DB2 for z/OS リクエスターの名前を指定します。

2. サーバー・データベースに接続します。次の 2 つの方法のどちらかで、サーバー・データベース内の QMF に接続できます。
 - リクエスター・データベース内の QMF を開始して、QMF ホーム・パネルが表示されたら、QMF CONNECT コマンドを使用してサーバーに接続します。
 - リクエスター・データベース内の QMF の開始時に、DSQSDBNM プログラム・パラメーターを指定します。QMF は、ホーム・パネルを表示する前に、このパラメーターで指定されたサーバー・ロケーションに接続します。このパラメーターについて詳しくは、196 ページの『初期データベース接続の指定』を参照してください。

例えば、LOCL という名前のローカル DB2 for z/OS データベース内の QMF を開始して、すぐに REMT という名前のリモート DB2 for Linux, UNIX, and Windows データベースに接続するには、QMF の開始に使用するコマンドに次のプログラム・パラメーターを指定します。

`DSQSSUBS=LOCL,DSQSDBNM=REMT`

3. QMF ホーム・パネルの「接続先」の下に表示されているデータベース名が、リモート・データベース・サーバーの正しい名前であることを確認します。ホーム・パネルの例については、99 ページの図 14 を参照してください。
4. QMF 視点が正しくインストールされていることを確認します。コマンド LIST TABLES (OWNER=Q NAME=DSQ% を発行して、リストに示された視点が接続先のデータベース・タイプに適切であることを確認します。

499 ページの『QMF 視点』には、プラットフォーム別に、QMF LIST コマンドをサポートするデフォルト視点がリストされています。

リスト内に適切な視点がない場合は、DSQ1BVW インストール・ジョブを再実行します。サーバーが DB2 for z/OS であり、拡張 LIST コマンド機能をインストールした場合は、DSQ1BVW の実行後に DSQ1BUDV インストール・ジョブを実行します。

5. 取り消しファンクション・キー (デフォルトで F12) を押して、QMF ホーム・パネルに戻ります。終了ファンクション・キー (デフォルトで F3) を押して、QMF セッションを終了します。

次のタスク

これらの作業が完了した後で、選択したインストール・パスに関連付けられているインストール手順に戻って、インストールの次のステップを判別してください。

関連概念:

187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターについて説明します。

157 ページの『第 9 章 QMF の開始』

QMF は、z/OS からのみ開始できます。QMF は、TSO、ISPF の下にバッチ・ジョブとして、あるいは CICS の下にセットアップすることができます。

関連タスク:

93 ページの『リクエスター・データベースへの QMF インストールのテスト』
リクエスター・データベースでの QMF のインストールをテストするには、以下の
手順に従ってください。

第 6 章 移行およびフォールバックの考慮事項

QMF バージョン 12.1 のインストール後またはバージョン 11 への移行後に、サイト固有の条件下で新しいインストール済み環境を十分にテストするために、QMF の旧リリースのオブジェクトにアクセスし、それらのオブジェクトを使用することが必要になる場合があります。

また、何らかの理由で旧リリースにフォールバックすることを決定した場合は、QMF バージョン 12.1 で作成したオブジェクトまたはプログラムを、QMF の旧リリースで使用する必要がある場合があります。

関連タスク:

90 ページの『サーバー・データベースでの旧リリースへのフォールバック』
QMF 互換モード・リリース (QMF バージョン 3.3、6.1、7.1、7.2、バージョン 8.1 互換モード、またはバージョン 9.1 互換モード) から QMF バージョン 12.1 に移行中にエラーが検出された場合は、旧リリースに戻すことができます。

QMF の旧リリースのプロファイルおよびオブジェクトへのアクセス

QMF の旧リリースで作成したオブジェクトに簡単にアクセスできるかどうかは、2 つのリリースが同じデータベースにあるかどうかによって異なります。

QMF の旧リリースが含まれているデータベースに QMF をインストールした場合、旧リリースのオブジェクトおよびプログラムは新しいリリースで使用可能です。

旧リリースが含まれていないデータベースに QMF バージョン 12.1 をインストールし、別のデータベースにある旧リリースのオブジェクトにアクセスする必要がある場合は、プロファイルを再作成して、表、視点、および QMF オブジェクトを移行することができます。

別のデータベースにある QMF プロファイルへのアクセス

新たにインストールしたリリースと同じデータベース内にない、旧リリースのプロファイルを使用するには、プロファイルを再作成します。

このタスクについて

QMF にアクセスするには、QMF プロファイルが必要です。プロファイル情報は、Q.PROFILES コントロール表に保管されています。

手順

新たにインストールしたリリースと同じデータベース内にない、旧リリースのプロファイルを使用するには、以下のいずれかの方法でプロファイルを再作成します。

- 旧リリースが QMF バージョン 8.1 またはバージョン 9.1 新機能モードや QMF バージョン 10、バージョン 11.1、またはバージョン 11.2 である場合は、QMF EXPORT コマンドおよび IMPORT コマンドを使用して、旧 QMF リリ

ースを含むデータベースから Q.PROFILES 表のコピーをエクスポートし、新しいリリースの QMF を含むデータベースにインポートすることで、プロファイルを移行できます。

- QMF のどのリリースの旧インストール済み環境でも、「SQL 照会」パネルから複数の INSERT ステートメントを発行して、Q.PROFILES 表に適切なデータを再作成できます。

関連概念:

226 ページの『QMF ユーザー・プロファイルの作成』

すべての QMF ユーザーは、ユーザー・プロファイルにアクセスする必要があります。このユーザー・プロファイルは、特定のユーザーからの個々の入力を 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 (例えば、プリンター出力の宛先、または入力の大文字変換の有無など) を制御します。

関連タスク:

『表、視点、および QMF オブジェクトの移行』

QMF EXPORT コマンドおよび IMPORT コマンドを使用して、表、視点、および QMF オブジェクトを移行できます。その代わりに DB2 ユーティリティを使用しても構いません。

表、視点、および QMF オブジェクトの移行

QMF EXPORT コマンドおよび IMPORT コマンドを使用して、表、視点、および QMF オブジェクトを移行できます。その代わりに DB2 ユーティリティを使用しても構いません。

手順

- QMF を使用して表、視点、および QMF オブジェクトを移行するには、旧リリースの QMF から表をエクスポートし、新しいリリースでこれらの表をインポートします。

DB2 QMF High Performance Option (HPO) をインストール済みで、データベース・タイプが DB2 for z/OS の場合は、HPO オブジェクト・マネージャーを使用して、ある DB2 サブシステムから別のサブシステムにオブジェクトを移行することもできます。

- DB2 を使用して表、視点、および QMF オブジェクトを移行するには、以下の作業を行います。
 1. アプリケーション・プログラム DSNTIAUL など、DB2 で提供されるお好きなツールまたはユーティリティを使用して、表をアンロードします。
 2. 新しいリリースの QMF がインストールされている DB2 サブシステムに表をロードします。DB2LOAD ユーティリティなどのツールまたはユーティリティを使用します。
 3. 使用可能なネットワーキング機能を使用して、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、新しいリリースの QMF が含まれているシステムに送信します。対象のオブジェクトが DB2 for z/OS にない場合は、作業しているデータベースのタイプで使用可能なシステム固有のユーティリティを使用してください。

次のタスク

インポートされた表の視点、索引、同義語、および権限を新しいデータベースで再作成します。

関連情報:

 DB2 資料

オブジェクトを移行するために必要なプログラム、ユーティリティ、および権限に関する情報を検索します。

リリース間のオブジェクトの互換性

異なる QMF のリリースで作成したオブジェクトを使用できるかどうかは、オブジェクトによって参照されるコマンドや機能が両方のリリースで同じように動作するかどうかによります。

QMF バージョン 12.1 のインストール・プロセスでは、バージョン 3.3、6.1、7.1、7.2、8.1、9.1、および 10.1 からの移行がサポートされます。非互換性に対する計画に役立つよう、移行元のリリースと QMF バージョン 12.1 の間で追加または変更されたフィーチャーおよび機能を必ず把握しておいてください。

関連概念:

471 ページの『付録 A. 旧リリースでの変更の要約』

QMF バージョン 12 リリース 1 は、バージョン 8.1 NFM、バージョン 9.1 NFM、バージョン 10.1、バージョン 11.1、およびバージョン 11.2 からの移行をサポートします。旧リリースの機能拡張が、QMF の移行計画に影響を与えることがあります。

QMF バージョン 12 リリース 1 と旧リリースとの前方互換性

旧リリースで作成したほとんどのオブジェクトは、変更なしで QMF バージョン 12.1 で使用できます。

ただし、旧リリースのオブジェクトを使用する際には、いくつかの例外と考慮事項に注意が必要です。

QMF コマンド

- 旧リリースの QMF コマンドはすべて、省略形およびオプションがバージョン 12.1 で有効な場合に限り、QMF バージョン 12.1 で発行できます。
- バージョン 12.1 でのコマンド、構文、省略形、およびオプションのリストについては、「DB2 QMF 解説書」を参照してください。このガイドは、ユーザー、プログラマー、およびデータベース管理者がこの情報について理解するために役立つことを目的としています。
 - QMF for TSO/CICS でのコマンドの構文および使用法
 - 照会での SQL キーワードの使用法
 - 書式、報告書、および図表 (取扱コードと編集コードを含む) の使用法
- QMF コマンド・インターフェースについて詳しくは、「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。このガイドには、以下の項目に関してアプリケーション・プログラマーに役立つ情報があります。

- アプリケーション・プログラミング設計の意思決定
 - さまざまなプログラミング手法の中からの選択
 - QMF コマンドおよび呼び出し可能インターフェースの使用法の理解
 - 2 か国語使用のアプリケーションの作成
- 文書編集インターフェースについて詳しくは、「DB2 QMF の使用の手引き」を参照してください。このガイドは、以下の項目に関してエンド・ユーザーに役立つことを目的にしています。
 - データの取り出し
 - データを報告書または図表にフォーマット設定
 - データの編集とその他の基本 QMF タスク

QMF プロシージャ

プロシージャ内の QMF コマンドは、『QMF コマンド』セクションで説明されている制限の対象になります。

旧リリースのプロシージャの一部は、ユーザー定義コマンド (QMF コマンド同義語と呼ばれる) としての verb も使用されている verb を含むコマンドを発行する場合は、正しく機能しません。問題が発生する可能性をなくすには、すべてのネイティブの QMF コマンドの前に QMF と入力します。コマンド同義語の詳細については、299 ページの『コマンド同義語のカスタマイズ』を参照してください。

アプリケーション

アプリケーション内の QMF コマンドは、『QMF コマンド』セクションで説明されている制限の対象になります。

QMF バージョン 12.1 と旧リリースとの互換性

QMF バージョン 12.1 で作成された一部のオブジェクトは、旧リリースでは使用できません。フォールバックする予定がある場合は、非互換性について考慮してください。

特に記述のない限り、旧バージョンとの互換性の問題は、QMF バージョン 12.1 でのオブジェクトの使用にのみ該当します。機能の要約の情報を使用して、バージョン 12.1 で使用する機能がバージョン 12.1 より前のリリースで同じように動作するかどうかを判別することができます。

旧バージョンとの互換性については、以下のいずれかのセクションを参照してください。

QMF コマンド

以下のコマンドまたはコマンド・オプションは QMF バージョン 12.1 で追加または変更されたため、旧リリースでは使用できません。これらのコマンドやオプションを含むプロシージャまたはアプリケーションは、旧リリースで実行しようとする前に、変更する必要があります。

- QBE および指示照会のための RUN QUERY での TABLE キーワード

以下は、QMF バージョン 12.1 での新しい大域変数です。これらの変数を参照するコマンドは、旧リリースでは使用できません。

- DSQEC_DS_SUPPORT: この大域変数によって、QMF は QMF Data Service データ・ソースにアクセスできます。
- DSQEC_BUFFER_SIZE: この大域変数は、QMF 内部記憶域バッファ・サイズを変更します。
- DSQEC_TRACE_MODULE: この大域変数によって、管理者は、QMF モジュール名ごとの診断のトレースを制限できます。

SQL ステートメント

QMF Data Service データ・ソースに対して 3 部構成の名前を参照する照会は、以前のリリースでは使用できません。旧リリースにフォールバックする前に、このステートメントを含む照会、プロシージャ、またはアプリケーションを変更してください。

書式

2 進列データに「C」編集コードまたは「CW」編集コードを指定している書式は、以前の QMF リリースでは使用できない可能性があります。

QMF プログラム・パラメーター

以前の QMF リリースは、DSQSMTHD プログラム・パラメーターでは始動できない可能性があります。バージョン 12.1 で追加されたマルチスレッド・サポートに依存する照会、プロシージャ、またはアプリケーションは、以前のリリースでは使用できない可能性があります。

2 GB 境界より上の記憶域の要件に関する変更

LOB データ・タイプおよび XML データ・タイプの取り出しには、2 GB 境界より上の記憶域が必要です。LOB データ・タイプおよび XML データ・タイプと、64 ビット・スピル・サポートのために、2 GB 境界より上の記憶域の量 (MEMLIMIT) を大きくすることができます。

QMF では、2 GB 境界より上の記憶域が使用されます。アドレス・スペースがメモリー・オブジェクトに対して一度に使用できる実記憶域および補助記憶域の量を制御するために、お客様の環境で、1 つのアドレス・スペース用に 2 GB 境界より上の使用可能な仮想ページの総数を設定するインストール・デフォルト MEMLIMIT が確立されている場合があります。このデフォルトが十分ではない場合、QMF を始動できません。

z/OS V1R10.0 以降において、現在の MEMLIMIT のデフォルトは 2G です。これより前の z/OS バージョンのデフォルトは 0 で、これはどのアドレス・スペースも 2 GB 境界より上の仮想ストレージを使用することはできないことを意味します。お客様の環境の MEMLIMIT 設定が、QMF 用に少なくとも 2G であることを確認する必要があります。広範囲にわたるオブジェクトのスケラビリティには、この量が推奨されます。QMF は 2 GB 境界より上の記憶域を必要な量だけ、また必要である間だけ使用します。ジョブまたはプロセスが完了した後、記憶域はすぐに解放されます。

MEMLIMIT パラメーターを設定するには、以下のいずれかの方法を使用します。

- SMFPRMxx PARMLIB メンバーの MEMLIMIT パラメーターにインストール・デフォルトを設定します。
- SET SMF または SETSMF コマンドを実行します。
- MEMLIMIT パラメーターまたは REGION=0 のどちらかを、ログオン・プロシージャ TSO JCL またはジョブ JCL に追加します。
- MEMLIMIT を IEFUSI 出口ルーチンに指定します。これを行うと、この MEMLIMIT 設定が他の MEMLIMIT 設定すべてをオーバーライドします。

次の例は、TSO ログオン・プロシージャに追加された MEMLIMIT パラメーターを示しています。

```
//TSOPROC EXEC PGM=IKJEFT01,REGION=0M,DYNAMNBR=175,  
// PARM='%LOGINIT',TIME=1440,MEMLIMIT=2G
```

MEMLIMIT について詳しくは、ご使用の z/OS のバージョンに対応する IBM z/OS 資料を参照してください。

注: これらの要件の変更による結果として、既存の QMF TSO アドレス・スペース割り振りを変更しなければならない場合もあります。

第 7 章 QMF の旧リリースの削除

QMF の旧リリースを削除する手順は、QMF がインストールされているデータベースのタイプによって異なります。リリースの削除は、このリリースへの接続が不要であること、および保管するすべてのものをこのリリースから移行済みであることを確認した後でのみ行うようにしてください。

始める前に

QMF の旧リリースをデータベースから削除する前に必ず、以下の条件を確認してください。

- そのリリースでは接続性は必要ありません。例えば、QMF Version 9.1 をサーバーから削除しようとしているとします。その場合、チェックすべきことは、リクエスター・データベース内の Version 9.1 インストール環境から CONNECT コマンドをサーバーに送信しないかどうかです。
- 旧リリースから保存するすべてのプログラム、表、視点、照会、書式、プロシージャ、およびその他のオブジェクトを移行してテストが完了した状態です。

関連概念:

113 ページの『第 6 章 移行およびフォールバックの考慮事項』

QMF バージョン 12.1 のインストール後またはバージョン 11 への移行後に、サイト固有の条件下で新しいインストール済み環境を十分にテストするために、QMF の旧リリースのオブジェクトにアクセスし、それらのオブジェクトを使用することが必要になる場合があります。

DB2 for z/OS データベースからの QMF の削除

DB2 for z/OS データベースから QMF の旧リリースを削除する手順は、QMF の新旧リリースが同じサブシステムに存在しているかどうかによって異なります。

始める前に

重要: このトピック内の作業を開始できるのは、QMF の旧バージョンがこのデータベース内で既に不要になっていることを確認してからです。旧バージョンが依然として必要になるのは、そのバージョンがローカルでアクセス中である場合か、または同じリリースのリモート・インストール環境との接続が必要となる場合です。

新旧のリリースが同じサブシステムにある場合

QMF の旧リリースを削除するには、そのリリースからアプリケーション・プランを解放し、ライブラリーを削除し、すべてのパッケージを除去します。この手順は、DB2 for z/OS データベース上の QMF の両方のリリースが同じサブシステムに存在する場合に適用されます。

旧アプリケーション・プランの解放

DSQ1DEL1 ジョブを編集して DSQ1JFPL ジョブを実行することにより、新リリースと同じサブシステム内に存在する旧リリースの QMF のアプリケーション・プランを解放します。

手順

1. QMF1210.SDSQSAPE(DSQ1JFPL) を編集する。
2. お客様システム環境のニーズに適合するようにジョブ・ステートメントを変更する。
3. ジョブのインストリーム・プロシージャ内のパラメーターの値を検査 (必要に応じて変更) する。

```
//DSQ1JFPL PROC RGN='2048K',      Job-step region size
//  QMFTPRES='QMF1210',          QMF
//  prefix
//  DB2EXIT='DSN1110.SDSNEXIT',   DB2 for z/OS exit library
//  DB2LOAD='DSN1110.SDSNLOAD'   DB2 for z/OS program library
```

4. QMF1210.SDSQSAPE(DSQ1DEL1) を編集する。
5. 括弧内で指定する DSN SYSTEM 値を DB2 for z/OS サブシステムの名前で置き換えます。FREE PLAN 値を旧リリースのアプリケーション・プラン名で置き換えます。下の表は、QMF バージョン 12.1 への移行がサポートされているリリースのデフォルトのアプリケーション・プラン名を示しています。

表 24. QMF アプリケーション・プラン名のデフォルト値

| 旧リリース | デフォルトのアプリケーション・プラン名 |
|----------------|---------------------|
| QMFバージョン 11.2 | QMF1120 |
| QMF バージョン 11.1 | QMF11 |
| QMF バージョン 10.1 | QMF10 |
| QMF バージョン 9.1 | QMF910 |
| QMF バージョン 8.1 | QMF810 |
| 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 |

例えば、DB2P と呼ばれる DB2 for z/OS サブシステム内で QMF バージョン 9.1 から QMF バージョン 12.1 に移行している場合は、ジョブ DSQ1DEL1 内で次の変更を行ってバージョン 9 プランを取り除きます。

```
DSN SYSTEM(DB2P)
FREE PLAN(QMF910)
```

6. 新旧リリースが存在している DB2 for z/OS サブシステムが開始されていない場合は、開始します。
7. ジョブ QMF1210.SDSQSAPE(DSQ1JFPL) をサブミットします。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します。

SMP/E を使用して旧リリースのライブラリーを削除する

SMP/E を使用して、バージョン 12.1 への移行元である QMF リリースのライブラリーを削除します。旧アプリケーション・プランの解放後に、ライブラリーを削除してください。

このタスクについて

重要: 現行ライブラリーを削除しないよう、接頭部には特に注意してください。バージョン 12.1 ライブラリーのデフォルト接頭部は QMF1210 です。

以下のライブラリーは、QMF バージョン 8.1、バージョン 9.1、バージョン 10.1、およびバージョン 11.1 で使用されます。バージョン 8.1 の接頭部は QMF810、バージョン 9.1 の接頭部は QMF910、バージョン 10.1 の接頭部は QMF1010、バージョン 11.1 の接頭部は QMF1110 です。

- *prefix*.SDSQCHRT
- *prefix*.ADSQCHRT
- *prefix*.SDSQCLTE
- *prefix*.SDSQDBRM
- *prefix*.ADSQDBMD
- *prefix*.DSQDEBUG
- *prefix*.SDSQEXCE
- *prefix*.SDSQEXIT
- *prefix*.SDSQLOAD
- *prefix*.ADSQMACE
- *prefix*.SDSQMAPE
- *prefix*.ADSQMAPE
- *prefix*.SDSQMLBE
- *prefix*.ADSQOBJ
- *prefix*.SDSQPLBE
- *prefix*.ADSQPMSE
- *prefix*.DSQPNLE
- *prefix*.SDSQPVRE
- *prefix*.ADSQPVRE
- *prefix*.SDSQSAPE
- *prefix*.SDSQSLBE
- *prefix*.SDSQUSRE
- *prefix*.DSQUCFRM

下の表に、QMF バージョン 7.2 以前のリリースのライブラリーを示します。

表 25. QMF バージョン 7.2 以前のリリースのライブラリーの名前

| バージョン 3.3 ライブラリー | バージョン 6.1 ライブラリー | バージョン 7.1 ライブラリー | バージョン 7.2 ライブラリー |
|------------------|------------------|------------------|------------------|
| QMF330.DSQCHART | QMF610.SDSQCHRT | QMF710.SDSQCHRT | QMF720.SDSQCHRT |
| QMF330.DSQCLSTE | QMF610.SDSQCLTE | QMF710.SDSQCLTE | QMF720.SDSQCLTE |

表 25. QMF バージョン 7.2 以前のリリースのライブラリーの名前 (続き)

| バージョン 3.3 ライブラリー | バージョン 6.1 ライブラリー | バージョン 7.1 ライブラリー | バージョン 7.2 ライブラリー |
|------------------|------------------|------------------|------------------|
| QMF330.DSQDBRM | QMF610.SDSQDBRM | QMF710.SDSQDBRM | QMF720.SDSQDBRM |
| QMF330.DSQDBRMD | QMF610.ADSQDBMD | QMF710.ADSQDBMD | QMF720.ADSQDBMD |
| | QMF610.DSQDEBUG | QMF710.DSQDEBUG | QMF720.DSQDEBUG |
| QMF330.DSQEXECE | QMF610.SDSQEXCE | QMF710.SDSQEXCE | QMF720.SDSQEXCE |
| | | | QMF720.SDSQEXIT |
| QMF330.DSQLOAD | QMF610.SDSQLOAD | QMF710.SDSQLOAD | QMF720.SDSQLOAD |
| QMF330.DSQMACE | QMF610.ADSQMACE | QMF710.ADSQMACE | QMF720.ADSQMACE |
| QMF330.DSQMAPE | QMF610.DSQMAPE | QMF710.DSQMAPE | QMF720.DSQMAPE |
| QMF330.DSQMLIBE | QMF610.SDSQMLBE | QMF710.SDSQMLBE | QMF720.SDSQMLBE |
| QMF330.DSQOBJ | QMF610.ADSQOBJ | QMF710.ADSQOBJ | QMF720.ADSQOBJ |
| QMF330.DSQPLIBE | QMF610.SDSQPLBE | QMF710.SDSQPLBE | QMF720.SDSQPLBE |
| QMF330.DSQPMSE | QMF610.ADSQPMSE | QMF710.ADSQPMSE | QMF720.ADSQPMSE |
| QMF330.DSQPNLE | QMF610.DSQPNLE | QMF710.DSQPNLE | QMF720.DSQPNLE |
| QMF330.DSQPVARE | QMF610.DSQPVARE | QMF710.DSQPVARE | QMF720.DSQPVARE |
| QMF330.DSQSAMPE | QMF610.SDSQSAPE | QMF710.SDSQSAPE | QMF720.SDSQSAPE |
| QMF330.DSQSLIBE | QMF610.SDSQSLBE | QMF710.SDSQSLBE | QMF720.SDSQSLBE |
| QMF330.DSQUSERE | QMF610.SDSQUSRE | QMF710.SDSQUSRE | QMF720.SDSQUSRE |
| QMF330.DSQTLIBE | | | |
| QMF330.DSQUCFRM | QMF610.DSQUCFRM | QMF710.DSQUCFRM | QMF720.DSQUCFRM |

旧リリースからのパッケージの除去

旧リリースのライブラリーを削除後に、旧リリースの QMF からすべてのパッケージを除去します。

手順

1. 以下の照会を実行して、特定リリースのすべての QMF パッケージをリストします。

```
SELECT SUBSTR(NAME,1,10)
FROM SYSIBM.SYSPACKAGE
WHERE NAME LIKE 'DSQX%'
```

このステートメントでは、X はリリースに関連付けられているバージョン固有の文字です。次の表には、QMF バージョン 12.1 への移行がサポートされている移行元の各リリースごとに、パッケージ名の最初の 4 文字が記載されています。

表 26. QMF バージョン 12.1 より前のリリースのパッケージ ID

| 今後、不要となる QMF リリース | パッケージ名の最初の文字 |
|-------------------|--------------|
| バージョン 3.3 | DSQA |
| バージョン 6.1 | DSQB |
| バージョン 7.1 | DSQC |
| バージョン 7.2 | DSQD |
| バージョン 8.1 | DSQE |
| バージョン 9.1 | DSQF |
| バージョン 10.1 | DSQG |
| バージョン 11.1 | DSQH |
| バージョン 11.2 | DSQI |

2. 各パッケージに対して DROP ステートメントを発行します。

例

QMF バージョン 10 に関連付けられているパッケージをリストするには、以下のステートメントを発行します。

```
SELECT SUBSTR(NAME,1,10)
FROM SYSIBM.SYSPACKAGE
WHERE NAME LIKE 'DSQG%'
```

新旧のリリースが異なるサブシステムにある場合

QMF の新旧リリースが異なる DB2 for z/OS サブシステムに存在しており、旧リリースを削除する場合は、QMF のすべてのトレースと QMF によって作成されたオブジェクトを削除します。

始める前に

重要: この作業は、QMF の新旧リリースが異なるサブシステムに存在しており、旧リリースへの接続が不要である場合にのみ行ってください。

手順

1. QMF バージョン 12.1 でもそのまま使用する、旧リリースからの QMF 照会、プロシージャ、または書式を バージョン 12.1 サブシステムへ移行する。
2. 121 ページの『SMP/E を使用して旧リリースのライブラリーを削除する』の説明に従って、QMF の旧リリースのライブラリーを削除する。
3. 122 ページの『旧リリースからのパッケージの除去』の説明に従って、旧リリースの QMF パッケージを除去する。
4. ジョブ DSQ1DELA をカスタマイズおよび実行することによって、QMF 構成体および QMF オブジェクトを旧リリースから削除する。このジョブは、QMF コントロール表を削除します。これには QMF オブジェクト・コントロール表、およびそこに含まれる照会、プロシージャ、または書式があります。
 - a. QMF1210.SDSQSAPE(DSQ1DELA) を編集する。
 - b. お客様システム環境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。
 - c. ジョブのインストリーム・プロシージャ内のパラメーター値を検査 (必要に応じて変更) する。

```
// DSQ1DELA PROC RGN='2048K',      Job-step region size
// QMFTPRE='QMF1210',             QMF prefix
// DB2EXIT='DSN1110.SDSNEXIT',    DB2 for z/OS exit library
// DB2LOAD='DSN1110.SDSNLOAD'     DB2 for z/OS program library
```

- d. メンバー QMF1210.SDSQSAPE(DSQ1DEL1) を編集する。このジョブは、旧リリースの QMF アプリケーション・プランを取り除きます。

ジョブ DSQ1DEL1 で、括弧内に指定された DSN SYSTEM 値を DB2 for z/OS サブシステムの名前で置き換えます。FREE PLAN 値を QMF の旧リリースのアプリケーション・プラン名で置き換えます。デフォルトの QMF アプリケーション・プラン名は 120 ページの表 24 に示されています。例えば、DB2P と呼ばれる DB2 for z/OS サブシステム内で QMF バージ

ョン 9.1 から QMF バージョン 12.1 に移行している場合は、ジョブ DSQ1DEL1 内で次の変更を行ってバージョン 9 プランを取り除きます。

```
DSN SYSTEM(DB2P)
FREE PLAN(QMF910)
```

- e. メンバー QMF1210.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。

各ユーザーの SAVE DATA コマンドおよび IVP に関する表スペースが、旧リリースで使用されていなかった場合は、次のステートメントを削除してください。

```
DROP STOGROUP DSQSGDEF
```

- f. メンバー QMF1210.SDSQSAPE(DSQ1DEL3) を編集する。このメンバーには、QMF コントロール表に対してユーザー管理データ・セットを削除するステートメントが入っています。データ・セットが DB2 for z/OS により管理されている場合は、このステップを行う必要はありません。

データ・セットが DB2 for z/OS によって管理されていない場合は、ジョブ内の各ステートメントにある QMFDSN を、削除対象の QMF リリースで使用していたハイレベル修飾子に変更します。

- 5. QMF の旧リリースが存在しているサブシステムを始動して、ジョブ QMF1210.SDSQSAPE(DSQ1DELA) を実行します。

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します。

関連概念:

113 ページの『第 6 章 移行およびフォールバックの考慮事項』

QMF バージョン 12.1 のインストール後またはバージョン 11 への移行後に、サイト固有の条件下で新しいインストール済み環境を十分にテストするために、QMF の旧リリースのオブジェクトにアクセスし、それらのオブジェクトを使用することが必要になる場合があります。

DB2 for iSeries または DB2 for Linux, UNIX, and Windows データベースからの QMF の削除

この手順は、QMF の新旧リリースが同じデータベースにあるか、異なるデータベースにあるかにより異なります。

このタスクについて

このトピックの手順では、QMF の旧リリースを DB2 for Linux, UNIX, and Windows または DB2 for iSeries データベースから削除します。

重要: このトピック内の作業を開始できるのは、QMF の旧バージョンがサーバー内で今後不要となることを確認してからです。同じ旧リリース・レベルで QMF への接続が必要な場合、旧バージョンは引き続き必要です。

新旧のリリースが同じリモート・サーバーにある場合

QMF の新旧リリースが同じリモート・サーバーにある場合は、旧リリースからすべての QMF パッケージを除去することにより、QMF の旧リリースを削除します。このケースでは、QMF コントロール表およびサンプル表は QMF バージョン 12.1 では必要となります。

このタスクについて

122 ページの表 26には、QMF バージョン 12.1 への移行がサポートされている移行元の QMF リリースに関連付けられた、パッケージ ID のバージョン固有文字がリストされています。

手順

特定リリースの全 QMF パッケージをリストするには、ご使用のサーバーに対して該当するステートメントを実行します。X を、削除対象の QMF リリースに関連付けられたパッケージ ID のバージョン固有文字で置き換えてください。

このステートメントの後に削除する各パッケージの DROP ステートメントを続けます。

- DB2 for Linux, UNIX, and Windows サーバーの場合:

```
SELECT SUBSTR(PKGNAME,1,10)
FROM SYSCAT.PACKAGES
WHERE PKGNAME LIKE 'DSQX%'
```

- DB2 for iSeries サーバーの場合:

```
SELECT SUBSTR(PACKAGE_NAME,1,10)
FROM QSYS2.SYSPACKAGE_
WHERE PACKAGE_NAME LIKE 'DSQX%'
```

例

例えば、DB2 for iSeries サーバー上の QMF バージョン 9 パッケージをリストするには、次のステートメントを発行します。

```
SELECT SUBSTR(PACKAGE_NAME,1,10)
FROM QSYS2.SYSPACKAGE_
WHERE PACKAGE_NAME LIKE 'DSQF%'
```

新旧のリリースが異なるサーバーにある場合

以下のタスクを実行して、QMF の旧バージョンを iSeries または Linux、UNIX、および Windows 版データベースから削除します。

始める前に

旧リリースからまだ使用する必要のある QMF 照会、プロシージャ、または書式を QMF バージョン 12.1 に移行する。オブジェクトの移行方法の詳細については、113 ページの『第 6 章 移行およびフォールバックの考慮事項』を参照してください。

このタスクについて

重要: このトピックの手順は、QMF の全トレースをリモート・サーバーから削除します。この作業を行うのは、新旧リリースが別サーバーにあり、QMF の旧リリースへの接続が今後不要となる場合だけです。

QMF コントロール表およびパッケージの削除

この手順は、QMF コントロール表を削除します。これには QMF オブジェクト・コントロール表、およびそこに含まれる照会、プロシージャー、または書式があります。

このタスクについて

旧バージョンの QMF を含む DB2 for Linux, UNIX, and Windows サーバーまたは DB2 for iSeries サーバーに対して、以下の手順を実行します。

手順

1. QMF1210.SDSQSAPE(DSQ1EDX1) を編集します。
2. *ssid* を DB2 for z/OS リクエスターの 4 文字の名前に変更し、*location* をサーバー・データベースの名前に変更します。
3. ジョブ内のコメントをチェックして、その他の値をカスタマイズする必要があるかを決定します。
4. ジョブ QMF1210.SDSQSAPE(DSQ1EDX1) をサブミットします。
5. DSQCDROP ジョブ・ステップで、戻りコード 0 または 4 について確認します。SYSTERM の完了メッセージを検討します。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。エラーを訂正してジョブを再実行します。

QMF サンプル表の削除

この手順では、サーバーからすべての QMF サンプル表を除去します。

手順

1. QMF1210.SDSQSAPE(DSQ1EDX2) を編集します。
2. *ssid* を DB2 for z/OS リクエスターの 4 文字の名前に変更し、*location* をサーバー・データベースの名前に変更します。
3. ジョブ内のコメントをチェックして、その他の値をカスタマイズする必要があるかを決定します。
4. ジョブ QMF1210.SDSQSAPE(DSQ1EDX2) をサブミットする。
5. DSQCDROP ジョブ・ステップで、戻りコード 0 または 4 について確認します。SYSTERM の完了メッセージを検討します。

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。エラーを訂正してジョブを再実行します。

DB2 (VSE および VM 版) データベースからの QMF の削除

この手順は、QMF の新旧リリースが同じデータベースにあるか、異なるデータベースにあるかにより異なります。

このタスクについて

このトピックの手順では、QMF の旧リリースを DB2 (VSE および VM 版) データベースから削除します。

重要: このトピック内の作業を開始できるのは、QMF の旧バージョンがサーバー内で今後不要となることを確認してからです。旧バージョンが依然として必要となるのは、同じリリースの QMF システム環境と接続する必要がある場合です。

新旧のリリースが同じ DB2 (VSE および VM 版) サーバーにある場合

QMF の新旧リリースが同じ DB2 (VSE および VM 版) サーバーにある場合は、旧リリースからすべての QMF パッケージを除去することにより、QMF の旧リリースを削除します。このケースでは、QMF コントロール表およびサンプル表は QMF バージョン 12.1 では必要となります。

このタスクについて

122 ページの表 26には、QMF バージョン 12.1 への移行がサポートされている移行元の QMF リリースに関連付けられた、パッケージ ID のバージョン固有文字がリストされています。

手順

特定リリースの全 QMF パッケージをリストするには、次のステートメントを実行します。X を、削除対象の QMF リリースに関連付けられたパッケージ ID のバージョン固有文字で置き換えてください。

このステートメントの後に削除する各パッケージの DROP ステートメントを続けます。

```
SELECT SUBSTR(PROGNAME,1,10)
FROM SYSTEM.SYSPROGAUTH
WHERE PROGNAME LIKE 'DSQX%'
```

例

例えば、以下のステートメントを発行して、DB2 (VSE および VM 版) サーバーから QMF バージョン 9 パッケージをリストします。

```
SELECT SUBSTR(PROGNAME,1,10)
FROM SYSTEM.SYSPROGAUTH
WHERE PROGNAME LIKE 'DSQF%'
```

新旧のリリースが異なるサーバーにある場合

QMF を DB2 (VM または VSE 版) サーバーから削除するには、次のステップを完了してください。

このタスクについて

重要: このトピックの手順では、QMF の全トレースをリモート VM または VSE サーバーから削除します。この作業を行うのは、QMF の新旧リリースが別サーバーにあり、QMF の旧リリースへの接続が今後不要となる場合だけです。

手順

1. 次の照会を実行して、旧 QMF パッケージをリストします。次に、その照会結果のリストにあるパッケージ名を除去します。削除対象の QMF リリースに対応する 122 ページの表 26 を見て、該当するバージョン固有文字に DSQX の X を置き換えます。

```
SELECT SUBSTR(PROGNAME,1,10)
FROM SYSTEM.SYSPROGAUTH
WHERE PROGNAME LIKE 'DSQX%'
```

例えば、以下のステートメントを発行して、DB2 for VM または VSE サーバーから QMF バージョン 9 パッケージをリストします。

```
SELECT SUBSTR(PROGNAME,1,10)
FROM SYSTEM.SYSPROGAUTH
WHERE PROGNAME LIKE 'DSQF%'
```

2. 次の照会を実行して、QMF 操作で使用される視点をリストします。次に、結果視点名をドロップします。

```
SELECT * FROM SYSTEM.SYSVIEWS
WHERE VCREATOR = 'Q'
```

3. 次の照会を実行して、QMF コントロール表をリストします。次に、照会した結果として表示された表名をドロップします。

```
SELECT * FROM SYSTEM.SYSCATALOG
WHERE CREATOR = 'Q'
```

4. 次の照会を実行して、QMF で使用される DB スペースをリストします。次に、照会した結果として表示された DB スペース名をドロップします。

```
SELECT * FROM SYSTEM.SYSDBSPACES
WHERE DBSPACENAME LIKE 'DSQ%'
```

第 8 章 オプション機能のインストール

QMF の基本インストールを完了後に、オプション機能をインストールすることができます。

QMF Analytics for TSO の構成

QMF Analytics for TSO の構成では、ファイルのカスタマイズおよび割り振りが必要です。

始める前に

QMF Analytics for TSO を使用する前に、以下を行う必要があります。

- IBM Graphical Data Display Manager - PGF (GDDM-PGF) のインストール検査手順 (IVP) をインストール、構成、および実行します。
- QMF Analytics for TSO がアクセスするエミュレーターがデータをグラフィカル (図表および統計グラフ) に表示できることを確認します。
- QMF Analytics for TSO V12 のインストールおよび IVP を実行します。

QMF Analytics for TSO のロード・ライブラリーに関する考慮事項

QMF Analytics for TSO に必要なモジュールは、接頭部「DYQ」を持ち、QMF for TSO ロード・ライブラリー SDSQLOAD 内にあります。

QMF Analytics for TSO を使用する場合、DSQLLIB を使用したロード・ライブラリーの割り振り方法は、SDSQLOAD には使用できません。

QMF Analytics for TSO で使用されるファイルの割り振り

QMF Analytics for TSO を使用するには、標準の QMF for TSO ファイルおよび QMF Analytics for TSO 構成ファイルを割り振る必要があります。

このタスクについて

QMF Analytics for TSO 構成ファイルは h1q.SDYQCFG(DYQCFG) であり、DDNAME には DYQCFG が使用されます。これを割り振ると、QMF Analytics for TSO 用のデフォルト構成が行われます。

手順

DDNAME DYQCFG は、以下のいずれかの方法で割り振ることができます。

- QMF プロシージャまたは QMF コマンド行から以下のようなコマンドを使用して割り振ります。

```
TSO ALLOC FI(DYQCFG) DSN('QMF1110.SDYQCFG(DYQCFG)')
```

- REXX で標準の QMF 割り振りを使用して割り振ります。

```
"ALLOC FI(DYQCFG) DA('QMF1110.SDYQCFG(DYQCFG)') SHR REUSE"
```

タスクの結果

QMF Analytics for TSO 構成ファイルを割り振りました。

次のタスク

デフォルト構成がご使用の環境に適さない場合は、カスタマイズすることができます。

QMF Analytics for TSO 用のファイル・システムのカスタマイズ

QMF Analytics for TSO は、zSeries ファイル・システム (zFS) を使用して、ユーザーの作業ファイルを保管します。各ユーザーの作業ファイル用のスペースを zFS ディレクトリーに用意する必要があります。

QMF Analytics for TSO および作業ファイル

QMF Analytics for TSO は、作業ファイルを使用して、図表および統計分析の生成中にデータを操作します。ユーザーが互いのデータを表示できないように、各ユーザーの作業ファイルを分離しておくことが重要です。

ZFS 作業ファイル使用のデフォルト・ロケーションは、hlq.SDYQCFG(DYQCFG) 構成ファイル内の「:asfiles.ZFS」構成エントリーによって決まります。

QMF Analytics for TSO は、ファイル・システム (zFS ファイルまたは HFS ファイル) 上のユーザーのホーム・ディレクトリーにある /QAT/ という名前のディレクトリーに作業ファイルを保管することで、ユーザーの作業ファイルを分離します。

システム上の各ユーザーが固有のホーム・ディレクトリーを使用してセットアップされている限り、このデフォルト構成を使用することができます。ただし、ユーザーが共通のホーム・ディレクトリーを共有するようご使用の環境が構成されている場合、または単にデフォルト構成で指定されたロケーションを使用したくない場合は、QMF Analytics for TSO が作業ファイルを保管するディレクトリー・ロケーションを設定します。これを行うには、QMF Analytics for TSO 構成ファイル (DYQCFG) 内の :asfiles エントリーを編集します。

重要: :asfiles. エントリーは、QMF Analytics for TSO 構成ファイル内で変更する必要がある唯一の行です。他の行を変更したり、これらの指示に従わずに :asfiles. エントリーを変更したりすると、QMF Analytics for TSO は正しく機能しません。

出荷時の :asfiles. エントリーのデフォルト設定は :asfiles.ZFS です。前述のとおり、この設定を使用すると、QMF Analytics for TSO は作業ファイルを [home directory]/QAT/ に保管します。

QMF Analytics for TSO 作業ファイルを別の場所に置くには、以下のように :asfiles.ZFS 設定を編集してください。

- 作業ファイルを [user's_home_directory]/QAT/ 内に保管するには、設定を :asfiles.ZFS のままにします。

- 作業ファイルを [user's_home_directory]/root/QAT/ 内に保管するには、
:asfiles.ZFS root (/ を付けない) を指定します。
- 作業ファイルを [file_system_root]/root/QAT / 内に保管するには、
:asfiles.ZFS /root (/ を付ける) を指定します。

root および /root の長さは 227 文字に制限されており、多数のレベルのサブディレクトリーを含めることができることに注意してください。末尾のスラッシュ / はオプションであり、ない場合でも QMF Analytics for TSO はスラッシュがあるものと想定します。

/root の使用時、または各ユーザーのホーム・ディレクトリーが固有ではない場合に、パス名を固有なものにするには、パス名内に &SYSUID. を使用することができます。QMF Analytics for TSO は、これを解釈する際に、&SYSUID. を現行ユーザーのユーザー ID (小文字) で置き換えます。

次の例は、:asfiles.ZFS のカスタマイズ方法を示しています。

注: QMF Analytics for TSO では、階層ファイル・システム (HFS) もサポートされています。HFS を使用している場合は、:asfiles の場合と全く同じセットアップ手順が適用されます。

```
*-----*
* QAT Configuration Entry for use with zFS/HFS      *
*-----*
:nick.QAT      :type.ENV
                :asfiles.ZFS products/workfiles/&SYSUID./QMF
                :library.ZFS
                :log.OFF
                :sortname.ICEMAN
```

この設定では、ユーザー「Alan」および「FINANCE」の作業ファイルは以下のよう
に保管されます。

- Alan の場合:

```
[home_directory]/products/workfiles/alan/QMF/QAT/
```

- FINANCE の場合:

```
[home_directory]/products/workfiles/finance/QMF/QAT/
```

前に / が付いたパス (:asfiles.ZFS /products/ など) を指定した場合は、ユーザーのホーム・ディレクトリーではなく /products/ ディレクトリーがファイル・システムのルート・ディレクトリーに置かれます。

QMF Analytics for TSO サンプル表のインストール

QMF Analytics for TSO 機能を解説するために資料で言及されている QMF Analytics for TSO サンプル表を、オプションでインストールすることができます。これらのサンプル表は、

Q.CASHFLOW、Q.CLIMATE_10YR、Q.CLIMATE_USA、および Q.WORLDINFO です。

始める前に

QMF Analytics for TSO を DB2 for z/OS または DB2 for Linux, UNIX, and Windows データベースにインストールした場合は、QMF Analytics for TSO サンプル表をインストールする前に基本 QMF サンプル表をインストールします。DB2 for z/OS では、基本サンプル表をインストールするジョブは DSQ1EIVS です。DB2 for Linux, UNIX, and Windows では、基本サンプル表をインストールするジョブは DSQ1EDJ4 です。QMF Analytics for TSO サンプル表を DB2 for iSeries データベースにインストールする前に、基本サンプル表をインストールする必要はありません。

手順

QMF Analytics for TSO サンプル表をインストールするには、以下のようになります。

ジョブ QMF1210.SDSQSAPE(DSQ1EIVQ) を実行します。

DB2 for z/OS および DB2 for Linux, UNIX, and Windows では、ジョブ DSQ1EIVQ は、ジョブ DSQ1EIVS または DSQ1EDJ4 によって作成された DSQ1STBT 表スペースにサンプル表をインストールします。

QMF Analytics for TSO が正しくインストールされているかを検査

QMF Analytics for TSO が正しくインストールされていることを検査でき、また照会結果の図表および統計分析を作成することで製品機能をテストできます。

始める前に

インストールを検査する前に、以下の作業を行ってください。

- IBM Graphical Data Display Manager - PGF (GDDM-PGF) のインストール検査手順 (IVP) をインストール、構成、および実行します。
- QMF Analytics for TSO がアクセスするエミュレーターがデータをグラフィカル (図表およびグラフ) に表示できることを確認します。
- QMF for TSO V12.1 をインストールしたこと、および QMF for TSO IVP を実行したことを確認します。

このタスクについて

検査手順では、QMF for TSO に付属の以下の 2 つのサンプル・データ表に対して照会を実行します。

- Q.STAFF

この表には、従業員の給与、部門、歩合、雇用日などの人事関連のデータがあります。

- Q.CLIMATE_10YR

この表には、長年にわたって収集された降雨量、晴れ日数、気温などの気候および天気関連のデータがあります。

手順

QMF Analytics for TSO が正しくインストールされていることを検査するには、以下のようにします。

1. QMF を開始します。
2. コマンド行で **DISPLAY Q.STAFF** と入力します。
3. QMF コマンド行で **SHOW ANALYTICS** と入力して Enter を押し、QMF Analytics for TSO を開始します。
4. 「QMF Analytics ホーム」ページの「図表」パネルで、「タワー」にタブで移動し、Enter を押します。
5. 最初のデータ入力フィールドにカーソルを置いて、「リスト」ファンクション・キー (F4) を押します。「列選択」パネルの「**COMM**」に移動し、Enter を押します。

QMF Analytics for TSO によって再度「図表」パネルが表示され、データ入力フィールドに「**COMM**」が挿入されます。

6. 「タワー図表」の「パラメーター選択」パネルにある 2 番目のデータ入力フィールドに移動し、「リスト」ファンクション・キー (F4) を押します。「列の選択」パネルの「**SALARY**」に移動し、Enter を押します。

QMF Analytics for TSO によって再度「図表」パネルが表示され、データ入力フィールドに「**SALARY**」が挿入されます。

7. X 軸のデータ入力フィールドにカーソルを置いて、「リスト」ファンクション・キー (F4) を押します。「列の選択」パネル上の「**DEPT**」に移動し、Enter を押します。

QMF Analytics for TSO によって再度「図表」パネルが表示され、データ入力フィールドに「**DEPT**」が挿入されます。

8. 「実行」ファンクション・キーを押して、図表を作成します。
9. 「パラメーター選択」パネルに戻るには、F12 を押します。
10. 「QMF Analytics ホーム」パネルに戻るには、「終了」ファンクション・キー (F3) を押します。
11. QMF コマンド行で **DISPLAY Q.CLIMATE_10YR** と入力します。
12. コマンド行で **SHOW ANALYTICS** と入力します。
13. 「QMF Analytics ホーム」ページの「統計」セクションにある「基本」に移動し、Enter を押します。
14. 最初のデータ入力フィールドにカーソルを置いて、「リスト」ファンクション・キー (F4) を押します。
15. 「列の選択」パネルの「**RAINFALL**」に移動し、Enter を押します。

QMF Analytics for TSO によって再度「統計」パネルが表示され、データ入力フィールドに「**RAINFALL**」が挿入されます。

16. 降雨の基本統計の要約を見るには、「実行」ファンクション・キー (F2) を押します。
17. 降雨データの基本統計を生成するには、「統計」ファンクション・キー (F10) を押します。

QMF Data Service (QDS) の構成

QMF for z/OS および QMF Enterprise Edition のユーザーは、QMF Data Service フィーチャーを使用して、DB2 以外のデータ (VSAM、IMSTM、順次ファイル、SMF データ、SYSLOG データなど) にアクセスすることができます。

始める前に

QMF for z/OS および QMF Enterprise Edition のユーザーが使用可能な QMF Data Services コンポーネントがインストールされていて、QMF for TSO/CICS 環境で有効であることが必要です。データ・ソースが QMF Data Service に定義されている必要があります。QMF for TSO/CICS 大域変数 DSQEC_DS_SUPPORT は値 1 に設定されている必要があります。QDS コンポーネントのインストールおよび構成 (パッケージのバインドや適切な QDS ライブラリーの割り振りなど) に関する詳しい説明については、「IBM DB2 QMF Data Service Getting Started Guide」を参照してください。QDS と QMF for TSO/CICS の構成を完了するためには、以下の作業を実行する必要があります。

このタスクについて

QMF for TSO/CICS SQL 照会によって、3 部構成の名前を使用して QMF Data Service 定義のデータ・ソースにアクセスできます。QMF Data Service データ・ソースは、SQL 照会における 3 部構成の表名によってアクセスされます。

例えば、次の照会を実行することによって、VDBX という名前の QMF Data Service サーバーに定義されている VSAM データ・セットにアクセスできます。

```
SELECT * FROM VDBX.DVSQL.VSAM_IMITMTRN
```

QMF Data Service は、サーバーに存在する 1 つ以上のソースを結合することができます。QMF Data Service によって受け入れられる SQL は、DB2 for z/OS によって受け入れられる SQL のサブセットです。受け入れられる SQL 構文については、「QMF Data Service SQL Guide」を参照してください。

QMF Data Service を使用するためには、QMF for TSO/CICS の大域変数 DSQEC_DS_SUPPORT を 1 に設定する必要があります。デフォルト値は 0 で、QMF Data Service へのアクセスを許可しないことを意味します。

DSQEC_DS_SUPPORT 大域変数を 1 に設定すると、SQL 照会は QMF Data Service コンポーネントによって分析されます。照会で参照されているオブジェクトが QMF Data Service コンポーネントに対して定義されている場合、照会全体が QDS によって実行されます。照会で参照されているオブジェクトのいずれも、QDS に対して定義されているオブジェクトにアクセスしない場合、照会は現行の DB2 Connect によって実行されます。

QMF 各国語機能のインストール

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

始める前に

- まだ実行済みでない場合は、NLF のインストール先のデータベースに QMF バージョン 12.1 (英語) のインストール、準備、およびテストを行ってください。このインストールは基本製品です。
 - リクエスターのインストール・パスについては、20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』を参照してください。
 - QMF CONNECT コマンドによってアクセスされるサーバーのインストール・パスについては、26 ページの『ロードマップ 2: QMF CONNECT コマンドでアクセスされるサーバー・データベースでの QMF のインストール』を参照してください。
 - サーバーが 3 部構成の名前によってのみアクセスされる場合、40 ページの『インストール・パス G: 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーの準備』を既に完了していれば、これ以上の NLF 関連のステップを実行する必要はありません。
- NLF をインストールするデータベースに対して適切な権限を保持していることを確認してください。詳細については、17 ページの『QMF のインストールに必要な権限』を参照してください。

このタスクについて

一部のモジュール、ライブラリー、およびジョブの名前には、*n* 文字が含まれます。この文字は、NLF 用の 1 文字の言語 ID を表す変数です。この文字があった場合は、NLF に関連付けられているこの言語 ID を置き換えてください。これらの ID は、136 ページの表 27 に表示されています。

手順

QMF NLF をインストールするには、以下の手順に従ってください。

1. 補足データについては、NLF プログラム・ディレクトリーをお読みください。複数のプログラム・ディレクトリーがあります。必ず、ご使用の QMF のエディションに適した各国語版プログラム・ディレクトリーを参照してください。プログラム・ディレクトリーは、NLF リリース間で更新され、PTF および APAR の説明を含む有用な情報が収められています。
2. プログラム・ディレクトリーで説明されているように、SMP/E を使用して、QMF バージョン 12.1 用の NLF ライブラリーを配布メディアから、基本製品をインストールしたデータベースにロードします。インストールしようとしている各 NLF に対して NLF プログラム・ディレクトリーで説明されているすべての SMP/E 作業を完了したことを確認してください。

NLF の JCL および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。

下の表には、QMF を使用できる各言語に関する情報があります。機能変更 ID (FMID) により、各 QMF NLF が SMP/E で識別されます。

1 文字の言語 ID (言語 ID) が、データ・セット、ライブラリー、およびモジュール名で使用されます。QMF は、特定の QMF コントロール表にある列

2 内の変換された名前を使用します。英語は、完全性のためにのみ含まれています。英語は各国語機能ではありません。

NLF のインストールでは、QMF バージョン 12.1 のサンプル・ライブラリー QMF1210.SDSQSAPE および QMF1210.SDSQSAP n 、およびロード・モジュール・ライブラリー QMF1210.SDSQLOAD を使用する必要があります。

表 27. QMF NLF の言語 ID と FMID

| 各国語機能 (National Language Feature) | この NLF に QMF が使用する名前 | 言語 ID | QMF バージョン 12.1 FMID |
|-----------------------------------|----------------------|-------|---------------------|
| 英語 | ENGLISH | E | HSQCC10 |
| U/C 英語 | UPPERCASE | U | JSQCC51 |
| デンマーク語 | DANSK | Q | JSQCC55 |
| カナダ・フランス語 | FRANCAIS CANADIEN | C | JSQCC5G |
| フランス語 | FRANCAIS | F | JSQCC56 |
| ドイツ語 | DEUTSCH | D | JSQCC57 |
| イタリア語 | ITALIANO | I | JSQCC58 |
| 日本語 | NIHONGO | K | JSQCC59 |
| 韓国語 | HANGEUL | H | JSQCC5A |
| ブラジル・ポルトガル語 | PORTUGUES | P | JSQCC5B |
| スペイン語 | ESPANOL | S | JSQCC5C |
| スウェーデン語 | SVENSKA | V | JSQCC5D |
| スイス・フランス語 | FRANCAIS (SUISSE) | Y | JSQCC5E |
| スイス・ドイツ語 | DEUTSCH (SCHWEIZ) | Z | JSQCC5F |

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) に関連付けます。QMF のシステム・リリース・レベルは P115 です。

3. NLF のインストール先であるデータベースを開始します。サーバーとして機能するデータベースに NLF をインストールする場合は、サーバーとリクエスター間の DRDA 通信を使用して、そのサーバーへの QMF バージョン 12.1 の NLF のインストールを行います。そのため、以下の手順を実行する必要があります。
 - a. NLF のインストール先のサーバーと基本 QMF バージョン 12.1 製品をインストール済みのリクエスターとの間で DRDA 通信を定義およびテストします。DRDA の定義およびテスト方法の詳細については、「DB2 資料」を参照してください。
 - b. リクエスター・データベースおよびサーバー・データベースの両方を開始します。リクエスター・データベース内の DB2 for z/OS サブシステムは、分散データ機能 (DDF) を使用して開始する必要があります。

NLF は、その各国語で QMF を始動する各 DB2 サブシステムにインストールする必要があります。必要に応じて複数の NLF を各サブシステムにインストールすることができます。

4. インストールしようとしている NLF に対するインストール・パラメーターをカスタマイズします。このカスタマイズは、ステップ 6 のジョブの中で行います。55 ページの表 11 には、QMF インストール・ジョブ内のインストール・パラメーターが記載されています。一般的には、カスタマイズされるパラメーターには、QMFTPRES、DB2EXIT、DB2LOAD、SSID、および LOCATION があります。ジョブごとに、RGN パラメーターのジョブ・ステップ領域サイズは 2048K (この設定がデフォルト) です。

NLF 固有のコマンド同義語を保管する Q.COMMAND_SYNONYM_n 表のスペース割り振りは、1 次スペース割り振りに対して 100、2 次スペース割り振りに対して 20 が設定されます。表 Q.COMMAND_SYNONYMX_n のインデックスの 1 次割り振りと 2 次割り振りは、同一サイズです。

5. ジョブ QMF1210.SDSQSAP_n(DSQ1_nPNL) を実行することによって、NLF 用の VSAM パネル・ライブラリーにデータを設定します。このジョブの詳細については、QMF プログラム・ディレクトリーを参照してください。
6. インストールする NLF ごとに、下の表にあるインストール・ジョブを、示されている順序で実行します。

TSO バッチ・モードで実行するすべてのステップに対して、システム・メッセージ内のステップ完了コードを検査します。完了メッセージは、SYSTSPRT または SYSTERM の出力の中にあります。戻りコード 0 または 4 について確認してください。

それぞれのジョブが完了したら、SYSTERM 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。SYSPRINT には、IBM ソフトウェア・サポート用の詳細な診断情報が収容されています。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します。

表 28. QMF 各国語機能をリクエストまたはサーバー・データベースにインストールするためのジョブ・シーケンス

| ジョブ名 | 説明 |
|---|--|
| 次の各ジョブは、ターゲット・データベースに QMF NLF コントロール表を作成し、Q.PROFILES 表に NLF をサポートする 2 つの行を追加します (TSO 用の行と CICS 用の行)。これらの行は、QMF ユーザー ID 値 SYSTEM が付けられて、この表の CREATOR 列に挿入されます。 | |
| QMF1210.SDSQSAP _n (DSQ1 _n UPO) | このジョブは、NLF を DB2 for z/OS データベースにインストールしている場合に実行します。 |
| QMF1210.SDSQSAP _n (DSQ1 _n DJ2) | このジョブは、NLF を DB2 for Linux, UNIX, and Windows データベースにインストールしている場合に実行します。 |
| QMF1210.SDSQSAP _n (DSQ1 _n AS2) | このジョブは、NLF を DB2 for iSeries データベースにインストールしている場合に実行します。 |
| QMF1210.SDSQSAP _n (DSQ1 _n CCS) | Q.COMMAND_SYNONYM _n という名前のコマンド同義語表を NLF 環境用に作成します。 |

表 28. QMF 各国語機能をリクエスターまたはサーバー・データベースにインストールするためのジョブ・シーケンス (続き)

| ジョブ名 | 説明 |
|----------------------------|--|
| | <p>次のジョブは、既存の QMF NLF サンプル表を削除し、新しいバージョンをインストールします。</p> <p>旧リリースの NLF サンプル表は、新しい QMF NLF インストールでも有効です。これらのジョブは、何らかの理由によりサンプルを削除して再インストールしなければならない場合のために示されています。NLF をインストールする場所にあるすべてのサンプル表で、SELECT 特権が PUBLIC に付与されます。これらのジョブを必要に応じていつでも再実行できます。</p> <p>大文字英語またはスウェーデン語の NLF ではサンプル表は使用不可です。</p> |
| QMF1210.SDSQSAPn(DSQ1nDSJ) | DB2 for z/OS データベースからの既存の QMF NLF サンプル表の削除 |
| QMF1210.SDSQSAPn(DSQ1nIVS) | <p>QMF NLF サンプル表を DB2 for z/OS データベースにインストールします。CDS および CDP パラメーターは、小数点に使用する句読記号を識別します。この値は、DB2 のインストール時に指定された DECPOINT オプションと一致する必要があります。</p> <ul style="list-style-type: none"> • ピリオドの場合は、現行値をそのままにしてください。 • コンマの場合は、CDS を 6、CDP を 7 に変更してください。 |
| QMF1210.SDSQSAPn(DSQ1nDX2) | <p>次のデータベースからの既存の QMF NLF サンプル表の削除:</p> <ul style="list-style-type: none"> • DB2 for iSeries サーバー • DB2 for Linux, UNIX, and Windows サーバー <p>サンプル表を VM または VSE データベースから削除するには、削除する表に対して DROP ステートメントを発行します。</p> |
| QMF1210.SDSQSAPn(DSQ1nAS4) | QMF NLF サンプル表を DB2 for iSeries データベースにインストールします。 |
| QMF1210.SDSQSAPn(DSQ1nDJ4) | QMF NLF サンプル表を DB2 for Linux, UNIX, and Windows データベースにインストールします。 |

7. QMF Analytics for TSO サンプル表をインストールする場合は、QMF1210.SDSQSAPn(DSQ1nIVQ) ジョブを実行します。

DSQ1nIVQ ジョブは、対応する QMF NLF に付属しています。

重要: DB2 for z/OS または DB2 for Linux, UNIX, and Windows サーバーの場合、ジョブ DSQ1nIVQ を実行する前に、DSQ1nIVS または DSQ1nDJ4 ジョブを実行して、基本 QMF サンプル表をインストールします。

8. QMF NLF を実行中の環境に対して、その QMF NLF をカスタマイズします。

| オプション | 説明 |
|--------------------|--|
| QMF が TSO 内で実行中の場合 | <p>66 ページの『TSO ログオン・プロシーチャーの準備』で準備した QMF 基本 (英語) 製品用の TSO ログオン・プロシーチャーを編集します。ここで強調表示されている各 NLF ステートメントを、ログオン・プロシーチャーで各英語ステートメントに連結します。</p> <pre> //ADMGGMAP DD DSN=QMF1210.SDSQMAPn,DISP=SHR // DD DSN=QMF1210.SDSQMAPE,DISP=SHR //ISPPLIB DD DSN=QMF1210.SDSQPLBn,DISP=SHR // DD DSN=QMF1210.SDSQPLBE,DISP=SHR //ISPMLIB DD DSN=QMF1210.SDSQMLBn,DISP=SHR // DD DSN=QMF1210.SDSQMLBE,DISP=SHR //ISPTLIB DD DSN=QMF1210.SDSQTLBn,DISP=SHR // DD DSN=QMF1210.SDSQTLBE,DISP=SHR // DD DSN=ISP.SISPTENU,DISP=SHR //SYSPROC DD DSN=SYS2.CLIST,DISP=SHR // DD DSN=QMF1210.SDSQCLTn,DISP=SHR // DD DSN=QMF1210.SDSQCLTE,DISP=SHR //SYSEXEC DD DSN=SYS2.EXEC,DISP=SHR // DD DSN=QMF1210.SDSQEXCn,DISP=SHR // DD DSN=QMF1210.SDSQEXCE,DISP=SHR //DSQPNLn DD DSN=QMF1210.DSQPNLn,DISP=SHR // DD DSN=QMF1210.DSQPNLE,DISP=SHR </pre> |

| オプション | 説明 |
|----------------------------|---|
| <p>QMF が CICS 内で実行中の場合</p> | <ol style="list-style-type: none"> 1. ジョブ DSQ1nADM を実行することによって、QMF/GDDM マップを GDDM ADMF データ・セットにロードします。必要に応じてパラメーター QMFTPRE、GDDMADM、および RGN を変更して、インストールしている NLF に合うようにします。このジョブについては、76 ページの『GDDM ADMF データ・セットへの QMF GDDM マップのロード』の基本製品で説明されています。 2. ジョブ DSQ1nCSD をカスタマイズおよび実行して、CICS システム定義ファイル内に QMF NLF 用のエントリを作成します。このジョブについては、74 ページの『CICS に対して QMF を記述』の基本製品で説明されています。 このジョブに対する戻りコードは、0 または 4 です。 3. QMF NLF パネル・ファイルを既存の JCL に割り振るための行を追加します。この JCL は、QMF を含む CICS 領域を開始するために使用します。この JCL は、78 ページの『CICS 始動ジョブ・ストリームの更新』に示されています。 QMF NLF パネル・ファイルに行を追加するには、まず、CICS 開始 JCL 内の QMF 基本製品に対するステートメントを見つけます。 //DSQPNLE DD DSN=QMF1210.DSQPNLE,DISP=SHR 次に、このステートメントの前に、QMF NLF パネル・ファイル用の次のステートメントを追加します。 //DSQPNLn DD DSN=QMF1210.DSQPNLn,DISP=SHR |

9. TSO 下の QMF を始動するには、94 ページの『TSO での QMF の開始』の手順を使用します。
10. NLF バージョンの QMF 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には、必要に応じて、既存の NLF 照会およびプロシージャーを削除し、バージョン 12.1 NLF オブジェクトを作成します。
 - a. 既存の NLF バージョンのサンプル照会およびプロシージャーを削除するには、次のコマンドを QMF for TSO コマンド行から発行します。デフォルト接頭部 QMF1210 ではなく、サイト固有の QMF ライブラリー接頭部を使用している場合は、プロシージャーを保存する前に、コマンド内で、また各プロシージャー全体を通して、QMF1210 を置き換えてください。

IMPORT PROC FROM QMF1210.SDSQSAPn(DSQ1nSQD)'
SAVE PROC AS Q.DSQ1nSQD
RUN PROC
 - b. 次のコマンドを発行してバージョン 12.1 NLF サンプル照会およびプロシージャーを作成します。

```

IMPORT PROC FROM 'QMF1210.SDSQSAPn(DSQ1nSQI)
SAVE PROC AS Q.DSQ1nSQI
RUN PROC

```

例えば、既存のドイツ語サンプル照会およびプロシーチャーを削除するには、次のコマンドを発行します。

```

IMPORT PROC FROM QMF1210.SDSQSAPD(DSQ1DSQD)'
SAVE PROC AS Q.DSQ1DSQD
RUN PROC

```

バージョン 12.1 ドイツ語サンプル照会およびプロシーチャーを作成するには、次のコマンドを発行します。

```

IMPORT PROC FROM 'QMF1210.SDSQSAPD(DSQ1DSQI)
SAVE PROC AS Q.DSQ1DSQI
RUN PROC

```

このジョブの実行中に障害が発生した場合は、エラーを訂正し、DSQ1nSQD プロシーチャーを再実行します。その後、DSQ1nSQI プロシーチャーを再実行します。サンプル照会およびプロシーチャーの詳細については、102 ページの『インストール検査手順に必要なサンプル照会およびサンプル・プロシーチャーのインストール』を参照してください。

11. 98 ページの『QMF パネル・ライブラリーの正しいバージョンの確認』の手順を使用して、正しいバージョンの QMF NLF パネル・ライブラリーをテストします。ホーム・パネルおよびヘルプ・パネルの両方が、インストールした言語で表示されることを確認します。
12. ご使用の環境に適したインストール検査手順を実行します。QMF を CICS 下で実行する場合は、TSO の IVP を完了してから CICS の IVP を完了してください。

| オプション | 説明 |
|---------------------|--|
| TSO 下の QMF NLF の場合 | <p>QMF コマンド行から次のコマンドを発行して、IVP を実行します。必要に応じて、サイトの要件に応じて手順をカスタマイズしてください。例えば、デフォルトの QMF1210 以外のライブラリー接頭部を使用している場合は、保存する前に、インストール検査手順全体で QMF1210 を置き換えます。</p> <pre> DISPLAY Q.DSQ1nIVP SAVE PROC AS Q.DSQ1nIVP (SHARE=YES) RUN PROC </pre> |
| CICS 下の QMF NLF の場合 | <p>98 ページの『CICS での QMF の開始』の手順を使用して CICS を開始します。次に、QMF コマンド行から次のコマンドを発行して、IVP を実行します。必要に応じて、サイトの要件に応じて手順をカスタマイズしてください。例えば、デフォルトの QMF1210 以外のライブラリー接頭部を使用している場合は、保存する前に、インストール検査手順全体で QMF1210 を置き換えます。</p> <pre> DISPLAY Q.DSQ1nIVC SAVE PROC AS Q.DSQ1nIVC (SHARE=YES) RUN PROC </pre> |

IVP でテストされる機能の詳細については、次のトピックのいずれかを参照してください。

- 105 ページの『TSO 用 IVP の実行』
- 108 ページの『CICS 用 IVP の実行』

IVP がエラーなく実行したときには、NLF インストールは完了です。対話式 IVP が TSO 上で完了した後に、オプションでバッチ・モード IVP を実行することができます。詳細については、106 ページの『TSO でのバッチ操作のテスト』を参照してください。

13. インストール後に、必要に応じて QMF NLF の旧リリースのライブラリーを削除します。これらのライブラリーの英語版は、121 ページの『SMP/E を使用して旧リリースのライブラリーを削除する』にリストされています。それぞれの名前の最後の E を、ご自身が使用する NLF の各国語 (NLID) で置き換えてください。

次のタスク

QMF セッション時に表示される各国語を変更するには、現在の 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。

関連資料:

226 ページの『Q.PROFILES 表について』

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・プロフィールです。

関連情報:

 DB2 資料

DECPOINT オプションに関する情報を検索します。

 DB2 照会報告書作成プログラム・ライブラリー

QMF バージョン 12.1 プログラム・ディレクトリーに関する情報を参照してください。

QMF 各国語機能の移行

既存の各国語機能 (NLF) が組み込まれている QMF インストール済み環境を移行する場合は、正しいバージョンの DSQPNL n データ・セット (n は各国語 ID) が割り振られていることを確認してください。バージョン 12.1 への移行時には、NLF 移行プロセスを完了するために、いくつかのジョブを実行する必要があります。

手順

NLF を移行するには、以下のようにします。

ジョブ QMF1210.SDSQSAP n (DSQ1 n RTS) を実行します。

このジョブは、新しいバージョン 12.1 コマンド同義語を作成します。

QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ)

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

始める前に

まだ実行済みでない場合は、QMF バージョン 12.1 (英語) を、このインターフェースのインストール先となる DB2 for z/OS リクエスター・データベースに対してインストール、準備、およびテストを行ってください。

リクエスターのインストール・パスについては、20 ページの『ロードマップ 1: スタンドアロンまたはリクエスター・データベースでの QMF のインストール (DB2 for z/OS のみ)』を参照してください。QMF は、DB2 for z/OS データベースからのみ始動できます。

このタスクについて

DB2 for z/OS ストアード・プロシージャを呼び出すことができるソフトウェア・プログラム (QMF for Workstation など) は、QMF for TSO を始動して、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。事前定義プロシージャまたは照会の名前をインターフェースに渡して、作業を実行し、必要な出力を返すことができます。また、インターフェースに渡されるプロシージャは、バッチ・ジョブを開始することもできます。これにより、CALL ステートメントを発行できる DB2 クライアントはいずれも、QMF for TSO バッチ・サービスにアクセスできます。

QMF バッチ・モードと同じように、ストアード・プロシージャ・インターフェースでは、ユーザーが QMF と対話することはできません。QMF に渡されるプロシージャは、QMF パネルを表示するコマンドを発行できません (例えば、LIST など)。

手順

ストアード・プロシージャ・インターフェースのインストール手順は、次のとおりです。

1. サイトで DB2 for z/OS ストアード・プロシージャを始動するために使用する WLM アドレス・スペースを構成してください。

QMF ストアード・プロシージャ・インターフェースは、REXX ストアード・プロシージャ Q.DSQQMFSP によって使用可能になります。このストアード・プロシージャは、QMF の外部にあり、APF 許可されていない WLM 管理アドレス・スペースで実行する必要があります。

一般に、DB2 for z/OS ストアード・プロシージャ (ネイティブ SQL プロシージャを除く) は、WLM 管理のストアード・プロシージャ・アドレス・スペース内で実行する必要があります。それぞれのストアード・プロシージャを

WLM アプリケーション環境に割り当てる必要があります。これは、ストアード・プロシージャの作業を適切な WLM 管理のアドレス・スペースに経路指定します。

QMF を実行することになる WLM 管理アドレス・スペース始動用 JCL に、次の変更を行います。

- a. NUMTCB パラメーターを 1 に設定します。Q.DSQQMFSP プロシージャは REXX ストアード・プロシージャであり、すべての REXX ストアード・プロシージャは、NUMTCB 値に 1 を割り当てる必要があります。
- b. TSO で QMF を実行するためのサイト要件に合う領域サイズを指定します。

最小記憶域要件については、50 ページの『記憶域のアドレッシング要件』に説明されています。これらの記憶域の数値に、QMF 以外のアプリケーション用に TSO 領域内に予約する記憶域の大きさを追加します。この記憶域は、3 に説明されているように、DSQSCMD n EXEC に DSQSRSTG パラメーターを使用して指定します。

- c. QMF1210.SDSQEXCE ライブラリーを SYSEXEC DD 名に割り振ります。QMF を英語および各国語環境の両方でストアード・プロシージャとして実行する場合は、QMF1210.SDSQEXC n も割り振ります。ここで、 n は、1 文字の各国語 ID です。
- d. QMF1210.SDSQLOAD ロード・ライブラリーを STEPLIB DD 名に割り振ります。
- e. QMF1210.SDSQMAPE ライブラリーを ADMGGMAP DD 名に割り振ります。QMF を英語および各国語環境の両方でストアード・プロシージャとして実行する場合は、QMF1210.SDSQMAP n も割り振ります。ここで、 n は、1 文字の各国語 ID です。
- f. 次のデータ・セットを割り振ります。

DSQDEBUG

QMF トレース出力を受け取ります。

SYSTSPRT

PTF または ALL がトレース・オプションとして指定されているときには、REXX トレース出力を受け取ります。

- g. GDDM パラメーターを、他の QMF for TSO のインストールと同じ方法で、カスタマイズします。
2. QMF ストアード・プロシージャ (Q.DSQQMFSP) を始動したユーザー ID が、QMF ライブラリー (WLM アドレス・スペースの始動用 JCL で割り振ったライブラリー) への読み取りアクセス権限を持っていることを確認してください。
3. QMF1210.SDSQEXC n ライブラリー内にある DSQSCMD n EXEC 内のパラメーター値を改訂します。 n を、QMF を始動する言語を表す 1 文字の各国語 ID に置き換えます。

ストアード・プロシージャのプログラム・ロジックは、DSQSCMD n EXEC を使用して、初期化のためにプログラム・パラメーターを QMF に渡します。DSQSCMD n EXEC 内で、次のパラメーターの値が必要です。

DSQSSUBS

Q.DSQQMFSP ストアド・プロシージャをインストールしようとするサブシステムの名前。

DSQSPLAN

Q.DSQQMFSP ストアド・プロシージャをインストールしようとする DB2 for z/OS サブシステム内の QMF アプリケーション・プランの名前。

QMF バージョン 12.1 アプリケーション・プランのデフォルト名は QMF12 です。

オプションで次のプログラム・パラメーターを設定することができます。

DSQSBSTG または DSQSRSTG

報告書操作に使用する仮想記憶域の大きさ。

DSQSMRFI

データベースのフェッチおよび挿入のための複数行処理を使用可能にします。

DSQSPILL および DSQSPTYP

アクティブな記憶域で、もう不要となったデータ用の特別な記憶域を使用できるようにします。

DSQSDBCS

DBCS データを非 DBCS ディスプレイ装置から印刷できるようにします。

DSQSCMD n EXEC 内のその他すべてのプログラム・パラメーターは、ストアド・プロシージャとして QMF を始動時に、無視されるかまたはデフォルト値のみを使用するかのいずれかです。

4. DSQ1BSP インストール・ジョブを実行して、QMF ストアド・プロシージャを定義します。
5. WLM アドレス・スペースをリフレッシュまたは再開します。
6. このインターフェースを使用して QMF を始動するすべてのユーザーに対して Q.DSQQMFSP プロシージャに EXECUTE 特権を付与します。以下に例を示します。

```
GRANT EXECUTE ON PROCEDURE Q.DSQQMFSP TO userid
```

7. QMF をストアド・プロシージャとして始動するすべてのユーザーが、有効な QMF プロファイルを持っていることを確認してください。

QMF は、Q.PROFILES 表の CREATOR 列の値を使って WLM 管理のアドレス・スペースが開始された許可 ID を、認証します。QMF は、最初にアドレス・スペースが開始された許可 ID の CREATOR 列を検査します。CREATOR 列内の値でこの ID に対応する値がない場合は、QMF は CREATOR=SYSTEM が指定された Q.PROFILES 表内の行を検査します。SYSTEM 行が検出されない場合は、QMF 初期化はエラーで終了します。

アドレス・スペースが開始された許可 ID に関連付けられているプロファイル設定 (または、特定のプロファイルがこのユーザー ID に関連付けられていない場合は、SYSTEM 行内のプロファイル) は、QMF セッション期間中のデフォルト設定として機能します。トレース・レベルは、CALL ステートメント上で渡

されるため、プロファイルの TRACE オプションの初期値は無視されます。
QMF がストアード・プロシージャとして実行しているときにプロンプトが出
されないため、CONFIRM オプションの値 YES も無視されます。

8. ストアード・プロシージャ・インターフェースのインストールが正常に完了したことを確認します。

- 次の CALL ステートメントを QMF for Workstation または別のクライアント・プログラムから発行します。このステートメントは、英語版の QMF for TSO を開始します。

```
CALL Q.DSQQMFSP('','PTF','','',?)
```

このステートメントは、Q.DSQQMFSP プロシージャに関するサービス関連
ストリングを返します。ストリングの形式は次のとおりです。

```
DSQQMFSP exec level: QMF VnRn WIMnnnnn mm-dd-yyyy DSQA0_QMF_VER_RLS:  
QMFVnRn.n DSQ_PRODUCT_RELEASE: nn Current date and time: dd mmm yyyy hh:mm:ss
```

- QMF サンプル照会をインストールした場合、次の CALL ステートメントを発行して、QMF のストアード・プロシージャとしての開始、および Q.SAMPLE_SELECT1 照会の最初のパラメーターとしての受け渡しをテストできます。

このステートメントは、トレース・ログは指定しませんが、ストアード・プロシージャの実行の最後に返されるメッセージは、照会が正常に実行されたかどうかを示します。

```
CALL Q.DSQQMFSP('Q.SAMPLE_SELECT1','','',?)
```

トレース・ログを結果セットとして受け取るには、次のステートメントを発行します。

```
CALL Q.DSQQMFSP('Q.SAMPLE_SELECT1','L2','','',?)
```

タスクの結果

DSQA0_STO_PROC_INT 大域変数は、QMF が DB2 for z/OS ストアード・プロシージャとして始動されたかどうかを記録します。

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャとしての開始』

QMF for TSO のストアード・プロシージャ・インターフェースでは、DB2 for z/OS ストアード・プロシージャを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

454 ページの『トレース・データ・セットの割り振り』

QMF の実行方法に応じて、DSQUDUMP データ・セットまたは DSQDEBUG データ・セットが割り振られていることを確認します。

関連タスク:

64 ページの『GDDM 外部デフォルトのカスタマイズ』

QMF for TSO を実行している場合、データベースからの情報を GDDM で正しく

表示するには、いくつかの外部デフォルトをカスタマイズする必要があります。

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

関連情報:

 DB2 資料

WLM 環境のセットアップに関する情報を検索します。

拡張 LIST コマンド機能のインストール (z/OS のみ)

拡張 LIST コマンド機能がインストールされている場合、表と視点に対する特権は PUBLIC の代わりに、ユーザーの 1 次または 2 次許可 ID にのみ付与する必要があります。この許可 ID は、OWNER=ALL オプションを LIST TABLES または LIST ALL コマンドに指定した場合にこのコマンドで生成されるオブジェクト・リストに含まれています。

このタスクについて

以下の表および視点は、OWNER が ALL に設定されている場合にオブジェクト・リストに表示されます。

- ユーザーの 1 次または 2 次許可 ID によって所有されている表および視点
- SELECT およびその他の特権がユーザーの 1 次または 2 次許可 ID に付与されている表および視点
- 特権が PUBLIC に付与されている表および視点

RACF グループ名は、2 次許可 ID として使用できます。

拡張 LIST コマンド機能は、DB2 for z/OS データベース使用時にのみ指定可能であり、この機能が必要な各データベース (リクエスターとサーバーの両方) にインストールする必要があります。このトピックでは、その機能に対して QMF で提供されるユーザー定義関数 (UDF) をインストールおよびアクティブ化する方法について説明します。

手順

拡張オブジェクト・リストをインストールしてアクティブにするには、以下の手順を実行します。

1. UDF 用に環境をセットアップします。このステップでは、WLM 設定済みアドレス・スペースで DB2 ストアード・プロシージャおよび UDF 用の環境のセットアップ方法と保守方法を記載してあります。通常はシステム管理者がこのステップを実行します。
2. QMF で提供される 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 を追加します。DSQABA1E は、QMF ロード・ライブラリー QMF1210.SDSQLOAD にあります。DSQABA1E メンバーを SDSQLOAD から WLM 設定済みストアード・プロシージャ・アドレス用の

STEPLIB 連結内のロード・ライブラリーにコピーします。この STEPLIB 連結は、このアドレス・スペースを開始するために使用される JCL に定義されています。

3. QMF で提供される UDF が実行される WLM 環境名を決定します。
4. QMF1210.SDSQSAPE ライブラリー内にある DSQ1BUDF ジョブを実行することにより、QMF UDF を DB2 に定義します。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。ただし、実行する前にこのジョブをカスタマイズすることが必要な場合があります。この方法については、ジョブの JCL の先頭に記載されたコメントを参照してください。
5. 登録をテストします。次のステップで QMF リスト視点を変更する前に、上記ステップが正常に実行され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 ステートメントを実行します。

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

データベースは、このステートメントを実行したユーザーに対して、有効な許可名のリストを返します。以下に例を示します。

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

6. QMF1210.SDSQSAPE ライブラリー内にあるジョブ DSQ1BUDV を実行することによって、QMF リスト視点を変更して QMF UDF を実行します。
7. オプション: QMF で表および視点のリスト生成に使用される視点をカスタマイズした場合は、次の SELECT ステートメントの例を検討してください。この例は、QMF で提供される UDF を使用するように変更されています。この例は、ユーザーのカスタマイズされた視点の変更役に役立ちます。

```
SELECT
  VARCHAR(RTRIM(T.CREATOR)),
  T.NAME, 'TABLE', T.TYPE
  , ' ', ' ', ' ',
  VARCHAR(RTRIM(SUBSTR(T.REMARKS, 1, 254))),
  ' ', ' ', ' ', VARCHAR(RTRIM(SUBSTR(T.LABEL, 1, 30)))
  , SUBSTR(T.LOCATION, 1, 16),
  VARCHAR(RTRIM(T.TBCREATOR)), T.TBNAME
FROM SYSIBM.SYSTABLES T
  ,( SELECT
      DISTINCT
      VARCHAR(RTRIM(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*')) AS U

      )
  ) AS UAT ("CREATOR", "NAME")
```

```

WHERE
(T.CREATOR=UAT.CREATOR OR T.CREATOR=CURRENT SCHEMA)
AND T.NAME=UAT.NAME
AND T.TYPE IN ('T', 'V', 'H');

```

8. 新しい機能をテストします。
 - a. 特定の QMF ユーザーの 1 次許可 ID に対して表および視点の特権を付与します。このユーザーの 2 次許可 ID に対して異なる表セットに特権を付与します。
 - b. 1 次許可 ID で QMF を始動します。
 - c. コマンド LIST TABLES (OWNER=ALL を発行します。DSQEC_LIST_OWNER 大域変数をこの値に設定することによって、ALL を LIST コマンドの OWNER オプションとして指定できます。リストは、特権を付与したすべての表を含めなければなりません。

タスクの結果

次の表は、拡張 LIST コマンド機能を要約する表です。これらのジョブを表にリストされている順序で実行し、それぞれのジョブがエラーなしに完了することを確認します。

表 29. 拡張 LIST コマンド機能をインストールするためのジョブ・シーケンス

| ジョブ名 | 説明 |
|----------|--|
| DSQ1BUDF | <p>拡張 QMF LIST コマンド機能を作成します。 command function.</p> <p>この機能が必要な各 DB2 for z/OS リクエスターまたはサーバーで、このジョブを実行します。</p> <p>データベース・モードまたはリリース・レベルをアップグレードするときには、このジョブを再実行してください。</p> |
| DSQ1BUDV | <p>拡張 LIST コマンドをサポートするために必要な QMF 視点を作成します。</p> <p>この機能が必要な各 DB2 for z/OS リクエスターまたはサーバーで、このジョブを実行します。</p> <p>データベース・モードまたはリリース・レベルをアップグレードするときには、このジョブを再実行してください。</p> |

次のタスク

デフォルトの LIST コマンドをサポートする視点をリストアするジョブ DSQ1BVW を実行することによって、いつでもデフォルトの QMF LIST コマンド機能に戻すことができます。

関連概念:

246 ページの『新しい視点をデフォルトの視点にする』

デフォルトの視点の代わりに作成した視点を使用するには、SET GLOBAL コマンドを使用して適切な大域変数を新しい視点名に設定します。

43 ページの『QMF を除く DB2 for z/OS のアップグレード時に実行する作業』
QMF がインストールされている DB2 for z/OS データベースのモードまたはリリースのいずれかをアップグレードしても、QMF はアップグレードしない場合は、2

つの QMF バインド・ジョブを実行する必要があります。

関連タスク:

245 ページの『ユーザーのオブジェクト・リストのカスタマイズ』

DB2 は許可情報を SYSIBM.SYSTABAUTH 表から収集するため、表リストおよび列情報に関して QMF から提供されるデフォルト視点を使用すると、処理時間が長くな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追加のセキュリティ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を作成することを検討してください。

関連情報:

 DB2 資料

2 次許可 ID として使用される RACF グループ名に関する情報と、DB2 ストアード・プロシージャおよび UDF 用の WLM 設定済みアドレス・スペースをセットアップする方法に関する情報を検索します。

QMF Data Service インターフェースの構成

QMF for z/OS V12 および QMF Enterprise Edition V12 には、QMF Data Service と呼ばれる新規コンポーネントが追加されています。QMF for TSO/CICS のユーザーは、QMF Data Service を使用して、DB2 データと DB2 以外のデータ (VSAM、IMS™、順次ファイル、SMF データ、SYSLOG データなど) にアクセスすることができます。アクセス可能なデータ・ソースの完全なリストについては、「DB2 QMF Data Service Solution Configuration Guide」を参照してください。

始める前に

- QMF Data Service のインストール検査手順 (IVP) をインストール、構成、および実行します。詳しくは、「DB2 QMF Data Service Customization Guide」を参照してください。
- QMF for TSO ユーザーであれば、少なくとも 1 つの DB2 for z/OS リクエストターで、QMF for TSO V12.1 のインストール検査手順をインストール、構成、および実行します。
- QMF for CICS ユーザーであれば、QMF for CICS V12.1 のインストール検査手順をインストール、構成、および実行します。

手順

QMF Data Service を対象にした QMF for TSO/CICS 接続のカスタマイズには、追加のパッケージのバインド、ロード・ライブラリーの割り振り、また CICS 使用のケースでは、既存の QMF for CICS 領域への QMF Data Service プログラム定義の追加が関係します。

1. QMF Data Service アクセスに必要なデータベース・パッケージをバインドします。このパッケージは、QMF for TSO または QMF for CICS を介して QMF Data Service サーバーに接続する元の DB2 for z/OS システムすべてでバインドする必要があります。
2. **QMF for TSO 実装のみ:** 既存の QMF for TSO 環境を更新し、QMF Data Service ライブラリー CQD.SCQDLOAD を STEPLIB または ISPLLIB のどちらかに追加します。

3. QMF for CICS 実装のみ: CICS プログラム定義を定義し、DFHRPL を更新するための以下のステップを、正常にインストールされてテストされている既存の QMF for CICS 領域に追加する必要があります。

- QMF1210.SDSQSAPE(DSQ1ECSQ) に見つかったジョブを、そのジョブの先頭にあるコメントに従ってカスタマイズします。ジョブを実行依頼して正常終了させます。
- QMF Data Service ライブラリー CQD.SCQDCLOUD を DFHRPL 連結に追加します。次に例を示します。

```
//DFHRPL DD ...
//      DD DSN=QMF1210.SDSQEXIT,DISP=SHR
//      DD DSN=QMF1210.SDSQLOAD,DISP=SHR
//      DD DSN=GDDM.SADMMOD,DISP=SHR
//      DD DSN=DSN1210.SDSNEXIT,DISP=SHR
//      DD DSN=DSN1210.SDSNLOAD,DISP=SHR
//      DD DSN=CQD.SCQDCLOUD,DISP=SHR
```

4. QMF Data Service 接続を確認します。検査を開始するには、QMF を始動してください。

注: サンプル仮想表 *staffvs* が QMF Data Service サーバーにあることを確認します。表 *staffvs* は、QMF Data Service サーバーのインストールおよび検査の間に作成されているはずですが、*staffvs* については、「DB2 QMF Data Service Customization Guide」でセクション『Verifying the server installation』を参照してください。

5. コマンド行で、次のコマンドを入力します。

```
SET GLOBAL (DSQEC_DS_SUPPORT=1
```

6. 次の SQL 照会 (*cdqs* は QMF Data Service サーバーの名前) を実行します。

```
SELECT * FROM cdqs.CQDSQL.STAFFVS
```

次のような報告書が表示されます。

| REPORT | LINE 1 | POS 1 | 159 |
|--------|--|-------|-----|
| | STAFFVS | | |
| | STAFFVS DATA STAFFVS STAFFVS STAFFVS STAFFVS | | |
| | KEY NAME DATA DATA DATA DATA | | |
| | ID L NAME DEPT JOB YRS | | |
| | ----- | | |
| | 10 7 SANDERS 20 MGR 7 | | |
| | 20 6 PERNAL 20 SALES 8 | | |
| | 30 8 MARENGHI 38 MGR 5 | | |
| | 40 7 O'BRIEN 38 SALES 6 | | |
| | 50 5 HANES 15 MGR 10 | | |
| | 60 7 QUIGLEY 38 SALES 0 | | |
| | 70 7 ROTHMAN 15 SALES 7 | | |
| | 80 5 JAMES 20 CLERK 0 | | |
| | 90 7 KOONITZ 42 SALES 6 | | |
| | 100 5 PLOTZ 42 MGR 7 | | |
| | 110 4 NGAN 15 CLERK 5 | | |
| | 120 8 NAUGHTON 38 CLERK 0 | | |
| | 130 9 YAMAGUCHI 42 CLERK 6 | | |
| | 140 5 FRAYE 51 MGR 6 | | |
| | 150 8 WILLIAMS 51 SALES 6 | | |
| | 160 8 MOLINARE 10 MGR 7 | | |
| | 170 8 KERMISCH 15 CLERK 4 | | |
| | 180 8 ABRAHAMS 38 CLERK 3 | | |
| | 190 7 SNEIDER 20 CLERK 8 | | |
| | 200 8 SCOUTTEN 42 CLERK 0 | | |

| | | | | | |
|-----|---|-----------|----|-------|----|
| 210 | 2 | LU | 10 | MGR | 10 |
| 220 | 5 | SMITH | 51 | SALES | 7 |
| 230 | 9 | LUNDQUIST | 51 | CLERK | 3 |
| 240 | 7 | DANIELS | 10 | MGR | 5 |
| 250 | 7 | WHEELER | 51 | CLERK | 6 |
| 260 | 5 | JONES | 10 | MGR | 12 |
| 270 | 3 | LEA | 66 | MGR | 9 |
| 280 | 6 | WILSON | 66 | SALES | 9 |
| 290 | 5 | QUILL | 84 | MGR | 10 |
| 300 | 5 | DAVIS | 84 | SALES | 5 |
| 310 | 6 | GRAHAM | 66 | SALES | 13 |
| 320 | 8 | GONZALES | 66 | SALES | 4 |
| 330 | 5 | BURKE | 66 | CLERK | 1 |
| 340 | 7 | EDWARDS | 84 | SALES | 7 |
| 350 | 6 | GAFNEY | 84 | CLERK | 5 |

*** END ***

| | | | | | |
|------------|-----------|--------|---------|----------|---------|
| 1=Help | 2= | 3=End | 4=Print | 5=Chart | 6=Query |
| 7=Backward | 8=Forward | 9=Form | 10=Left | 11=Right | 12= |

OK, this is the REPORT from your RUN command.
COMMAND ==>

QMF Data Service の使用について詳しくは、『QMF Data Service (QDS) の構成』を参照してください。

第 2 部 QMF for TSO/CICS の管理

| | | | |
|---|-----|---|-----|
| 第 9 章 QMF の開始 | 157 | QMF 以外のアプリケーションに使用するた めの TSO 領域からの仮想記憶域の予約 . . . | 199 |
| TSO での QMF の開始 | 157 | 仮想記憶域で不要になったデータ用の追加記憶域 の取得 | 200 |
| 必要なファイルとリソースの割り振り | 157 | 拡張仮想記憶域への報告書データの書き込み (TSO のみ) | 201 |
| QMF の CLIST および EXEC ライブラリーを TSO へ追加 | 157 | TSO でのファイルへの報告書データの書き込 み | 202 |
| TSO CALL コマンドを使用した QMF の開始 | 158 | CICS でのファイルへの報告書データの書き 込み | 206 |
| DSQQMFE モジュールを使用した QMF の直接 開始 | 158 | QMF をバッチ・モードで実行している場合 の報告書データの書き込み | 207 |
| バッチ環境での QMF の開始 | 159 | 予備ファイルの問題のトラブルシューティン グ | 207 |
| QMF の DB2 for z/OS ストアード・プロシー ジャーとしての開始 | 160 | フェッチおよび挿入操作のパフォーマンスの制御 | 209 |
| CALL ステートメントの形式 | 160 | 報告書待ち時間の制御 | 209 |
| QMF for TSO の QMF for Workstation か らの開始と線形手順の実行 | 163 | 複数行のフェッチと挿入のサポートの使用可能化 | 211 |
| QMF for TSO の QMF for Workstation か らの開始とロジックを持つプロシージャーの 実行 | 165 | 複数のデータベース・スレッドの使用 | 212 |
| ISPF 上での QMF の開始 | 167 | QMF アクティビティの自動化 | 213 |
| ISPF メニューからの QMF の開始 | 167 | 操作モードの指定 (対話式またはバッチ) | 214 |
| QMF プログラム・ライブラリーを割り振る | | QMF の開始時に実行する初期プロシージャーの 指定 | 215 |
| LIBDEF ステートメントの使用 | 168 | バッチ・モードでの初期プロシージャーの実 行 | 216 |
| ISPF でのバッチ・モードによる QMF の開始 | 171 | 初期プロシージャーを使用した対話式 QMF 作業の実行 | 216 |
| ISPF のもとでの QMF の開始の例 | 172 | 初期プロシージャーへの変数値の引き渡し | 217 |
| ネイティブ z/OS バッチでの QMF の開始 | 173 | トレース・オプションの設定 | 221 |
| CICS での QMF の開始 | 174 | TSO のトレースの設定 | 221 |
| QMF 開始時の大域変数および QMF セッション動 作の初期化 | 175 | CICS のトレースの設定 | 222 |
| DSQUOPTS ルーチンによる大域変数の設定 | 176 | 非 DBCS ディスプレイ装置からの DBCS データの 印刷 | 223 |
| DSQUOPTS ルーチンで設定できる変数 | 176 | 第 11 章 ユーザーの登録と特権の設定 | 225 |
| 大域変数表を使用した大域変数の設定 | 179 | アプリケーション・プランおよびパッケージに対す るアクセスの制御 | 225 |
| Q.GLOBAL_VARS 表の構造 | 180 | 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 | 225 |
| デフォルト・システム初期化プロシージャーによ る初期化 | 182 | アプリケーション・プランおよびパッケージに対 するアクセスの取り消し | 225 |
| 独自の初期化プロシージャーの作成 | 184 | QMF ユーザー・プロファイルの作成 | 226 |
| 第 10 章 起動時のプログラム・パラメーターおよ びプリファレンスの設定 | 187 | Q.PROFILES 表について | 226 |
| プログラム・パラメーターの要約 | 187 | サイトのためのプロファイル構造の確立 | 233 |
| データベースおよび環境のパラメーターの設定 | 195 | ユーザー・プロファイルの追加 | 234 |
| TSO で QMF を開始する DB2 (z/OS 版) サブ システムの名前の指定 | 195 | ユーザー・プロファイルの更新 | 236 |
| 初期データベース接続の指定 | 196 | SET PROFILE コマンドの使用 | 236 |
| QMF アプリケーション・プラン名の指定 | 197 | SQL UPDATE ステートメントの使用 | 237 |
| TSO で QMF プロファイル・キーとして使用す る ID の指定 | 197 | SYSTEM プロファイルの更新 | 237 |
| 報告書用記憶域の定義 | 198 | ユーザー・プロファイルの削除 | 237 |
| 固定量の報告書用仮想記憶域の定義 | 199 | 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付与 | 238 |
| 可変量の報告書用仮想記憶域の定義 | 199 | | |
| 使用可能な記憶域の比率として報告書用記憶 域を定義 | 199 | | |

| | |
|---|------------|
| QMF コマンドおよび機能に必要な特権 | 238 |
| 特権の付与と取り消し | 240 |
| SQL GRANT ステートメント | 241 |
| SQL REVOKE ステートメント | 242 |
| オブジェクトの標準の設定および非コミット読み取りの許可 | 243 |
| 他のユーザーとの QMF オブジェクトの共用 | 243 |
| 非コミット読み取りの許可 | 244 |
| ユーザーのオブジェクト・リスト | 245 |
| ユーザーのオブジェクト・リストのカスタマイズ | 245 |
| ユーザー独自のリスト視点の作成 | 245 |
| 新しい視点をデフォルトの視点にする | 246 |
| QMF LIST および DESCRIBE コマンドのデフォルトの動作 | 247 |
| コマンドに使用されるデフォルトの視点 | 247 |
| デフォルトの視点名を保管する大域変数 | 250 |
| オブジェクト・リストの記憶域要件 | 251 |
| 第 12 章 データベース内のオブジェクトの作成と保守 | 253 |
| 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 | 253 |
| 表を作成するための手順 | 253 |
| SAVE DATA および IMPORT コマンド用の表スペースの割り当て | 254 |
| プロファイルの SPACE フィールドに有効な値 | 254 |
| 表スペース・スキームを選択するときを考慮する要因 | 255 |
| ユーザーへの表の作成権限の付与 | 256 |
| 機密データをフィルターするための視点の使用 | 257 |
| 視点の作成 | 257 |
| 視点の特権の付与 | 258 |
| QMF オブジェクト・カタログの保守 | 259 |
| Q.OBJECT_DIRECTORY 表の構造 | 259 |
| Q.OBJECT_DATA 表の構造 | 263 |
| Q.OBJECT_REMARKS 表の構造 | 264 |
| QMF オブジェクト・カタログ用の表スペースの拡大 | 264 |
| QMF オブジェクトのリスト | 267 |
| QMF オブジェクトの表示 | 267 |
| QMF オブジェクトの所有権の移譲 | 268 |
| 廃止された QMF オブジェクトの削除 | 269 |
| z/OS データ・セットからの照会、書式、およびプロシージャのインポート | 269 |
| DB2 for z/OS サブシステムの保守 | 269 |
| データ・セットの管理 | 270 |
| QMF コントロール表の保守 | 271 |
| QMF コントロール表のモニターと再編成 | 271 |
| バッファー・プールの切り替え | 272 |
| 第 13 章 印刷および図表作成機能のセットアップ | 273 |
| 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 | 273 |
| GDDM サービスを使用した印刷の処理 | 274 |

| | |
|---|------------|
| QMF が GDDM ニックネームとインターフェースをとる方法 | 274 |
| GDDM がニックネームを検索する場所 | 275 |
| さまざまなプリンター・ファミリーのニックネームの例 | 275 |
| 特定プリンターのニックネーム定義の例 | 277 |
| 印刷を処理するための GDDM サービスのセットアップ | 278 |
| 正しいタイプの GDDM 装置の選択 | 278 |
| プリンター・ニックネーム指定の作成 | 278 |
| GDDM デフォルト・モジュールのニックネームでの更新 | 280 |
| ニックネームのアクティブ化 | 281 |
| 印刷を処理するための QMF サービスの使用 | 283 |
| ネイティブ z/OS バッチ、TSO、および ISPF で印刷するための QMF サービス | 283 |
| CICS の印刷での QMF サービスの使用 | 284 |
| ユーザーが QMF を終了することなく印刷できるようにする | 285 |
| オブジェクト・タイプごとの印刷の要件 | 287 |
| 図表作成機能の有効化 | 288 |
| TSO および ISPF での図表サポートの有効化 | 288 |
| CICS での図表サポートの有効化 | 289 |
| 第 14 章 コマンド同義語 | 291 |
| QMF に用意されているデフォルト同義語の使用 | 291 |
| デフォルト同義語のリスト | 291 |
| DPRE 同義語: ISPF を使用した印刷報告書のレビュー | 292 |
| DPRE の使用 | 293 |
| DPRE のカスタマイズ | 293 |
| 同義語のガイドライン | 294 |
| 同義語 verb | 294 |
| VERB 列に関する規則 | 294 |
| コマンド同義語 verb として基本 QMF verb を使用 | 294 |
| 同義語オブジェクト名 | 295 |
| 同義語定義 | 295 |
| 定義の指針 | 295 |
| 例: 線形プロシージャを同義語定義で使用 | 296 |
| 変数を同義語定義で使用 | 297 |
| 変数 &ALL | 297 |
| 情報を SYNONYM_DEFINITION 列にキー入力 | 298 |
| コマンド同義語のカスタマイズ | 299 |
| コマンド同義語表の作成 | 299 |
| コマンド同義語定義を表に入力 | 301 |
| 同義語のアクティブ化 | 301 |
| コマンド同義語表の保守の最小化 | 302 |
| 1 つの同義語表を全ユーザーに割り当てる | 303 |
| 同義語表の視点を個々のユーザーに割り当てる | 303 |
| 共用または私用で使用する同義語 | 303 |
| 共用またはグループ使用のための同義語 | 304 |
| 許可表と対になった同義語 | 304 |
| 第 15 章 カスタム・ファンクション・キー | 307 |

| | |
|--|------------|
| QMF ファンクション・キーのカスタマイズ | 307 |
| パネル ID の表示 | 307 |
| フルスクリーン・パネル ID | 307 |
| ウィンドウ・パネル ID | 308 |
| カスタマイズしたいキーの選択 | 311 |
| フルスクリーン・パネルのデフォルト・キー | 311 |
| ウィンドウ・パネルのデフォルト・キー | 312 |
| ファンクション・キー表の作成 | 313 |
| ユーザーのファンクション・キー定義を表に入力 | 314 |
| コマンドとファンクション・キーの結び付け | 314 |
| ファンクション・キーにラベルを付けて画面 | 315 |
| 上に配置 | 315 |
| 新規ファンクション・キー定義のアクティブ化 | 316 |
| ファンクション・キー表に関する問題の検査と診断 | 317 |
| キー定義の例 | 318 |
| 第 16 章 QMF 書式のためのカスタム編集出口ルーチン | 321 |
| 編集出口ルーチンおよび QMF | 321 |
| 出口ルーチンとの間で受け渡されるフィールド | 323 |
| インターフェース制御ブロックのフィールド | 323 |
| 入出力域を特徴付けるフィールド | 325 |
| 編集コードの選択 | 326 |
| 2 バイト文字セット・データおよび編集ルーチン | 327 |
| DBCS データ、および編集ルーチンが受け取るもの | 328 |
| 編集ルーチンは必ず正しい結果を戻すこと | 329 |
| 日付、時刻、タイム・スタンプ・データ、および編集ルーチン | 329 |
| 日付、時間、およびタイム・スタンプ情報で必要な形式 | 330 |
| TSO での日時データ用 DB2 出口 | 331 |
| プログラミング言語の編集ルーチン | 332 |
| 編集ルーチンをハイレベル・アセンブラで作成 | 332 |
| TSO、ISPF、およびネイティブ z/OS用のアセンブラー編集ルーチンの作成 | 333 |
| アセンブラー編集ルーチンを CICS 用に作成 | 336 |
| 編集ルーチンを PL/I で作成 | 339 |
| TSO、ISPF、またはネイティブ z/OS の CEEPIPI Language Environment インターフェースおよび PL/I 編集ルーチン | 339 |
| PL/I 編集ルーチンを CICS 用に作成 | 342 |
| 編集ルーチンを COBOL で作成 | 346 |
| TSO、ISPF、またはネイティブ z/OS の IGZERRE インターフェースおよび COBOL 編集ルーチン | 346 |
| TSO、ISPF、またはネイティブ z/OS の CEEPIPI Language Environment インターフェースおよび COBOL 編集ルーチン | 349 |
| COBOL 編集ルーチンを CICS 用に作成 | 352 |
| 第 17 章 QMF リソースの制御 | 357 |
| QMF for TSO/CICS に付属のデフォルト管理プログラム出口ルーチン | 357 |

| | |
|---|------------|
| 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 | 357 |
| 管理プログラムがリソース限界を知る方法 | 358 |
| 管理プログラムがリソース限界の到達を知る方法 | 361 |
| デフォルト管理プログラム出口によるリソース制限 | 361 |
| デフォルトのリソース限界 | 362 |
| デフォルトの限界のアクティブ化 | 363 |
| 独自のリソース限界の定義 | 364 |
| 独自のリソース・コントロール表の作成 | 366 |
| 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 | 368 |
| QMF と管理プログラムとの対話方法 | 369 |
| 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 | 370 |
| QMF が管理プログラムを呼び出す時点 | 370 |
| 管理プログラム出口ルーチンに入ると起きること | 373 |
| 機能呼び出しのアドレス可能度の確立 | 378 |
| デフォルトの管理プログラム出口ルーチンの変更または独自のルーチンの作成 | 379 |
| リソース制御情報を管理プログラム出口に渡す | 380 |
| DXEGOVA 制御ブロックの構造 | 380 |
| リソース・コントロール表のアドレッシング | 384 |
| DXEXCBA 制御ブロックの構造 | 385 |
| QMF セッション中のリソース制御情報の保管 | 394 |
| 取り消されたアクティビティに対するメッセージ | 394 |
| 管理プログラム出口ルーチンの変換、アSEMBル、リンク・エディット | 396 |
| TSO、ISPF、およびネイティブ z/OS バッチ用管理プログラム出口ルーチンのアSEMBルおよびリンク・エディット | 396 |
| CICS での管理プログラム出口ルーチンの変換、アSEMBル、リンク・エディット | 397 |
| ユーザー・アクティビティの取り消し | 399 |
| DB2 リソース限定機能の使用 | 400 |
| 管理プログラム間の相違点 | 401 |
| DB2 管理プログラムでモニター可能な QMF コマンド | 401 |
| DB2 管理プログラムにより照会が取り消された場合の QMF の対応方法 | 403 |
| QMF コマンドの DB2 管理の構成 | 403 |
| 第 18 章 バッチ・モードでの QMF の実行 | 407 |
| z/OS での QMF のバッチ・プログラムとしての実行 | 407 |
| バッチ・モードで操作する権限 | 407 |
| QMF バッチ・ジョブを実行するための JCL | 408 |
| QMF バッチのネイティブ z/OS での実行 | 411 |
| TSO での QMF バッチの実行 | 413 |
| QMF BATCH コマンドを使用した ISPF での QMF バッチの実行 | 414 |
| ISPF でバッチ・アプリケーションを使用する権限 | 414 |

| | | | |
|---------------------------------|------------|---------------------|-----|
| ISPF でのバッチ・アプリケーションの使用 | 415 | 個々の QMF モジュールのトレース | 459 |
| ISPF でのバッチ・アプリケーションの開始 | 416 | QMF トレース・データの表示 | 460 |
| バッチ・コマンド・プロンプト・パネルのフ | | QMF サービス・レベルの判別 | 461 |
| ィールド | 417 | トレース機能の停止 | 463 |
| バッチ・アプリケーションの変更 | 420 | エラー・ログ報告書の検査 | 463 |
| ISPF または TSO のフォアグラウンドでの | | 環境に固有の診断の使用 | 464 |
| QMF バッチ・ジョブの開始 | 424 | TSO における問題の診断 | 465 |
| バッチ・モードのプロシージャまたはアプリケ | | CICS における問題の診断 | 465 |
| ーションのデバッグ | 424 | IBM への問題報告 | 466 |
| リモート DB2 クライアントからの QMF バッチ・ | | 既に報告されている問題の検索 | 467 |
| ジョブの開始 | 425 | IBM ソフトウェア・サポートでの作業 | 467 |
| QMF の CICS でのバッチ・トランザクションとし | | | |
| ての実行 | 426 | | |
| CICS での端末からのバッチ・アクティビティ | | | |
| の実行 | 427 | | |
| CICS での端末を使用しないバッチ・アクティビ | | | |
| ティの実行 | 427 | | |
| CICS でのプロシージャのデバッグ | 427 | | |
| 第 19 章 問題のトラブルシューティングと診断 | 429 | | |
| QMF サービスの適用 | 429 | | |
| 一般的な問題の訂正 | 430 | | |
| 初期化時に発生する可能性があるエラー | 430 | | |
| QMF が現行のロード・モジュールを使用し | | | |
| ていないときに発生する問題 | 431 | | |
| 異常終了 | 433 | | |
| QMF メッセージ | 436 | | |
| QMF の開始に関する問題 | 440 | | |
| コマンド同義語、ファンクション・キー、お | | | |
| よびリソース・コントロール表に関する問題 | 441 | | |
| QMF の開始後の警告メッセージ | 441 | | |
| 正しくない出力 | 442 | | |
| 印刷の問題 | 442 | | |
| GDDM エラー | 442 | | |
| QMF エラー | 444 | | |
| 表示エラー | 445 | | |
| 記憶域関連問題の解決 | 445 | | |
| QMF パフォーマンスの管理 | 445 | | |
| 動的ステートメントの EXPLAIN 情報の取り込 | | | |
| み | 446 | | |
| QMF 照会を照会加速に適格にする | 446 | | |
| 記憶域の問題の解決 | 447 | | |
| リソース競合の問題の解決 | 449 | | |
| DSQEC_BUFFER_SIZE 大域変数設定による | | | |
| QMF パフォーマンスの向上 | 449 | | |
| 診断援助機能の使用 | 449 | | |
| メッセージ・サポートを使用した問題の診断 | 450 | | |
| エラー・メッセージを出した QMF 機能の判 | | | |
| 別 | 450 | | |
| システム・エラー・メッセージの処理 | 450 | | |
| SQL 戻りコードの処理 | 451 | | |
| 診断情報を取り込むための割り込みの作成 | 452 | | |
| トレース機能 | 454 | | |
| トレース・データ・セットの割り振り | 454 | | |
| トレース機能の開始 | 455 | | |
| トレース出力の適切な詳細レベルの設定 | 457 | | |

第 9 章 QMF の開始

QMF は、z/OS からのみ開始できます。QMF は、TSO、ISPF の下にバッチ・ジョブとして、あるいは CICS の下にセットアップすることができます。

TSO での QMF の開始

TSO で QMF を開始するには、以下のようないくつかの方法があります。ファイルおよびリソースを割り振り、ライブラリー CLIST と EXEC を TSO に追加することにより、TSO で開始するように QMF を設定します。

必要なファイルとリソースの割り振り

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・プロシーチ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チャーは端末モニター・プログラム (TMP) を呼び出します。

TSO セッションの最中、TMP はユーザーとディスプレイ装置間の基本インターフェースです。サイトで IBM が提供しているプロシーチャーの代わりに独自の TMP を使用している場合、このトピックで紹介する情報の一部は該当しない場合があります。

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 の中には、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。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 の要件を満たしていることを確認する必要があります。

QMF の CLIST および EXEC ライブラリーを TSO へ追加

サイトで作成された DD ステートメントを、TSO の EXEC の検索順序のために使用します。

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。これらは、TSO によって使用されるデータ・セットです。REXX EXEC について、サイトでの検索順序を知らない場合、SYSEXEC と SYSPROC の両方に対して SDSQEXCE を割り振ります。

```
//*****  
//*          DATA SETS USED BY TSO          *  
//SYSPROC DD DSN=SYS2.CLIST,DISP=SHR        * CLIST Library  
//          DD DSN=QMF1210.SDSQCLTE,DISP=SHR  
//SYSEXEC DD DSN=SYS2.EXEC,DISP=SHR  
//          DD DSN=QMF1210.SDSQEXCE,DISP=SHR
```

図 16. 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 を開始するには、QMF ロード・ライブラリーの名前を指定し、データ・セット名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。

この例では、TSO CALL コマンドを使用して QMF を開始します。

```
CALL 'QMF1210.SDSQLOAD(DSQMFE)' 'DSQSMODE=I,DSQSSUBS=DB2T'
```

QMF ロード・ライブラリーは、CALL コマンドの実行中はタスク・ライブラリーとして割り振られます。ただし、これらの製品にプログラム・インターフェースをロードさせるには、QMF に、DB2 および GDDM のライブラリーへのアクセス権限を付与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。

重要: TSO CALL コマンドは、ISPF の下で実行中の CLIST からは使用しないでください。ISPF のもとで QMF を開始する場合は、ISPF SELECT PGM メソッドを使用して QMF を開始する必要があります。

関連概念:

167 ページの『ISPF 上での QMF の開始』
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。

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り、TSO の下で QMF を開始することができます。

例えば、次のコマンドは、開始方法と、ISPF から独立して作動している QMF にパラメーターを渡す方法を示しています。

```
DSQQMFE DSQSBSTG=50000,DSQSMODE=B
```

この例のパラメーターは、次のとおりです。

- DSQSBSTG パラメーター (報告書の最大記憶域) には値 50,000 を渡します
- DSQSMODE (操作モード) には値 B (バッチ) を指定します

CLIST または EXEC から開始し、初期プロシーチャーを指定するには、次のようなコマンドを発行します。

```
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以下のことを行います。

- 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
```

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

- 0 実行は正常終了しました。
- 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 サーバー・エラーが発生しました。 QMF は開始されませんでした。メッセージが表示されていない場合は、メッセージのトレース出力を確認して、問題の診断に役立ててください。

関連概念:

460 ページの『QMF トレース・データの表示』

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る必要があります。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

バッチ環境での QMF の開始

ISPF サービスを使用せずに QMF を開始するには、バッチ・ジョブの JCL の SYSTSIN データ・セットにステートメントを設定します。

例えば、次のコマンドで、DSQSMODE=B は適切な操作モードを確立し、DSQSRUN パラメーターは実行するプロシージャーの所有者と名前を指定します。

```
DSQQMFE DSQSMODE=B,DSQSRUN=auth_id.proc_name
```

ステートメントには、必須の DSQSMODE および DSQSRUN パラメーターの他に、他のパラメーター値を含めることもできます。

関連概念:

407 ページの『第 18 章 バッチ・モードでの 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。したがって、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。

187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターについて説明します。

QMF の DB2 for z/OS ストアード・プロシージャーとしての開始

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

インターフェースのコンポーネントをインストールし、正常に動作することを検証する手順については、ストアード・プロシージャー・インターフェースのインストール手順に説明があります。QMF がストアード・プロシージャーとして動作しているときは、QMF バッチ・モードと同様に、パネルを表示することはできません。したがって、QMF の開始後に実行される QMF プロシージャーはどれもが LIST や DISPLAY などのコマンドは発行できません。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

425 ページの『リモート DB2 クライアントからの QMF バッチ・ジョブの開始』
QMF ストアード・プロシージャー DSQQMFSP がインストールされた DB2 データベースに接続されているすべての DB2 クライアントで、TSO バッチ・ジョブ用に QMF を開始することができます。バッチ・ジョブは、QMF ストアード・プロシージャーを実行する z/OS システムから開始されます。

CALL ステートメントの形式

このトピックでは、必要な CALL ステートメントの形式について説明します。

QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する CALL ステートメントの形式は以下のとおりです。

```
▶—CALL—Q.DSQQMFSP—▶  
▶—( 'object-name'—'trace-level'—'L2-destination'—'language'—status-message— )▶
```

object-name

QMF が開始した後に、実行する QMF プロシージャーまたは照会の名前を指定する入力パラメーター。受け渡す名前と置換変数値を含むパラメーター値の最大のサイズは 32,704 バイトです。

すべてのタイプの QMF 照会が受け入れられます。プロシージャーは、QMF 線形プロシージャーまたはロジックを持つプロシージャーのいずれかです。

このパラメーターで指定された照会またはプロシージャーは、ストアード・プロシージャー・インターフェース・コンポーネントがインストールされているサブシステム内の QMF カタログ内に存在している必要があります。

指定したオブジェクトが照会の場合は、1 つの結果セットが返されます。最大 21 セットまでの結果セットがプロシージャーから返されます (*trace-level* が L2 で、*L2-destination* がブランクまたは NULL のときに、最後の結果セットとして返されるトレース出力を含む)。QMF は、プロシージャー内に次のコマンドが出現するたびに呼び出し側プログラムに結果セットを返します。

```
PRINT REPORT (PRINTER=' ')
```

DSQEC_CC 大域変数を 0 に設定して、返された結果セットから紙送り制御文字を除去します。

PRINTER オプションは、コマンド自体に設定するか、またはプロシージャー内で次のコマンドを発行して QMF プロファイルの PRINTER オプションを設定することができます。

```
SET PROFILE (PRINTER=' ')
```

QMF は、ストアード・プロシージャーとして始動された場合はパネルを表示できないため、変数値に対するプロンプトはありません。このオブジェクトが照会の場合は、照会内のすべての変数値は CALL ステートメント上で渡される必要があります。このオブジェクトがプロシージャーの場合は、プロシージャーには SET GLOBAL コマンドを指定して、必要な値が要求される前にその値を設定できます。また、値は CALL ステートメント上で渡すことができます。

このパラメーターは、*trace-level* が PTF に設定されているときには無視されません。

trace-level

トレース詳細のレベルを指定する入力パラメーター。有効な値は、次のとおりです。

ブランク

QMF トレース出力が生成されません。

NULL

QMF トレース出力が生成されません。

NONE

QMF トレース出力が生成されません。

L2

このオプションは、QMF メッセージおよびコマンドを最高の詳細レベルでトレースします。トレース出力の送信先は、*L2-destination* がどのように設定されているかによって異なります。

ALL

このオプションは、最も詳細なレベルまで QMF アクティビティーをトレースします。このトレースには、プログラム初期化エラー、およびユーザー・プロファイルの確立前に発生する可能性があるその他のエラーも含まれます。トレース出力は、QMF ストアード・プロシージャー・インターフェースのインストール時に定義した DSQDEBUG DD カードに送信されます。

PTF このオプションを使用して、ストアード・プロシージャ・インターフェースが正常に実行していることを確認します。本書または IBM ソフトウェア・サポートによる指示がない限り、このオプションは使用しないでください。

object-name パラメーターにより指定された初期プロシージャ内で SET PROFILE (TRACE コマンドを指定して、ストアード・プロシージャ・セッションの期間にトレース詳細のレベルを変更できます。これが行われる条件として、トレース結果が DSQDEBUG データ・セットに出力されるように設定されている必要があります。*trace-level* が L2 に設定されている場合 SET PROFILE (TRACE コマンドが受け入れられるには、*L2-destination* は、DSQDEBUG に設定されている必要があります。

L2-destination

trace-level が L2 に設定されているときのトレース・ログの宛先を指定する入力パラメーター。有効な値は、次のとおりです。

DSQDEBUG

QMF ストアード・プロシージャ・インターフェースのインストール時に定義した DSQDEBUG DD カードにトレース出力を送信します。このオプションを使用するのは、*object-name* パラメーターで渡されたプロシージャが TRACE オプション指定の SET PROFILE コマンドを含んでいると考えられる場合です。

ブランク

トレース出力をストアード・プロシージャの実行からの最後の結果セットとして返します。

NULL

トレース出力をストアード・プロシージャの実行からの最後の結果セットとして返します。

DSQSCMD n EXEC に現れる DSQSDBG パラメーターは、その値をこの入力パラメーターから取得します。このパラメーターは、ストアード・プロシージャ・インターフェースのコンテキストの外側に外部的に設定することはできません。

language

QMF が実行する言語を指定する入力パラメーター。

1 文字の各国語 ID を指定します。この入力パラメーターのブランクまたは NULL 値は、QMF を英語で呼び出します。

status-message

これは出力パラメーターであり、そこにはプロシージャまたは照会 (*object-name* パラメーターで渡す) の実行によって発行された最後のメッセージが含まれます。出力パラメーターを定義する方法は、CALL ステートメントを発行するソフトウェア・プログラムによって異なります。例えば、QMF for Workstation では、状況メッセージを含んでいる出力パラメーターは、疑問符 (?) 文字として定義されます。

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

457 ページの『トレース出力の適切な詳細レベルの設定』

すべての QMF 機能を詳細にトレースすることができ、また個々の QMF 機能をトレースすることができます。

459 ページの『個々の QMF モジュールのトレース』

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。

460 ページの『QMF トレース・データの表示』

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る必要があります。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

461 ページの『QMF サービス・レベルの判別』

ターゲット・ゾーンおよび配布ゾーンに対する SMP/E 報告書の実行は常に、サービス・レベルを判別するための最適な方法です。ただし、QMF サービス・レベルを他の方法で判別することもできます。

関連資料:

463 ページの『エラー・ログ報告書の検査』

Q.ERROR_LOG 表は、リソースおよびその他の問題に関する情報をログに記録する QMF コントロール表です。

QMF for TSO の QMF for Workstation からの開始と線形手順の実行

以下の例は、QMF for TSO を QMF for Workstation から開始し、従業員名、部門、および職種について特定の基準を満たす行に対して Q.STAFF サンプル表を照会する単純な線形手順を実行する方法を示しています。

このタスクについて

この照会の名前は STAFFQUERY です。

```

SELECT * FROM Q.STAFF
WHERE
NAME = &NAME AND
DEPT = &DEPT AND
JOB = &JOB

```

この照会を実行する QMF プロシージャー例の名前は STAFFPROC です。

```

SET GLOBAL (DSQEC_CC=0 --Turn off carriage control for printing
RUN QUERY STAFFQUERY (&&NAME = &NAME &&DEPT = &DEPT &&JOB = &JOB
PRINT REPORT (PR= ' '

```

呼び出し側のプログラムに返される各結果セットにつき 1 つの PRINT REPORT コマンドをコーディングします。結果セットで報告書出力を受け取るには、PRINT REPORT コマンドの PRINTER オプションにブランクのストリングを設定する必要があります。プロシージャー内に複数の PRINT REPORT コマンドがある場合でも、最初の PRINT REPORT コマンドのコードの前に SET PROFILE コマンドを発行すれば PRINTER オプションを設定することができるので、PRINTER オプションを何回も指定する必要はありません。

プロシージャーには結果セットを返す PRINT REPORT コマンドを最大で 20 個含めることができます。結果セットから紙送り制御文字を除去するには、上の例に示すように、DSQEC_CC 大域変数を 0 に設定してください。

手順

QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始し、STAFFPROC プロシージャー例を実行するには、QMF for Workstation から以下のいずれかのようなコマンドを発行します。

- 次のように、変数値の区切り文字として括弧を使用することができます。

```
CALL Q.DSQMFSP('STAFFPROC(&NAME=('PERNAL'),&DEPT=(20),&JOB=('SALES'))','L2',' ','E',?)
```

この CALL ステートメントは、

- QMF for Workstation に対して、Q.STAFF サンプル表から以下の行を含む結果セットを返します。

| ID | NAME | DEPT | JOB | YEARS | SALARY | COMM |
|----|--------|------|-------|-------|----------|--------|
| 20 | PERNAL | 20 | SALES | 8 | 18171.25 | 612.45 |

- メッセージとコマンドをトレースする *trace-level* パラメーターに値 L2 を指定します。
- *L2-destination* パラメーターはブランクのままにします。これは、QMF に対して、最後の結果セットとしてトレース出力を返すことを指示します。したがって、STAFFPROC プロシージャーが終了したとき、合計で 2 つの結果セットが返されます。
- *language* パラメーターに値 E を指定することにより、QMF が動作する言語として英語を指定します。
- 出力パラメーターを定義する方法は、CALL ステートメントを発行するソフトウェア・プログラムによって異なります。例えば、QMF for Workstation では、出力パラメーターは疑問符 (?) 文字として定義されます。
- また、次のように、区切り文字として括弧を使用せずに変数値を渡すこともできます。

```
CALL Q.DSQMFSP('STAFFPROC(&NAME='perna1',&DEPT=20,&JOB='SALES'),'L2','','E',?)
```

- 次の例では、CALL ステートメントはアポストロフィを含む変数値を渡します。

```
CALL Q.DSQMFSP('STAFFPROC(&NAME='O''BRIEN',&DEPT=38,&JOB='SALES'),'L2','','E',?)
```

このステートメントは、QMF for Workstation に最初の結果セットとして、次の行を返します。

| ID | NAME | DEPT | JOB | YEARS | SALARY | COMM |
|----|---------|------|-------|-------|----------|--------|
| 40 | O'BRIEN | 38 | SALES | 6 | 18006.00 | 846.55 |

最後の結果セットには、実行されたストアード・プロシージャのトレース出力が含まれます。

QMF for TSO の QMF for Workstation からの開始とロジックを持つプロシージャの実行

この例では、QMF for Workstation から、QMF for TSO をストアード・プロシージャとして開始します。続いて、デフォルト値を設定するロジックを持つプロシージャを実行し、大域変数値を検索して、照会を実行します。

このタスクについて

ここに示す、ロジックを持つプロシージャ STAFFPROC は、以下の作業を行います。

- すべての QMF SAVE コマンドで、SHARE=YES パラメーターをデフォルトにします。
- DSQEC_CC 大域変数に 0 を設定して、呼び出し側のプログラムに返される結果セットから紙送り制御文字を除去します。
- QMF for TSO が DB2 for z/OS ストアード・プロシージャとして開始されたかどうかを示す DSQAO_STO_PROC_INT 大域変数の値を検索します。
- トレース・ログに、QMF for TSO が DB2 for z/OS ストアード・プロシージャとして開始されたかどうかを示すメッセージ・テキストを書き込みます。
- 次のような STAFFQUERY という名前の照会を実行します。

```
SELECT * FROM Q.STAFF
WHERE
NAME = &NAME AND
DEPT = &DEPT AND
JOB = &JOB
```

- 照会結果を結果セットとして呼び出し側のプログラムに返します。次のように、返す必要のある各結果セットにつき 1 つの PRINT REPORT コマンドをコーディングします。

```
PRINT REPORT (PRINTER=' ')
```

プロシージャ内に複数の PRINT REPORT コマンドがある場合は、次の例に示すように、SET PROFILE コマンドを発行してプロファイルの PRINTER オプションに空白のストリングを設定できるため、このオプションを各 PRINT REPORT コマンドで指定する必要がなくなります。プロシージャには結果セットを返す PRINT REPORT コマンドを最大で 20 個含めることができます。

```

/* REXX */
"SET GLOBAL (DSQEC_SHARE = 1"
"SET GLOBAL (DSQEC_CC=0"
"GET GLOBAL ("SPINT"=DSQAO_STO_PROC_INT)"
IF SPINT = 1 THEN
DO
"MESSAGE (TEXT 'YOU ARE RUNNING IN THE QMF STORED PROC INT.'"
"SET PROFILE (PR = ' ')"
END
ELSE
DO
"MESSAGE (TEXT 'YOU ARE NOT RUNNING IN THE QMF STORED PROC INT.'"
END

"RUN QUERY STAFFQUERY (&NAME = &NAME &DEPT = &DEPT &&JOB = &JOB"
"PRINT REPORT"

```

図 17. ロジックを持つプロシージャー STAFFPROC

手順

QMF for Workstation から QMF for TSO を開始して、ロジックを持つプロシージャーを実行するには、以下の例に従ってください。

1. QMF for Workstation から次のようなコマンドを発行します。このコマンドは、DB2 for z/OS ストアード・プロシージャーとして QMF for TSO を開始し、ロジックを持つプロシージャー例を実行します。

```
CALL Q.DSQMFSP('STAFFPROC(&NAME='0''BRIEN',&DEPT=38,&JOB='SALES'),'L2','','E',?)
```

この CALL ステートメントは、

- 変数値の区切り文字として二重引用符を使用します。
- QMF for Workstation に対して、Q.STAFF サンプル表から以下の行を含む結果セットを返します。

| ID | NAME | DEPT | JOB | YEARS | SALARY | COMM |
|----|---------|------|-------|-------|----------|--------|
| 40 | O'BRIEN | 38 | SALES | 6 | 18006.00 | 846.55 |

- メッセージとコマンドをトレースする **trace-level** パラメーターに値 L2 を指定します。
- **L2-destination** パラメーターはブランクのままにします。これは、QMF に対して、最後の結果セットとしてトレース出力を返すことを指示します。したがって、STAFFPROC プロシージャーが終了したとき、合計で 2 つの結果セットが返されます。
- **language** パラメーターに値 E を指定することにより、QMF が動作する言語として英語を指定します。

出力パラメーターの定義方法は、CALL ステートメントを発行するソフトウェア・プログラムによって異なります。例えば、QMF for Workstation では、出力パラメーターは疑問符 (?) 文字として定義されます。

2. QMF の開始時に照会のテキスト全体を渡すには、変数のみを含む照会を作成します。以下に例を示します。

```
&QUERY
```

3. 次の例に示すように、照会を実行する RUNQPROC という名前のプロシージャーを作成します。線形手順でも同じ結果を得られます。

```
/* REXX */
"SET PROFILE (PR = ' ')"
"RUN QUERY RUNQ(&&QUERY=&QUERY)"
"PRINT REPORT"
```

以下の例では、CALL ステートメントは、QMF for Workstation を開始し、照会テキスト全体を最初のパラメーターの一部として渡して RUNQPROC プロシージャを実行します。

```
CALL Q.DSQMFSP('RUNQPROC(&QUERY=((SELECT CURRENT SERVER FROM SYSIBM.SYSDUMMY1)))','L2','','E',?)
```

ISPF 上での QMF の開始

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。この方法として、以下のことを実行できます。

- 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、それ以外の方法のいずれかを使用した QMF を開始することもできます。例えば、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。

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 CALL ステートメントまたは TSO コマンドを使用すると、結果は予測不能になる可能性があります。

制約事項:

- 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。例えば、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)
```

- 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 を開始するメニュー・オプションを構成することができます。

ISPF マスター・アプリケーション・メニューの次のサンプル定義は、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 DB2T 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(ISP0PRIM) NEWAPPL'
              2, 'PGM(DSQMFE) NEWAPPL(DSQE) PASSLIB PARM(DSQSSUBS=DB2T)'
              /*                                */
              /* ADD OTHER APPLICATIONS HERE */
              /*                                */
              P, 'PANEL(ISPOPT)'
              X, 'EXIT'
              ' ' ' ' ' '
              ' ' ' ' ' '
              *, '?' )
)END

```

図 18. サンプルのマスター・アプリケーション・メニュー

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。例えば、DB2T がテスト DB2 サブシステム、DB2P が実動サブシステムであるとしします。この場合、メニューには 2 つ (各サブシステムごとに 1 つずつ) のオプションを追加できます。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 を作成する場合があります。メニューの PROC セクションの行は、次のようになります。

```

2, 'PGM(DSQMFE) NEWAPPL(DSQE) PASSLIB PARM(DSQSSUBS=DB2T)'
3, 'PGM(DSQMFE) NEWAPPL(DSQE) PASSLIB PARM(DSQSSUBS=DB2P)'

```

QMF プログラム・ライブラリーを割り振る LIBDEF ステートメントの使用

ISPF セッション中に、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場合は、プログラム・ライブラリーを DSQLLIB という固有の QMF DD 名に割り振ります。

ISPF LIBDEF ステートメントの LIBRARY オプションの ID 値として DD 名 DSQLLIB を指定します。

例えば、QMF および DB2 製品ライブラリーを割り振るには、TSO ALLOCATE および ISPF LIBDEF ステートメントを作成します。


```
ALLOC FI(DSQLLIB) DA('QMF1210.SDSQEXIT','QMF1210.SDSQLOAD', +  
'DSN1110.SDSNEXIT','DSN1110.SDSNLOAD') SHR REUSE  
LIBDEF ISPLLIB LIBRARY ID(DSQLLIB)
```

GDDM 外部デフォルト・モジュール ADMADFT は、LIBDEF ステートメントで DSQLLIB 用に定義された検索順序では見つかりません。GDDM モジュール用のサンプル割り振りステートメントについては、TSO ログオン・プロシーチャーの準備に関する情報内に説明があります。

ISPF LIBDEF サービスを使用してプログラム・ライブラリーを割り振るには、次に示すような CLIST を作成します。以下の CLIST では、ISPF が既に稼働中であり、他の ISPF リソースは既に割り振り済みであると仮定されています。

```

/*****/
/* Allocate QMF and DB2 programs to DSQLLIB */
/*****/
ALLOC FI(DSQLLIB) SHR REUSE +
      DA('QMF1210.SDSQEXIT', +
         'QMF1210.SDSLOAD', +
         'DSN1110.SDSNEXIT', +
         'DSN1110.SDSNLOAD')
/*****/
/* Allocate QMF libraries used for GDDM */
/*****/
ALLOC FI(ADMGGMAP) DA('QMF1210.SDSQMAPE') SHR REUSE
ALLOC FI(ADMCFORM) DA('QMF1210.SDSQCHR') SHR REUSE
ALLOC FI(DSQCFRM) DA('QMF1210.DSQCFRM') SHR REUSE
ALLOC FI(ADMGDF) DA('QMF1210.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('QMF1210.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(10,20) 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('QMF1210.SDSQPLBE')
ISPFEXE LIBDEF ISPLLIB DATASET ID('QMF1210.SDSQSLBE')
ISPFEXE LIBDEF ISPLLIB DATASET ID('QMF1210.SDSQMLBE')
ISPEXEC LIBDEF ISPTLIB DATASET ID('QMF1210.SDSQTLBE')
/*****/
/* 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 ISPLLIB LIBRARY ID( )
ISPEXEC LIBDEF ISPTLIB 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(DSQCFRM)
FREE FI(ADMGDF)

```

図 19. ISPF LIBDEF CLIST

上記の CLIST は、予備データのファイル割り振りを示しています (DSQSPILL)。予備データにファイルを割り振らない場合、データは拡張記憶域に予備書き込みすることができます。

関連タスク:

66 ページの『TSO ログオン・プロシージャの準備』

ユーザーの記憶域やその他の要件をサポートするために TSO ログオン・プロシージャを変更する必要があります。この手順では、ロード・ライブラリーおよびデータ・セットの割り振りを行います。

201 ページの『拡張仮想記憶域への報告書データの書き込み (TSO のみ)』

QMF for TSO では、QMF が実行されているシステムで利用可能な拡張記憶域が非常に限られている場合を除き、予備データ用の拡張記憶域が使用されます。

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。

手順

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(DSQMFE) NEWAPPL(DSQE) PARM(...DSQSMODE=B,DSQSRUN=auth_ID.proc_name)
```

PARM キーワードの後に続くパラメーターは、バッチ・モードを確立し (DSQSMODE=B)、実行するプロシージャを指定し (DSQSRUN=auth_ID.proc_name)、そのプロシージャの変数を含むことができます。

PARM キーワードの後の省略符号は、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現しています。

DSQSRUN パラメーターに指定する値は、所有者の許可 ID を含む必要があります。例えば、以下のステートメントでは、プロシージャ PROCA の所有者がユーザー JONES であることを示しています。

```
ISPSTART PGM(DSQMFE) NEWAPPL(DSQE) PARM(DSQSMODE=B,DSQSRUN=JONES.PROCA)
```

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。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 データ・セット内のすべてのコマンドが実行済みになると、ジョブ・ステップは終了します。

- 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。

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。

以下の状況では、ISPF 下で QMF を開始することが必要になることがあります。

- ISPF を CLIST から開始し、QMF を初期ダイアログとして指定する場合:

```
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSIROW=150)
```

このステートメントは、DSQSIROW (報告書の最初の画面が表示される前に取り出される行数の制御) に値 150 を渡します。

- ISPF 内で作動する CLIST から開始する場合:

```
ISPEXEC SELECT PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSSUBS=SFDX)
```

このステートメントは、ローカル DB2(z/OS 版) サブシステムに名前 SFDX 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)'  
                :  
                :  
                :)
```

このコードは、DSQSPILL パラメーターに値 NO を渡し、報告書データでは予備記憶域を使用しないことを示します。

- CLIST から開始し、初期プロシージャーを指定する場合:

```
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以下のことを行います。

- 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)
```

初期プロシージャーでは、変数の前に正しい数のアンパーサンドを付加する必要があります。

関連概念:

217 ページの『初期プロシージャーへの変数値の引き渡し』

DSQSRUN パラメーターで初期プロシージャーの名前を指定する際、そのプロシージャーに入れる 1 つ以上の変数の値も指定できます。

ネイティブ z/OS バッチでの QMF の開始

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ことができます。

パネルの保管場所、パネルのファイル名、他の表や QMF オブジェクトの名前と場所、および予備ファイルの場所 (QMF がデータを拡張記憶域ではなくファイルに予備書き込みするように構成されている場合) を定義する JCL を作成します。

ネイティブ z/OS バッチで QMF コマンドを発行するには、以下の QMF プログラム・パラメーターを使用します。

- DSQSMODE

バッチ・モードの場合は、値 B を使用します。(I は対話型操作を指定します。) このパラメーターの短縮名は M です。このサンプル・コマンドでは、呼び出しコマンドに M=B を指定することにより、モードをバッチに設定します。

- DSQSRUN

このパラメーターを使用して、QMF の開始時に実行される初期プロシージャの名前を指定します。このパラメーターの短縮名は I です。このサンプル・コマンドでは、I パラメーターにプロシージャ X を指定します。

以下の例では、プロシージャ X が終了すると、QMF が終了します。ジョブの戻りコードを調べて、ジョブが正常に実行されたか確認することができます。

```
//RUNQMF EXEC PGM=DSQQMFE,PARM='M=B,I=X,P=QMF12,S=DSN'  
//*****  
//* Program load libraries  
//*****  
//STEPLIB DD DSN=QMF1210.SDSQLOAD,DISP=SHR  
// DD DSN=DSN1110.SDSNEXIT,DISP=SHR  
// DD DSN=DSN1110.SDSNLOAD,DISP=SHR  
// DD DSN=GDDM.SADMMOD,DISP=SHR  
//*****  
//* QMF/GDDM maps  
//*****  
//ADMGMAP DD DSN=QMF1210.SDSQMAPE,DISP=SHR  
//*****  
//* Data sets used by QMF *  
//*****  
//DSQPRINT DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330)  
//DSQDEBUG DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=122,BLKSIZE=1210)  
//DSQDUMP DD SYSOUT=*,DCB=(RECFM=VBA,LRECL=125,BLKSIZE=1632)  
//DSQSPILL DD DSN=&&SPILL,DISP=(NEW,DELETE),  
// UNIT=SYSDA,SPACE=(TRK,(100),RLSE),  
// DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)
```

図 20. ネイティブ z/OS のバッチで QMF プロシージャを実行するための 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。

関連概念:

187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターについて説明します。

CICS での QMF の開始

QMF を CICS 用にカスタマイズすると、CICS ディスプレイから QMF を開始することができます。

CICS で QMF を開始するには、トランザクション名 QMF n を使用します。 n は 1 文字の各国語機能 ID で、QMF が実行される言語を示します。あるいは、CICS START コマンドを使用するアプリケーションを作成することもできます。次のコマンドは、QMF for CICS でのデフォルト (英語) のトランザクションを示しています。

QMF parameters

QMF はトランザクション ID、*parameters* は必要なプログラム・パラメーターを示します。

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、ID で指定される装置が使用可能になっている必要があります。また、その ID が、START コマンドが発行されるシステムでローカルリモートのいずれとして定義されているか確認する必要があります。

以下の例では、CICS の下で QMF を開始させる方法を示します。

- CICS の初期化画面からの開始:

```
QMFE DSQSIROW=150,DSQSBSTG=500000,DSQSPILL=NO
```

このステートメントは、DSQSIROW (報告書の最初の画面が表示される前に取り出される行数) に値 150 を渡します。また、このステートメントは、DSQSBSTG (報告書用の最大記憶域) に値 500,000 バイトを渡し、QMF 予備ファイルを使用しないことを指定します (DSQSPILL=NO)。

- CICS 初期化画面からの開始と初期プロシージャの指定:

```
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以下のことを行います。

- 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
```

初期プロシージャーでは、変数の前に正しい数のアンパーサンドを付加する必要があります。

関連概念:

217 ページの『初期プロシージャーへの変数値の引き渡し』

DSQSRUN パラメーターで初期プロシージャーの名前を指定する際、そのプロシージャーに入れる 1 つ以上の変数の値も指定できます。

187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターについて説明します。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

QMF 開始時の大域変数および QMF セッション動作の初期化

いくつかの方法を使用して、QMF の開始時に QMF 大域変数を設定できます。この方法には、DSQUOPTS ルーチン、Q.GLOBAL_VARS 大域変数表、またはデフォルト・システム初期化プロシージャーの変更があります。このプロシージャーは、ホーム・パネルが表示される前にユーザーの QMF セッションの他の側面を設定するためにも使用できます。

QMF は開始時に、以下の大域変数設定のソースを、QMF で定義されているデフォルト値から順に検査します。大域変数が複数の場所で定義されている場合、QMF は検出した最後の値を使用します。

1. QMF で定義されているデフォルト大域変数値。
2. DSQUOPTS 出口ルーチンで定義されている大域変数値。
3. SYSTEM の USERID を使用して Q.GLOBAL_VARS 大域変数表で定義されている大域変数値。これらは、管理者が設定した大域変数値です。
4. ユーザーの USERID を使用して Q.GLOBAL_VARS 大域変数表で定義されている大域変数値。これらは、前のセッションで保存された大域変数値です。
5. デフォルト・システム初期化プロシージャー (Q.SYSTEM_INI) で定義されている大域変数値。

制約事項: 大域変数が、UPDATEABLE 列に値 0 を持つ Q.GLOBAL_VARS 表で定義されている場合、その大域変数は Q.SYSTEM_INI プロシージャーでは設定できません。また、Q.SYSTEM_INI プロシージャーでは、QMF で読み取り専用として定義されている大域変数 (DSQAO で始まる大域変数など) を更新することはできません。

DSQUOPTS ルーチンによる大域変数の設定

DSQUOPTS 出口ルーチンでの設定によって、QMF 開始時に特定の`大域変数の値`がオーバーライドされます。デフォルト DSQUOPTS ルーチンはニーズに合わせて変更できます。

このタスクについて

デフォルト・バージョンの DSQUOPTS は、QMF1210.SDSQLOAD データ・セットに付属しています。

手順

DSQUOPTS 出口ルーチンを変更するには、以下の手順に従ってください。

1. 必要に応じて、アセンブラー・ソースを変更します。出口のソースは QMF1210.SDSQSRE(DSQUOPTS) に含まれています。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。
2. モジュールをアセンブルし、リンク・エディットします。DSQUOPTS をアセンブルし、リンク・エディットするサンプル・ジョブは、メンバー QMF1210.SDSQSAPE(DSQUOPT) に含まれています。
3. QMF 出口ライブラリーを、使用する初期化プロシージャに割り振って、変更された出口が QMF の初期化プロセス中に参照されるようにします。

タスクの結果

サンプル・ジョブを使用して DSQUOPTS 出口ルーチンをアセンブルし、リンク・エディットすると、変更された DSQUOPTS ロード・モジュールがデフォルトの QMF の出口ライブラリー QMF1210.SDSQEXIT に保管されます。

DSQUOPTS ルーチンで設定できる変数

DSQUOPTS 出口ルーチンは、いくつかの変数に関する設定を提供します。DSQUOPTS 出口を使用して、初期化時に他の変数値を設定することもできます。

デフォルト版の DSQUOPTS 出口ルーチンは、以下の変数値を設定します。

DISADM

QMF が、ユーザー ID が QMF 管理者権限を持っているかどうかを確認するかしないかを指定します。QMF 大域変数 DSQEC_DISABLEADM を使用して、この値を設定することもできます。

有効な値は以下のとおりです。

NO QMF 管理者権限の確認を有効にします。この値がデフォルトです。

DISADM 変数が NO に設定されている場合、DSQEC_DISABLEADM 大域変数の値は 0 です。

YES

QMF 管理者権限の確認を無効にします。

DISADM 変数が YES に設定されている場合、DSQEC_DISABLEADM 大域変数の値は 1 です。

QMF は、QMF の初期化時に、QMF を開始するユーザーの許可 ID が Q.PROFILES 表に対して INSERT か DELETE のいずれかの特権を持っているかどうかを確認します。許可 ID がこれらの特権のいずれかを持つ場合、QMF はユーザーが QMF 管理者であると見なし、この権限を反映するために大域変数 DSQAO_QMFADM に 1 を設定します。QMF 管理者は、所有していない QMF オブジェクトをすべての QMF ユーザーと共有することなく、これらのオブジェクトに対して以下の操作を実行できます。

- SAVE
- ERASE
- IMPORT
- EXPORT
- DISPLAY

QMF 管理者は、他のユーザーが所有する ANALYTIC オブジェクトに対して DISPLAY コマンドを、また ANALYTIC オブジェクトおよび FOLDER オブジェクトに対して ERASE コマンドを発行することもできます。

初期化時の確認中に QMF を開始したユーザーが QMF 管理者権限を持っていなかったことがわかると、INSERT 特権、DELETE 特権、あるいはその双方に対して、-551 SQL コードが発行されます。この SQL コードは、次の環境下で予期されます。メッセージがユーザーに示されたり、QMF トレース・データ・セット (DSQDEBUG) またはエラー・ログ表 (Q.ERROR_LOG) で収集されることはありません。これは、エラーではないからです。メッセージによって、QMF の初期化が停止されたり、QMF の実行中に問題が発生することもあります。ただし、何らかの理由でこの SQL コードを抑止するには、管理者権限の確認を無効にする必要があります。

QMF 管理者権限の確認を無効にすると、他のユーザーのオブジェクトを自由に操作する機能が制限されることがあります。QMF 管理者権限がない場合、操作するすべてのオブジェクトがすべてのユーザーで共有されている必要があります。オブジェクトを共有するには、すべてのオブジェクトを SHARE=YES パラメーターを使用して保存するようユーザーに要求するか、DSQEC_SHARE 変数を設定して SHARE パラメーターのデフォルト値を YES に変更することができます。この値を指定すると、すべてのオブジェクトがすべてのユーザーで共有されるようになります。そのためユーザーは、共有を望まないオブジェクトを保存する場合は、SHARE=NO を明示的に指定する必要があります。DSQEC_SHARE 変数を設定すれば、ユーザー支援を行うときに、ユーザー ID を切り替えたり、オブジェクトを共有するためにかかる時間を節約することができます。

SHARE

QMF Analytics for TSO で SAVE コマンドの SHARE パラメーターのデフォルト値および「保管」ファンクション・キーを設定します。QMF 大域変数 DSQEC_SHARE を使用して、この値を設定することもできます。

有効な値は以下のとおりです。

NO QMF SAVE コマンドの SHARE パラメーターのデフォルト設定を NO にします。この値がデフォルトです。

SHARE 変数が NO に設定されている場合、DSQEC_SHARE 大域変数の値は 0 です。

YES

SHARE パラメーターのデフォルト設定を YES にします。

SHARE 変数が YES に設定されている場合、DSQEC_SHARE 大域変数の値は 1 です。

LASTRUN

Q.OBJECT_DIRECTORY 表の LAST_USED 列を更新するコマンドまたはコマンド・セットを決定します。この値は、LIST コマンドで生成されるオブジェクト・リストの「最終使用日」フィールドに表示されます。QMF 大域変数 DSQEC_LAST_RUN を使用して、この値を設定することもできます。LAST_USED 列の値は、発行されたコマンドが正常に行われたかどうかとは無関係に更新されます。ただし、LAST_USED 列は即時に更新されないことがあります。また、QMF が異常終了すると、列が更新されない場合があります。

有効な値は以下のとおりです。

ZERO

以下のいずれかのコマンドが実行されたとき、Q.OBJECT_DIRECTORY 表の LAST_USED 列を更新します。

- CONVERT
- DISPLAY
- EXPORT
- IMPORT
- LAYOUT
- PRINT
- RUN
- SAVE

この値がデフォルトです。

LASTRUN 変数が ZERO に設定されている場合、DSQEC_LAST_RUN 大域変数の値は 0 です。

ONE

LAST_USED 列の更新を RUN、SAVE、および IMPORT コマンドのみに制限します。

LASTRUN 変数が ONE に設定されている場合、DSQEC_LAST_RUN 大域変数の値は 1 です。

TWO

LAST_USED 列の更新を RUN コマンドのみに制限します。

LASTRUN 変数が TWO に設定されている場合、DSQEC_LAST_RUN 大域変数の値は 2 です。

OTC_LICENSE

この変数はサポートされておらず、互換性のためにのみ組み込まれています。有効な値は NOT_USED のみです。

関連概念:

18 ページの『QMF 管理に必要な権限』
特定のデータベース・オブジェクトに対して複数の GRANT ステートメントを発行することを避けるには、大部分の QMF 管理作業およびカスタマイズ作業に、DBADM または同等の権限を持つ許可 ID を使用する必要があります。ユーザーがデータを保管するすべてのデータベースで、このレベルの権限が必要です。

大域変数表を使用した大域変数の設定

Q.GLOBAL_VARS 表を使用して、QMF の開始時に使用される初期大域変数値を設定することができます。これらの大域変数をセッション中に変更可能にするかどうかを指定することもできます。

このタスクについて

ユーザー定義大域変数と読み取り専用でない QMF 事前定義大域変数の両方に、初期値を設定することができます。

手順

Q.GLOBAL_VARS 表の初期大域変数値を設定するには、以下の手順に従ってください。

1. 行を追加する SQL INSERT ステートメントを Q.GLOBAL_VARS 表に作成します。すべての列に、以下のように値を指定します。

VARNAME

大域変数の名前を指定します。

UPDATEABLE

QMF の初期化後、ユーザーが大域変数を変更できないように設定するには、値 0 を指定します。QMF セッション中に大域変数を更新できるように設定するには、値 1 を指定します。

USERID

デフォルトで値 SYSTEM が使用され、指定する必要はありません。

VARVALUE

大域変数の値を指定します。

2. INSERT ステートメントを実行します。

例

Q.GLOBAL_VARS 表への行の挿入については、次のステートメント例を参照してください。

表 30. Q.GLOBAL_VARS に行を追加するサンプル INSERT ステートメント

| INSERT ステートメント | 説明 |
|--|--|
| INSERT INTO Q.GLOBAL_VARS (VARNAME,VARVALUE,UPDATEABLE) VALUES('DSQDC_SCROLL_AMT','CSR',1) | DSQDC_SCROLL_AMT 大域変数を CSR に設定し、ユーザーが大域変数を変更できることを指定します。 |

表 30. Q.GLOBAL_VARS に行を追加するサンプル INSERT ステートメント (続き)

| INSERT ステートメント | 説明 |
|--|---|
| INSERT INTO Q.GLOBAL_VARS (VARNAME,VARVALUE,UPDATEABLE) VALUES('DSQEC_DISABLEADM','0',0) | DSQEC_DISABLEADM 大域変数を 0 に設定し、ユーザーが大域変数を変更できないことを指定します。 |
| INSERT INTO Q.GLOBAL_VARS (VARNAME,VARVALUE,UPDATEABLE) VALUES('MYVAR1','VALUE OF MYVAR1',1) | MYVAR1 大域変数を VALUE OF MYVAR1 に設定し、ユーザーが大域変数を変更できないことを指定します。 |
| INSERT INTO Q.GLOBAL_VARS (VARNAME,VARVALUE,UPDATEABLE) VALUES('MYVAR2','VALUE OF MYVAR2',2) | MYVAR2 大域変数を VALUE OF MYVAR2 に設定し、ユーザーが大域変数を変更できないことを指定します。 |

Q.GLOBAL_VARS 表の構造

Q.GLOBAL_VARS 表には、大域変数値およびセッション変数値が保管されます。この表は QMF のインストール時に作成され、DB2 (VSE および VM 版) での場合上を除き、QMF を開始するために必要です。

QMF 管理者は、Q.GLOBAL_VARS 表を使用して、QMF の開始時に設定される大域変数値を指定することができます。定義済みの各大域変数の UPDATEABLE 列の値に応じて、ユーザーは QMF の初期化後にこれらの大域変数値を変更できます。デフォルトでは、大域変数値は、ユーザーがリセットするか QMF セッションを終了するまで保持されます。ただし、DSQEC_USERGLV_SAV 大域変数は、あるセッションから別のセッションまで大域変数値を保持するように設定できます。この場合、ユーザーの大域変数値もこの表に保管されます。

DSQEC_SESSGLV_SAV 大域変数は、1 つのセッション内および複数のセッション間の両方で、QMF パネル上のユーザーの入力を保存するよう設定することもできます。この場合、ユーザーのパネル入力は、接頭部 DXY を持つセッション変数として Q.GLOBAL_VARS 表に保存されます。

表は空にできますが、列に NULL 値を入れることはできません。

表 31. Q.GLOBAL_VARS 表の構造

| 列名 | データ・タイプと長さ | 機能および使用できる値 |
|---------|------------|---|
| VARNAME | CHAR(18) | 機能 大域変数またはセッション変数の名前を指定します。 値 任意の有効な大域変数名またはセッション変数名。 |

表 31. Q.GLOBAL_VARS 表の構造 (続き)

| 列名 | データ・タイプと長さ | 機能および使用できる値 |
|------------|--------------|---|
| UPDATEABLE | CHAR(1) | <p>機能</p> <p>QMF の初期化後にユーザーが大域変数値を変更できるかどうかを指定します。</p> <p>値</p> <p>0 QMF セッション中に大域変数を更新できないことを指定します。</p> <p>1 QMF セッション中に大域変数を更新できることを指定します。</p> <p>現行 QMF セッションの状態に関する情報を記録する変数など、一部の QMF 大域変数は読み取り専用です。読み取り専用の大域変数を Q.GLOBAL_VARS 表に含める場合、変数は、UPDATEABLE 設定とは無関係に初期化時に設定されません。QMF 初期化も失敗します。</p> <p>セッション間で保存される大域変数の場合、この値は、現在ユーザーが変数を更新できるかどうかに従って設定されます。変数を更新できない場合、この値は 0 に設定されます。それ以外の場合は 1 に設定されます。</p> |
| USERID | VARCHAR(128) | <p>機能</p> <p>QMF ユーザー ID を指定します。</p> <p>値</p> <p>SYSTEM</p> <p>この値は、管理者が初期大域変数値を表に追加する場合に使用する必要があります。この値は、大域変数がすべてのユーザーに適用されることを指定します。</p> <p>SYSTEM がデフォルト値です。</p> <p>userid 大域変数およびセッション値が表に追加される際に、この値はユーザーの QMF ID に設定されます。</p> <p>USERID フィールドおよび VARNAME フィールドは、各大域変数項目の主キーとして使用されます。</p> |

表 31. Q.GLOBAL_VARS 表の構造 (続き)

| 列名 | データ・タイプと長さ | 機能および使用できる値 |
|----------|--|---|
| VARVALUE | VARCHAR(2000) インストール時に、VARVALUE 列は長さ 2000 で定義されます。ただし、18 から 32555 までの任意の長さを使用することができます。 | 機能 大域変数またはセッション変数の値を指定します。 値 任意の有効な大域変数値。 |

デフォルト・システム初期化プロシージャによる初期化

QMF のデフォルト・プロシージャの名前は Q.SYSTEM_INI です。Q.SYSTEM_INI プロシージャは、ホーム・パネルが表示される前に、ユーザーに実行権限が与えられた任意の QMF コマンドまたは任意の保管照会文を実行することができます。

このタスクについて

Q.SYSTEM_INI プロシージャは QMF の初期化が完了した直後、かつ DSQSRUN プログラム・パラメーターで指定された QMF 初期プロシージャが実行される前に実行されます。したがって、DSQSRUN パラメーターで指定されたプロシージャにセッション制御が含まれている場合、それらは Q.SYSTEM_INI プロシージャに移動させることを考慮してください。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。

このサンプル Q.SYSTEM_INI プロシージャは QMF に付属しており、SHARE=YES をデータベースに保存されるすべての QMF オブジェクトのデフォルトにします。

```

-- 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

```

図 21. QMF に付属する Q.SYSTEM_INI プロシージャ

QMF セッション中は、Q.SYSTEM_INI プロシージャ、およびこのプロシージャで使用される、あるいは呼び出されるすべてのオブジェクトは、他のすべての 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を持持ちます。Q.SYSTEM_INI プロシージャは、QMF セッションの開始のたびに QMF によって実行が試行されることを除けば、通常の QMF プロシージャです。したがって、プロシージャが正常に実行されるようにするには、QMF は QMF 管理者権限を持つ許可 ID の下で開始しておく必要があります。

Q.SYSTEM_INI プロシージャは存在しているけれども、制限されている (共有されていない) 場合は、結果はその他の QMF プロシージャ・オブジェクトの場合と同様になります。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。

手順

デフォルトのシステム初期化プロシージャを使用するには、以下の手順を実行します。

1. プロシージャをインストールします。QMF に付属しているサンプル・プロシージャの名前は DSQ0BINI です。これは、QMF1210.SDSQSAPE (DSQ0BINI) にあります。
 - QMF バージョン 12.1 を前の QMF リリースを含むデータベースにインストールした場合は、サンプルをインストールする前に QMF 内から下のコマンドを発行して既存のシステム初期化プロシージャを確認します。必ず、管理者権限を持つ QMF ユーザー ID の下でデータベースに接続してください。

```
DISPLAY Q.SYSTEM_INI
```
 - 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は、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、次のコマンドを続けます。

```
IMPORT PROC FROM 'QMF1210.SDSQSAPE(DSQ0BINI)'
```

2. プロシージャを Q.SYSTEM_INI の名前で、将来、実行元となるデータベースに保存します。デフォルトのシステム初期化プロシージャが自動的に実行できるようになるためには、プロシージャは Q.SYSTEM_INI という名前で保存されている必要があります。すべての QMF ユーザーとプロシージャを共有します。
 - a. プロシージャを共有するには、それを QMF で表示してから、コマンド SAVE PROC (SHARE=YES を発行します。
 - b. プロシージャを保存する際に、プロシージャを説明するコメントを追加します。以下に例を示します。

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

次のタスク

Q.SYSTEM_INI プロシージャに関する問題をトラブルシューティングするために、QMF L2 トレース・オプションを使用して、発行されたコマンドおよびメッセージを表示することができます。

Q.SYSTEM_INI プロシージャは、他のプロシージャで置き換えることはできませんが、他のプロシージャを呼び出すことはできます。デフォルトのシステム初期化プロシージャがお客様のサイトの要件に合わない場合は、独自のバージョンを作成することができます。

関連概念:

17 ページの『QMF のインストールおよび管理に必要な権限』

QMF のインストールおよび一般的な QMF 管理には、特定の権限が必要です。

関連タスク:

454 ページの『トレース機能』

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。

独自の初期化プロシージャの作成

デフォルト・バージョンがお客様のサイトの要件に合わない場合は、独自のプロシージャを作成することができます。

独自の QMF セッション・プロシージャは、特定の QMF 大域変数またはプロファイル値の設定程度の簡単なものにする 것도、QMF へのカスタマイズ・インターフェースを備えた複雑なものにする 것도できます。独自の初期化プロシージャを呼び出すには、Q.SYSTEM_INI プロシージャを使用します。Q.SYSTEM_INI プロシージャは、サイトの特定のプロシージャで置き換えることはできませんが、他のプロシージャを呼び出すために使用することができます。

独自のシステム初期化プロシージャの作成方法の例については、以下のいずれかのセクションを参照してください。

- 185 ページの『ユーザー固有のデフォルトを設定するプロシージャの作成』
- 185 ページの『オブジェクト・リストを表示するプロシージャの作成』

ユーザー固有のデフォルトを設定するプロシージャの作成

Q.SYSTEM_INI プロシージャは、別のプロシージャを呼び出すことができます。呼び出されるプロシージャとしては、QMF ユーザーが作成、所有、および更新するユーザー・プロシージャでも使用できます。

各ユーザーは、固有の SQL ID の下で動作します。SQL ID は、オブジェクト所有者が明示的に指定されていない場合、デフォルトのオブジェクト所有者です。したがって、各ユーザーのシステム初期化プロシージャで同じ名前を使用することができますが、各ユーザーのニーズによっては、それを変更することもできます。以下の例では、大域変数を設定してから、ユーザー固有のセッション・プロシージャを呼び出すシステム初期化プロシージャを示します。この例では、ユーザー固有のセッション・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 set up QMF session defaults for
-- every QMF user and then calls a user procedure named USER_INI that sets
-- individual QMF session defaults.
--
QMF SET GLOBAL (DSQEC_NLFCMD_LANG=1)  -- Process English commands
QMF RESET PROC                        -- Hide contents of this procedure
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)  -- Sort object list by date modified
QMF SET GLOBAL (DSQEC_RESET_RPT=1)    -- Prompt for report completion
RUN USER_INI                          -- Run user's session procedure
QMF END                                -- Display QMF home panel first
QMF SET GLOBAL (DSQEC_NLFCMD_LANG=0)  -- Return to presiding language
```

図 22. ユーザー定義プロシージャを呼び出す Q.SYSTEM_INI の例

以下の USER_INI プロシージャの例は、Q.SYSTEM_INI プロシージャの例で参照されています。

```
PROC                WILLIAMS.USER_INI          LINE 1
-- This QMF procedure example shows how to set up 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 procedure
QMF SET PROFILE (SPACE=MYSPACE)       -- Store data in table space MYSPACE
QMF SET PROFILE (PRINTER=MYROOM)      -- Print reports on a printer named MYROOM
QMF SET GLOBAL (DSQDC_LIST_ORDER=3A)  -- Sort object list by object name
QMF SET GLOBAL (DSQEC_RESET_RPT=2)    -- Always reset reports
QMF SET GLOBAL (DSQEC_SHARE=1)        -- Always share this user's QMF objects
QMF SET GLOBAL (DSQEC_NLFCMD_LANG=0)  -- Return to presiding language
```

図 23. システム初期化プロシージャから呼び出されるユーザー・プロシージャの例

オブジェクト・リストを表示するプロシージャの作成

次の例は、QMF ホーム・パネルに代わってオブジェクトのリストを表示するシステム初期化プロシージャです。

```

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
-- panel.
--
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)  -- Sort object list by object name
QMF SET GLOBAL (DSQEC_NLFCMD_LANG=0)  -- Return to presiding language
QMF LIST ALL                          -- List all objects

```

図 24. QMF ホーム・パネルではなく Q.SYSTEM_INI を使用したオブジェクト・リストの表示

関連タスク:

245 ページの『ユーザーのオブジェクト・リストのカスタマイズ』

DB2 は許可情報を SYSIBM.SYSTABAUTH 表から収集するため、表リストおよび列情報に関して QMF から提供されるデフォルト視点を使用すると、処理時間が長くな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追加のセキュリティ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を作成することを検討してください。

第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターについて説明します。

QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する際には、ストアード・プロシージャー・インターフェースのインストール手順で説明されているように、これらの一部のプログラム・パラメーターを設定することができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

プログラム・パラメーターの要約

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

下の表に、QMF プログラム・パラメーターの長い名前および短縮名、およびプログラム・パラメーターを使用できる環境を示します。CICS でのみ使用されるパラメーターには、短縮名がありません。

表 32. プログラム・パラメーター

| 長い名前 | 短縮名 | 環境 | 説明 |
|----------|-----|-----------|--|
| DSQSBSTG | B | TSO, CICS | <p>制御または設定 報告書操作に使用される、ユーザーごとの仮想記憶域の最大量。</p> <p>有効な値</p> <p>0 - 2147483647 記憶域をバイト単位で指定します。</p> <p>0KB - 2097152KB 記憶域を KB 単位で指定します。</p> <p>0MB - 2048MB 記憶域を MB 単位で指定します。</p> <p>0GB - 2GB 記憶域を GB 単位で指定します。</p> <p>1% - 100% 使用可能な記憶域のパーセントを指定します。 パーセントは TSO でのみ有効です。</p> <p>デフォルト</p> <p>0 TSO、ISPF、またはネイティブ z/OS で QMF を開始した場合 これらの環境で、このパラメーターに 15000 未満の値を設定した場合は、自動的に 15000 が使用されます。</p> <p>500000 CICS で QMF を開始した場合</p> <p>詳細については、199 ページの『固定量の報告書用仮想記憶域の定義』を参照してください。</p> |
| DSQSDBCS | K | TSO, CICS | <p>制御または設定 非 DBCS ディスプレイ装置からの DBCS データの印刷</p> <p>有効な値 YES (非 DBCS ディスプレイ装置からの DBCS データの印刷をサポートする) または NO (このサポートを設定しない)</p> <p>デフォルト NO</p> <p>詳細については、223 ページの『非 DBCS ディスプレイ装置からの DBCS データの印刷』を参照してください。</p> |

表 32. プログラム・パラメーター (続き)

| 長い名前 | 短縮名 | 環境 | 説明 |
|----------|-----|-----------|--|
| DSQSDBNM | D | TSO, CICS | <p>制御または設定</p> <p>QMF ホーム・パネルが表示される前に接続する初期データベース・ロケーションの名前</p> <p>有効な値</p> <p>いずれかの有効なデータベース名。ロケーション名は 16 文字まで指定できます。QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合は、このパラメーターを使用しないでください。</p> <p>デフォルト</p> <p>CICS の場合は、現在 CICS で使用されているデフォルト・データベース。それ以外の場合は、使用されているサブシステムのデフォルト・データベース。</p> <p>詳細については、196 ページの『初期データベース接続の指定』を参照してください。</p> |
| DSQSDBQN | — | CICS | <p>制御または設定</p> <p>QMF トレース・データに使用される CICS リソースの名前</p> <p>有効な値</p> <p>CICS キュー命名規則に従う任意の名前</p> <p>デフォルト</p> <p>DSQD</p> <p>詳細については、222 ページの『CICS のトレースの設定』を参照してください。</p> |
| DSQSDBQT | — | CICS | <p>制御または設定</p> <p>QMF トレース・データに使用される CICS リソースのタイプ</p> <p>有効な値</p> <p>TS (CICS 一時記憶域キューを使用) または TD (CICS 一時データ・キューを使用)</p> <p>デフォルト</p> <p>TD</p> <p>詳細については、222 ページの『CICS のトレースの設定』を参照してください。</p> |

表 32. プログラム・パラメーター (続き)

| 長い名前 | 短縮名 | 環境 | 説明 |
|-----------|-----|-----------|---|
| DSQSDEBUG | T | TSO, CICS | <p>制御または設定 トレース詳細レベル</p> <p>有効な値 ALL (すべての機能を最も詳細なレベルでトレースする) または NONE (トレース・データは記録されない)。QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合は、このパラメーターを使用しないでください。</p> <p>デフォルト NONE</p> <p>詳細については、221 ページの『TSO のトレースの設定』 および 222 ページの『CICS のトレースの設定』を参照してください。</p> |
| DSQSFISO | ISO | TSO, CICS | <p>制御または設定 DATE(ISO) オプションおよび TIME(ISO) オプションを使用してプリコンパイルされた QMF パッケージを使用するかどうか。このパラメーターは DSQAO_DSQSFISO 大域変数を設定します。</p> <p>有効な値</p> <p>YES DATE(ISO) オプションおよび TIME(ISO) オプションを使用してプリコンパイルされた QMF パッケージを使用します。この値を使用すると、形式を指定しない CHAR(<i>datetime-expression</i>) 関数の結果は ISO 形式になります。</p> <p>NO DATE(ISO) オプションおよび TIME(ISO) オプションを使用してプリコンパイルされていない QMF パッケージを使用します。この値を使用すると、形式を指定しない CHAR(<i>datetime-expression</i>) 関数の結果は、DSNHDECP モジュールで指定された形式になります。</p> <p>デフォルト YES</p> |
| DSQSIROW | F | TSO, CICS | <p>制御または設定 報告書の最初の画面が表示される前に、データベースから取り出される行数</p> <p>有効な値 0 から 99999999</p> <p>デフォルト 100</p> <p>詳細については、209 ページの『報告書待ち時間の制御』を参照してください。</p> |

表 32. プログラム・パラメーター (続き)

| 長い名前 | 短縮名 | 環境 | 説明 |
|----------|-----|-----------|---|
| DSQSMODE | M | TSO, CICS | <p>制御または設定 操作モード</p> <p>有効な値 B (バッチ操作) または I (対話式操作)。 QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合は、このパラメーターを使用しないでください。</p> <p>デフォルト I (QMF が呼び出し可能インターフェースから開始される場合は B)</p> <p>詳細については、 214 ページの『操作モードの指定 (対話式またはバッチ)』を参照してください。</p> |
| DSQSMRFI | MR | TSO, CICS | <p>制御または設定 DB2 for z/OS の複数行フェッチと挿入機能の使用。このパラメーターは DSQAO_DSQSMRFI 大域変数を設定します。</p> <p>有効な値 YES (複数行のフェッチ/挿入を使用) または NO (単一行のフェッチ/挿入を使用)。照会アクセラレーターを使用している場合は、この値を YES に設定する前に、その行セット・カーソル機能を確認してください。一部の照会アクセラレーターでは、行セット・カーソルを使用して実行される照会 (複数行フェッチで使用) はサポートされません。</p> <p>デフォルト NO</p> <p>詳細については、 211 ページの『複数行のフェッチと挿入のサポートの使用可能化』を参照してください。</p> |

表 32. プログラム・パラメーター (続き)

| 長い名前 | 短縮名 | 環境 | 説明 |
|----------|-----|-----------|---|
| DSQSMTHD | MT | TSO | <p>制御または設定</p> <p>ユーザーごとの第 2 DB2 for z/OS スレッドの使用。このパラメーターは DSQAO_DSQSMTHD 大域変数を設定します。</p> <p>有効な値</p> <p>YES (第 2 DB2 for z/OS スレッドの使用) または NO (単一の DB2 for z/OS スレッドの使用)。第 2 スレッドは、RUN QUERY コマンドと DISPLAY TABLE コマンド、およびオープン・カーソルによるそれ以降の報告書のスクロール (BOTTOM、TOP、FORWARD、BACKWARD、RIGHT、および LEFT) の間のみアクティブです。完全な報告書を使用するコマンド (EXPORT DATA、SAVE DATA、PRINT REPORT、DPRE など) も第 2 スレッドを使用します。</p> <p>デフォルト</p> <p>NO</p> |
| DSQSPILL | L | TSO, CICS | <p>制御または設定</p> <p>アクティブな記憶域で不要になったデータを保持する追加記憶域の使用</p> <p>有効な値</p> <p>YES (アクティブな記憶域で不要になったデータに予備記憶域を使用する) または NO (予備記憶域を使用しない)</p> <p>デフォルト</p> <p>QMF が TSO で実行されている場合は YES、QMF が CICS で実行されている場合は NO。DSQSPILL が YES の場合は、DSQSPTYP を使用して予備データに使用する記憶域のタイプを指定します。</p> <p>詳細については、200 ページの『仮想記憶域で不要になったデータ用の追加記憶域の取得』を参照してください。</p> |
| DSQSPLAN | P | TSO | <p>制御または設定</p> <p>QMF アプリケーション・プランの名前</p> <p>有効な値</p> <p>任意の有効な DB2 アプリケーション・プラン名このパラメーターは、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合、DSQSCMDn EXEC 内に必要です。</p> <p>デフォルト</p> <p>QMF12</p> <p>詳細については、197 ページの『QMF アプリケーション・プラン名の指定』を参照してください。</p> |

表 32. プログラム・パラメーター (続き)

| 長い名前 | 短縮名 | 環境 | 説明 |
|----------|-----|-----|---|
| DSQSPRID | U | TSO | <p>制御または設定</p> <p>QMF for TSO のインストールのための QMF プロファイル・キー</p> <p>有効な値</p> <p>PRIMEID (QMF はユーザーの DB2 1 次許可 ID を Q.PROFILES 表の CREATOR 列に対して認証します) または TSOID (QMF はユーザーの TSO ログオン ID を Q.PROFILES 表の CREATOR 列に対して認証します)。QMF for TSO を DB2 for z/OS スタート・プロシージャーとして開始する場合は、値 PRIMEID のみが許可されます。</p> <p>デフォルト</p> <p>PRIMEID</p> <p>詳細については、197 ページの『TSO で QMF プロファイル・キーとして使用する ID の指定』を参照してください。</p> |
| DSQSPTYP | ST | TSO | <p>制御または設定</p> <p>アクティブな記憶域で不要なデータを保持する拡張記憶域の使用。DSQPILL パラメーターが YES に設定されている場合のみにサポートされます</p> <p>有効な値</p> <p>FILE (ファイルにデータを予備書き込みする) または 64BIT (拡張仮想記憶域にデータを予備書き込みする)</p> <p>デフォルト</p> <p>FILE</p> <p>詳細については、201 ページの『拡張仮想記憶域への報告書データの書き込み (TSO のみ)』を参照してください。</p> |

表 32. プログラム・パラメーター (続き)

| 長い名前 | 短縮名 | 環境 | 説明 |
|----------|-----|-----------|--|
| DSQSRSTG | R | TSO | <p>制御または設定</p> <p>QMF 以外のアプリケーションに使用するために TSO 領域から予約する記憶域のバイト数</p> <p>QMF が QMF セッションで最初の報告書を生成する際、このパラメーターで指定された記憶域容量を、その時点で QMF で使用可能なフリー連続記憶域容量から差し引きます。残りの記憶域は、以降の QMF セッションで QMF 報告書操作に使用可能になります。</p> <p>有効な値 0 から 99999999</p> <p>デフォルト 0</p> <p>詳細については、199 ページの『QMF 以外のアプリケーションに使用するための TSO 領域からの仮想記憶域の予約』を参照してください。</p> |
| DSQSRUN | I | TSO, CICS | <p>制御または設定</p> <p>QMF ホーム・パネルが表示される前に、QMF の開始時に実行する QMF プロシージャの名前</p> <p>有効な値 任意の有効なプロシージャ名。QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始する場合は、このパラメーターを使用しないでください。</p> <p>デフォルト 初期プロシージャを実行しない</p> <p>詳細については、215 ページの『QMF の開始時に実行する初期プロシージャの指定』を参照してください。</p> |
| DSQSSPQN | — | CICS | <p>制御または設定</p> <p>DSQSPILL が YES に設定されている場合に、アクティブな記憶域で不要になったデータを保持する CICS キューの名前</p> <p>有効な値 CICS キュー命名規則に従う任意の名前</p> <p>デフォルト DSQSmnnn (nnnn は CICS 端末 ID)</p> <p>詳細については、206 ページの『CICS でのファイルへの報告書データの書き込み』を参照してください。</p> |

表 32. プログラム・パラメーター (続き)

| 長い名前 | 短縮名 | 環境 | 説明 |
|----------|-----|-----|--|
| DSQSSUBS | S | TSO | <p>制御または設定</p> <p>TSO で QMF を開始する DB2 サブシステムの名前</p> <p>有効な値</p> <p>任意の有効な DB2 サブシステム名。このパラメーターは、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合、DSQSCMDn EXEC 内に必要です。</p> <p>デフォルト</p> <p>DSN</p> <p>詳細については、『TSO で QMF を開始する DB2 (z/OS 版) サブシステムの名前の指定』を参照してください。</p> |

関連情報:

 DB2 資料

有効なプラン名に関する情報を検索します。

データベースおよび環境のパラメーターの設定

これらのプログラム・パラメーターは、QMF を実行するデータベースおよび環境に関連する設定およびプリファレンスを制御します。

TSO で QMF を開始する DB2 (z/OS 版) サブシステムの名前の指定

QMF は TSO で実行されている DB2 for z/OS サブシステム内で開始される必要があります。DSQSSUBS パラメーターを使用して、QMF を開始する DB2 for z/OS サブシステムの名前を指定します。

データ共有環境では、DB2 サブシステム名の代わりに、DB2 グループ接続名を使用できます。

QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合は、DSQSCMDn EXEC 内にこのパラメーターの値が必要です。

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

初期データベース接続の指定

ローカルの DB2 for z/OS サブシステム内で QMF を開始して、QMF ホーム・パネルが表示される前に、すぐに別のデータベースに接続することができます。

手順

QMF を開始して初期データベース接続を指定するには、以下の手順に従ってください。

1. DSQSSUBS パラメーターを、QMF を開始する DB2 for z/OS サブシステムの名前に設定します。
2. DSQSDBNM パラメーターを、QMF ホーム・パネルが表示される前に接続したいロケーションの名前に設定します。場所名には 16 文字まで指定できます。

タスクの結果

QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始する際に、ストアード・プロシージャ・インターフェースにパラメーター値を渡す DSQSCMDn EXEC 内にこのパラメーターが指定されている場合、このパラメーターは無視されます。

関連概念:

195 ページの『TSO で QMF を開始する DB2 (z/OS 版) サブシステムの名前の指定』

QMF は TSO で実行されている DB2 for z/OS サブシステム内で開始される必要があります。DSQSSUBS パラメーターを使用して、QMF を開始する DB2 for z/OS サブシステムの名前を指定します。

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャとしての開始』

QMF for TSO のストアード・プロシージャ・インターフェースでは、DB2 for z/OS ストアード・プロシージャを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

QMF アプリケーション・プラン名の指定

QMF アプリケーション・プランのデフォルト名は QMF12 です。QMF アプリケーション・プランに違う名前を選択した場合は、このパラメーターで正しい名前を渡す必要があります。

QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合は、DSQSCMD n EXEC 内にこのパラメーターの値が必要です。

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

TSO で QMF プロファイル・キーとして使用する ID の指定

QMF for TSO の各ユーザーに固有のプロファイルを作成することを選択する場合は、表の CREATOR 列の値はすべて TSO ログオン ID にするか、または DB2 1 次許可 ID にすることができます。QMF でユーザーの TSO ログオン ID を使用してユーザーを認証することを選択する場合は、QMF の開始時にこの選択を指定する必要があります。

QMF を開始するとき、QMF ではプログラムが開始されたユーザー ID を、Q.PROFILES 表の CREATOR 列の値に対して認証します。QMF は最初に、プロ

グラムが開始された特定ユーザー ID の CREATOR 列を検査します。この ID に対応する行が表にない場合、QMF は CREATOR=SYSTEM が指定された Q.PROFILES 表内の行を検査します。

QMF でユーザーの TSO ログオン ID を使用してユーザーを認証するように設定する場合は、QMF の開始時に DSQSPRID プログラム・パラメーターに値 TSOID を入力する必要があります。

DSQSPRID=TSOID

このパラメーターのデフォルトは値 PRIMEID であり、QMF はユーザーの DB2 1 次許可 ID を使用してユーザーを認証します。

TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合は、値 PRIMEID のみが DSQSCMDn EXEC 内で有効です。

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

報告書用記憶域の定義

ユーザーがデータを検索、インポート、またはエクスポートする QMF タスクを実行する際に、データは操作が完了するまで記憶域に保持されます。CICS と TSO の両方で、これらの操作に使用する仮想記憶域容量の固定上限を設定することができます。この仮想記憶域は、アクティブ記憶域とも呼ばれます。TSO では、報告書に使用する使用可能な記憶域のパーセンテージまたは部分を指定することもできます。

このタスクについて

TSO と CICS の両方で QMF のパフォーマンスを向上させるために、アクティブな記憶域で不要になったデータの保持に使用する追加記憶域を指定することもできます。このタイプの記憶域は、予備記憶域と呼ばれます。データは、拡張仮想記憶

域またはファイルに予備書き込みすることができます。

固定量の報告書用仮想記憶域の定義

DSQSBSTG パラメーターを使用して、QMF が報告書操作で使用する仮想記憶域の固定バイト数を指定します。

DSQSBSTG パラメーターは 0 から 2147483647 の間の値に設定します。この値は、QMF での報告書操作に使用する仮想記憶域の最大バイト数を表します。このパラメーターに 15000 未満の値を設定した場合は、自動的に 15000 が使用されます。

DSQSBSTG パラメーターのデフォルトは次のとおりです。

- TSO、ISPF、またはネイティブ z/OS から QMF を開始した場合は 0
- CICS から QMF を開始した場合は 500000

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

可変量の報告書用仮想記憶域の定義

TSO では、報告書に使用できる仮想記憶域量を、使用可能な総記憶域の比率または部分として定義することができます。

DSQSBSTG プログラム・パラメーターを使用して、報告書に使用される記憶域を、使用可能な記憶域の比率として直接定義することができます。DSQSRSTG プログラム・パラメーターを使用して、QMF 以外のアプリケーション用に予約する記憶域量を定義できます。残りの仮想記憶域量は報告書操作に使用できます。

使用可能な記憶域の比率として報告書用記憶域を定義

TSO の場合のみ、報告書に使用できる記憶域量を、使用可能な総仮想記憶域の比率として定義することができます。

手順

報告書用記憶域の量を、使用可能な記憶域の比率として定義するには、以下のようになります。

DSQSBSTG パラメーターを 1% から 100% の範囲内の値に設定します。% 記号を数値の後に続ける必要があります。値と記号の間にスペースを入れてはいけません。

QMF 以外のアプリケーションに使用するための TSO 領域からの仮想記憶域の予約

QMF for TSO の DSQSRSTG パラメーターを使用して、TSO コマンド、REXX、または ISPF などの QMF 以外のアプリケーションで使用するために TSO 領域に予約する仮想記憶域の最大容量を指定します。この予約された記憶域は、QMF 報告書操作に使用されなくなります。

このタスクについて

このパラメーターに設定する記憶域の容量は、他のプログラムおよび報告書の作成に影響する可能性があります。QMF で新しい報告書が作成される際に、QMF アドレス・スペースで使用可能な記憶域容量が決まります。QMF は、DSQSRSTG パラメーターで指定された記憶域容量を、その時点で QMF で使用可能なフリー記憶域容量から差し引きます。残りの仮想記憶域量は報告書操作に使用できます。

手順

QMF 以外のアプリケーションに使用するために TSO 領域から予約する記憶域の容量を設定するには、以下の手順に従ってください。

1. DSQSBSTG パラメーターが 0 (デフォルト値) に設定されていることを確認します。QMF for TSO で DSQSBSTG がゼロ以外の値に設定されていると、QMF は指定の記憶域容量を割り振り、DSQSRSTG パラメーターを無視します。
2. DSQSRSTG パラメーターを 0 から 99999999 の間の値に設定します。この値は、QMF に関連しないアプリケーション用に予約する仮想記憶域の最大バイト数を表します。

DSQSBSTG と DSQSRSTG 両方のパラメーターに 0 を指定すると、QMF 以外のアプリケーションのために記憶域は予約されません。この値は、QMF セッション中に z/OS、TSO コマンド、REXX、ISPF、または QMF に関連しないその他のサービスを使用しないユーザーにとって十分と考えられます。ただし、最も一般的なユーザーであっても、サイト定義の QMF コマンドを発行する際に、気付かずに QMF に関連しないプログラムを使用することがあるため注意が必要です。このようなコマンドは QMF アプリケーションによって実行されますが、一般的に QMF に関連しないプログラムが広く使用されています。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。

関連概念:

199 ページの『固定量の報告書用仮想記憶域の定義』

DSQSBSTG パラメーターを使用して、QMF が報告書操作で使用する仮想記憶域の固定バイト数を指定します。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

仮想記憶域で不要になったデータ用の追加記憶域の取得

アクティブな記憶域への要求は他のプログラムの操作に影響するため、QMF では追加の記憶域 (予備記憶域と呼ばれます) を指定して、アクティブな記憶域で不要になったデータを保管することができます。

このタスクについて

QMF 報告書が返される時、またはユーザーが IMPORT または EXPORT コマンドを発行したとき、これらの操作に必要なデータは仮想記憶域に保持されます。こ

の仮想記憶域は、アクティブな記憶域とも呼ばれます。予備記憶域を使用すると、QMF は同一データを複数回コピーするためにデータベースに戻る必要がなくなるため、対話式 QMF セッションでのパフォーマンスが向上します。代わりに、データは予備記憶域から検索されて、必要に応じてアクティブな記憶域に戻されます。

データは、以下のいずれかのイベントが発生する前に、必要に応じて予備記憶域に書き込まれます。

- QMF 報告書がリセットされた (RESET DATA コマンドにより)、または新しい報告書が作成された (例えば、QMF RUN QUERY、DISPLAY TABLE、または IMPORT DATA コマンドを発行した場合)。
- 照会が終了し (要求された行がすべて検索され)、DATA オブジェクトが完了した。
- 予備データ用に指定した記憶域の最大容量が使用された。

予備記憶域には 2 つのタイプがあります。QMF for TSO では、予備データ用の拡張記憶域を使用できます。QMF for TSO および QMF for CICS の両方で、アクティブな記憶域が使い果たされた場合に使用される予備データ用ファイルを割り振ることができます。

関連タスク:

『拡張仮想記憶域への報告書データの書き込み (TSO のみ)』

QMF for TSO では、QMF が実行されているシステムで利用可能な拡張記憶域が非常に限られている場合を除き、予備データ用の拡張記憶域が使用されます。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

拡張仮想記憶域への報告書データの書き込み (TSO のみ)

QMF for TSO では、QMF が実行されているシステムで利用可能な拡張記憶域が非常に限られている場合を除き、予備データ用の拡張記憶域が使用されます。

このタスクについて

手順

QMF for TSO で拡張仮想記憶域に報告書データを予備書き込みするには、次の手順に従います。

1. QMF の開始時に、次のプログラム・パラメーターを設定します。
 - DSQSPILL=YES
 - DSQSPTYP=64BIT
2. 拡張記憶域管理機能への各要求ごとに QMF が獲得する拡張記憶域のメガバイト数を DSQEC_EXTND_STG 大域変数に設定します。拡張記憶域を必要とする操作を実行すると、その操作が完了するまで、または拡張記憶域を使い果たすまで、QMF は拡張記憶域管理機能に対して指定容量を求める要求を繰り返し発行します。この大域変数のデフォルト値は 25 です。これは、QMF が各要求ごとに 25 MB の記憶域を要求することを示します。

この大域変数を設定する場合は、QMF ユーザーが作業する DATA オブジェクトの平均サイズを考慮してください。平均サイズが非常に大きい場合に DSQEC_EXTND_STG 変数に小さすぎる値を設定すると、QMF は DATA オブジェクトの完了までに拡張記憶域管理機能に呼び出しを数多く発行しなければならないため、全般的なパフォーマンスに影響する可能性があります。

TSO でのファイルへの報告書データの書き込み

QMF ユーザーが大容量の仮想記憶域を必要とするデータ・タイプ (XML または LOB データ・タイプなど) で作業する場合は、ファイルではなく拡張記憶域にデータを予備書き込みするように QMF を構成する必要があります。

このタスクについて

QMF 報告書がリセットされた場合 (RESET DATA コマンドにより)、または新しい報告書が作成された場合 (例えば、RUN QUERY、DISPLAY TABLE、または IMPORT DATA コマンドを発行した場合)、QMF は TTR ゼロで予備ファイルをオープンおよびクローズして、スペース割り振りを 1 次割り振りにリセットします。

関連概念:

207 ページの『予備ファイルの問題のトラブルシューティング』

予備ファイルを作成すると記憶域の問題を解決できますが、その他の問題が発生する場合があります。通常は、スペースを追加し、ユーザーの予備ファイルを移動し、パフォーマンスに影響する検討要因を変更することにより、これらの問題から復旧することができます。

予備ファイルに必要なスペースの見積もり:

予備記憶域が使い果たされると、QMF は予備ファイルにすでに取得済みのデータを使用せずに、データベースから再度データを取得し、仮想記憶域を使用してそのデータを保持するため、予備記憶域を十分に準備することが重要です。

このタスクについて

手順

以下の手順を使用して、予備ファイルに必要なスペース容量を計算します。

1. DATA オブジェクトに表れる可能性がある最長行の幅 (W) を計算する。このためには、フィールド幅 (バイト数) を加算します (次の表を使用)。定義列、および LOB または XML データを含む列は、予備ファイルに書き込まない。LOB 列および XML 列は、取り出されて使用時に報告書記憶域に移動される際に、拡張記憶域に保管されます。

表 33. さまざまなデータ・タイプのフィールドの幅: 予備ファイルの必要サイズを計算する際は、この表を使用してデータの行幅の見積もりに役立ててください。

| フィールド・タイプ | フィールド幅 (バイト) |
|---|--------------|
| CHAR(<i>n</i>)、BINARY(<i>n</i>) | <i>n</i> +2 |
| VARCHAR(<i>n</i>)、VARBINARY(<i>n</i>) | <i>n</i> +4 |
| DATE | 12 |
| DECFLOAT(16) | 8 |

表 33. さまざまなデータ・タイプのフィールドの幅 (続き): 予備ファイルの必要サイズを計算する際は、この表を使用してデータの行幅の見積りに役立ててください。

| フィールド・タイプ | フィールド幅 (バイト) |
|---------------------------------|--|
| DECFLOAT(34) | 16 |
| 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 |
| BIGINT | 22 |
| TIME | 10 |
| TIMESTAMP(0) | 21 |
| TIMESTAMP(n) | $22 + n$ ($n= 1$ から 12) |
| TIMESTAMP(0) WITH TIME ZONE | 29 |
| TIMESTAMP(n) WITH TIME ZONE | $30 + n$ ($n= 1$ から 12) |
| 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。

2. W の値を使用して、1 ページ当たりの行数 (R)、または 1 行当たりのページ数 (P) を計算する。
 - W が 4096 以下の場合、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。
 - 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 より大きい場合、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複数のページにわたるようにします。各行は、ページの先頭から始まります。
3. 手順 1 (202 ページ) と手順 2 の値を使用して、予備ファイルに必要なページ数を計算する。
 - 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 より大きい場合は、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。

下表に、それぞれの予備ファイルの計算例を示します。

表 34. 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 x 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 行/バッファースの場合:

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

R=6000、2 バッファース / 行の場合:

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

4. 同時 QMF ユーザー数を見積もって、これらのユーザー全員の個別予備ファイルを収容できることに加えて、他のプログラムからの記憶域への要求も収容するのに十分な予備記憶域を準備してください。

関連タスク:

201 ページの『拡張仮想記憶域への報告書データの書き込み (TSO のみ)』

QMF for TSO では、QMF が実行されているシステムで利用可能な拡張記憶域が非常に限られている場合を除き、予備データ用の拡張記憶域が使用されます。

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 ステートメントによって、予備ファイルを割り振ることができます。

始める前に

QMF の開始時に次のプログラム・パラメーター値が設定されていることを確認します。

- DSQSPILL=YES
- DSQSPTYP=FILE

これらの値は両方とも、これらのプログラム・パラメーターのデフォルト値です。

このタスクについて

重要: 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場合、

低下することがあります。この場合は、入出力異常終了が発生します。QMF は、DCB SYNAD 出口用のエラー処理ルーチンを備えており、DCB 異常終了出口によって提供された情報を使用してこれらの入出力異常終了から復旧します。次に、QMF 予備ファイルの使用を停止して、データベースからデータを再度フェッチします。DCB 異常終了 (B37 異常終了) をインターセプトする z/OS ツールを使用している場合、QMF 予備ファイルをそのような操作から除外することを確認します。除外しない場合、QMF が予備ファイルを適切に管理できなくなり、予期しない結果が発生するだけでなく、エラーのトレースおよび診断が困難になります。

手順

予備ファイルを割り振る際には、次のガイドラインに従ってください。

- ユーザーのセッション中に保持される一時データ・セットとしてファイルを割り振ります。
- DASD が 16,777,215 トラック以内の単一ボリュームである場合は、ファイルを仮想入出力装置に割り振ることができます。この割り振りは、割り振りステートメント例の UNIT=SYSVIO ステートメントに示されています。または、予備ファイルを他の DASD 記憶域に割り振ることもできます。予備ファイルを端末装置またはディスプレイ装置に割り振ることはできません。

QMF 予備ファイルは、BSAM アクセス方式を使用する直接アクセス用に割り振られます。QMF 予備ファイルの管理には、z/OS の NOTE、POINT、READ、WRITE、および CHECK サービスが使用されます。これらのシステム・サービスの使用に適用される z/OS の制約事項はすべて、QMF 予備ファイルの使用にも適用されます。

- ブロックごとに 1 レコードの固定長レコードを指定します。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。)
- 下の例に示すように、RLSE キーワードを使用して記憶域の解放を指定します。
- CLIST 内で予備ファイルを割り振るには、以下の例のように、ATTR ステートメントを ALLOC ステートメントの前に発行します。ATTR ステートメントは必須ではありませんが、指定可能です。

```
ATTR SPILL RECFM(F) LRECL(4096) BLKSIZE(4096)
ALLOC FILE(DSQSPILL) UNIT(SYSVIO) SPACE(10,20) RELEASE +
NEW DELETE USING(SPILL)
```

- ユーザーの QMF セッション中に予備ファイルを割り振りまたは再割り振りする必要がある場合は、以下の手順に従います。
 1. QMF RESET DATA コマンドを発行して、現行の予備ファイルを閉じる。
 2. FREE コマンドを発行して、現行の予備ファイルの割り振りを解放する。
QMF 内からこのコマンドを発行する場合は、以下の例のように、QMF TSO コマンドを前に付ける必要があります。

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

3. 前の例に示されているように、ATTR ステートメントおよび ALLOC ステートメントを使用して新しい予備ファイルを割り振る。ユーザーが新規報告書を作成すると、QMF は予備ファイル指定を使用します。

例

予備ファイルの割り振りステートメントの例は、QMF に付属のサンプル・ログオン・プロシージャにあります。このサンプルでは、予備ファイルは DD 名 DSQSPILL に割り振られています。

```
//DSQSPILL DD DSN=&&SPILL,DISP=(NEW,DELETE),  
//          UNIT=SYSVIO,SPACE=(TRK,(10,20),RLSE),  
//          DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)
```

このステートメントの SPACE オペランドにより、以下のように、セッション中の予備ファイルの記憶域所要量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 10 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どきのみに限られる。
- RLSE キーワードによって、ユーザーのデータ・オブジェクトのリセット時に、QMF に 2 次エクステントをすべて解放させる。例えば、ユーザーが新規の照会を実行し、新規の報告書が戻されるときに、DATA オブジェクトがリセットされます。

CICS でのファイルへの報告書データの書き込み

CICS では、予備データは一時記憶域キューに入れられる必要があります。

手順

CICS で報告書データの書き込みにファイルの使用を指定するには、次の手順に従います。

1. 予備ファイルに必要なスペースを見積もる。202 ページの『予備ファイルに必要なスペースの見積もり』を参照してください。
2. 同時 QMF ユーザー全員の個別予備ファイルを収容できることに加えて、QMF 以外のトランザクションからの要求も収容できるように DFHTEMP 記憶域を拡大する。
3. QMF の開始時に、次のプログラム・パラメーターを設定します。
 - DSQSPILL=YES
 - DSQSPTYP=FILE
 - DSQSSPQN=*queuename*

タスクの結果

例えば、英語で QMF を開始するとき、CICS キューに名前 MYDATA を指定するには、以下のように入力します。

```
QMFE DSQSPILL=YES, DSQSPTYP=FILE, DSQSSPQN=MYDATA
```

QMF をバッチ・モードで実行している場合の報告書データの書き込み

予備記憶域が最もパフォーマンス向上に役立つのは、対話式 QMF セッションの場合、すなわち DSQSMODE パラメーターが I に設定されている場合です。ただし、QMF を非対話式で実行する (DSQSMODE パラメーターが B に設定されている) 場合も、報告書の生成にデータの受け渡しが複数回必要なときには、予備記憶域の使用が同じくパフォーマンスを向上させることができます。

DATA 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 (例えば、RUN QUERY コマンドに続いて SAVE DATA コマンドを実行する場合)。

データの受け渡しが複数回必要になるのは、以下の場合です。

- 同じデータに関して、異なる書式で複数の報告書を印刷する必要がある場合。
- 報告書で PCT、CPCT、TCPCT、または TPCT 編集コードを使用する場合。
- 報告書の幅が印刷幅よりも大きいため、QMF がページの分割を行う必要がある場合。

QMF をバッチ・モードで実行する場合は、行われる作業に基づいて DSQSPILL プログラム・パラメーターを設定する必要があります。バッチ・ジョブで印刷用に大きい DATA 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。この場合は、QMF の開始時に DSQSPILL=NO を指定します。

CICS アプリケーション内からバッチ QMF セッションを開始し、CICS 端末 ID を指定しないように選択した場合は、DSQSSPQN パラメーターの値を明示的に指定する必要があります。指定しない場合、QMF は開始されません。

予備ファイルの問題のトラブルシューティング

予備ファイルを作成すると記憶域の問題を解決できますが、その他の問題が発生する場合があります。通常は、スペースを追加し、ユーザーの予備ファイルを移動し、パフォーマンスに影響する検討要因を変更することにより、これらの問題から復旧することができます。

詳しくは、以下のいずれかのセクションを参照してください。

- 『DASD ボリューム上のスペースが少な過ぎることが原因の問題』
- 208 ページの『予備ファイル用の 2 次スペース割り振りの増加が原因の問題』
- 208 ページの『パフォーマンスの考慮事項』

DASD ボリューム上のスペースが少な過ぎることが原因の問題

複数のユーザーが同一の QMF ログオン・プロシージャーを使用して予備ファイルの問題に直面し、しかもこの共通のログオン・プロシージャーが予備ファイルをすべて特定の 1 つの DASD ボリュームに割り振っている場合は、このボリューム上のスペースが不十分なことが問題の原因になっている可能性があります。そのような場合は、このログオン・プロシージャー中の予備ファイルの DD ステートメントを変更することによって、問題を解決できることがあります。新しい DD ステ

ートメントでは、特定のボリュームに対する参照に代えて、非特定のボリューム参照を行うこととなります。

予備ファイル用の 2 次スペース割り振りの増加が原因の問題

予備ファイルの 2 次割り振りを増やすことで、1 人のユーザーの予備ファイル問題を解決できる場合がありますが、そうすることによって、他のユーザーに対して予備ファイルの問題が生じる可能性があります。2 次割り振りを増やす必要がある場合は、該当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。例えば、大きい報告書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、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、その報告書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できなかった場合です。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。

パフォーマンスの考慮事項

条件付き形式設定または列定義（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。

データを後で取り出すことができるように予備ファイル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。最初にデータをフェッチした後、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。

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 パラメーターの値（報告書の最初の画面を表示する前に取り出される初期行数）。
- 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。
- 報告書を完了させるために追加のデータ受け渡しが必要な場合に、データがどこから検索されるか。
 - データベース（すべてのデータがメモリーおよび予備ファイルに適合するとは限りません）
 - 仮想メモリーおよび予備ファイル
 - 仮想メモリーのみ
- 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 コマンドは、DATA オブジェクトの開始点からやり直す必要がある場合があります。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 の割り振りが不十分なときに報告書が非常に大きいと、BACKWARD コマンドが使用されるたびに DATA オブジェクト全体が 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。DSQSIROW パラメーターの値が十分に大きく、報告書の最初の画面が表示される前に DATA オブジェクト全体を仮想メモリーに取り出すことが可能な場合は、データベースのロックが解除されて、表示された報告書をスクロールする際により高いパフォーマンスを得ることができます。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。ただし、DSQSIROW を高い値に設定すると、報告書の最初の画面の表示が遅くなる場合もあります。

大きな応答セットを作成する、RUN QUERY コマンドに続くコマンドのパフォーマンスを向上させるために、または大きな応答セットに対して SAVE DATA コマンドを実行しているときの記憶域問題を避けるために、複数のデータベース・スレッドを使用することができます。エンド・ユーザー SQL コマンドを個別のデータベース・スレッドで実行するために、このオプションの第 2 スレッドが QMF プログラム・パラメーター DSQSMTHD (YES/NO、デフォルトは NO) によって制御されます。このプログラム・パラメーターによって、QMF 管理者は、QMF が追加のデータベース・スレッドを使用するか、またはデフォルト設定どおりに単一スレッドで動作するかを制御することができます。注: ユーザーごとに 2 つのスレッドが使用されていると、一般的な DB2 リソース消費 (CTHREAD 制限など) に影響が及ぶ可能性があります。

関連概念:

『報告書待ち時間の制御』

DSQSIROW パラメーター。QMF で最初の報告書画面が表示される前に、データベースが取り出す行数を制御します。

関連タスク:

212 ページの『複数のデータベース・スレッドの使用』

大きな応答セットを作成する、RUN QUERY コマンドに続くコマンドのパフォーマンスを向上させるために、または大きな応答セットに対して SAVE DATA コマンドを実行しているときの記憶域問題を避けるために、エンド・ユーザー SQL コマンドを個別のデータベース・スレッドで実行することができます。

フェッチおよび挿入操作のパフォーマンスの制御

フェッチおよび挿入の操作時には、QMF の 2 つのプログラム・パラメーターがデータベース・パフォーマンスの制御に役立ちます。

報告書待ち時間の制御

DSQSIROW パラメーター。QMF で最初の報告書画面が表示される前に、データベースが取り出す行数を制御します。

DSQSIROW パラメーターを使用して、報告書の最初の画面がユーザーに表示される前に、QMF が DATA 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 DATA 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

DSQSIROW の値により、QMF がデータベースからの行のフェッチを停止し、報告書を表示するタイミングが決まります。ただし、QMF は、DSQSIROW 値に関係なく、最初のデータ画面が表示される前に、常に 4 KB のバッファを十分埋めることのできる行をフェッチします。例えば、バッファを埋めるために必要な行数が 62 行で、DSQSIROW を 50 に設定したとします。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。

報告書書式設定オプションの中には、PCT および ACROSS 取扱コードのように、QMF が最初の画面を表示する前に、すべてのデータのフェッチ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。

DSQSIROW の値が小さい場合のパフォーマンス

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。使用する DSQSIROW パラメータの値が小さ過ぎる場合、QMF は、報告書の最初の画面の表示前に、DATA オブジェクトを完了できないことがあります。DATA 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。必要であれば、非コミット読み取りを許可することでこの問題を回避できます。

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 DATA 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、要求された行がすべて取り込まれたかどうかに関係なく、DATA オブジェクトをクリアする。
- いずれかの SAVE コマンド (例えば、SAVE DATA または SAVE FORM) を使用して、残りの行を DATA 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。

バッチ・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 によるデータのフェッチ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 値が大きい場合のパフォーマンス

DSQSIROW パラメーターに使用する値が大き過ぎると、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。さらに、DSQSIROW で指定した行数を取り出すほど十分な記憶域がない場合は、記憶域がいっぱいになります。記憶域がいっぱいな状態で、CICS で QMF を実行している場合、CICS は記憶域が使用可

能になるのを待ちます。TSO では、記憶域がいっぱいであることを示すメッセージが QMF から発行されます。

関連概念:

244 ページの『非コミット読み取りの許可』

ユーザーの QMF セッションが非コミット読み取りを行えるようにしたい場合は、Q.SYSTEM_INI プロシージャ内の大域変数 DSQEC_ISOLATION の値を指定することができます。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

複数行のフェッチと挿入のサポートの使用可能化

DSQSMRFI パラメーター。データベースがフェッチおよび挿入の際に処理する行が複数行か、単一行かを制御します。

DSQSMRFI パラメーター (MR) を使用して、QMF がデータベースのフェッチおよび挿入操作で複数行の処理を使用するかどうかを制御します。複数行の処理を行うと、単一の SQL ステートメントで複数のデータ行の処理が可能になり、ネットワーク・トラフィックの削減につながります。また、次のコマンドで複数行サポートを使用すると、パフォーマンスがより向上します。

- BOTTOM
- FORWARD
- DISPLAY TABLE
- DPRE
- EXPORT DATA および EXPORT TABLE
- IMPORT TABLE
- PRINT REPORT および PRINT TABLE
- RUN QUERY または RUN PROC (検索操作の指定時)
- SAVE DATA

次のような状況では、自動的に単一行フェッチが使用されます。

- コマンドで XML または LOB データの処理が必要な場合
- リクエスター・データベース内で MR=YES を設定して QMF を開始し、QMF CONNECT コマンドを使用して DB2 for z/OS バージョン 8.1.5 (またはそれ以降) 以外のサーバーに接続し、フェッチまたは挿入操作を必要とするコマンドの発行に進む場合

制約事項:

- MR が YES に設定されていて、3 部構成の名前を含む QMF コマンドを使用する場合は、コマンドの開始元となるリクエスターと、コマンドの送信先となるサーバーの両方が DB2 for z/OS 8.1.5 (またはそれ以降) である必要があります。そうでない場合、コマンドは失敗します。3 部構成の名前を DB2 for z/OS バージョン 8.1.5 (またはそれ以降) 以外のサーバーに送信する QMF コマンドを発

行する場合は、MR=NO を設定して QMF を開始してください。3 部構成の名前を含むコマンドは、DB2 (VSE および VM 版) サーバーには送信できません。

- 照会アクセラレーターの機能によっては、QMF DSQSMRFI (MR) プログラム・パラメーターが重要になる場合があります。一部の照会アクセラレーターでは、行セット・カーソルを使用して実行される照会はサポートされません。DSQSMRFI (MR) プログラム・パラメーターを YES に設定して QMF が開始された場合、QMF は行セット・カーソルを使用します。行セット・カーソルを使用して実行される照会が照会アクセラレーターでサポートされていない場合は、MR=NO を指定して QMF を開始してください。
- QMF では、XML データ・タイプをサポートするデータベース・リリースに接続されている場合にのみ、XML データを使用した操作がサポートされます。

関連タスク:

449 ページの『DSQEC_BUFFER_SIZE 大域変数設定による QMF パフォーマンスの向上』

DSQEC_BUFFER_SIZE 大域変数の値を大きくすると、QMF パフォーマンスは向上します。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

複数のデータベース・スレッドの使用

大きな応答セットを作成する、RUN QUERY コマンドに続くコマンドのパフォーマンスを向上させるために、または大きな応答セットに対して SAVE DATA コマンドを実行しているときの記憶域問題を避けるために、エンド・ユーザー SQL コマンドを個別のデータベース・スレッドで実行することができます。

このタスクについて

複数のスレッドを使用するために、オプションの第 2 スレッドが QMF プログラム・パラメーター DSQSMTHD (YES/NO、デフォルトは NO) によって制御されます。このプログラム・パラメーターによって、QMF 管理者は、QMF が追加のデータベース・スレッドを使用するか、またはデフォルト設定どおりに単一スレッドで動作するかを制御することができます。

第 2 スレッドを使用するエンド・ユーザー SQL には、以下の QMF コマンドが含まれます。

- RUN QUERY (TABLE キーワードが付いた RUN QUERY は除く)
- DISPLAY TABLE
- 報告書をスクロールするコマンド (BOTTOM、TOP、FORWARD、BACKWARD、RIGHT、および LEFT)
- 完全な報告書を必要とするコマンド (EXPORT DATA、SAVE DATA、PRINT REPORT、DPRE、CONNECT、ERASE TABLE)

DSQSMTHD が YES に設定されていると、DSQAO_CURSOR_OPEN 状態大域変数の値は、エンド・ユーザー SQL 第 2 スレッドの状況を反映します。

DSQSMTHD は、TSO のもとで動作している QMF (QMF CAF インターフェース) にも適用可能です。QMF が CICS のもとで動作している場合、または QMF がストアード・プロシージャ・インターフェースのもとで動作している場合、DSQSMTHD 設定は無視されます。

DSQSMTHD が YES に設定されていると、大域変数 DSQEC_RESET_RPT の効果 (不完全な報告書が未解決である場合にプロンプトを出す) は変わります。リモート・データベースに対する CONNECT、または同じデータベース内での新しいユーザー ID およびパスワードへの CONNECT を実行すると、第 2 スレッドも接続されます。オープンしている報告書オブジェクトがあれば、それらは CONNECT の前に完了していなければなりません。DSQEC_RESET_RPT を使用すると、この状態についてのアラートを受け取ることができます。

ERASE TABLE コマンドがユーザー・スレッドを使用していなければ、引き続き、不完全なデータ・オブジェクトが完了することが必要です。このことによって、デッドロックおよびタイムアウト状態が回避されます。

不完全な報告書 (DSQSMTHD=YES) のケースでは、報告書に対して COMMIT を実行 (ロックを解放) し、スレッドを削除するコマンドのリストは次のとおりです。

- RESET DATA
- RUN QUERY (SELECT ステートメント)
- PRINT TABLE
- DISPLAY TABLE
- EXPORT TABLE
- CONNECT
- ERASE TABLE

不完全なデータのプロンプト (DSQEC_RESET_RPT) も (RESET DATA の場合を除いて) 発生します。

注: ユーザーごとに 2 つのスレッドが使用されていると、一般的な DB2 リソース消費 (CTHREAD 制限など) に影響が及ぶ可能性があります。

QMF アクティビティの自動化

DSQSMODE パラメーターおよび DSQSRUN パラメーターにより、QMF アクティビティを自動化して、リソースおよび時間を節約することができます。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

操作モードの指定 (対話式またはバッチ)

プログラム・パラメーター DSQSMODE を使用して、QMF での作業を実行するバッチ・セッションを開始することで、リソースおよび時間を節約できます。トランザクションの実行中、システムは他の作業のために解放されます。

始める前に

QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合は、このパラメーターを指定しないでください。バッチ・モードを示す値 B は、インターフェースを開始する DSQSCMDn EXEC 内に既に指定されています。

このタスクについて

ユーザーが実行する必要がある一部の照会および報告書作成タスクには、QMF との対話を必要としないものがあります。例えば、ある販売員が、一連の表で会計状況を照会するために、数日ごとに同じ QMF プロシージャーを使用する場合です。データは変わっても、そのデータへのアクセスに必要なプロシージャーとタスクは同じままです。

手順

QMF をバッチ・モードで開始するには、以下の手順に従ってください。

1. DSQSMODE パラメーターに値 B を指定します。

QMF の対話式操作を指定するには、DSQSMODE パラメーターに値 I を使用します。これはデフォルト値です。

2. バッチ・セッションでは QMF パネルが表示されないため、DSQSRUN パラメーターを使用して、必要な作業を行ってから、プログラムを終了する初期プロシージャーを実行します。

例えば、次のコマンドは英語でバッチ・セッションを開始し、ユーザー JONES が所有する STARTPROC という名前の初期プロシージャーを実行します。

```
DSQQMFE M=B,I=JONES.STARTPROC
```

3. 必要な場合は、DSQSDBNM パラメーターを追加して、接続する初期データベースを指定します (デフォルトのデータベース・ロケーションを使用したくない場合)。

関連概念:

215 ページの『QMF の開始時に実行する初期プロシージャーの指定』

DSQSRUN パラメーターは、QMF の開始直後に実行する QMF プロシージャーの名前を渡してから、QMF を終了します。

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

QMF の開始時に実行する初期プロシージャの指定

DSQSRUN パラメーターは、QMF の開始直後に実行する QMF プロシージャの名前を渡してから、QMF を終了します。

例えば、英語で QMF を開始して、STARTPROC という名前の初期プロシージャを実行するには、次のコマンドを発行します。

```
DSQQMFE I=STARTPROC
```

他のユーザーがプロシージャ名を使用して QMF を開始する場合は、そのプロシージャ名をその所有者の SQL 許可 ID で修飾します。例えば、ユーザー JONES が STARTPROC プロシージャを所有している場合は、次のように入力します。

```
DSQQMFE I=JONES.STARTPROC
```

初期プロシージャの名前を渡すと、QMF は RUN PROC コマンドを発行します。このコマンドにより、指定した名前のプロシージャが実行されます。

重要: QMF では、ユーザー ID とプロシージャ名構文にブランクを入れることはできません。例えば、QMF は以下のプロシージャ名を認識しません。

```
DSQQMFE I=JONES. STARTPROC
```

埋め込みブランク付きのプロシージャ名を使用するには、以下のように、その名前を引用符で囲む必要があります。

```
DSQQMFE I=JONES.'START PROC'
```

- 以下の場合に DSQSRUN パラメーターを使用できます。
- バッチ・モードで QMF のアクティビティを自動化して、対話式で実行する場合に通常使用されるリソースを節約する場合。
- ユーザーが事前定義されたプロシージャの範囲内で対話式 QMF 作業を実行できるようにし、その後そのプロシージャで指定された作業の完了時に QMF を終了する場合。

QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始する場合は、このパラメーターを指定しないでください。この方法で QMF を開始する場合は、CALL ステートメントを使用して、QMF の開始時に実行する初期照会の名前、またはプロシージャ名を渡します。

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

バッチ・モードでの初期プロシージャーの実行

プロシージャーをバッチ・モードで実行できるようにするには、DSQSMODE パラメーターに値 B を使用し、DSQSRUN パラメーターを使用してプロシージャーの名前を指定します。

例えば、毎週月曜日の朝に、在庫状況報告書を作成する必要があるとします。毎週日曜日の夜に、INVENTORY という表の同じ列からデータを取り出す照会を実行する必要があります。その場合の照会は、以下のようになります。

```
SELECT * FROM INVENTORY
WHERE STOCK < 20
```

この例では、この照会の名前を INVENTORY_QUERY とします。この照会を実行して状況報告書を印刷する場合に使用するプロシージャーは、以下のようになります。これらの例について、この QMF プロシージャーを INVENTORY_PROC と呼びびます。

```
RUN QUERY INVENTORY_QUERY
PRINT REPORT
EXIT
```

このプロシージャーには EXIT コマンドが含まれます。これは、QMF をバッチ・モードで実行する際に QMF セッションを終了するユーザーがいないためです。EXIT により QMF セッションは終了し、QMF で保持されていたリソースは解放されます。バッチ・モードで実行される初期プロシージャー内には、常に EXIT コマンドを使用してください。

報告書の作成に伴うタスクは変更されない (変更されるのはデータのみ) ため、以下のように DSQSRUN パラメーターを使用して、日曜日の夜に勤務時間外の INVENTORY 表を照会し、報告書を印刷できます。

```
QMFn I=INVENTORY_PROC,M=B
```

初期プロシージャーを使用した対話式 QMF 作業の実行

対話式 QMF セッションで初期プロシージャーを使用して、エンド・ユーザー用の QMF タスクを事前定義できます。これにより、エンド・ユーザーは必要なデータのみ簡単にアクセスできるようになります。

例えば、ある QMF エンド・ユーザーが毎週月曜日の朝に在庫状況報告書を作成することになっているとします。このユーザーは、在庫が少ないことを示す値は分かりませんが、状況報告書の作成方法がよく分かりません。このような場合、このユーザーが在庫が少ないことを示す値のみを入力すれば済むような変数を照会に入れることができます。ここでは、この照会を INVENTORY_QUERY と呼びます。

```
SELECT * FROM INVENTORY
WHERE STOCK < &LOWSTOCK
```

ユーザーが印刷前にデータの表示を必要とする可能性があるため、以下のように、INVENTORY_PROC プロシージャには EXIT コマンドが含まれていません。

```
RUN QUERY INVENTORY_QUERY
```

次に、以下のように DSQSMODE パラメーターを指定せずに DSQSRUN パラメーターを使用して、ユーザー用の対話式セッションが開始されるようにできます。

```
QMFn I=INVENTORY_PROC
```

INVENTORY_PROC プロシージャは、ユーザーに &LOWSTOCK 変数値を求めるプロンプトを出します。

対話式セッションでは、表示された報告書の確認が終わったらコマンド行で EXIT と入力するようにユーザーに指示してください。初期プロシージャは、EXIT コマンドが発行されるまで繰り返し実行されます。したがって、報告書パネルから終了ファンクション・キーを押すと、初期プロシージャが再実行され、QMF ホーム・パネルは表示されません。

また、DSQSRUN パラメーターを使用する場合は、DSQEC_RERUN_IPROC 大域変数が 0 に設定されていること、さらに、現行オブジェクトが QMF ホーム・パネルでないことを確認してください。

関連概念:

『初期プロシージャへの変数値の引き渡し』

DSQSRUN パラメーターで初期プロシージャの名前を指定する際、そのプロシージャに入れる 1 つ以上の変数の値も指定できます。

初期プロシージャへの変数値の引き渡し

DSQSRUN パラメーターで初期プロシージャの名前を指定する際、そのプロシージャに入れる 1 つ以上の変数の値も指定できます。

DSQSRUN パラメーターに変数を指定する際には、以下の規則に従ってください。

- 変数パラメーター・リストは括弧で囲む。
- 変数名の前にアンパーサンド (&) を付け、ストリングの形式を *variable_name=value* にする。
- プロシージャ名と変数パラメーター・リストを結合した文字数の合計を 98 文字以下にする。
- 複数の変数パラメーター指定は、単一コンマ、1 つ以上のブランク、またはコンマとブランクの組み合わせを使用して区切る。

下の表に、各環境で変数を使用する際に必要なアンパーサンドの数を示します。

表 35. プログラム変数の前に付ける必要のあるアンパサンドの数

| 環境 | 追加するアンパサンドの数 | 例 |
|-----------------------------|--------------|--------------------|
| CICS | 0 | &variable=value |
| ISPF を使用しない TSO | 0 | &variable=value |
| ISPF を使用した TSO | 1 | &&variable=value |
| ISPF を使用しない TSO (CLIST を使用) | 2 | &&&variable=value |
| ISPF を使用した TSO (CLIST を使用) | 3 | &&&&variable=value |

初期プロシージャの名前を指定すると、QMF は、このプロシージャを実行する RUN PROC コマンドを発行します。プロシージャに変数を使用する場合、これらの変数に指定する値は、RUN コマンドでの変数の引き渡しに使用される構文に準拠したものでなければなりません。

例えば、組織の従業員に関する情報について頻繁に必要となるのが 2 つの部分であるとして、1 つは従業員の名前で、もう 1 つは多様に変わります。このような場合は、NAME 列の値を戻して、あとは他の列の名前の変数を使用する照会を定義できます。下の図に、照会およびプロシージャの例を示します。また、この図には、DSQSRUN パラメーターの入力時に変数の値を渡す方法、およびその結果として QMF が発行する RUN PROC コマンドについても示されます。

```

照会 (JONES.QUERY2)
    SELECT NAME, &COL FROM Q.STAFF
プロシージャ (JONES.PROC2)
    RUN QUERY JONES.QUERY2 (&&COL=&COL)
DSQSRUN パラメーター
    QMFn I=JONES.PROC2(&COL=YEARS)
結果の RUN コマンド
    RUN PROC JONES.PROC2 (&COL=YEARS)

```

図 25. DSQSRUN パラメーターでの列名の引き渡し

下の図に似た例を示しますが、この例では、プロシージャに 1 つの列名を渡す代わりに、従業員の名前、部門、および従業員の給与を返す複数の列名を渡すことができます。

```
照会 (JONES.QUERY3)
      SELECT &COLS FROM Q.STAFF
プロシージャ (JONES.PROC3)
      RUN QUERY JONES.QUERY3 (&&COLS=&COLS)
DSQSRUN パラメーター
      QMFn I=JONES.PROC3(&COLS=((DEPT,NAME,SALARY)))
結果の RUN コマンド
      RUN PROC JONES.PROC3(&COLS=((DEPT,NAME,SALARY)))
```

図 26. DSQSRUN パラメーターでの複数の列名の引き渡し

次の 4 つの例では、通常は照会の WHERE キーワードの後に指定する情報の引き渡し方法を示します。

これらの例には文字ストリングが含まれています。これらのストリングには、RUN PROC コマンドの処理時に QMF が値を評価する方法のため、特殊構文が必要となります。また、ストリングには、例のように、特殊文字 (コンマ、ブランク、括弧、二重または単一引用符、アポストロフィ、および等号)を含めることもできます。

これら 4 つのうち最初の例は、ある組織のすべての管理者の名前および従業員番号を返す照会を示しています。DSQSRUN パラメーターで文字ストリング MGR を渡す場合は、必ずその値を単一引用符で囲みます。

```
照会 (JONES.QUERY4)
      SELECT JOB, NAME, ID FROM Q.STAFF WHERE JOB=&JOB
プロシージャ (JONES.PROC4)
      RUN QUERY JONES.QUERY4 (&&JOB=&JOB)
DSQSRUN パラメーター
      QMFn I=JONES.PROC4(&JOB='MGR')
結果の RUN コマンド
      RUN PROC JONES.PROC4 (&JOB='MGR')
```

図 27. DSQSRUN パラメーターでの文字ストリング (単一引用符で囲む) の引き渡し

4 つの例のうちの 2 番目は、コンマを含む変数値の引き渡し方法を示しています。値 SAN JOSE, CA は、コンマが含まれているため、単一引用符で囲みます。

照会 (JONES.QUERY5)
SELECT * FROM Q.APPLICANT WHERE ADDRESS=&CITY
プロシージャ (JONES.PROC5)
RUN QUERY JONES.QUERY5 (&CITY=&CITY)
DSQSRUN パラメーター
QMFn I=JONES.PROC5(&CITY='SAN JOSE, CA')
結果の RUN コマンド
RUN PROC JONES.PROC5 (&CITY='SAN JOSE,CA')

図 28. DSQSRUN パラメーターでのコンマを含むストリングの引き渡し

文字ストリングを使用した 4 つの例のうちの 3 番目は、単一引用符を含む変数値 (例えば名前のアポストロフィ) の引き渡し方法を示しています。DSQSRUN パラメーターでこの値を渡す場合は、必ず値を単一引用符で囲み、アポストロフィには 1 つではなく 2 つの単一引用符を使用してください。

照会 (JONES.QUERY6)
SELECT * FROM Q.STAFF WHERE NAME=&NAME
プロシージャ (JONES.PROC6)
RUN QUERY JONES.QUERY6 (&NAME=&NAME)
DSQSRUN パラメーター
QMFn I=JONES.PROC6(&NAME='O'BRIEN')
結果の RUN コマンド
RUN PROC JONES.PROC6 (&NAME='O'BRIEN')

図 29. DSQSRUN パラメーターでのストリングの一部としてのアポストロフィの引き渡し

4 つの例のうちの最後は、RUN コマンドでの複数の変数値の引き渡し方法を示しています。

照会 (JONES.QUERY7)
SELECT * FROM Q.STAFF WHERE DEPT IN &DEPT AND JOB = &JOB
プロシージャ (JONES.PROC7)
RUN JONES.QUERY7 (&DEPT=&V1 &&JOB=&V2)
DSQSRUN パラメーター
QMFn I=JONES.PROC7(&V1=(((10,38))) &V2='MGR')
結果の RUN コマンド
RUN PROC JONES.PROC7(&V1=(((10,38))) &V2='MGR')

図 30. DSQSRUN パラメーターでの複数の変数値の引き渡し

関連資料:

187 ページの『プログラム・パラメーターの要約』
QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

トレース・オプションの設定

QMF にはトレース機能が用意されており、QMF およびユーザーのアクティビティの追跡に役立ちます。

このタスクについて

DSQSDBUG パラメーターを使用して、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。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も含めて、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には、DSQSDBUG パラメーターに値 ALL を指定します。この値は、以下の 2 つの異なるタイプの初期化コマンドに示されています。コマンドの *n* を、使用中の言語に対応する 1 文字の各国語 ID で置き換えてください。

```
DSQQMF $n$  T=ALL  
QMF $n$  T=ALL
```

DSQSDBUG に NONE を設定した場合、トレース出力の詳細レベルは、QMF セッションの実行が対話式であるか、バッチ・モードであるかによって次のように異なります。

- 対話式セッションとバッチ・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。
- バッチ・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。

このパラメーターを用いて設定したトレース・レベルは、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。

関連概念:

457 ページの『トレース出力の適切な詳細レベルの設定』

すべての QMF 機能を詳細にトレースすることができ、また個々の QMF 機能をトレースすることができます。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

TSO のトレースの設定

QMF for TSO でトレースを設定するには、QMF の開始時に DSQSDBUG=ALL を指定します。

QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合は、DSQSDBUG パラメーターを指定しないでください。この方法で QMF を開始する場合は、CALL ステートメントを使用して、トレースの詳細レベルを指定する入力パラメーターを渡します。

QMF の開始後、必要に応じて 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。また、SET PROFILE コマンドを使用して、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。

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

454 ページの『トレース機能』

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。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

CICS のトレースの設定

QMF for CICS でのトレースを設定するには、いくつかのパラメーターを設定します。

手順

QMF for CICS でトレースを指定するには、以下の手順に従ってください。

1. QMF の開始時に DSQSDEBUG=ALL を指定する。
2. DSQSDBQT パラメーターを使用して、トレース・データに使用する記憶域のタイプを指定する。

- 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・レベルのトレースには、一時記憶域を使用します。ALL などの他のタイプのトレースでは、より多くのトレース・データが取り込まれるため、より多くの記憶域を必要とします。トレース出力が CICS の一時記憶域キューの最大サイズを超えると考えられる場合は、一時データ・キュー (値 TD) を使用してください。

- 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 以外の名前を指定する場合は、そのキューを初めて使用する前に、CICS DCT でそのキューを事前定義しておく必要があります。
3. DSQSDBQN パラメーターを使用して、トレース・データを保管する CICS 記憶域キューの名前を指定する。
- 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。
 - 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。例えば、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
```
 - 一時データ・キューは、DD 名に関連付ける必要があります。DSQDEBUG は、トレース・データ・セットのデフォルト名です。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。

タスクの結果

QMF の開始後、必要に応じて 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。また、SET PROFILE コマンドを使用して、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。

関連タスク:

454 ページの『トレース機能』

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。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

非 DBCS ディスプレイ装置からの DBCS データの印刷

大文字、日本語、または韓国語の各国語機能 (NLF) を使用する場合は、2 バイト文字セット (DBCS) データの印刷が必要になる場合があります。

手順

非 DBCS ディスプレイ装置から DBCS データを印刷するには、DSQSDBCS プログラム・パラメーターを YES に設定します。

例

サポート対象のユーザーが非 DBCS ディスプレイ装置を使用しており、非数値列に DBCS データが入った表 (表名 DBCSTABLE) を印刷する必要があるとします。以下のステートメントは、消去された CICS 画面から大文字の NLF を開始し、ユーザーが PRINT DBCSTABLE (PRINTER=*printername* などのコマンドを使用して DBCSTABLE を印刷できるようにします。

```
QMFU K=YES
```

関連概念:

273 ページの『第 13 章 印刷および図表作成機能のセットアップ』

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。

関連資料:

187 ページの『プログラム・パラメーターの要約』

QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターのクイック・リファレンスが用意されています。

第 11 章 ユーザーの登録と特権の設定

ユーザーが QMF をアクセスおよび使用できるようにするには、QMF アプリケーション・プランと、パッケージおよびデータベース・オブジェクトへのアクセス権限を付与する必要があります。また、ユーザー・プロファイルも作成する必要があります。ユーザーがオブジェクトをより効率的に処理できるように、ユーザーのオブジェクト・リストをカスタマイズすることもできます。

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制御

QMF へのアクセスを制御するには、QMF アプリケーション・プランおよびパッケージへのアクセス権限を必要に応じて付与および取り消し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

QMF アプリケーション・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。QMF にアクセスするには、すべての QMF ユーザーが QMF アプリケーション・プランおよびパッケージへのアクセス権限を持っていることが必要です。

インストール時に、EXECUTE 特権が QMF プラン (デフォルトで QMF12 と名前が付けられる) およびパッケージの PUBLIC に付与されます。PUBLIC から QMF プランのアクセスを取り消すことができます。ただし、次の例に示すように、その後各 QMF ユーザーに対して個々に SQL GRANT ステートメントを発行する必要があります。

```
GRANT EXECUTE ON PLAN qmfplanname TO userid
```

QMF へのアクセスを自分が作成した QMF プロファイル構造を介して制限することもできます。

関連概念:

『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取り消し』

QMF アプリケーション・プランおよびパッケージに対する GRANT ステートメントを取り消すには、SYSADM (または同等の) 権限を持っている必要があります。

233 ページの『サイトのためのプロファイル構造の確立』

ユーザーは、総称プロファイルまたは固有のプロファイルを使用して QMF にアクセスできます。必要に応じて、QMF アクセスを固有のプロファイルを持つユーザーに制限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取り消し

QMF アプリケーション・プランおよびパッケージに対する GRANT ステートメントを取り消すには、SYSADM (または同等の) 権限を持っている必要があります。

QMF アプリケーション・プランへのユーザーのアクセスを取り消すには、次のようなステートメントを発行できます。

```
REVOKE EXECUTE ON PLAN qmflplanname FROM userid
```

バージョン 12.1 のデフォルトの QMF プラン名は QMF12 です。

ユーザーの EXECUTE 特権が複数のユーザーによって付与されることが可能である場合は、次の SQL ステートメントを使用してその特権を取り消します。

```
REVOKE EXECUTE ON PLAN qmflplanname FROM userid BY ALL
```

これらの特権を PUBLIC から取り消しするには、ステートメント内で PUBLIC を *userid* と置き換えます。

除去されるユーザーが元は QMF 管理者で、QMF プランおよびパッケージへのアクセス権を他のユーザーに付与していた場合、この元管理者からアクセス権を除去すると、他のユーザーに付与されていたアクセス権も除去されます。

QMF ユーザー・プロファイルの作成

すべての QMF ユーザーは、ユーザー・プロファイルにアクセスする必要があります。このユーザー・プロファイルは、特定のユーザーからの個々の入力を 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 (例えば、プリンター出力の宛先、または入力の大文字変換の有無など) を制御します。

Q.PROFILES 表について

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。

下の表に、Q.PROFILES コントロール表の列を示します。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。リストされているデフォルトは、英語 QMF 環境のためのものです。

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。例えば、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 Q.PROFILES 表を参照してください。

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。

QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始した場合、QMF を開始する CALL ステートメント上で指定した入力パラメーターは、表に示されているように QMF プロファイル内のいくつかの設定をオーバーライドします。

表 36. Q.PROFILES 表の構造

| 列名 | データ・タイプと長さ | NULL の 使用 | 機能および使用できる値 |
|---------|--|--------------|--|
| CREATOR | VARCHAR(128) 例外: CREATOR 列は、DB2 (VSE および VM 版) データベースに接続されている場合は CHAR(8) として定義されます。 | 不可 | <p>機能</p> <p>プロファイルを所有するユーザーの ID を示す。</p> <p>値 SYSTEM (デフォルト)、DB2 1 次許可 ID または SQL 許可 ID、または TSO ログオン ID (DSQSPRID パラメーターが TSOID に設定されている場合)。</p> <p>SYSTEM 行は、英語および各 NLF ごとに Q.PROFILES 表と共に出荷される。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> <p>QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始した場合、CREATOR は、Q.DSQQMFSP プロシージャが実行する WLM 管理のアドレス・スペースを開始する SYSTEM または許可 ID のいずれかでなければなりません。その他の値は、QMF 初期化エラーの原因となります。</p> |
| CASE | CHAR(18) | 可 | <p>機能</p> <p>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。</p> <p>TSO において、大/小文字混合パスワードでの RACF サポートを利用する場合は、値 MIXED を使用する。これを使用しないと CONNECT コマンドは失敗する。この場合、QMF は大文字のコマンドのみを認識するため、QMF ユーザーは、すべての入力データを大文字で入力するようにする。</p> <p>これらの値について詳しくは、DB2 QMF 解説書を参照してください。NLF ユーザーの場合、CASE は別のデフォルトを持っていることがある。</p> |
| DECOPT | CHAR(18) | 可 | <p>機能</p> <p>数字報告書列に QMF が使用す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 解説書を参照してください。NLF ユーザーの場合、DECOPT は変換されて別のデフォルトを持っていることがある。</p> |

表 36. Q.PROFILES 表の構造 (続き)

| 列名 | データ・タイプと長さ | NULL の 使用 | 機能および使用できる値 |
|----------|------------|--------------|---|
| CONFIRM | CHAR(18) | 可 | <p>機能</p> <p>CONFIRM をサポートする QMF コマンドで、確認プロンプトのデフォルトの処置を指定する。コマンドで CONFIRM オプションを指定しない場合は、このデフォルトが適用されます。CONFIRM の値は、QMF が次のことを行うコマンドを完了する前に確認パネルを表示するかどうかを示します。</p> <ul style="list-style-type: none"> データベース内のオブジェクトを変更または置き換える。 DB2 リソース限定機能 (DB2 ガバナー) に定義されている割り振られたリソースを超えるコスト予測がある。 ファイルまたはデータ・セットを置き換える (例えば、EXPORT コマンドの場合) <p>値</p> <p>YES (デフォルト) 確認パネルを表示します。このパネルで、コマンドを実行する前にコマンドを取り消す機会が提供されます。</p> <p>値 YES は、QMF for TSO が DB2 for z/OS ストアード・プロシージャーとして開始されている場合は無視されます。</p> <p>NO 確認パネルを表示せずにコマンドを実行します。</p> <p>データベースを変更する複数の SQL ステートメントを含む照会を実行すると、1 つの確認パネルが表示されます。このプロンプトに対する対応として提供した応答は、照会内のすべての SQL ステートメントによって作成される変更に対して適用されます。</p> |
| WIDTH | CHAR(18) | 可 | <p>機能</p> <p>1 ページ当たりの印刷列数を制御する。</p> <p>値 22 から 999。デフォルト値は 132 です。</p> |
| LENGTH | CHAR(18) | 可 | <p>機能</p> <p>1 ページ当たりの印刷行数を制御する。</p> <p>値 1 から 999、または改ページをしない場合は CONT。デフォルトは 60 です。</p> |
| LANGUAGE | CHAR(18) | 可 | <p>機能</p> <p>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> |

表 36. Q.PROFILES 表の構造 (続き)

| 列名 | データ・タイプと長さ | NULL の 使用 | 機能および使用できる値 |
|-------|------------|--------------|---|
| SPACE | CHAR(50) | 可 | <p>機能</p> <p>SAVE DATA または IMPORT コマンドで作成した表のデータベース内のデフォルトの記憶域スペースを指定する。</p> <p>値 現行データベース・ロケーションの有効な記憶域スペース名です。データベースのタイプによって、データベース名、表スペース名、またはデータベース名と表スペース名の組み合わせになります。以下の保管場所を指定できます。</p> <p>以下の保管場所を「値」に指定できます。</p> <p>databaseName.spaceName</p> <p>データベースおよび表スペースの両方で QMF が特定の表スペースに表を保存するように指定します。</p> <p>例えば、次の照会は、データベース DBASE1 の表スペース TSPACE1 に表を保存するように、SMITH という QMF ユーザーのプロファイルを構成します。</p> <pre>UPDATE Q.PROFILES SET SPACE='DBASE1.TSPACE1' WHERE CREATOR='SMITH' AND TRANSLATION='ENGLISH'</pre> <p>ただし、すべてのユーザーが同じ表スペースを使用している場合は、リソースの競合が発生する可能性があります。</p> <p>DATABASE "databaseName"</p> <p>DATABASE キーワードの後にデータベース名を引用符で囲んで指定すると、各表は DB2 によってその表用に暗黙的および排他的に作成された別々の表スペースに作成されます。このような暗黙的な表スペースは、デフォルトの LOCKSIZE、BUFFERPOOL、STOGROUP、およびスペース属性を持ち、作成された表の名前と一致する名前を持ちます。</p> <p>例えば、次の Q.PROFILES 表の SPACE フィールドの値は、各表を DSQDBDEF データベース内の別々の表スペースに保存します。</p> <pre>DATABASE "DSQDBDEF"</pre> <p>各表スペースを 1 つの表に制限すると、リソースの競合が最小限になります。パーティション化された表スペース・スキームおよびユニバーサル表スペース・スキームのどちらも、表スペース当たり 1 つの表の制限を課しません。</p> <p>NULL または空白</p> <p>QMF プロファイルの SPACE フィールドが NULL または空白の場合は、SAVE DATA または IMPORT TABLE コマンドがデフォルトで DSNDB04 データベース内の暗黙的に生成された表スペース名に表を作成します。</p> |

表 36. Q.PROFILES 表の構造 (続き)

| 列名 | データ・タイプと長さ | NULL の 使用 | 機能および使用できる値 |
|---------|------------|--------------|--|
| TRACE | CHAR(18) | 可 | <p>機能</p> <p>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。デフォルト値は、DSQSMODE パラメーターに指定された値により異なる。例えば、DSQSMODE パラメーターが B に設定されているときは、トレース・レベルは L2 であり、それ以外の場合は、NONE である。</p> <p>NLF で変換される値は ALL と NONE だけである。</p> <p>QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始した場合、トレース・オプションの値は QMF を開始する CALL ステートメントの trace-level パラメーターから取得されます。SET PROFILE (TRACE コマンドを object-name パラメーターで指定される初期プロシージャー内に含めて、トレース出力が DSQDEBUG データ・セットに出力されるように設定されている限り、ストアード・プロシージャー・セッションの期間にトレース詳細のレベルを変更できます。trace-level が L2 に設定されている場合、SET PROFILE (TRACE コマンドが受け入れられるには、<i>L2-destination</i> は、DSQDEBUG に設定されている必要があります。</p> <p>詳しくは、457 ページの『トレース出力の適切な詳細レベルの設定』および 459 ページの『個々の QMF モジュールのトレース』を参照してください。</p> |
| PRINTER | CHAR(8) | 可 | <p>機能</p> <p>プリンター出力の経路を制御する。</p> <p>値 NULL (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。GDDM によって定義されたプリンターに出力を送信するには、GDDM ニックネームを使用します。</p> <p>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> <p>QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始して、レポート出力を結果セットで受け取るようにする場合は、このプロファイル・オプションをブランクのストリングとして設定するか、または QMF の開始後に実行する初期プロシージャーでブランクのストリングを SET PROFILE コマンド上に渡すようにする必要があります。</p> |

表 36. Q.PROFILES 表の構造 (続き)

| 列名 | データ・タイプと長さ | NULL の 使用 | 機能および使用できる値 |
|-------------|--|--------------|---|
| TRANSLATION | CHAR(18) | 不可 | <p>機能</p> <p>使用されている言語が、英語または各国語機能 (NLF) であるかを示す。</p> <p>値 英語 (デフォルト) または QMF が NLF 用に使用する名前。</p> <p>QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始した場合、言語値は、QMF を開始する CALL ステートメントに渡される language パラメーターから取得され、プロファイルからは取得されません。</p> <p>NLF について詳しくは、136 ページの表 27 を参照してください。</p> |
| PFKEYS | VARCHAR(261) 例外: PFKEYS 列は、DB2 (VSE および VM 版) データベースに接続されている場合は CHAR(31) として定義されます。 | 可 | <p>機能</p> <p>ユーザーのカスタマイズされたファンクション・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。この値がブランクまたは NULL (デフォルト) の場合は、デフォルトのキーが使用される。</p> |
| SYNONYMS | VARCHAR(261) 例外: SYNONYMS 列は、DB2 (VSE および VM 版) データベースに接続されている場合は CHAR(31) として定義されます。 | 可 | <p>機能</p> <p>ユーザーのカスタマイズ済みコマンド定義が保管されている表または視点を指定する。ユーザーがコマンド同義語を入力する場合、同義語定義は、ここで指定された表に保管されている必要があります。保管されていない場合、コマンドは失敗します。</p> <p>値 有効な DB2 表名または視点名のいずれか。</p> <p>ここで指定した表は、299 ページの『コマンド同義語のカスタマイズ』で説明されている構造に適合している必要があります。デフォルトの同義語を使用するか、同義語の独自の表を作成するか、独自の表を作成して QMF 提供の同義語を追加することができます。</p> <p>デフォルトの QMF コマンド同義語は Q.COMMAND_SYNONYMS 表に保管されます。NLF ユーザーの場合、デフォルトの同義語は Q.COMMAND_SYNONYM_# 表に保管されます。# は 136 ページの表 27 の 1 文字の言語 ID です。</p> <p>この値がブランクまたは NULL (デフォルト) の場合は、カスタマイズされた定義は使用されない。</p> |

表 36. Q.PROFILES 表の構造 (続き)

| 列名 | データ・タイプと長さ | NULL の 使用 | 機能および使用できる値 |
|----------------|------------|--------------|--|
| RESOURCE_GROUP | CHAR(16) | 可 | <p>機能</p> <p>デフォルトの管理プログラム出口ルーチンがどのようにユーザーのリソースまたはコマンドを制限するかを制御する。</p> <p>値 任意の有効なりソース・グループ名。この値がブランクまたは NULL (デフォルト) の場合、QMF は、ユーザーの 1 次許可 ID または SQL 許可 ID を DSQSPRID プログラム・パラメーターの値に従ってここで使用しようとする。DSQSPRID が TSOID に設定されている場合は、QMF は、TSO ログオン ID を RESOURCE_GROUP 列の値として使用しようとする。DSQSPRID が PRIMEID に設定されている場合は、QMF は、DB2 または SQL 1 次許可 ID を RESOURCE_GROUP 列の値として割り当てようとする。これらのケースでは、ユーザーのセッションは管理しない (ただし、TSO ID または 1 次許可 ID または SQL 許可 ID が有効なりソース・グループ名の場合を除く)。</p> <p>ユーザーの QMF セッションに使用中のリソース・グループは、DSQAP_RESOURC_GRP 大域変数に記録されません。</p> |
| MODEL | CHAR(8) | 可 | <p>機能</p> <p>データ・アクセスのモデルを指定する。</p> <p>値 この列には、リレーショナル・データを示して、常に値 REL を使用する。</p> |
| ENVIRONMENT | CHAR(8) | 可 | <p>機能</p> <p>操作環境を指定する。</p> <p>値 新しいインストールの有効な値は、TSO または CICS。デフォルト値は TSO。 Q.PROFILES 表が旧リリースから移行済みの場合は、この列内に TSO または CICS に加えて次の値が表示されます。</p> <p>CMS</p> <p>QMF バージョン 7.2 以前で使用されます。</p> <p>CICSMVS</p> <p>QMF バージョン 7.2 以前で使用され、z/OS での QMF インストール済み環境を VSE でのインストール済み環境と区別します。</p> <p>CICSVSE</p> <p>QMF バージョン 7.2 以前で使用され、VSE での QMF インストール済み環境を z/OS でのインストール済み環境と区別します。</p> <p>WINDOWS</p> <p>QMF for Workstation および QMF for WebSphere® で使用されます。</p> |

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

サイトのためのプロファイル構造の確立

ユーザーは、総称プロファイルまたは固有のプロファイルを使用して QMF にアクセスできます。必要に応じて、QMF アクセスを固有のプロファイルを持つユーザーに制限することができます。

総称プロファイル

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ことができます。

Q.PROFILES 表は、この行に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。お客様サイトの要件を満たす総称プロファイルを作成するために、デフォルト値を変更することができます。

固有のプロファイル

総称 QMF プロファイルを保持しないことを選択した場合は、各ユーザーについて Q.PROFILES 表に固有の行を作成する必要があります。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。

サイト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、他のユーザーには総称プロファイルを使用できるようにすることができます。

固有のプロファイルを持つユーザーに対する QMF 使用の制限

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。以下は、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。SYSTEM 行をドイツ語バージョンの Q.PROFILES 表から削除するステートメントは、NLF の行を削除する方法の例として提供されています。TRANSLATION 列の値には、QMF が NLF に使用する名前を使用してください。

重要: 英語環境の場合も 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。

表 37. QMF の使用を、固有のプロファイルを持つユーザーに制限

| 基本 QMF (英語) | ドイツ語 NLF |
|---|---|
| DELETE FROM Q.PROFILES WHERE CREATOR='SYSTEM' AND TRANSLATION='ENGLISH' | DELETE FROM Q.PROFILES WHERE CREATOR='SYSTEM' AND TRANSLATION='DEUTSCH' |

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する必要があります。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。

関連タスク:

『ユーザー・プロファイルの追加』

作成する固有のユーザー・プロファイルごとに、Q.PROFILES 表内に行を追加します。

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

関連資料:

226 ページの『Q.PROFILES 表について』

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。

ユーザー・プロファイルの追加

作成する固有のユーザー・プロファイルごとに、Q.PROFILES 表内に行を追加します。

このタスクについて

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。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=userid、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

QMF は、先行するリストのペアの 1 つに一致する CREATOR と ENVIRONMENT のための値を見つける必要があります。見つからない場合、QMF 初期化はエラーとなって、QMF ホーム・パネルを表示せずに終了します。

Q.PROFILES 表の CREATOR 列内の値が、DB2 1 次許可 ID ではなく TSO ログオン ID である場合は、QMF の開始時に DSQSPRID パラメーターを TSOID に設定してください。これを行わないと、ユーザーが QMF にログオンしようとしたときに Q.PROFILES 表での認証が失敗します。

手順

ユーザー・プロファイルを Q.PROFILES 表に追加するには、以下の手順に従ってください。

1. 行を追加する SQL INSERT ステートメントを作成します。

ヒント: メンテナンス・タスクをより効率的に実行するために、単一の SQL 照会は複数の SQL メンテナンス・ステートメントを含むことができます。複数のステートメントを含む照会を作成するには、ステートメントの間にセミコロンを入れて、DSQEC_RUN_MQ 大域変数を 1 に設定します。複数のステートメントを含む照会の作成方法、および使用可能なステートメント・タイプの制約に関する情報は、RUN コマンドの説明に記載されています。また、標準プロファイルの説明するテンプレート照会を作成し、あるプロファイルから別のプロファイルへと変わる値 (CREATOR 列の値など) に置換変数を使用することもできます。

2. 少なくとも、表の CREATOR 列、ENVIRONMENT 列、および TRANSLATION 列の値を以下のように指定します。

- CREATOR

総称 QMF プロファイルを作成している場合は、値 SYSTEM を指定します。固有のプロファイルを作成している場合は、ユーザーの許可 ID を指定します。データベースは、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。

- ENVIRONMENT

CICS または TSO の値を指定してください。TSO の値は、TSO、ISPF、またはネイティブ z/OS を含みます。

- TRANSLATION

使用している NLF に ENGLISH または翻訳された値を指定してください。例えば、フランス語ユーザーのプロファイル行は、TRANSLATION 列に FRANCAIS を含みます。

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。

同じユーザーに対して複数の NLF プロファイルを確立することができます。例えば、ある各国語について 1 組のプロファイル値を、別の各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。

3. RUN QUERY コマンドを発行してプロファイル行を追加します。

例

下の表は、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)、および SCHMIDT という SQL 許可 ID を持つユーザー (ドイツ語 NLF) のための固有プロファイルを TSO 環境に作成する SQL の例を示しています。

表 38. TSO でのユーザー・プロファイルの作成

| 基本 QMF (英語) | ドイツ語 NLF |
|---|--|
| <pre>INSERT INTO Q.PROFILES (CREATOR, LANGUAGE, SPACE, TRANSLATION, PFKEYS, SYNONYMS, RESOURCE_GROUP, ENVIRONMENT) VALUES ('JONES', 'PROMPTED', 'SAVEIT' 'ENGLISH', 'PFKEYS', 'COMMAND_SYNONYMS' 'NONPRIME', 'TSO')</pre> | <pre>INSERT INTO Q.PROFILES (CREATOR, LANGUAGE, SPACE, TRANSLATION, PFKEYS, SYNONYMS, RESOURCE_GROUP, ENVIRONMENT) VALUES ('SCHMIDT', 'MENUE', 'STUTZBER' 'DEUTSCH', 'DEUTASTEN', 'COMMAND_SYNONYM_D', 'SCHICHT', 'TSO')</pre> |

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

関連資料:

226 ページの『Q.PROFILES 表について』

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。

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

SET PROFILE を使用すると、QMF コマンド行から最小限の入力を行うだけでコマンドを入力できるため、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。変更した値を保存するには、SAVE PROFILE コマンドを使用してください。

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。

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。

ヒント: メンテナンス・タスクをより効率的に実行するために、単一の SQL 照会は複数の SQL メンテナンス・ステートメントを含むことができます。複数のステートメントを含む照会を作成するには、ステートメントの間にセミコロンを入れて、DSQEC_RUN_MQ 大域変数を 1 に設定します。複数のステートメントを含む照会の作成方法、および使用可能なステートメント・タイプの制約に関する情報は、RUN コマンドの説明に記載されています。

既存のユーザー・プロファイルを更新するには、下の表に示すような SQL UPDATE ステートメント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 QMF (英語) でのユーザー JONES の場合のステートメントの例で、右側はドイツ語 NLF でのユーザー SCHMIDT の場合の同じステートメントの例です。

表 39. UPDATE ステートメントを使用して Q.PROFILES 表のユーザー・プロファイルを更新

| 基本 QMF (英語) | ドイツ語 NLF |
|--|--|
| UPDATE Q.PROFILES SET SYNONYMS='COMMAND_SYNONYMS' WHERE CREATOR='JONES' AND TRANSLATION='ENGLISH' | UPDATE Q.PROFILES SET SYNONYMS='GUMMOW.XYZ' WHERE CREATOR='SCHMIDT' AND TRANSLATION='DEUTSCH' |

重要: Q.PROFILES 表で、UPDATE、DELETE、および INSERT ステートメントを実行する場合は、常にそのステートメントに TRANSLATION 列を含めてください。含めない場合、QMF は指定した変更をすべての言語環境に適用します。

SYSTEM プロファイルの更新

Q.PROFILES 表の 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。

例えば、サイトに 2 つのユーザー・グループがあるとします。PRIME グループは、大部分の QMF ユーザーを含み、全員が通常の勤務時間中にシステムを使用しています。NONPRIME グループは、少数のユーザーを含み、全員が勤務時間外にシステムを使用しています。PRIME は、Q.PROFILES の SYSTEM 行の RESOURCE_GROUP フィールドのデフォルト値であるとして、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。

ユーザー・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、そのプロファイルに関連付けられている 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、他のユーザーに移行されていることが確かな場合のみです。

ユーザー・プロフィールを削除するには、以下のような DELETE ステートメントを使用します。いくつかの DELETE ステートメントを SQL 照会に含めて保守タスクの速度を上げることができます。

表 40. QMF ユーザー・プロフィールの削除

| 基本 QMF (英語) | ドイツ語 NLF |
|--|--|
| DELETE FROM Q.PROFILES WHERE CREATOR='JONES' AND TRANSLATION='ENGLISH' | DELETE FROM Q.PROFILES WHERE CREATOR='SCHMIDT' AND TRANSLATION='DEUTSCH' |

重要: 各 DELETE ステートメントが単一の NLF 環境でユーザー・プロフィールを削除したい場合は、TRANSLATION 列に値を指定します。TRANSLATION 列に値を指定しないと、QMF はすべての言語環境のプロフィールを削除します。

プロフィー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存を必要とするものが何も入っていないければ、DROP TABLESPACE 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。

関連情報:

 DB2 資料

DB2 のカスタマイズに関する情報を検索します。

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付与

照会やプロシージャなどの 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。

このタスクについて

アクセスする必要がある QMF およびデータベース・オブジェクトをユーザーが操作できるようにするには、それらに対するアクセス権をユーザーに付与する必要があります。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。

QMF コマンドおよび機能に必要な特権

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するプロセスは、各ユーザーが実行する必要があるタスクを判別することにより開始されます。続いて、QMF 照会およびコマンドを実行し、QMF 機能を実行するために、ユーザーに必要な SQL 特権を確認します。

QMF を使用するために必要な特権について詳しくは、以下のセクションを参照してください。

- 239 ページの『QMF コマンドに必要な SQL 特権』
- 240 ページの『指示照会および QBE 照会に必要な SQL 特権』

- 240 ページの『表エディターに必要な SQL 特権』

QMF コマンドに必要な SQL 特権

次の表を使用して、ユーザーが必要な 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 41. 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に必要な SQL 特権 |
|---|---|
| DISPLAY <i>tablename/viewname</i> | SELECT |
| DRAW <i>tablename/viewname</i> | SELECT |
| EDIT TABLE <i>tablename/viewname</i> | 必要な特権は表エディター・モードによって異なる。 |
| EXPORT TABLE <i>tablename/viewname</i> | SELECT |
| IMPORT TABLE <i>tablename/viewname</i> | <p>表がある場合は、SELECT、DELETE、および INSERT 特権が必要。コメントを組み込むには、表の所有権か、または表が常駐するデータベースに対する DBADM (または同等の) 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、ユーザーはデータベースに対する DBADM 特権または同等の特権、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、または表が作成されるデータベースに対する CREATETAB 特権を持っている必要がある。</p> <p>CICS では、IMPORT コマンドの使用を控えめにします。これは、同じアドレス・スペース内のほかのユーザーの QMF パフォーマンスに影響を与える可能性があるからです。QMF は、QSAM で操作をするときに GET/PUT サービスを使用します。このサービスでは入出力操作中に、同じ CICS 領域内の他の QMF ユーザーをロックアウトしてしまう可能性があります。</p> |
| PRINT <i>tablename/viewname</i> | SELECT |
| RUN 照会 | 照会で使用される任意の特権。 |
| RUN プロシージャ | プロシージャ内のコマンドで使用される任意の特権。 |
| SAVE DATA | <p>表がある場合は、その表に対する SELECT、DELETE、および INSERT 特権が必要。</p> <p>表がない場合は、データベースに対する CREATETAB 特権か DBADM (または同等の) 権限、または関連付けられているユーザー・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。</p> <p>コメントを組み込むには、表の所有権か、または表が常駐するデータベースに対する DBADM (または同等の) 権限のどちらかを持っている必要がある。</p> |
| LIST <i>tablename/viewname</i> | SELECT |

指示照会および QBE 照会に必要な SQL 特権

下の表を使用して、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 42. QMF 照会のタイプおよびそれに対応する SQL

| 照会タイプ | 照会に必要な SQL 特権 |
|--------|---------------|
| プロンプト | SELECT |
| QBE I. | INSERT |
| QBE P. | SELECT |
| QBE U. | UPDATE |
| QBE D. | DELETE |

表エディターに必要な SQL 特権

次の表を使用して、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 43. 表エディター・コマンドおよびそれと対応する SQL

| 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 |

関連概念:

256 ページの『ユーザーへの表の作成権限の付与』

表を作成するユーザーに付与する必要がある権限にはさまざまなものがあります。QMF ユーザー・プロファイルの SPACE フィールドにデータベースおよび表スペース名を指定する場合は、1 つの特権セットが必要です。データベースのみを指定して値を指定しない場合は、それとは別の特権セットが必要です。

関連情報:

 DB2 資料

SELECT、INSERT、UPDATE、または DELETE などの SQL 特権に関する情報と、表をインポートするために必要な特権および権限に関する情報を検索します。

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所有し、データベースに保存します (他の所有者を持つ表を作成する場合を除く)。オブジェクト所有者は、これらのオブジェクトに対する SQL 特権を他のユーザーに付与したり取り消したりすることができます。

DBADM または同等の権限を持っていれば、彼らがこの権限を持っている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。DBADM または同等の権限を持っておらず、オブジェクトを所

有していない場合は、そのオブジェクトに対して持っている特権の付与オプションを持っていない限りなりません。(付与オプションは、GRANT ステートメントで WITH GRANT OPTION 文節を使用することによって指定されます)。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、書式名、および分析オブジェクト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。例えば、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。QMF セッション中、そのユーザーが次のようなコマンドを出すとして

```
RUN QUERY (FORM=FORMA
```

このコマンドで、次の SQL 照会が実行されます。

```
SELECT * FROM TABLEA
```

RUN コマンドでは、照会 JONES.QUERY が参照され、書式は、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。

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、DBADM 権限を持つ必要があります。

SQL GRANT ステートメント

SQL GRANT ステートメントを使用して、表または視点に SQL SELECT、UPDATE、INSERT、DELETE、およびその他の特権を付与することができます。

例えば、ユーザー JONES が表 ORDER_BACKLOG に対して表エディターの変更モードを使用する必要があるとします。ORDER_BACKLOG 表に対する UPDATE 特権を JONES に付与するには、次のステートメントを発行します。

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

WITH GRANT OPTION 節は、JONES が ORDER_BACKLOG 表に関して付与されたすべての SQL 特権を他のユーザーに付与できることを示します。

すべてのローカル 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。例えば、次のステートメントを使用して、ORDER_BACKLOG 表に対する INSERT 特権をローカル・データベース内のすべてのユーザーに付与し、これらの各ユーザーが他のユーザーに INSERT 特権を付与できるようにします。

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

分散データを使用可能にしている DB2 for z/OS サブシステムのローカル・ユーザーおよびリモート・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。例えば、次のステートメントでは、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。すべてのユーザーがサンプル表に対する SELECT 特権を持つように、同様のステートメントが、QMF のインストール中にすべての QMF サンプル表に対して実行されます。

ヒント: メンテナンス・タスクをより効率的に実行するために、単一の SQL 照会は複数の SQL メンテナンス・ステートメントを含むことができます。複数のステートメントを含む照会を作成するには、ステートメントの間にセミコロンを入れて、DSQEC_RUN_MQ 大域変数を 1 に設定します。複数のステートメントを含む照会の作成方法、および使用可能なステートメント・タイプの制約に関する情報は、RUN コマンドの説明に記載されています。または、頻繁に変更する照会の部分 (特権のタイプ、オブジェクト名、および許可 ID など) の代わりに QMF 変数を使用するテンプレート照会を作成することができます。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。

関連情報:

 DB2 資料

GRANT ステートメントに関する情報を検索します。

SQL REVOKE ステートメント

REVOKE ステートメントを発行して特権を除去します。

例えば、次のステートメントは、表 SMITH.TABLEA に対する、ユーザー BAKER の SELECT 特権を取り消します。

```
REVOKE SELECT ON TABLE SMITH.TABLEA FROM BAKER
```

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。

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。SQL 許可 ID が SYSADM または同等の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。SYSADM または同等の権限を使用することで、特権の付与元が複数の場合でも、その特権を取り消すことができます。例えば、BAKER が表 SMITH.TABLEA に対して SELECT 特権を持っているとします。QMF 管理者は、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。QMF 管理者には SYSADM 権限があるので、次のステートメントを実行することができます。

```
REVOKE SELECT ON TABLE SMITH.TABLEA FROM BAKER BY ALL
```

節 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。

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。

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、例えば、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。

データベース特権はカスケード構造になっています。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。

特権の逸失が多数のユーザーに波及する可能性があるのは特に、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。例えば、ある表に対する SELECT 特権が失われると、誰もプログラムを実行できません。

特権のカスケードによる問題が発生する可能性がより高くなるのは、多くのユーザーがデータベース特権を付与できる場合です。そのため、組織内のどのユーザーにこの作業の責任を持たせるかについては、慎重に検討してください。

ヒント: 一度に多くのユーザーから異なる特権を取り消す必要がある場合は、単一の SQL 照会に複数の REVOKE ステートメントを含めることができます。複数のステートメントを含む照会を作成するには、ステートメントの間にセミコロンを入れて、DSQEC_RUN_MQ 大域変数を 1 に設定します。複数のステートメントを含む照会の作成方法、および使用可能なステートメント・タイプの制約に関する情報は、RUN コマンドの説明に記載されています。または、頻繁に変更する照会の部分 (特権のタイプ、オブジェクト名、および許可 ID など) の代わりに QMF 変数を使用するテンプレート照会を作成することができます。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もできます。

関連情報:

 DB2 資料

REVOKE ステートメントに関する情報を検索します。

オブジェクトの標準の設定および非コミット読み取りの許可

QMF 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。そのため、そ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。非コミット読み取りを許可することは、分散環境において有益です。

他のユーザーとの QMF オブジェクトの共用

自分が所有する QMF オブジェクトに他の QMF ユーザーがアクセスできるように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、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。QMF Analytics for TSO では、「保管」ファンクション・キーを通じて使用可能な SHARE オプションによって、図表または統計仕様へのアクセスを可能にすることができます。図表または統計仕様は、分析オブジェクトとして保管されています。すべての QMF フォルダー・オブジェクトには、実際の QMF オブジェクト (照会、プロシージャー、書式、または分析オブジェクト) に関する情報が含まれています。実際の QMF オブジェクトは、そのオブジェクトの所有者および SHARE オプションを識別します。

標準的な名前を使用し、照会、書式、プロシージャー、表、および分析オブジェクトの目的を他のユーザーのために記述するコメントを付けるように、ユーザーに促してください。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。

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。例えば、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 31. QMF オブジェクトの共用

ANALYTIC オブジェクトを保管するには、QMF Analytics for TSO で「保管」ファンクション・キーを使用する必要があることに注意してください。

デフォルトは大域変数 DSQEC_SHARE によって定義されます。

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 SAVE コマンドを使用して、オブジェクトの共用状況を常に変更できます。

```
DISPLAY ORDER_QUERY  
SAVE QUERY AS ORDER_QUERY (SHARE=NO)
```

図 32. QMF オブジェクトの共用状況の変更

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大域変数 DSQEC_ISOLATION の値を指定することができます。

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)

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使用している場合は、ユーザーが表示するすべてのレポートは、レポートが表示された後にデータベースから削除されたデータを含むことがあります。

'1' 分離レベル CS (カーソル固定)

これがデフォルトです。カーソル固定を使用しているときには、QMF は、レポート内のデータに影響するすべてのデータベース・コマンドが完了するまでレポートを表示しません。

ユーザーのオブジェクト・リスト

QMF ユーザーは、定期的にデータベースに保存した表および視点をリストしたり、表または視点の用途または表または視点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。

ユーザーのオブジェクト・リストのカスタマイズ

DB2 は許可情報を SYSIBM.SYSTABAUTH 表から収集するため、表リストおよび列情報に関して QMF から提供されるデフォルト視点を使用すると、処理時間が長くな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追加のセキュリティ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を作成することを検討してください。

このタスクについて

ユーザーのオブジェクト・リストをカスタマイズするには、独自の視点を作成し、視点を使用するための SET GLOBAL コマンドを実行します。

次のタスク

デフォルトのオブジェクト・リスト機能をカスタマイズするのに加えて、1 次許可 ID または 2 次許可 ID のいずれかを許可する表および視点をリストする LIST コマンドに機能拡張をインストールできます。この機能拡張は、これらのオブジェクトをリスト結果内で表示したい場合に、これらのオブジェクトに対する PUBLIC 特権を付与する必要を削減します。

関連タスク:

147 ページの『拡張 LIST コマンド機能のインストール (z/OS のみ)』
拡張 LIST コマンド機能がインストールされている場合、表と視点に対する特権は PUBLIC の代わりに、ユーザーの 1 次または 2 次許可 ID にのみ付与する必要があります。この許可 ID は、OWNER=ALL オプションを LIST TABLES または LIST ALL コマンドに指定した場合にこのコマンドで生成されるオブジェクト・リストに含まれています。

ユーザー独自のリスト視点の作成

CREATE ステートメントを使用して、LIST コマンドまたは DESCRIBE コマンド用のユーザー独自の視点を作成します。

このステートメントでは、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。

```
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 33. 大域変数を使用したオブジェクト・リストのカスタマイズ

ヒント: メンテナンス・タスクをより効率的に実行するために、単一の SQL 照会は複数の SQL メンテナンス・ステートメントを含むことができます。複数のステートメントを含む照会を作成するには、ステートメントの間にセミコロンを入れて、DSQEC_RUN_MQ 大域変数を 1 に設定します。複数のステートメントを含む照会の作成方法、および使用可能なステートメント・タイプの制約に関する情報は、RUN コマンドの説明に記載されています。

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、QMF によって提供される、対応する視点と同じ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。
- OWNER、TNAME、TYPE、および CNAME には必ず値を指定します。これらの列を NULL にすることはできません。

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、SYSIBM.SYSTABLES または SYSIBM.SYSCOLUMNS (列情報の場合) 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、大域変数 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。

新しい視点をデフォルトの視点にする

デフォルトの視点の代わりに作成した視点を使用するには、SET GLOBAL コマンドを使用して適切な大域変数を新しい視点名に設定します。

以下に例を示します。

```
SET GLOBAL (DSQEC_TABS_LDB2 = QMFADM.LOCAL_DB2_TABLES
```

関連概念:

250 ページの『デフォルトの視点名を保管する大域変数』

LIST コマンドおよび DESCRIBE コマンドで使用されるデフォルト視点の名前は、QMF 大域変数に保管されます。

関連資料:

『コマンドに使用されるデフォルトの視点』

QMF は、インストール中にデフォルト視点を提供します。各視点がどのように使用されるかは、LIST または DESCRIBE コマンドを発行したかどうか、コマンドが送信された場所によって異なります。

QMF LIST および DESCRIBE コマンドのデフォルトの動作

ユーザーが表内の列に対して LIST TABLES コマンドまたは DESCRIBE コマンドを発行すると、QMF は一連のデータベース・カタログ表で定義されている視点から必要な情報を生成します。QMF 大域変数は、視点名を保管するために使用されます。

コマンドに使用されるデフォルトの視点

QMF は、インストール中にデフォルト視点を提供します。各視点がどのように使用されるかは、LIST または DESCRIBE コマンドを発行したかどうか、コマンドが送信された場所によって異なります。

視点を作成する SQL ステートメントは、各視点名の下に示されます。これらの視点定義は、ジョブ DSQ0BCTV 内でも見られます。これはインストール中にジョブ DSQ1BVW を実行すると実行されます。

デフォルトの視点は、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 の LIST コマンドまたは 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。

Q.DSQEC_TABS_LDB2L

視点 Q.DSQEC_TABS_LDB2L は、現在の場所 (現在接続しているデータベース) に送信された LIST コマンドで使用されます。この視点は、DB2 (VSE および VM 版) を除いたすべてのデータベース・タイプで使用されます。

この視点は、次の SQL ステートメントにより作成されます。これらのステートメントは、DB2 for z/OS データベースにのみ適用されます。DB2 for Linux, UNIX, and Windows および DB2 for iSeries 用の DDL はジョブ DSQ0BCTV にあります。

```
CREATE VIEW Q.DSQEC_TABS_LDB2L
  (OWNER,TNAME,TYPE,SUBTYPE,MODEL,RESTRICTED,REMARKS,
   CREATED,MODIFIED,LAST_USED,LABEL,LOCATION,OWNER_AT_LOCATION,
   NAME_AT_LOCATION)
AS SELECT
  VARCHAR(RTRIM(T.CREATOR)),
  T.NAME,'TABLE',T.TYPE
```

```

, ' ', ' ', ' ',
VARCHAR(RTRIM(SUBSTR(T.REMARKS, 1, 254))),
' ', ' ', ' ', ' ', VARCHAR(RTRIM(SUBSTR(T.LABEL, 1, 30)))
, SUBSTR(T.LOCATION, 1, 16),
VARCHAR(RTRIM(T.TBCREATOR)), T.TBNAME
FROM SYSIBM.SYSTABLES T
, ( SELECT DISTINCT
VARCHAR(RTRIM(TA.TCREATOR)), TA.TTNAME
FROM SYSIBM.SYSTABAUTH TA
WHERE TA.GRANTEETYPE=' '
AND TA.GRANTEE IN (
USER, CURRENT SQLID, CURRENT SCHEMA,
'PUBLIC', 'PUBLIC*' )
) AS UAT ("CREATOR", "NAME")
WHERE (T.CREATOR=UAT.CREATOR OR
T.CREATOR=CURRENT SCHEMA)
AND T.NAME=UAT.NAME
AND T.TYPE IN ('T', 'V', 'H');

```

Q.DSQEC_TABS_RDB2L

視点 Q.DSQEC_TABS_RDB2L は、LOCATION パラメーターを含む LIST コマンドで使用されます。LOCATION パラメーターを含む LIST コマンドを開始したり、送信することができるのは、DB2 for z/OS データベースのみです。

この視点は、次の SQL ステートメントにより作成されます。

```

CREATE VIEW Q.DSQEC_TABS_RDB2L
(OWNER, TNAME, TYPE, SUBTYPE, MODEL,
RESTRICTED, REMARKS, CREATED, MODIFIED,
LAST_USED, LABEL, LOCATION, OWNER_AT_LOCATION,
NAME_AT_LOCATION)
AS SELECT VARCHAR(RTRIM(T.CREATOR)), T.NAME, 'TABLE', T.TYPE
, ' ', ' ', ' ',
VARCHAR(RTRIM(SUBSTR(T.REMARKS, 1, 254))),
' ', ' ', ' ', ' ',
VARCHAR(RTRIM(SUBSTR(T.LABEL, 1, 30))),
SUBSTR(T.LOCATION, 1, 16),
VARCHAR(RTRIM(T.TBCREATOR)), T.TBNAME
FROM SYSIBM.SYSTABLES T
, ( SELECT DISTINCT
VARCHAR(RTRIM(TA.TCREATOR)), TA.TTNAME
FROM SYSIBM.SYSTABAUTH TA
WHERE TA.GRANTEETYPE=' '
AND TA.GRANTEE IN (
USER, CURRENT SQLID, CURRENT SCHEMA, 'PUBLIC*' )
) AS UAT ("CREATOR", "NAME")
WHERE (T.CREATOR=UAT.CREATOR OR
T.CREATOR=CURRENT SCHEMA)
AND T.NAME=UAT.NAME
AND T.TYPE IN ('T', 'V', 'H');

```

Q.DSQEC_COLS_LDB2L

視点 Q.DSQEC_COLS_LDB2L は、現在の場所 (現在接続しているデータベース) にある表に対して発行された DESCRIBE コマンドで使用されます。この視点は、DB2 (VSE および VM 版) を除いたすべてのデータベース・タイプで使用されます。

この視点は、次の SQL ステートメントにより作成されます。これらのステートメントは、DB2 for z/OS データベースにのみ適用されます。DB2 for Linux, UNIX, and Windows および DB2 for iSeries 用の DDL はジョブ DSQ0BCTV にあります。


```

CREATE VIEW Q.DSQEC_COLS_LDB2L
  ( OWNER, TNAME, CNAME, REMARKS, LABEL )
AS SELECT
  VARCHAR(RTRIM(C.TBCREATOR)),
  C.TBNAME,
  VARCHAR(RTRIM(SUBSTR(C.NAME, 1, 30))),
  VARCHAR(RTRIM(SUBSTR(C.REMARKS, 1, 254))),
  VARCHAR(RTRIM(SUBSTR(C.LABEL, 1, 30)))
FROM SYSIBM.SYSCOLUMNS C
  ,( SELECT DISTINCT
    VARCHAR(RTRIM(TA.TCREATOR)), TA.TTNAME
  FROM SYSIBM.SYSTABAUTH TA
  WHERE TA.GRANTEETYPE=' '
  AND TA.GRANTEE IN (
    USER,CURRENT SQLID,CURRENT SCHEMA,
    'PUBLIC','PUBLIC*' )
  ) AS UAT ("CREATOR","NAME")
  WHERE (C.TBCREATOR=UAT.CREATOR OR
    C.TBCREATOR=CURRENT SCHEMA)
  AND C.TBNAME=UAT.NAME;

```

Q.DSQEC_COLS_RDB2L

視点 Q.DSQEC_COLS_RDB2L を使用できるのは、DB2 for z/OS データベースの DESCRIBE コマンドのみです。

この視点は、次の SQL ステートメントにより作成されます。

```

CREATE VIEW Q.DSQEC_COLS_RDB2L
  ( OWNER, TNAME, CNAME, REMARKS, LABEL )
AS SELECT VARCHAR(RTRIM(C.TBCREATOR)),
  C.TBNAME,
  VARCHAR(RTRIM(SUBSTR(C.NAME, 1, 30))),
  VARCHAR(RTRIM(SUBSTR(C.REMARKS, 1, 254))),
  VARCHAR(RTRIM(SUBSTR(C.LABEL, 1, 30)))
FROM SYSIBM.SYSCOLUMNS C
  ,( SELECT
    VARCHAR(RTRIM(TA.TCREATOR)), TA.TTNAME
  FROM SYSIBM.SYSTABAUTH TA
  WHERE TA.GRANTEETYPE=' '
  AND TA.GRANTEE IN (
    USER,CURRENT SQLID,'PUBLIC*'
  )
  ) AS UAT ("CREATOR","NAME")
  WHERE C.TBCREATOR=UAT.CREATOR AND C.TBNAME=UAT.NAME

```

Q.DSQEC_ALIASESL

視点 Q.DSQEC_ALIASESL は、表のリストに対する別名のリスト、あるいは DB2 (VSE および VM 版) を除いたすべてのデータベース・タイプに対する別名の列情報のみを選択します。

この視点は、次の SQL ステートメントにより作成されます。これらのステートメントは、DB2 for z/OS データベースにのみ適用されます。DB2 for Linux, UNIX, and Windows および DB2 for iSeries 用の DDL はジョブ DSQ0BCTV にあります。

```

CREATE VIEW Q.DSQEC_ALIASESL
  ( OWNER, TNAME, TYPE, SUBTYPE, MODEL,
  RESTRICTED, REMARKS, CREATED, MODIFIED,
  LAST_USED, LABEL, LOCATION,
  OWNER_AT_LOCATION, NAME_AT_LOCATION )
AS SELECT
  VARCHAR(RTRIM(T.CREATOR)),T.NAME,'TABLE',T.TYPE
  ,',',',',',VARCHAR(RTRIM(SUBSTR(T.REMARKS, 1, 254))),
  ',',' ',',',',',',

```

```

        VARCHAR(RTRIM(SUBSTR(T.LABEL, 1, 30))),
        SUBSTR(T.LOCATION, 1, 16),
        VARCHAR(RTRIM(T.TBCREATOR)),T.TBNAME
FROM SYSIBM.SYSTABLES T
WHERE T.CREATOR
      IN (USER,CURRENT SQLID,CURRENT SCHEMA)
AND T.TYPE = 'A';

```

Q.DSQEC_TABS_SQLL

視点 Q.DSQEC_TABS_SQLL は、現在接続している DB2 (VSE および VM 版) データベースに送信される LIST コマンドで使用されます。

この視点は、次の SQL ステートメントにより作成されます。

```

CREATE VIEW Q.DSQEC_TABS_SQLL
(OWNER,TNAME,TYPE,SUBTYPE,MODEL,RESTRICTED,REMARKS,
CREATED,MODIFIED,LAST_USED,LABEL,LOCATION,
OWNER_AT_LOCATION,NAME_AT_LOCATION)
AS SELECT
STRIP(CREATOR),TNAME,'TABLE',TABLETYPE,' ',' ',
REMARKS,' ',' ',' ',
TLABEL,' ',' ',' '
FROM SYSTEM.SYSCATALOG, SYSTEM.SYSTABAUTH
WHERE CREATOR = TCREATOR AND TNAME=TTNAME AND
GRANTEETYPE = ' ' AND
GRANTEE IN (USER,'PUBLIC')

```

Q.DSQEC_COLS_SQLL

視点 Q.DSQEC_COLS_SQLL は、現在接続している DB2 (VSE および VM 版) データベースに送信される DESCRIBE コマンドで使用されます。

この視点は、次の SQL ステートメントにより作成されます。

```

CREATE VIEW Q.DSQEC_COLS_SQLL
(OWNER, TNAME, CNAME, REMARKS,LABEL)
AS SELECT
STRIP(CREATOR), TNAME, CNAME, REMARKS, CLABEL
FROM SYSTEM.SYSCOLUMNS, SYSTEM.SYSTABAUTH
WHERE TCREATOR = CREATOR AND TTNAME = TNAME AND
GRANTEETYPE = ' ' AND GRANTEE IN (USER,'PUBLIC')

```

デフォルトの視点名を保管する大域変数

LIST コマンドおよび DESCRIBE コマンドで使用されるデフォルト視点の名前は、QMF 大域変数に保管されます。

LIST コマンドの場合:

- コマンドがローカル DB2 for z/OS データベースに送信された場合 (または CONNECT コマンドを発行して DB2 (VSE および VM 版) 以外のリモート・データベースに接続し、LIST TABLES コマンドを発行した場合)、QMF は、DSQEC_ALIASES および DSQEC_TABS_LDB2 大域変数内で指定された視点を使用します。デフォルトでは、DSQEC_ALIASES 大域変数は Q.DSQEC_ALIASESL に設定され、DSQEC_TABS_LDB2 大域変数は、Q.DSQEC_TABS_LDB2L に設定されます。
- LOCATION パラメーターを含む LIST TABLES コマンドを発行する場合、QMF は DSQEC_TABS_RDB2 大域変数内で指定された視点を使用します。これはデフォルトで視点 Q.DSQEC_TABS_RDB2L に設定されます。LOCATION パラメーターを含む LIST コマンドを開始したり、送信することができるのは、DB2 for z/OS データベースのみです。

- CONNECT コマンドを使用して DB2 (VSE および VM 版) データベースに接続し、LIST TABLES コマンドを発行する場合、QMF は DSQEC_TABS_SQL 大域変数内で指定された視点を使用します。これはデフォルトで Q.DSQEC_TABS_SQLL です。

DESCRIBE コマンドの場合:

DESCRIBE コマンドを発行するときに、QMF が必要な情報を生成するために使用する視点もコマンドが送信されるデータベースに依存します。

- コマンドがローカル DB2 for z/OS データベースに送信された場合 (または CONNECT コマンドを発行して DB2 (VSE および VM 版) 以外のリモート・データベースに接続して DESCRIBE コマンドを発行した場合)、QMF は DSQEC_COLS_LDB2 大域変数内で指定された視点を使用します。この変数はデフォルトで、Q.DSQEC_COLS_LDB2L に設定されます。
- LOCATION パラメーターを含む LIST TABLES コマンドを発行して、次にリスト内の表または視点の 1 つに対して DESCRIBE コマンドを発行する場合、QMF は DSQEC_COLS_RDB2 大域変数で指定された視点を使用します。この変数はデフォルトで Q.DSQEC_COLS_RDB2L に設定されます。LOCATION パラメーターを含む LIST コマンドを開始したり、送信することができるのは、DB2 for z/OS データベースのみです。
- CONNECT コマンドを使用して DB2 (VSE および VM 版) データベースに接続し、LIST TABLES コマンドに続けて DESCRIBE コマンドをリスト内の表または視点の 1 つに発行する場合、QMF は DSQEC_COLS_SQL 大域変数内で指定された視点を使用します。この変数はデフォルトで、Q.DSQEC_COLS_SQLL に設定されます。

関連資料:

247 ページの『コマンドに使用されるデフォルトの視点』

QMF は、インストール中にデフォルト視点を提供します。各視点がどのように使用されるかは、LIST または DESCRIBE コマンドを発行したかどうか、コマンドが送信された場所によって異なります。

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。

各行の記憶域要件を以下に示します。

- 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、FORM、および ANALYTIC のオブジェクト情報 (63 バイト)

- 表示データと制御情報は、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。

関連資料:

499 ページの『QMF 視点』

以下の視点が QMF に付属しています。QMF はこれらの視点を、表に示されているプラットフォーム上で使用し、QMF LIST コマンドが発行されたときにオブジェクト・リストを作成します。

第 12 章 データベース内のオブジェクトの作成と保守

どのように QMF およびデータベース・オブジェクトをセットアップして保守するかは、ユーザーが QMF ならびにデータのセキュリティーおよびシステムのパフォーマンスをどの程度簡単に使用できるかに影響します。

ユーザーがデータベース内で表を作成できるようにする

サイトの要件によって、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。

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、SQL 照会パネルから、またはデータベースから直接実行します。

ヒント: 単一の SQL 照会は、複数の SQL メンテナンス・ステートメントを含むことができます。複数のステートメントを含む照会を作成するには、ステートメントの間にセミコロンを入れて、DSQEC_RUN_MQ 大域変数を 1 に設定します。複数のステートメントを含む照会の作成方法、および使用可能なステートメント・タイプの制約に関する情報は、RUN コマンドの説明に記載されています。

- QMF DISPLAY コマンドと、その後続く SAVE DATA コマンド。

SAVE DATA コマンドの構文およびオプションに関する情報は、SAVE コマンドの説明に含まれています。

- QMF IMPORT コマンド

表を作成するための手順

表を作成するための手順は、管理者がユーザーのために表を作成するのか、ユーザーが自分用の表を作成するのかによって異なります。

手順

- 管理者がユーザーのために表を作成する場合は、以下の手順に従ってください。
 1. 最初に使用する前に、表スペースを作成してデータベースに定義します。
 2. 表を作成するには、SQL CREATE TABLE ステートメントを発行するか、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 コマンドを実行します。
 3. DB2 パフォーマンスの向上を図るために、作成する表に 1 つまたは複数の索引を作成します。
 4. 表にデータを入れます。DB2 for z/OS 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。

5. 表に関する DB2 特権および SQL 特権を、それらを必要とするユーザーに付与します。
- ユーザーが自身で表を作成する場合は、以下の手順に従ってください。
 1. ユーザーに DB2 CREATETS または CREATETAB 権限を付与します。CREATETAB 権限のみを付与するように選択した場合は、最初に使用する前に、表スペースを作成してデータベースに定義します。
 2. SPACE フィールドに関して SQL UPDATE ステートメントを使用して、ユーザーの QMF プロファイル内に表スペースを割り当てます。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。
 3. 表スペース内に自分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、ユーザーが自分用の表スペースを作成できるようにします。ユーザーは、自分が作成する表および表スペースに対して、自動的に SQL 特権をすべて持ちます。
 4. SQL CREATE TABLE ステートメント、QMF SAVE DATA コマンドと 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ガイドラインについて、教育を行います。
 5. 新しい表を作成するためにユーザーが SAVE DATA コマンドまたは IMPORT コマンドを発行する対象となる表または視点に対する DB2 特権および SQL 特権を付与します。少なくとも SELECT 特権を付与しないと、QMF は新しい表を作成するためのデータを読み取ることができません。

関連情報:

DB2 資料

表の論理設計に関する情報と、CREATE TABLE、CREATE INDEX、および表の作成に関連するその他の SQL ステートメントに関する情報を検索します。また、LOAD ユーティリティーに関する情報と、権限、特権、および GRANT ステートメントに関する情報を検索します。

SAVE DATA および IMPORT コマンド用の表スペースの割り当て

ユーザーが SAVE DATA または IMPORT TABLE コマンドを発行したときに、どのデータベースに表が保存されるかは、QMF プロファイルの SPACE オプションの設定方法によって異なります。

プロファイルの SPACE フィールドに有効な値

QMF SYSTEM プロファイルを使用しているユーザーは、デフォルトのデータベースおよび DSQDBDEF.DSQTSDEF の表スペースを持っています。

プロファイルの SPACE オプションを次の値で設定できます。

- *dbname.spacename*

データベースおよび表スペースの両方で QMF が特定の表スペースに表を保存するように指定します。このトピックを通して、これは明示的表スペースといえます。

例えば、次の照会は、SMITH という名前の QMF ユーザーのプロファイルがデータベース DBASE1 の表スペース TSPACE1 に表を保存するように構成します。

```
UPDATE Q.PROFILES
  SET SPACE='DBASE1.TSPACE1'
  WHERE CREATOR='SMITH' AND TRANSLATION='ENGLISH'
```

ただし、すべてのユーザーが同じ表スペースを使用している場合は、リソースの競合が発生する可能性があります。

- DATABASE "*databasename*"

DATABASE キーワードの後にデータベース名を引用符で囲んで指定すると、各表は DB2 によってその表用に暗黙的および排他的に作成された別々の表スペースに作成されます。このような暗黙的な表スペースは、デフォルトの LOCKSIZE、BUFFERPOOL、STOGROUP、およびスペース属性を持ち、作成された表の名前と一致する名前を持ちます。

例えば、次の Q.PROFILES 表の SPACE フィールドの値は、各表を DSQDBDEF データベース内の別々の表スペースに保存します。

```
DATABASE "DSQDBDEF"
```

各表スペースを 1 つの表に制限すると、リソースの競合が最小限になります。パーティション化された表スペース・スキームおよびユニバーサル表スペース・スキームのどちらも、表スペース当たり 1 つの表の制限を課します。

- NULL または空白

QMF プロファイルの SPACE フィールドが NULL または空白の場合は、SAVE DATA または IMPORT TABLE コマンドがデフォルトで DSNDB04 データベース内の暗黙的に生成された表スペース名に表を作成します。

表スペース・スキームを選択するときを考慮する要因

必要に応じて表スペースを明示的に作成するか、または暗黙的に作成するように QMF を構成するかを決定する際には、以下の要因を考慮してください。

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。

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。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の場合も、保守を単純化することができます。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せん。

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、または暗黙的に作成されたユニバーサル表スペース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。

保全性とセキュリティー

一部の DB2 特権には、ユーザーが必要としない特権ではあるけれども、それをユーザーに付与する必要があるものがあります。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、選択した表スペース内での表の作成に限定することができます。暗黙表スペース・オプションを選択した場合は、データベースの表スペースを作成する特権をユーザーに付与する必要があり、この特権を SAVE および IMPORT コマンドで作成する表スペースに制限することはできません。

便宜 明示的に作成された表スペース DSQDBDEF.DSQTSDEF は、QMF SYSTEM プロファイルを使用しているユーザー用のデフォルトの表スペースです。この表スペースは、QMF のインストール過程で作成され、インストール検査手順で使用されます。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。

この表スペースは、表が基本的に読み取り専用の場合のみ、複数のユーザーで使用してください。それ以外の場合は、リソースの競合によって必要な更新が妨げられることがあります。

関連情報:

DB2 資料

表スペースの拡大手順に関する情報を検索します。

ユーザーへの表の作成権限の付与

表を作成するユーザーに付与する必要がある権限にはさまざまなものがあります。QMF ユーザー・プロファイルの SPACE フィールドにデータベースおよび表スペース名を指定する場合は、1 つの特権セットが必要です。データベースのみを指定して値を指定しない場合は、それとは別の特権セットが必要です。

明示的に作成された表スペースに必要な特権

少なくともユーザーは、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。

あるユーザーに表の作成を許可したいが、使用されるリソース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。

暗黙的に作成された表スペースに必要な特権

少なくともユーザーは、データベースに対する `CREATETAB` および `CREATETS` 特権を必要とする。

デフォルトの DB2 for z/OS データベース `DSNDB04` のユーザーは、すでに前述の特権の一部を持っていることがあります。データベース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。

重要: DB2 自体によって排他的に使用されるどのデータベースに対する特権も付与しないでください。このようなデータベースには、`DSNDB01`、`DSNDB03`、および `DSNDB05` があります。

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 を所有者修飾子 (オブジェクトの所有者) とする必要があります。他の ID を使用して、適切な `CREATE TABLE` ステートメントを実行することはできません。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、このユーザーは必要な `INSERT` 特権を持つこととなります。必要なのは、`CREATE TABLE` ステートメントを実行する特権だけです。

関連概念:

254 ページの『プロファイルの `SPACE` フィールドに有効な値』

`QMF SYSTEM` プロファイルを使用しているユーザーは、デフォルトのデータベースおよび `DSQDBDEF.DSQTSDEF` の表スペースを持っています。

関連情報:

 [DB2 資料](#)

`GRANT` ステートメントに関する情報を検索します。

機密データをフィルターするための視点の使用

視点を使用して特定のユーザーから機密データをフィルターすることができます。

視点の作成

視点を作成するんは、`CREATE VIEW` ステートメントを使用します。

例えば、人事情報が入った表 `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。行ごとに、視点では該当の従業員の名前、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。

以下のような `SQL` ステートメントを持つ視点を作成できます。

```
CREATE VIEW VIEWA AS
  SELECT NAME, DEPT, JOB, YEARS
  FROM SMITH.STAFF
```

視点を作成する場合は、ユーザーの SQL 許可 ID が、少なくとも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。

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。その除去された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様です。

関連情報:



DB2 資料

読み取り専用視点および CREATE VIEW ステートメントに関する情報を検索します。

視点の特権の付与

表と同様に、視点に SELECT、INSERT、UPDATE、および DELETE 特権を付与することができます。

SELECT 特権を持っていれば、ユーザーは、SELECT 照会および副照会での表と同じように、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、その視点の基本表のデータを変更することができます。

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。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 CREATE 定義ステートメントの FROM 文節で名前を指定する表および視点のことです。例えば、次のステートメント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, YEARS
  FROM SMITH.STAFF
```

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、以下の 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。

視点は作成したステートメントが 2 つ以上の表に結合されている場合は、自動的に読み取り専用になります。

- 視点の所有者が該当の基本オブジェクトに対しても同じ特権を持っています。

表と同様に、WITH GRANT OPTION 文節を指定して特権を他のユーザーに付与できることを保証します。

関連情報:

 DB2 資料

読み取り専用視点および CREATE VIEW ステートメントに関する情報を検索します。

QMF オブジェクト・カタログの保守

QMF の照会、書式、プロシージャー、分析、およびフォルダーに関する情報を保管する QMF オブジェクト・カタログは、定期的に圧縮および再編成する必要があります。

QMF カタログの定期保守には、新規の所有者へのオブジェクトの転送、あるいはその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。

照会、書式、プロシージャー、分析、およびフォルダーに関する QMF カタログ情報はすべて、以下の 3 つの QMF コントロール表のいずれかに保管されます。

- Q.OBJECT_DIRECTORY
- Q.OBJECT_DATA
- Q.OBJECT_REMARKS

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つことができます。また、ある所有者から別の所有者にオブジェクトを転送するために、これらの表の情報を使用する必要があることもあります。QMF カタログの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。

関連資料:

497 ページの『TSO および CICS での QMF コントロール表および表スペース』QMF に付属のコントロール表は以下のとおりです。

Q.OBJECT_DIRECTORY 表の構造

Q.OBJECT_DIRECTORY 表には、データベースにある QMF の照会、書式、プロシージャー、フォルダー、および分析オブジェクトごとに 1 行が含まれています。

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。

Q.OBJECT_DIRECTORY 表は、下の表に示す構造を持っています。

表 44. Q.OBJECT_DIRECTORY 表の構造

| 列名 | データ・タイプと長さ | NULL の使用 | 機能 / 値 |
|-------|---|----------|---|
| OWNER | VARCHAR(128) 例外: OWNER 列は、DB2 (VSE および VM 版) データベースに接続されている場合は CHAR(8) として定義されます。 | 不可 | オブジェクトの作成者の許可 ID を示す。 |
| NAME | VARCHAR(128) 例外: NAME 列は、DB2 (VSE および VM 版) データベースに接続されている場合は VARCHAR(18) として定義されます。 | 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、ANALYTIC、QUERY、または FOLDER) を示す。 例外: DB2 Server (VSE および VM 版). |

表 44. Q.OBJECT_DIRECTORY 表の構造 (続き)

| 列名 | データ・タイプと長さ | NULL の使用 | 機能 / 値 |
|-------------|------------|----------|--|
| SUBTYPE | CHAR(8) | 可 | <p>QUERY オブジェクトには以下のサブタイプが使用されます。</p> <ul style="list-style-type: none"> • SQL • QBE • PROMPTED <p>ANALYTIC オブジェクトには以下のサブタイプが使用されます。</p> <p>PLOT プロット・グラフ</p> <p>HIST ヒストグラム</p> <p>PIE 円グラフ</p> <p>TOWER タワー図表</p> <p>MIXED 混合図表</p> <p>MAP マップ</p> <p>UNIVAR 単変量曲線</p> <p>LINEAR 線形傾向</p> <p>DCF 割引キャッシュ・フロー</p> <p>BASIC 基本統計</p> <p>BIVAR 二変量曲線</p> <p>WILCOSR ウィルコクソン符号順位検定</p> <p>MANNW マン・ホイットニー U 検定</p> <p>その他すべてのオブジェクト・タイプの場合、このフィールドはブランクです。</p> |
| OBJECTLEVEL | INTEGER(4) | 不可 | <p>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。</p> |

表 44. Q.OBJECT_DIRECTORY 表の構造 (続き)

| 列名 | データ・タイプと長さ | NULL の使用 | 機能 / 値 |
|------------|------------|----------|--|
| RESTRICTED | CHAR(1) | 不可 | オブジェクトを共有していない場合 (QMF Analytics for TSO で QMF SAVE コマンドの SHARE パラメーターまたは SAVE パネルの SHARE オプションを使用) は YES。オブジェクトを他のユーザーと共有している場合は NO。 |
| MODEL | CHAR(8) | 可 | この値は常に 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 | <p>オブジェクトにアクセスしたコマンドが正常に実行されたかどうかとは無関係に、オブジェクトが最後に使用された時点を表す日付値を示す。ただし、LAST_USED 列はコマンドの発行直後には更新されないことがあります。また、QMF が異常終了すると、列が更新されない場合があります。</p> <p>デフォルトでは、次のコマンドは LAST_USED 値を変更します。</p> <ul style="list-style-type: none"> • CONVERT • DISPLAY • EXPORT • IMPORT • LAYOUT • PRINT • RUN • SAVE <p>DSQEC_LAST_RUN 大域変数を 1 に設定することによって、LAST_USED 列を更新するコマンドを、RUN、SAVE、および IMPORT のみに制限することができます。</p> <p>また、DSQEC_LAST_RUN 大域変数を 2 に設定することによって、LAST_USED 列を更新するコマンドを RUN のみに制限することができます。</p> <p>この値は、オブジェクトが複数回使用される場合でも、1 日に 1 回だけ更新されます。</p> |

Q.OBJECT_DATA 表の構造

Q.OBJECT_DATA 表には、データベースにある照会、書式、プロシージャー、フォルダー、および分析オブジェクトごとに 1 つ以上の行が含まれています。

各行には、各オブジェクト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、UNIQUE および CLUSTER 属性が指定されている索引 Q.OBJECT_OBJDATA があります。キー付きの列は OWNER、NAME、および SEQ です。

表は、下の表に示す構造を持っています。

表 45. Q.OBJECT_DATA 表の構造

| 列名 | データ・タイプと長さ | NULL の使用 | 機能 / 値 |
|----------|---|----------|---|
| OWNER | VARCHAR(128) 例外: OWNER 列は、DB2 (VSE および VM 版) データベースに接続されている場合は CHAR(8) として定義されます。 | 不可 | オブジェクトの作成者の許可 ID を示す。 |
| NAME | VARCHAR(128) 例外: NAME 列は、DB2 (VSE および VM 版) データベースに接続されている場合は VARCHAR(18) として定義されます。 | 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、ANALYTIC、QUERY、または FOLDER) を示す。 例外: QMF が DB2 Server (VSE および VM 版) に接続している場合、FOLDER は有効なタイプではありません。 |
| 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。例えば、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 | VARCHAR(3600) FOR BIT DATA | 可 | オブジェクトを構成するデータを含む。 重要: データは 2 進形式で保管されるため、この列を更新しないでください。さらに、APPLDATA 列は、コード・ページ (CCSID) 変換の対象とはなりません。 |

Q.OBJECT_REMARKS 表の構造

Q.OBJECT_REMARKS 表には、データベースにある照会、書式、プロシージャー、フォルダー、および分析オブジェクトごとに 1 行が含まれています。

各行には、QMF SAVE コマンドの発行時、または QMF Analytics for TSO での「保管」ファンクション・キーの使用時に入力されたコメントが含まれています。

Q.OBJECT_REMARKS 表には、UNIQUE および CLUSTER 属性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。

表は、下の表に示す構造を持っています。

表 46. Q.OBJECT_REMARKS 表の構造

| 列名 | データ・タイプと長さ | NULL の使用 | 機能 / 値 |
|---------|---|----------|---|
| OWNER | VARCHAR(128) 例外: OWNER 列は、DB2 (VSE および VM 版) データベースに接続されている場合は CHAR(8) として定義されます。 | 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(128) 例外: NAME 列は、DB2 (VSE および VM 版) データベースに接続されている場合は VARCHAR(18) として定義されます。 | 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、ANALYTIC、QUERY、または FOLDER) を示す。 例外: QMF が DB2 Server (VSE および VM 版) に接続している場合、FOLDER は有効なタイプではありません。 |
| REMARKS | VARCHAR(254) | 可 | オブジェクトが作成または置換された際にオブジェクトと共に保存されたコメントを含む。 |

QMF オブジェクト・カタログ用の表スペースの拡大

QMF オブジェクトが大きくなりすぎて、QMF オブジェクト・カタログ・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。

始める前に

必要なスペースの容量を予測するのを支援するために、現在使用しているスペースの YOU 用を決定することができます。スペースが DB2 によって管理されている場合、この情報は以下の方法で入手できます。

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='ttttttt' AND DBNAME='DSQDBCTL'
```

このステートメントで、*ttttttt* は表スペース名です。結果 (SPACE) によって、現在表スペースに割り振られている記憶域の K バイト (KB) 数が与えられます。

このタスクについて

下の表に、デフォルト表スペースを示します。

この手順に加えて、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 47. QMF オブジェクトに関する情報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 表および Q.OBJECT_DATA2 表 | 5120 ページ |

重要: QMF が以前にインストールされていなかった場合、QMF バージョン 12.1 は、DB2 によって管理される表スペース・データ・セットを作成します。QMF のインストール後に、QMF 記憶域グループを DB2 による管理からユーザー管理に変更しないでください。ただし、スペースがユーザー管理の場合は、データ・セット名が分かっている場合、スペース情報に関して TSO LISTCAT コマンドを使用することができます。

手順

QMF オブジェクト・カタログ・コントロール表の表スペースを拡大するには、以下の手順に従ってください。

1. 表スペースのイメージ・コピーを作成する。これは、プロシージャが失敗し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ます。

表スペースに使用されるデータ・セット管理のタイプを判別するには、次の照会を実行します。

```
SELECT STORTYPE
FROM SYSIBM.SYSTABLEPART
WHERE TSNAME='DSQTSCT3' AND DBNAME='DSQDBCTL'
```

この照会では、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 によって管理されることを示します。次の表は、QMF コントロール表のデフォルトのデータベース・パーティション・グループを示します。

表 48. QMF オブジェクトに関する情報を保管するコントロール表用のデータベース・パーティション・グループ

| データベース・パーティション・グループ名 | 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。 | 複数のデータベース・パーティションに配布可能。成長への対応力は低い。 |
| DSQTSOBJ | プロシージャ、照会、書式、フォルダー、および分析オブジェクトが保管される QMF オブジェクト・カタログ・コントロール表。 | 複数のデータベース・パーティションに配布可能。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、単一データベース・パーティションに制限するように定義する必要がある。 |
| DSQTSAMP | QMF サンプル表。 | 複数のデータベース・パーティションに配布可能。 |

3. 次のコマンドを使用してデータベースを停止する。

```
-STOP DATABASE(DSQDBCTL)
```

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、次の例のような DB2 ステートメントを出す。

```
ALTER TABLESPACE DSQDBCTL.tttttt
USING STOGROUP ssssss PRIQTY pppp SECQTY ssss
```

このステートメントで、*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 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 (キロバイト数) です。

- 表スペース・データ・セットが DB2 によって管理されている場合は、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
```

このステートメントで、*ttttt* は表スペース名です。文字ストリング *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 (KB 数) です。

5. 表スペース・データを移動する。表スペース記述を単に変更しただけでは、拡大は行われません。代わりに、表スペースを再度埋め込むことが必要です。
6. 次のステートメントでデータベースを開始する。

```
-START DATABASE(DSQDBCTL)
```

QMF オブジェクトのリスト

QMF 環境の維持に必要な情報を入手するために、QMF ユーザーがデータベースに保存した照会、書式、プロシージャー、フォルダー、分析オブジェクトをリストすることができます。

管理者権限を持っている場合は、以下の照会を使用して、自分が所有していない 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する照会、書式、プロシージャー、フォルダー、および分析オブジェクトのリスト

この照会で返されるオブジェクトのリストは、タイプ (ANALYTICS、FORM、PROC、または QUERY) 別にソートされ、タイプが ANALYTICS または QUERY の場合はさらにサブタイプ別にソートされます。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。共用状況が、表 Q.OBJECT_DIRECTORY の RESTRICTED 列に反映されます。値「Y」は、オブジェクトが共用されていないことを示します。

QMF オブジェクトの表示

QMF 環境の維持に必要な情報を入手するために照会、プロシージャー、書式、または分析オブジェクトをリストするだけでなく、DISPLAY コマンドを使用して詳細情報を得ることができます。

フォルダーにはオブジェクト・データが含まれていないため、DISPLAY は、QMF フォルダー・オブジェクトには有効なコマンドではありません。フォルダーには、他の QMF オブジェクト (照会、プロシージャー、書式、または分析オブジェクト) への参照のみが含まれています。

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を使用してオブジェクトの表示を試みます。

- 次のステートメントを発行してユーザーのオブジェクトを共用します。次に、それらを自身の ID から表示します。

```
UPDATE Q.OBJECT_DIRECTORY
SET RESTRICTED = 'N'
WHERE OWNER = 'userid'
```

重要: このステートメントを実行するのは、ユーザーのどのオブジェクトが制限され、どのオブジェクトが制限されていないかを追跡する必要がない場合に限りです。このステートメントの発行後は、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。

ヒント: 単一の SQL 照会は、複数の SQL メンテナンス・ステートメントを含むことができます。複数のステートメントを含む照会を作成するには、ステートメントの間にセミコロンを入れて、DSQEC_RUN_MQ 大域変数を 1 に設定します。複数のステートメントを含む照会の作成方法、および使用可能なステートメント・タイプの制約に関する情報は、RUN コマンドの説明に記載されています。

- 表示したいオブジェクトごとに QMF DISPLAY コマンドを発行する。

関連概念:

267 ページの『QMF オブジェクトのリスト』

QMF 環境の維持に必要な情報を入手するために、QMF ユーザーがデータベースに保存した照会、書式、プロシージャ、フォルダー、分析オブジェクトをリストすることができます。

QMF オブジェクトの所有権の移譲

QMF オブジェクト (照会、書式、プロシージャ、フォルダー、および分析オブジェクト) をあるユーザーから別のユーザーに移譲するには、3 つのステートメントを実行する必要があります。必ず、3 つのステートメントすべてを実行してください。

重要: まず、移行するオブジェクトと同じ名前で作成されているオブジェクトを新しい所有者が持たないようにします。そうでないと、QMF は、既存のオブジェクトを移行するオブジェクトで置き換えます。

ユーザー間で QMF オブジェクトを移行するには、以下のステートメントを実行します。

| | | |
|-----------------------------|-----------------------------|-----------------------------|
| UPDATE Q.OBJECT_DIRECTORY | UPDATE Q.OBJECT_REMARKS | UPDATE Q.OBJECT_DATA |
| SET OWNER = 'newuserid' | SET OWNER = 'newuserid' | SET OWNER = 'newuserid' |
| WHERE OWNER = 'olduserid' | WHERE OWNER = 'olduserid' | WHERE OWNER = 'olduserid' |
| AND NAME IN <i>namelist</i> | AND NAME IN <i>namelist</i> | AND NAME IN <i>namelist</i> |

これらのステートメントで、*namelist* は移行す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、名前はそれぞれ単一引用符で囲んでコンマで区切る必要があります。次に例を示します。

```
('QUERY1','QUERY2','FORMA','PROCB')
```

古い SQL 許可 ID で修飾されたオブジェクトを指名する照会またはプロシージャについては、修飾子の変更を忘れないで行ってください。例えば、MYQUERY を BAXTER から JONES に移行する場合は、名前を BAXTER.MYQUERY から JONES.MYQUERY に変更します。

オブジェクトを移行後、そのオブジェクトを共有する必要がある場合は、267 ページの『QMF オブジェクトの表示』に示されているような SQL ステートメントを使用して、RESTRICTED 列の値を N に変更します。

ヒント: 単一の SQL 照会は、複数の SQL メンテナンス・ステートメントを含むことができます。複数のステートメントを含む照会を作成するには、ステートメントの間にセミコロンを入れて、DSQEC_RUN_MQ 大域変数を 1 に設定します。複数のステートメントを含む照会の作成方法、および使用可能なステートメント・タイプの制約に関する情報は、RUN コマンドの説明に記載されています。

廃止された QMF オブジェクトの削除

特定ユーザーの QMF の照会、書式、プロシージャ、フォルダー、および分析オブジェクトをすべて削除するには、3 つのステートメントを実行する必要があります。必ず 3 つのステートメント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記が 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。

```
DELETE FROM Q.OBJECT_DIRECTORY      DELETE FROM Q.OBJECT_REMARKS      DELETE FROM Q.OBJECT_DATA
WHERE OWNER = 'olduserid'           WHERE OWNER = 'olduserid'         WHERE OWNER = 'olduserid'
```

図 35. QMF オブジェクト・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 表を日時でソートすることによって、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特定の日付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。

ヒント: 単一の SQL 照会は、複数の SQL メンテナンス・ステートメントを含むことができます。複数のステートメントを含む照会を作成するには、ステートメントの間にセミコロンを入れて、DSQEC_RUN_MQ 大域変数を 1 に設定します。複数のステートメントを含む照会の作成方法、および使用可能なステートメント・タイプの制約に関する情報は、RUN コマンドの説明に記載されています。

z/OS データ・セットからの照会、書式、およびプロシージャのインポート

z/OS データ・セット内に存在する QMF オブジェクトは、IMPORT コマンドを使用して QMF にインポートできます。

エクスポートされたオブジェクトが RACF の場合、これらのデータ・セット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。

DB2 for z/OS サブシステムの保守

保守の責任を共有する場合は、特殊な管理作業をユーザーに割り当て、そのユーザー独自の許可 ID で実行させることができます。

このタスクについて

このようなユーザーには、該当の作業に必要な照会およびユーティリティーを実行するために十分な DB2 権限のみを付与します。次に例を示します。

- 新しいユーザーの QMF プロファイルを作成するには、Q.PROFILES 表に対する INSERT 特権が必要です。
- 関連する表、索引、および表スペースを管理するには、データベースに対する DBADM または同等の権限が必要です。

他のユーザーに対してデータベースにより低い特権を付与することができるようにするには、WITH GRANT OPTION 文節を含めます。

- QMF オブジェクト・コントロール表をモニターおよび再編成できるようにするために、保管されているデータベースに対する STATS および REORG 特権が必要です。

特に断りがない限り、情報は DB2 for z/OS に関連しています。z/OS で、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。DB2 for Linux, UNIX, and Windows では、各サーバー (指定のロケーション) は単一のデータベースです。各データベース内で複数の表スペースを保守できます。

保守作業中にユーザーによる QMF へのアクセスを防ぐには、保守操作を始める前に、作業中の表スペースに対して DB2 -STOP DATABASE コマンドを発行してください。

関連情報:



DB2 資料

-STOP DATABASE コマンドに関する情報を検索します。

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と DB2 管理の場合があります。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。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有します。QMF コントロール表の記憶域グループは、DSQSGCTL です。QMF インストール・プロセスによってこの記憶域グループが作成されます。QMF のインストール後は、この記憶域グループを DB2 管理からユーザー管理に変更しないでください。

DB2 for Linux, UNIX, and Windows を使用している場合、記憶域グループは適用されません。

ユーザー管理のデータ・セットの VSAM クラスタ

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 ステートメントによって DB2 にリンクします。クラスターとその DB2 オブジェクト間のリンクは、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。

関連資料:

497 ページの『TSO および CICS での QMF コントロール表および表スペース』
QMF に付属のコントロール表は以下のとおりです。

QMF 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。

このタスクについて

SQL 保守ステートメントは、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。DB2 for Linux, UNIX, and Windows サーバーの場合、データベースがインストールされているオペレーティング・システムから DB2 コマンド行プロセッサを使用できます。

QMF 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状態をモニターすることにより、保守問題の発生を回避できます。

このタスクについて

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。

手順

1. コントロール表および索引で RUNSTATS ユーティリティを定期的に行い、特定の DB2 システム表に現行統計を追加します。
2. これらのシステム表を照会し、これらの統計を検討して、再編成が必要かどうかを判断します。
3. 再編成が必要な場合は、以下の手順を実行します。
 - a. REORG ユーティリティを実行する。
 - b. RUNSTATS ユーティリティを再実行する。
 - c. 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。

次のタスク

最も効率的な検索パスを選択できるように、再編成後には QMF などの最も重要なアプリケーションを再バインドしてください。

関連情報:

 DB2 資料

DB2 システム・カタログを使用したコントロール表の状態のモニターに関する情報

を検索します。

バッファーク・プールの切り替え

パフォーマンスの理由により、QMF 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ーク・プールを変更しなければならない場合があります。

例えば、コントロール表索引および表スペース用のバッファーク・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可能性があります。

バッファーク・プールの変更には、ALTER TABLESPACE および ALTER INDEX ステートメントを用います。新しいバッファーク・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。

関連情報:

 DB2 資料

ALTER TABLESPACE ステートメントおよび ALTER INDEX ステートメントに関する情報と、これらを実行するために必要な権限に関する情報を検索します。

第 13 章 印刷および図表作成機能のセットアップ

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・ユーザー用に印刷をセットアップする方法は、所有するプリンターのタイプと印刷するオブジェクトの種類に応じて異なります。このトピックは、QMF サービスと図形データ表示管理プログラム (GDDM) サービスのどちらを使用して印刷を処理するのがより効率的かを判断するために役立ちます。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。

関連タスク:

223 ページの『非 DBCS ディスプレイ装置からの DBCS データの印刷』

大文字、日本語、または韓国語の各国語機能 (NLF) を使用する場合は、2 バイト文字セット (DBCS) データの印刷が必要になる場合があります。

印刷に QMF または GDDM のいずれのサービスを使用するか の決定

GDDM サービスまたは QMF サービスのいずれを使用して印刷するかは、印刷する必要のあるオブジェクトのタイプと、使用できるプリンターや他のリソースのタイプによって決まります。

それぞれの要求にどちらの方法が適しているか、このトピックを読んで決定してください。

- 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、名前を装置にリンクすることができます。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 を使用する場合は、以下の点も考慮してください。

- 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信するプログラム、または CICS 提供のトランザクション CEBR を使用して印刷出力を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、印刷出力が一時記憶域キューの最大サイズを超えることが想定される場合は、印刷作業で GDDM を使用するか、QMF で使用する一時データ・キューを定義します。

GDDM サービスを使用した印刷の処理

GDDM サービス (QMF 以外) を使用して、ネイティブ z/OS バッチ、TSO、ISPF、および CICS で印刷を処理することができます。

このタスクについて

重要: このトピックの説明は、GDDM 製品に付属した GDDM のデフォルト値を使用する場合にのみ適用されます。

関連情報:

 [IBM Publications Center](#)
GDDM 資料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をとる方法

QMF は GDDM によって提供される標準インターフェースを介して GDDM ニックネームとインターフェースし、QMF が GDDM 印刷ファイルをオープンできるようにする呼び出しを発行します。

PRINT コマンドが開始される時、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 FSRCE 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。

GDDM がニックネームを検索する場所

PRINT コマンドでプリンター名を入力すると、GDDM は、CICS では単一のモジュールを検索し、ネイティブ z/OS バッチ、TSO、および ISPF では複数のデータ・セットを検索します。

プリンター・ニックネームを使用すると、複雑な印刷装置またはディスプレイ装置を定義でき、エンド・ユーザーの作業を単純化することができます。ニックネームは、GDDM に報告書のフォーマット設定方法および配布方法を指示する装置特性を定義します。またニックネームは、ローカル・デバイスおよびリモート・デバイスの両方を定義できます。

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、TSO、および ISPF で、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 ADMADFT を検索して、出力の方法および宛先が定義されているニックネームを探します。

CICS

CICS では、GDDM は、デフォルト・モジュール ADMADFC のみを検索します。GDDM は、ニックネームを使用して通信できるすべての装置 (ディスプレイ装置も含む) を認識します。

さまざまなプリンター・ファミリーのニックネームの例

以下の例では、さまざまなプリンター・ファミリーに GDDM プリンター・ニックネームを定義する方法を示しています。

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 の GDDM プリンターに定義するために、以下の例のように ADMMNICK 指定を使用するとします。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
```

- ネイティブ z/OS バッチ、TSO、および ISPF: TSO、ISPF、およびネイティブ z/OS バッチでニックネームを作成後、QMF PRINT コマンドを実行し、存在するニックネームを指定した結果として、一時データ・セットが作成されます。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 ニックネームを使って QMF オブジェクトを印刷する場合は、QMF に付属の GDDM マップ・グループを GDDM 印刷ユーティリティで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・グループを保持するデータ・セットの名前 (QMF1210.SDSQMAPE) が含まれます。

```
//ADMGGMAP DD DSN=QMF1210.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。

制約事項: CICS では、ADMMNICK 指定を作成した後、CICS にプリンターを定義することにより、名前に装置をリンクさせます。CICS RDO TERMINAL 定義内の TERMINAL 属性と、ADMMNICK 指定内の TONAME パラメーターが同じ値になっていることを確認してください。DFHTCT マクロを使用して CICS にプリンターを定義した場合は、ADMMNICK 指定内の TONAME パラメーターと、端末コントロール表内 (TCT) の TRMIDNT 属性が同じ値になっていることを確認してください。

ファミリー 3 GDDM プリンターのニックネーム例

ニックネーム 370PRINT をファミリー 3 の GDDM プリンターに定義するために、以下の例のように ADMMNICK 指定を使用するとします。

```
ADMMNICK NAME=370PRINT,TOFAM=3,DEVTOK=S3800W8,TONAME=370P
```

TSO、ISPF、およびネイティブ 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。続いて、選択したプリンターにフォーマット設定済みファイルを送信できます。

CICS で ADMMNICK 指定を作成したら、CICS に一時データ・キューを定義することによって、名前に装置をリンクさせます。ADMMNICK 指定の TONAME パラメーターと CICS RDO TDQUEUE 定義の TDQUEUE 属性が同じ値になっていることを確認してください。DFHDCT マクロを使用して一時データ・キューを定義した場合は、ADMMNICK 指定の TONAME パラメーターと DCT の DESTID 属性が同じ値になっていることを確認してください。

TSO、ISPF、またはネイティブ z/OS バッチのファミリー 4 GDDM プリンターのニックネーム例

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=A3820Q
```

出力は、デフォルトでは、DD 名 ADMIMAGE に書き込まれます。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 z/OS にスプールできます。ファミリー 4 プリンターは、CICS では使用できません。

関連概念:

277 ページの『特定プリンターのニックネーム定義の例』

以下の例は、ファミリー 1、2、または 3 の装置に使用できるプリンター・ニックネームを示しています。

関連情報:

 IBM Publications Center

有効な装置トークンに関する情報を GDDM 資料で検索します。

特定プリンターのニックネーム定義の例

以下の例は、ファミリー 1、2、または 3 の装置に使用できるプリンター・ニックネームを示しています。

3800、3812、または 3820 プリンター、8 行/インチ:

次の定義を使用して、ニックネーム GDDMPRT2 をファミリー 3 プリンター用に定義します。

```
GDDMPRT2 ADMMNICK TOFAM=3,DEVTOK=S3800N8,NAME=MYPRINT2
```

非 3800 システム・プリンター、132 桁、8 行/インチ:

次の定義を使用して、ニックネーム GDDMPRT3 をファミリー 3 プリンター用に定義します。

```
GDDMPRT3 ADMMNICK TOFAM=3,DEVTOK=S1403W8,NAME=MYPRINT3
```

リモート接続 3287 (図表の印刷に適切):

次の定義を使用して、ニックネーム GDDMPRT4 をファミリー 2 プリンター用に定義します。

```
GDDMPRT4 ADMMNICK TOFAM=2,DEVTOK=R87,NAME=MYPRINT4
```

印刷制御オプションのない任意の宛先:

次の定義を使用して、ニックネーム GDDMPRT5 をファミリー 3 プリンター用に定義します。

```
GDDMPRT5 ADMMNICK TOFAM=3,PROCOPT=((PRINTCTL,0)),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します。このキーワードは、数多くの印刷制御オプションの指定に使用します。例えば、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。

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。

GDDM-PCLK を使用する PC プリンター:

次の定義を使用して、ニックネーム PCPRINT をファミリー 1 プリンター用に定義します。

```
GDDMPRT6 ADMMNICK TOFAM=1,FAM=0,NAME=PCPRINT,TONAME=*,ADMPCPRT
```

このコマンドで、* はユーザーの現行装置またはデフォルト値を示しています。

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。

関連概念:

275 ページの『さまざまなプリンター・ファミリーのニックネームの例』

以下の例では、さまざまなプリンター・ファミリーに GDDM プリンター・ニックネームを定義する方法を示しています。

関連情報:

 IBM Publications Center

ファミリー 4 プリンターのニックネームおよび印刷制御オプションに関する情報を GDDM 資料で検索します。

印刷を処理するための GDDM サービスのセットアップ

GDDM サービスを使用して、ネイティブ z/OS バッチ、TSO、ISPF、および CICS で印刷を処理することができます。

手順

GDDM サービスを使用して印刷を処理するには、以下の手順に従ってください。

1. プリンターの GDDM ニックネームを選択します。
2. 『正しいタイプの GDDM 装置の選択』
3. 『プリンター・ニックネーム指定の作成』
4. 281 ページの『ニックネームのアクティブ化』
5. 280 ページの『GDDM デフォルト・モジュールのニックネームでの更新』
6. Q.PROFILES 表にあるユーザーの行の PRINTER フィールドを更新します。

関連概念:

275 ページの『GDDM がニックネームを検索する場所』

PRINT コマンドでプリンター名を入力すると、GDDM は、CICS では単一のモジュールを検索し、ネイティブ z/OS バッチ、TSO、および ISPF では複数のデータ・セットを検索します。

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

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・ディスプレイ装置などのディスプレイ装置も含まれることがあります。

ファミリー 2 装置

IBM 3270 ディスプレイ装置や待機プリンターなどの装置が含まれます。

ファミリー 3 装置

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

出力を印刷するために ADMOPUT および ADMOPUJ ユーティリティーを使用する必要があるプリンターです。これらのユーティリティーは GDDM により提供されます。ファミリー 4 装置は、CICS での印刷では使用できません。

プリンター・ニックネーム指定の作成

次の形式を使用して、GDDM プリンターのニックネームを指定します。

```
ADMMNICK NAME=nickname,TOFAM=family_type,DEVTOK=device_token,TONAME=name
```

図 36. プリンターのニックネームを指定する ADMMNICK 指定の形式

- 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 から 8 文字で示します。例えば、MYPRTR がニックネームの場合、ユー

ザーはコマンド PRINT REPORT (PRINTER=MYPRTR を入力できます。NAME は、単一名、コンマで区切られた名前のリスト、あるいは先頭または末尾に ? 文字が付けられた名前 (出力を類似の名前を持つ複数のプリンターに送信するためにワイルドカードとして使用されます) のいずれかです。

- 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を固有に識別します。
- TONAME は CICS のみで使用されます。このフィールドは、CICS 端末定義、または CICS 一時データ・キュー定義を指していますが、これらの定義は、CICS が GDDM とプリンター間の通信を適切に管理することを可能にします。TONAME パラメーターを使用して、装置のタイプによって異なる値を持つ 1 文字から 4 文字の CICS 定義名を指定します。
 - ニックネームでファミリー 1 または 2 プリンターを定義している場合、TONAME パラメーターには、対応する CICS 端末定義を指定する必要があります。

CICS オンライン・リソース定義 (RDO) を使用してプリンターを CICS に定義すると、CICS システム定義 (CSD) ファイルが更新され、TERMINAL 属性は TONAME と同じ値になります。

DFHTCT マクロを使用して CICS にプリンターを定義すると、CICS 端末コントロール表 (TCT) には項目が作成され、TRMIDNT フィールドは TONAME と同じ値になります。

- ニックネームでファミリー 3 を定義している場合、TONAME パラメーターには、対応する CICS 一時データ・キューを指定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 に一時データ・キューを定義すると、CICS システム定義 (CSD) ファイルが更新され、TDQUEUE 属性は TONAME と同じ値になります。

DFHDCT マクロを使用して CICS に一時データ・キューを定義すると、CICS 宛先コントロール表 (DCT) には対応する項目が作成され、DESTID フィールドは TONAME と同じ値になります。

ネイティブ z/OS バッチ、TSO、および ISPF へのニックネームの追加

ニックネームを、ADMDEFS のデータ・セットに追加します。GDDM は最初はこのデータ・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。ADMMNICK 指定の正しい形式については、278 ページの図 36 を参照してください。

CICS へのニックネームの追加

CICS でニックネームを作成するには、最初に、278 ページの図 36 に従って、GDDM 外部デフォルト・モジュール ADMADFC に GDDM ADMMNICK 指定

を定義する必要があります。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。

固有のラベルを構文に追加できます。例えば、GDDMPRT1 は以下の例にある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
```

1 つの定義で複数のニックネームの定義

ニックネーム定義で次のようにワイルドカード「?」を指定することにより、単一のニックネームを使用して複数のプリンター・アドレスを定義できます。

```
ADMNICK 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。以下に例を示します。

```
PRINT REPORT (PRINTER=MYPRINT2
```

この例では、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・セットに送信します。

関連概念:

277 ページの『特定プリンターのニックネーム定義の例』

以下の例は、ファミリー 1、2、または 3 の装置に使用できるプリンター・ニックネームを示しています。

275 ページの『さまざまなプリンター・ファミリーのニックネームの例』

以下の例では、さまざまなプリンター・ファミリーに GDDM プリンター・ニックネームを定義する方法を示しています。

関連情報:

 IBM Publications Center

GDDM 印刷ユーティリティおよびファミリー 4 印刷に関する情報を GDDM 資料で検索します。

GDDM デフォルト・モジュールのニックネームでの更新

プリンターのニックネーム指定を作成後、GDDM デフォルト・モジュール ADMADFC または ADMADFT を更新します。

このタスクについて

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ています。このモジュールは、SADMSAM データ・セットのメンバーとして保管されています。

手順

モジュールをニックネーム指定で更新するには、以下のようにします。

1. ソース・ファイルを編集してニックネームを追加します。

- ネイティブ z/OS バッチ、TSO、および ISPF では、外部デフォルト・モジュールは ADMADFT です。

- CICS では、外部デフォルト・モジュールは ADMADFC です。
2. モジュール内の ADMMDFT ステートメントの後に ADMMNICK 指定を入力します。
 3. 変更されたデフォルト・モジュールを再アSEMBルし、リンク・エディットします。

関連情報:

 IBM Publications Center

デフォルト・モジュールに関する情報を GDDM 資料で検索します。

ニックネームのアクティブ化

GDDM デフォルト・モジュールをニックネームの指定で更新した後は、このニックネームを使用できるようにテストし、取り込む必要があります。このプロセスは、独自の操作環境により異なります。

ネイティブ z/OS バッチ、TSO、および ISPF

ニックネーム定義をテストするには、それらを外部デフォルトのファイル内に入れ、これらを使用して印刷し、正しく作動することが確認できるまで行います。次に、これらを外部デフォルト・モジュールにアSEMBルします。

GDDM は外部デフォルト・モジュールを、データ・セットよりも効率的に使用して、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りません。

TSO、ISPF、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。ユーザーのログオン・プロシージャーに DD 名 ADMDEFS を追加するには、次のようなステートメントを使用します。

```
//ADMDEFS DD DSN=LOCAL.GDDM.NICKNAME,DISP=SHR
```

CICS

CICS では、ニックネームはユーザーのデフォルトの指定に組み込まれ、外部デフォルト・モジュール ADMADFC にアSEMBルされます。

ADMADFC モジュールを更新した後、CICS リソース定義を更新して、CICS がニックネームを管理対象の装置にリンクできるようにする必要があります。

- ファミリー 2 ニックネームを装置にリンク: QMF は、報告書での 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ニックネームを必要とします。CICS に VTAM または CICS の端末定義で指定されたプリンターがある場合、プリンターの GDDM ファミリー 2 装置の属性には待機 を指定する必要があります。ファミリー 2 装置を使用する場合、ADMMNICK 指定内の TONAME パラメーターには、CICS 端末定義を指定します (これに対して、ファミリー 3 装置では、CICS 一時データ・キューを指定します)。

例えば、次のニックネーム指定について考慮してください。

```
ADMMNICK NAME=GRAPHIC,TOFAM=2,DEVTOK=R87S,TONAME=GRAP
```

このニックネーム指定の場合は、下の例のようなマクロを使用して、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 37. ファミリー 2 GDDM プリンターのニックネームを CICS に定義

RDO を使用して、CICS にプリンターを定義するには、次の例に従ってください。

```
DEFINE TERMINAL(GRAP) TYPETERM(DFHSCSP) ...
```

- ファミリー 3 ニックネームを装置にリンク: ファミリー 3 装置を使用するには、次の構文を使用して GDDM ニックネーム表をセットアップします。ADMMNICK 指定に示されたパラメーターの意味については、278 ページの図 36を参照してください。

```
GDDMPRT ADMMNICK TOFAM=3,          X
          NAME=SYSVRT,             X
          DEVTOK=S1403W6,          X
          TONAME=SYSP
```

図 38. 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 文書 (IBM Publications Centerにあります) では、ニックネームをユーザーのデフォルト指定に組み入れて、ユーザーのデフォルト指定を外部デフォルト・モジュール ADMADFC にアセンブルするプロセスが説明されています。

TONAME パラメーターには、一致する CICS 一時データ・キューが必要です。RDO を使用すると、次に示すように、CICS に一時データ・キューを定義することができます。

```
DEFINE TDQUEUE(SYSP) TYPE(EXTRA) BLOCKSIZE(6050) DDNAME(ADMSYSP)
          RECORDFORMAT(VARIABLE) BLOCKFORMAT(BLOCKED)
          RECORDSIZE(260) TYPEFILE(OUTPUT) ...
```

一時データ・キューを CICS に DCT 項目として定義する方法については、283 ページの図 39を参照してください。

```

* THE GDDM NICKNAME IS SYSPRT AND THE
* LONGEST RECORD THAT CAN BE PRINTED
* IS 256.

      DFHDCT TYPE=SDSCI,DSCNAME=ADMSYSP,                X
      RECFORM=VARBLK,                                  X
      RECSIZE=260,BLKSIZE=6050,TYPEFLE=OUTPUT

      .
      .
* ENTRY FOR GDDM NICKNAME SYSPRT
SYSP    DFHDCT TYPE=EXTRA,DESTID=SYSP,DSCNAME=ADMSYSP,RSL=1

```

図 39. CICS DCT への TONAME 項目の追加

以下のように DD 名 ADMSYSP を CICS 始動 JCL に追加する必要もありません。

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

図 39 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。

印刷を処理するための QMF サービスの使用

(GDDM ではなく) QMF サービスを使用して、ネイティブ z/OS バッチ、TSO、ISPF、および CICS で印刷を処理することができます。

ネイティブ z/OS バッチ、TSO、および ISPF で印刷するための QMF サービス

DSQPRINT を使用して、報告書、表、SQL または QBE 照会、プロシージャー、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、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、JCL または CLIST を使用して DSQPRINT を割り振る必要があります。DSQPRINT を割り振る JCL の例を以下に示します。

```
//DSQPRINT DD DSN=&SYSUID..PRINT.DATA,DISP=MOD
```

以下の例は、CLIST で上記と同様のことを行ったものです。

```
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)
```

PRINT コマンドを発行する際には、紙送り制御 (CC) オプションに YES を設定してください。

ISPF を使用している場合は、QMF 提供の DPRE (印刷報告書の表示) コマンド同義語を使用して、報告書を印刷することなく、指定した幅と長さの値の効果を見ることができます。この視点は DSQPRINT を使用している場合にのみ適用されません。

関連概念:

292 ページの『DPRE 同義語: ISPF を使用した印刷報告書のプレビュー』
IBM は、報告書を印刷時と同じように表示する QMF コマンド同義語 DPRE を提供しています。DPRE コマンドを発行すると、QMF は報告書を完了し、DSQPRINT に割り振られたデータ・セットに報告書を印刷します。続いて、このデータ・セットを表示するために ISPF ブラウザーが呼び出されます。

CICS の印刷での QMF サービスの使用

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 一時記憶域キューには最大サイズがあり、データはローカル印刷の宛先にのみ経路指定されます。データを他の印刷宛先に経路指定する必要がある場合、または印刷出力が一時記憶域キューには大き過ぎる場合は、データを一時記憶域キューから一時データ・キューに経路指定するプログラム、または CICS 提供のトランザクション CEBR を使用して報告書を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 一時データ・キューは、CICS を開始する前にキューに定義された記憶域の量によってのみ制限されます。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。

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。例えば、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

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。以下に例を示します。

```
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。

大域変数を使用して印刷のためのキューを定義

PRINT コマンドでキュー・タイプとキュー名に値を指定しない場合、QMF は大域変数 DSQAP_CICS_PQTYPE および DSQAP_CICS_PQNAME 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、大域変数 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。

大域変数 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・キューの名前には 1 から 4 バイトを使用できます。一時記憶域キューの名前には 1 から 8 バイトを使用できます。デフォルトの一時記憶域キュー名は DSQPnnnn です。ここで nnnn はユーザーの 4 バイトの CICS 端末 ID です。例えば、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。同じ装置からの後続の PRINT コマンドでは、すべて同じキュー名が使用され、直前の報告書に追加されます。CICS コマンドについて詳しくは、DB2 QMF 解説書を参照してください。

CICS 一時記憶域キューからの報告書の表示

報告書を CICS 提供のトランザクション CEBR で表示することができます。

ユーザーが QMF を終了することなく印刷できるようにする

ユーザーが「印刷」ファンクション・キーを押すことで、ローカル印刷ユーティリティーを使用して QMF を終了することなくオブジェクトを印刷できるように、システムをカスタマイズすることができます。印刷のためのコマンド同義語を定義し、「印刷」ファンクション・キーをカスタマイズします。

手順

1. 現行オブジェクトをローカルで印刷するためのファイルを作成します。
 - ネイティブ z/OS バッチ、TSO、または ISPF で実行している場合

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。以下のサンプルは、PRTQMF という名前の QMF 呼び出し可能インターフェース・アプリケーションを使用したものです。

```
/* PRTQMF REXX EXEC for local DSPRINT */  
CALL DSQCIX "PRINT PROC (PRINTER=MYPRINT1"  
DSPRINT '&SYSUID..MYPRINT1.DATA'
```

この例では、定義された MYPRINT1 ニックネームを持っており、印刷出力を MYPRINT1.DATA と呼ばれるデータ・セットに出すことを想定します。

ユーザーによっては、PRINT コマンドの使用を回避して、ローカル印刷用へオブジェクトを単純にエクスポートすることを好む場合があります。この場合、次のような EXEC を使用します。

```
/* PRTQMF REXX EXEC for local print utilities called DSPRINT */  
CALL DSQCIX "EXPORT PROC TO MYPROC"  
DSPRINT '&SYSUID..MYPROC.PROC'
```

- CICS で実行している場合

PRT_QMF と呼ばれる QMF プロシージャを作成します。これはオブジェクトを一時記憶域に送り、その後オブジェクトを印刷するトランザクションを開始します。

```
PRINT REPORT (QUEUENAME=QMFREPT,QUEUETYPE=TS)  
CICS QMFP (FROM='QMFREPT')
```

2. 印刷のための QMF コマンド同義語を作成します。

以下のサンプル照会によって、PRTQMF プログラムを実行するためのコマンド同義語 PRTQMF が作成されます。

- ネイティブ z/OS バッチ、TSO、または ISPF で実行している場合

次の例のようなコマンド同義語を作成します。

```
INSERT INTO COMMAND_SYNONYMS (VERB, SYNONYM_DEFINITION, REMARKS)  
VALUES('PRTQMF','TSO PRTQMF','Print QMF Proc')
```

- CICS で実行している場合

次の例のようなコマンド同義語を作成します。

```
INSERT INTO COMMAND_SYNONYMS (VERB, SYNONYM_DEFINITION, REMARKS)  
VALUES('PRTQMF','RUN PRT_QMF','Print QMF Report')
```

3. このコマンド同義語を使用するようにファンクション・キーをカスタマイズします。各パネルのキーはカスタマイズする必要があります。

次の例では、ユーザーのプロファイルで PFKEYS 列値が PFKY_TABLE に設定されていることを想定しています。

- ネイティブ z/OS バッチ、TSO、または ISPF で実行している場合

「PROC」パネル上のファンクション・キーをカスタマイズします。次の例は、プロシージャ・パネル上のファンクション・キー 4 をカスタマイズするための照会を示しています。

```
INSERT INTO PFKY_TABLE (PANEL,ENTRY_TYPE,NUMBER,PF_SETTING)  
VALUES('PROC','K',4,'PRTQMF')
```

- CICS で実行している場合

報告書パネル上のファンクション・キーをカスタマイズします。REPORT
パネル上のファンクション・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 PFKEY_TABLE (PANEL,ENTRY_TYPE,NUMBER,PF_SETTING)
VALUES('REPORT','K',4,'PRTQMF')
```

4. QMF を再始動して、ファンクション・キーの変更を実施します。

関連タスク:

299 ページの『コマンド同義語のカスタマイズ』

デフォルトのコマンド同義語が要件に合わない場合は、独自の同義語を作成することが
できます。コマンド同義語が作成された後は、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。

307 ページの『QMF ファンクション・キーのカスタマイズ』

ユーザーがファンクション・キーを押すと QMF が発行するコマンド、およびファン
クション・キー表内のキー・ラベルをカスタマイズします。

オブジェクト・タイプごとの印刷の要件

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 49. 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・タイ プ | ニックネームの必要 性 | GDDM が制御を取得するとき | 出力の経路指定先 (z/OS の場合) |
|----------------|----------------|---|--|
| 図表 | あり | GDDM ICU は、PRINT コマ ンドが実行されるとき必ず制御を得 る。 | 出力は GDDM によって制御され る。 |
| 書式 | あり | 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 | なし | 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、GDDM は制御 を取得。 | 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 | なし | 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、GDDM は制御 を取得。 | 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 | なし | 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、GDDM は制御 を取得。 | 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 | あり | 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。 |

表 49. QMF オブジェクトおよびデータベース・オブジェクトのための印刷要件の要約 (続き)

| オブジェクト・タイプ | ニックネームの必要性 | GDDM が制御を取得するとき | 出力の経路指定先 (z/OS の場合) |
|------------|------------|---|---|
| 報告書 | なし | 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、GDDM は制御を取得。 | 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 | なし | 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、GDDM は制御を取得。 | 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 | なし | 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、GDDM は制御を取得。 | 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。 |

図表作成機能の有効化

QMF では、GDDM-PGF 製品に付属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することができます。

このタスクについて

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フォーマットを指定できます。ユーザーが QMF の DISPLAY CHART または EXPORT CHART コマンドを入力すると、図表形式とデータが組み合わされて、図形データ・ファイル (GDF) データが作成されます。図表フォーマットは、IBM 提供のものを使用することも、独自に作成することもできます。さらに、ユーザーは、新規に作成した図表フォーマットについて、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。

関連概念:

45 ページの『リクエスター (DB2 for z/OS) データベースのインストール前提条件』

スタンドアロン・データベースまたはリクエスター・データベースとして機能する DB2 for z/OS データベースに QMF をインストールするには、ハードウェアおよびソフトウェアの要件を満たす必要があります。

TSO および ISPF での図表サポートの有効化

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フォーマットを指定できます。

このタスクについて

ユーザーは、デフォルトの図表フォーマットを使用することも、独自の図表フォーマットを作成することもできます。さらに、ユーザーは、新規に作成した図表フォーマットについて、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。

手順

ユーザーが保管する図表フォーマットを保持するためのライブラリーを作成するには、以下のようにします。

1. 以下の DD ステートメントでライブラリーを作成します。

```
//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. ユーザーの QMF セッション用のライブラリーを割り振り、DD 名 DSQUCFRM を使用します。データ・セットはユーザーの TSO ログオン・プロシージャを使用して割り振ることができます。あるいは、ユーザーが QMF に到達するために呼び出す CLIST を使用して割り振ることができます。以下に例を示します。

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

次のタスク

デフォルトの図表形式は、デフォルトでは、ライブラリー QMF1210.SDSQCHRT 内にあります。このライブラリーを DD 名 ADMCFORM に割り振ります。ユーザー指定の図表形式はこのライブラリーとユーザー・ライブラリーの両方で検索されますが、最初に検索されるのは新規のライブラリーです。ユーザーが図表書式を保存する場合は必ず新規のライブラリーに保存され、QMF1210.SDSQCHRT に保存されることはありません。

このような仕組みにより、各ユーザーは、デフォルトの図表フォーマットおよびユーザーが保存したフォーマットの両方にアクセスできます。また、この仕組みにより、デフォルトの図表フォーマットが置き換えられることも回避できます。

CICS での図表サポートの有効化

QMF のインストール時に、デフォルトの図表形式を保持するために、データ・セットが作成されます。このデータ・セットは、CICS に、DSQUCFRM の名前を持つ CSD ファイル項目として記述されます。

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。DSQUCFRM データ・セットは、QMF から ICU を使用する際の図表フォーマットの保管に使用されるデフォルトの図表ライブラリーです。ICU パネル・ディレクトリーの拡張書式を使用すると、図表形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、CICS に記述して、QMF を実行している CICS 領域からアクセスできるようにする必要があります。図表ライブラリーは、CSD データ・セットのファイル項目で記述します。

QMF には、EXPORT CHART コマンドが用意されています。このコマンドは、図表全体を図形データ・フォーマット (GDF) で保管する場合に使用します。図表をエクスポートすると、GDF データは GDDM ADMF ライブラリーに保管されます。GDDM の ICU 機能を使用しても、図表全体を GDF で保管することができます。

関連情報:

 IBM Publications Center

拡張 ICU パネル・ディレクトリーに関する情報を GDDM 資料で検索します。

第 14 章 コマンド同義語

コマンド同義語を使用すると、独自の用語を定義し、それらを QMF、CICS、または TSO の各コマンドにリンクすることができ、QMF コマンドのカスタマイズに役立ちます。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があります。

QMF に用意されているデフォルト同義語の使用

QMF には、デフォルトでいくつかのコマンド同義語が用意されています。

デフォルト同義語のリスト

QMF にはいくつかのコマンド同義語が用意されており、これらの同義語は Q.COMMAND_SYNONYMS 表にあります。この表に対してアクセスするユーザーは、同義語を QMF コマンドであるかのように入力し、デフォルトのコマンド同義語の 1 つに関連付けられているアプリケーションを呼び出すことができます。

DB2 for Linux, UNIX, and Windows に接続している場合は、デフォルト同義語が表 Q.COMMAND_SYN_TSO に示されます。

これらのデフォルトの同義語は、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。QMF が CICS で実行されているときには、ISPF は利用できません。

DPRE (印刷報告書の表示)

印刷報告書の表示 REXX EXEC の同義語は DPRE です。この EXEC は、ユーザーの現行報告書を印刷書式で表示します。

BATCH (バッチ照会/プロシージャー)

バッチ・アプリケーションの同義語は BATCH です。このアプリケーションは、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。

ISPF (ISPF へのブリッジ)

ISPF へのブリッジの同義語は ISPF です。この同義語は、QMF の対話モードを一時的に離れ、ISPF/PDF セッションに「ブリッジ」できるようにします。セッションが完了したら、QMF の ISPF コマンドが実行されたポイントに戻ります。

LAYOUT (レイアウト書式)

レイアウト書式アプリケーションの同義語は LAYOUT です。このアプリケーションは、照会を実行せずに報告書をカスタマイズできるようにします。

RUNTSO

RUNTSO コマンドにより、QMF for Workstation および QMF for WebSphere のユーザーは、QMF for TSO に対するストアード・プロシージャー・インターフェースを使用して、照会およびプロシージャーを実行す

ることが可能になります。 RUNTSO コマンドを含む照会またはプロシージャーがストアード・プロシージャー・インターフェースから QMF for TSO に渡されると、QMF for TSO は、当初クライアントからの RUNTSO コマンドで渡されたパラメーターを使用して RUN コマンドを発行します。

ユーザーが QMF for Workstation または QMF for WebSphere のいずれかから RUNTSO コマンドを発行するには、ストアード・プロシージャー・インターフェースをインストールおよび構成しておく必要があります。

RU RU は RUN コマンドの同義語です。このコマンド同義語は、RUN コマンドの省略形として RU を使用する場合に必要です。

関連概念:

『DPRE 同義語: ISPF を使用した印刷報告書のプレビュー』

IBM は、報告書を印刷時と同じように表示する QMF コマンド同義語 DPRE を提供しています。 DPRE コマンドを発行すると、QMF は報告書を完了し、DSQPRINT に割り振られたデータ・セットに報告書を印刷します。続いて、このデータ・セットを表示するために ISPF ブラウザーが呼び出されます。

407 ページの『第 18 章 バッチ・モードでの 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。したがって、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。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

DPRE 同義語: ISPF を使用した印刷報告書のプレビュー

IBM は、報告書を印刷時と同じように表示する QMF コマンド同義語 DPRE を提供しています。 DPRE コマンドを発行すると、QMF は報告書を完了し、DSQPRINT に割り振られたデータ・セットに報告書を印刷します。続いて、このデータ・セットを表示するために ISPF ブラウザーが呼び出されます。

印刷報告書は、画面上に表示される報告書と正確に同じには見えません。表示報告書と印刷報告書には、以下の相違点があります。

- 1 つ以上の改ページを含む報告書は、スクロール可能な単一ページとして表示されます。 1 つ以上の改ページを含む印刷報告書には、複数のページが含まれません。
- ページ見出しと脚注は、表示報告書には 1 回表示されます。印刷報告書では、各ページの先頭と末尾に表示されます。
- 表示報告書を水平方向にスクロールしても、固定列は動きません。印刷報告書では、各ページの左側に固定列が繰り返し出力されます。

DPRE コマンドに関連付けられた同義語定義は、DSQA n RIC という名前の REXX EXEC を実行します (n は、QMF が実行されている言語に対応する 1 文字の各国

語 ID です)。DSQAnR1C は、DSQABR1C という名前の関連する EXEC を呼び出します。これらのアプリケーションは、すべてのユーザーが使用できるように共有されます。

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ます。

オブジェクトをロード後、次のコマンドを発行します。

```
DPRE
```

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。

各国語機能 (NLF) を使用する場合は、DPRE の変換済みコマンド同義語を発行して、印刷報告書を表示します。例えば、DPRE の変換済みドイツ語コマンド同義語は AGB です。他の言語環境での DPRE 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。n 文字を、ご使用の NLF 用の 1 文字の言語 ID に置き換えてください。

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、ユーザーのプロファイルから取り出されます。プロファイルに指定される WIDTH パラメーターは、論理レコード長 (LRECL) が 132 未満の場合に使用されます。そうでない場合は、レコード長 132 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。

エラーへの応答

DSQPRINT は、PRINTER=' ' が、QMF for TSO が ISPF の下で開始された 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。報告書が表示される前に何らかの理由で DPRE コマンドが停止した場合は、印刷出力は目的の場所に出力されない可能性があります。この問題を訂正するには、QMF を終了して新しいセッションを開始する必要があります。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

DPRE のカスタマイズ

TSO では、DSQABR1C EXEC を変更することで、DPRE コマンド同義語の動作をカスタマイズすることができます。

DSQABR1C EXEC には、プログラムの先頭にカスタマイズ・セクションが含まれています。このカスタマイズ・セクションを使用して、BROWSE データ・セットの

名前やプリンターの割り振りなどの設定を変更することができます。DSQABR1C EXEC には、カスタマイズ・セクションを変更するための命令が含まれています。

重要: ファイルを変更する際には、最初にその名前を変更し、オリジナル・ファイルおよび変更したファイルのバックアップ・コピーを必ず保存するようにしてください。

同義語のガイドライン

コマンド同義語には、verb、定義、およびオプションでオブジェクト名が含まれています。

同義語 verb

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 for CICS、または QMF for TSO に対するユーザー独自の語です。例えば、ユーザーの会社で計算結果を戻すプロシージャを実行する財務分析者がいる場合、QMF の基本 verb である RUN に対して、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 列の項目が有効であることを確認してください。

同義語表の VERB 列の項目は、以下の規則に従って入力する必要があります。

- 長さは 1 から 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ます。

表 50. コマンド同義語に有効なまたは無効な verb の例

| 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。

例えば、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には、コマンドの前に QMF を付けてください。例えば、同義語 DISPLAY が、QMF コマンドの RUN PROC SALES_REPORT を実行する同義語定義を表すとします。SALES_REPORT プロシージャは照会を実行し、報告書を GDDM 定義のプリンターで印刷します。DISPLAY の前に QMF の入力を忘れた場合は、必ずしも出したかったデータの形式で印刷報告書が受け取れるとは限りません。

いくつかの基本 QMF コマンドは、その後ろにキーワードまたはパラメーターを指定する必要があります。例えば、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。

同義語オブジェクト名

オブジェクト名はコマンド同義語ではオプションです。ただし、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。

同義語表の OBJECT 列の項目は、以下の規則に従って入力する必要があります。

- 長さは 1 から 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、二重引用符で囲まれなければならない。

QMF もデータベース・マネージャーも、名前を処理するとき、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。

表 51. コマンド同義語定義に有効なまたは無効な例

| 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) |

同義語定義

同義語定義は、ユーザーがコマンド同義語を入力したときに実行されるコマンド、プロシージャ、またはアプリケーションです。

定義の指針

SYNONYM_DEFINITION 列には、特定のコマンドを入力することができます。

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

- QMF プロシージャまたは照会を呼び出す RUN コマンド。

例えば、次のコマンドは、コマンド同義語 COMPUTE MONTHLY_SALES の同義語定義です。

```
RUN PROC JONES.SALES_DATA
```

- 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

例：線形プロシージャ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。

例えば、以下のプロシージャを検討します。

```
-- Procedure name: SALES_PROC  
RUN QUERY SALES_DATA  
PRINT REPORT (QUEUE=XYZ,QUEUETYPE=TS)  
TSO RPTX (FROM=('REPORTX, XYZ'))
```

図 40. 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 TSOTSO 仮想記憶域または CICS 一時記憶域キューに経路指定します。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、報告書を記憶域から事前定義印刷宛先へ経路指定します。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。

このプロシージャを実行する同義語の定義は、以下のようになります。

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

図 41. コマンド同義語を使用して線形プロシージャを実行する例

各国語機能 (NLF) を使用する場合、照会、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内のコマンドは、コマンド同義語を使用する前に、必ず変換されていなければなりません (プロシージャで DSQEC_NLFCMD_LANG 変数を 1 に設定した場合を除きます。その場合、ユーザーはコマンドを英語で発行する必要があります)。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。

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。

例えば、次の例は、表名に値 Q.STAFF を渡す定義を示しています。これは MYQUERY の実行時に評価されます。

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

図 42. 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
```

アンバーサンドは、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バーサンドになります。

変数 &ALL

&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。変数 &ALL 以外のすべての変数には、同義語定義でアンバーサンドを 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、同義語定義内の変数情報の所在を示すことができます。

この同義語定義は、&ALL 変数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 43. 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、次の例のようになります。

```
SELECT * FROM Q.STAFF  
WHERE DEPT=&DEPT and JOB=&EMPLOYEE_JOB
```

図 43 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'
```

変数 &ALL を同義語定義の中で使用するときは、以下のガイドラインを守ってください。

- 同義語定義では &ALL は一度だけ使用する。
- &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。

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。297 ページの図 43 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

コマンド同義語定義を同義語表に入力する際には、特定の規則に従う必要があります。

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。以下に例を示します。

```
RUN MYQUERY (&&NAMEVALUE='&ALL')
```

名前 MARENGHI を検索する場合、'MARENGHI' 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。例えば、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。

コマンド同義語のカスタマイズ

デフォルトのコマンド同義語が要件に合わない場合は、独自の同義語を作成することができます。コマンド同義語が作成された後は、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。

コマンド同義語表の作成

ユーザーが QMF セッションを開始すると、QMF は、ユーザーが自身のプロファイルの SYNONYMS フィールドに指定した名前を持つコマンド同義語表をロードします。

このタスクについて

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があるかどうかを検査します。一致するものが見つからない場合、QMF はそのコマンドを基本 QMF コマンドとして解釈します。QMF で同義語表が検査されないようにするには、文字 QMF を任意のコマンドの前に入力します。

手順

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ローカル DB2 for z/OS サブシステム内に獲得または追加します。使用可能な表スペースがない場合は、次に示すようなステートメントを使用して、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 44. 表スペースの作成

このステートメントは、表スペース DSQTSSN1 を作成します。この表スペースの記憶域グループとデータベースは、デフォルトの同義語表 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、以下のような CREATE TABLE ステートメントを実行します。COMMAND_SYNONYMS をユーザーの表の名前、また DSQTSSN1 の固有の表スペース名をユーザーの固有の表の名前で置き換えます。ステートメントの他の部分は、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 DSQTSSN1

```

図 45. 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、各同義語の目的を識別することができます。

3. 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・カタログに追加する必要はありません。ただし、表に関する 1 つのコメントと、列を記述するその他のコメントを追加することを検討してください。例えば、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てします。これを説明するコメントを加えるには、次のようなステートメントを含む照会を実行します。

```
COMMENT ON COLUMN COMMAND_SYNONYMS.AUTHID
IS 'PRIVATE SYNONYM: USE AUTH ID. PUBLIC SYNONYM: USE NULL'
```

後続の COMMENT ON ステートメントを発行することにより、現行のコメントを置き換えることができます。

4. QMF がコマンド同義語表を処理するとき、索引を作成して、初期化時のパフォーマンスが最大にします。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、重複が見つかったら、QMF トレース・データに警告メッセージを記録します。

関連情報:

 DB2 資料

COMMENT ON ステートメントに関する情報を検索します。

コマンド同義語定義を表に入力

コマンド同義語表を作成後、SQL INSERT ステートメントを使用して、同義語を表に入力します。表エディターを使用して表を更新することもできます。

この SQL INSERT ステートメントは、表に同義語を入力します。

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

図 46. コマンド同義語定義の作成

これが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。

次の例のステートメントは、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 47. 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。

これらの例にある同義語は、QMF が各同義語を正しく処理できるようにするための、同義語のガイドラインに従います。

関連概念:

294 ページの『同義語のガイドライン』

コマンド同義語には、verb、定義、およびオプションでオブジェクト名が含まれています。

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あてはまります。

このタスクについて

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。例えば、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。

手順

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、以下のようにします。

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します。この列は、VARCHAR(261) として定義されています。これにより、128 バイトの表所有者 ID と表名が使用できます。

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。

次の例は、COMMAND_SYNONYMS 表を英語のユーザー JONES に割り当て、GUMMOW.XYZ 表をドイツ語 NLF 環境のユーザー SCHMIDT に割り当てる方法を示しています。

表 52. ユーザーの QMF コマンド同義語のアクティブ化

| 基本 QMF (英語) | ドイツ語 NLF |
|--|--|
| <pre>UPDATE Q.PROFILES SET SYNONYMS='COMMAND_SYNONYMS' WHERE CREATOR='JONES' AND TRANSLATION='ENGLISH' AND ENVIRONMENT='TSO'</pre> | <pre>UPDATE Q.PROFILES SET SYNONYMS='GUMMOW.XYZ' WHERE CREATOR='SCHMIDT' AND TRANSLATION='DEUTSCH' AND ENVIRONMENT='TSO'</pre> |

前のステートメントは、QMF に既に登録されているユーザーに適用されます。類似のステートメントを使用して 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 ステートメントを使用します。

2. SQL SELECT 特権を PUBLIC に付与して、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します。以下に例を示します。

```
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 をその視点だけに許可し、使用の対象としない同義語にアクセスするのを阻止します。

次のタスク

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。

関連タスク:

303 ページの『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許容範囲内に抑えるには、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。

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、コマンド同義語表を再編成することが必要な場合があります。

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。

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ったものです。

このステートメントは、デフォルト名 Q.COMMAND_SYNONYMS を持つ表を、基本 (英語) QMF を使用するすべてのユーザーに割り当てます。

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

図 48. 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許容範囲内に抑えるには、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、各同義語についてパブリック (全ユーザー) またはプライベート (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。

手順

各同義語に共用または私用のフラグを立てる視点を作成して割り当てるには、以下の手順に従ってください。

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 NULL にしておく、それは共用同義語であることを示し、AUTHID 列にユーザー ID を入れると、それは私用同義語を示します。同じ同義語に多数の項目を割り当て、それぞれの項目を異なるユーザーに割り当てるのが可能です。
2. 以下のようなステートメントを使用して、同義語表に視点を作成します。このステートメントにより、ユーザー (図で *userid* で示されている) は、表中のすべてのパブリック同義語、および自分の SQL 許可 ID に対してプライベート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 49. 同義語の個人使用および共用を制御する視点の作成

共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。

このタスクについて

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、それぞれのパブリック同義語 1 つにつき 1 行を使用した行からなります。AUTHID はヌルか、そのユーザー・グループを一意的に識別する値を持ちます。

手順

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. 以下のようなステートメントを使用して、同義語表に視点を作成します。このステートメントは、共通ユーザー ID が DEPTD02 であるユーザー・グループ用に作成される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 50. 同義語のグループ使用および共用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。

以下のようなステートメントを使用して、コマンド同義語を保持するこの方法を実施できます。このステートメントは、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 51. 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 15 章 カスタム・ファンクション・キー

各 QMF パネルのファンクション・キーのデフォルト設定とラベルは、エンド・ユーザーが一般的に行う QMF のタスクの共通セットを表しています。ただし、各サイトの要件は固有であるため、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ときに実行される QMF コマンドの両方をカスタマイズするための手段が用意されています。

QMF ファンクション・キーのカスタマイズ

ユーザーがファンクション・キーを押すと QMF が発行するコマンド、およびファンクション・キー表内のキー・ラベルをカスタマイズします。

このタスクについて

QMF ファンクション・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 などのフルスクリーン・パネル) と 2 次パネル (ウィンドウ・ダイアログ・パネルとして開くパネル) 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。

パネル ID の表示

ファンクション・キー表の PANEL 列に、ファンクション・キーをカスタマイズするパネルのパネル ID を入力します。大域変数 DSQDC_SHOW_PANID を使用して、パネル ID を表示します。

このタスクについて

大域変数 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。

手順

パネル ID を表示するよう大域変数を設定するには、次のコマンドを使用します。

```
SET GLOBAL (DSQDC_SHOW_PANID=1
```

フルスクリーン・パネル ID

QMF の英語の基本製品のフルスクリーン・パネル ID

各 QMF 各国語機能 (NLF) の有効なフルスクリーン・パネル ID は、言語固有版の DSQ22957 メッセージに示されています。NLF のメッセージを確認するには、NLF の実行中に QMF コマンド行から HELP DSQ22957 コマンドを発行します。

ファンクション・キー表にファンクション・キー定義を入力する際には、以下に示すとおり、またはメッセージ・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。

- FORM.BREAK1
- FORM.BREAK2

- FORM.BREAK3
- FORM.BREAK4
- FORM.BREAK5
- FORM.BREAK6
- FORM.CALC
- FORM.COLUMNS
- FORM.CONDITIONS
- FORM.DETAIL
- FORM.FINAL
- FORM.MAIN
- FORM.OPTIONS
- FORM.PAGE
- GLOBALS
- HOME
- PROC
- PROFILE
- PROMPTED QUERY
- REPORT
- SQL QUERY

ウィンドウ・パネル ID

ファンクション・キー表にファンクション・キー定義を入力する際に使用するウィンドウ・パネル ID。

パネル ID が表示されるように大域変数 DSQDC_SHOW_PANID を設定した場合、これらの表に示されている各 ID には、表示時に 4 文字が先頭に付いていることがわかります。

表で名前が付けられていないウィンドウ・パネルには固有のパネル ID がいないため、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できます。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、ID の一部です。

詳しくは、以下のいずれかのセクションを選択してください。

- 『コマンド・ウィンドウ』
- 309 ページの『書式ウィンドウ』
- 309 ページの『大域変数ウィンドウ』
- 309 ページの『ヘルプおよびプロンプト・ウィンドウ』
- 309 ページの『ロケーション・ウィンドウ』
- 309 ページの『オブジェクト・リスト・ウィンドウ』
- 310 ページの『指示照会ウィンドウ』

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 |

大域変数ウィンドウ

| パネル ID | 表題または説明 |
|--------|--|
| GLADVA | 追加変数 |
| GLSHVA | 個々の大域変数の表示 (このラベルは、ユーザーが GLOBALS パネルの Show Field キーを押すと表示されるパネルに関連付けられません。) |
| GLXXXX | 大域変数ウィンドウ・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 | サブジェクト・ヘルプ・ウィンドウ・クラス (エラー・メッセージ・ヘルプ・パネルは、異なる命名規則に従っています。) |
| 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・リストのソート |
| 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 |
| QPCDIT | 条件コネクター |
| QPCFOR | 時間枠の変更 |
| QPCOCH | 列 - 変更内容 |
| QPCODE | 列記述 |
| 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 |
| QPERSP | 時間枠の指定 |
| QPEXPR | 式 |
| QPJOCO | 結合列 |
| QPJOTA | 結合表 |
| QPROBE | 行 - 間 |
| QPROCH | 行 - 変更内容 (左側) |
| QPROCT | 行 - 含む |
| QPROC1 | 行 - 比較演算子 1 |
| QPROC2 | 行 - 比較演算子 2 |
| QPROEN | 行 - 終了 |
| QPROEQ | 行 - 等しい |
| QPROGQ | 行 - より大きいか等しい |
| QPROGR | 行 - より大きい |
| QPROLQ | 行 - より小さいか等しい |
| QPROLS | 行 - より小さい |
| QPROST | 行 - 開始 |
| QPROWS | 行 (行条件) |
| QPSHFI | フィールド表示 |
| QPSHSQ | SQL 表示 |
| QPSOCH | ソート - 変更 |
| QPSORT | ソート |
| QPSPEC | 指定 |
| QPTABL | 表 |
| QPITAB | 時間枠を指定する表または視点の選択 |
| QPXXXX | PQ ウィンドウ・クラス |

カスタマイズしたいキーの選択

デフォルトのキー・ラベルはパネル・タイプによって異なります。カスタマイズしたいファンクション・キーを選択してください。

表エディター・パネルでファンクション・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 コマンドまたは地域別定義コマンドを選択して、変更するファンクション・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

下の表に、フルスクリーン・パネルのデフォルト・ファンクション・キー設定を示します。

表 53. フルスクリーン・パネルのデフォルトのファンクション・キーとコマンド

| デフォルト・キー・ラベル | キーを押したときに発行されるデフォルト・コマンド |
|--------------|--------------------------------|
| Backward | BACKWARD |
| Cancel | CANCEL |
| Change | CHANGE |
| Chart | DISPLAY CHART または SHOW CHART |
| Check | CHECK |
| Clear | CLEAR |
| Command | SHOW COMMAND |
| Comments | SWITCH COMMENTS |
| Delete | DELETE |
| Describe | DESCRIBE |
| Draw | DRAW |
| Edit Table | EDIT TABLE |
| End | END |
| Enlarge | ENLARGE |
| Form | DISPLAY FORM または SHOW FORM |
| Forward | FORWARD |
| Help | HELP |
| Insert | INSERT |
| Left | LEFT |
| List | LIST |
| Print | PRINT |
| Proc | DISPLAY PROC または SHOW PROC |
| Profile | DISPLAY PROFILE |
| Query | DISPLAY QUERY または SHOW QUERY |
| Reduce | REDUCE |
| Refresh | REFRESH |
| Report | DISPLAY REPORT または SHOW REPORT |
| Retrieve | RETRIEVE |
| Right | RIGHT |

表 53. フルスクリーン・パネルのデフォルトのファンクション・キーとコマンド (続き)

| デフォルト・キー・ラベル | キーを押したときに発行されるデフォルト・コマンド |
|--------------|--------------------------|
| 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

下の表に、ウィンドウ・パネル上のデフォルト・ファンクション・キー、および各キーが押されたときに実行されるコマンドを示します。

表 54. ウィンドウ・パネルのデフォルトのファンクション・キーとコマンド

| デフォルト・キー・ラベル | キーを押したときに発行されるデフォルト・コマンド |
|---------------------|--------------------------|
| Attribute | SPECIFY ATTRIBUTES |
| Backward | BACKWARD |
| Cancel | CANCEL |
| Clear | CLEAR |
| Command | SHOW COMMAND |
| Comments | SWITCH COMMENTS |
| Condition | SPECIFY CONDITION |
| Delete | DELETE |
| Describe | DESCRIBE |
| End | END |
| Exit | END |
| Forward | FORWARD |
| Help | HELP |
| Index | HELP INDEX |
| Keys | HELP KEYS |
| 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 |

表 54. ウィンドウ・パネルのデフォルトのファンクション・キーとコマンド (続き)

| デフォルト・キー・ラベル | キーを押したときに発行されるデフォルト・コマンド |
|--------------------|--------------------------|
| Show Field | SHOW FIELD |
| Show View | SHOW VIEW |
| Sort | SORT |
| Specify Attributes | SPECIFY ATTRIBUTES |
| Specify Condition | SPECIFY CONDITION |
| Switch | HELP SWITCH |

大域変数リスト・パネルでは、RESET GLOBAL は Delete ファンクション・キーが押されたときに実行されるコマンドです。

ファンクション・キー表の作成

どのファンクション・キーをカスタマイズしたいかを決定後、カスタマイズ済みファンクション・キー定義を適切なパネルにリンクする表を作成できます。

手順

ファンクション・キー表を作成するには、以下の手順に従ってください。

- 次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_PFKKEYS をユーザー独自の名前で置き換えます。DSQTSSN1 をユーザー独自の表スペースで置き換えます。

```
CREATE TABLE MY_PFKKEYS
(PANEL          CHAR(18)          NOT NULL,
ENTRY_TYPE     CHAR(1)           NOT NULL,
NUMBER         SMALLINT         NOT NULL,
PF_SETTING     VARCHAR(254),
REMARKS       VARCHAR(254))
IN DSQTSSN1
```

図 52. 機能キー表の作成

- 次のような 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_PFKKEYS 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。

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。例えば、MY_PFKKEYS は私用と共用のファンクション・キーを区別する 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、次の例のようなステートメントを含む照会を実行します。

```
COMMENT ON COLUMN MY_PFKEYS.AUTHID
IS 'PRIVATE PFKEY: USE AUTH ID. PUBLIC PFKEY: USE NULL'
```

後続の COMMENT ON ステートメントを発行することにより、現行のコメントを置き換えることができます。

3. 次のような 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は警告メッセージをログに記録し、それらの警告メッセージ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。

関連情報:

 DB2 資料

表スペースの作成に関する情報、およびシステム・カタログへのコメントの追加に関する情報を検索します。

ユーザーのファンクション・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、キー定義をファンクション・キー表に挿入します。

各ファンクション・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、画面に表示されるキー・ラベルを指定します。

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。関連付けられたラベルのないファンクション・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、関連付けられたコマンドのないラベルは非アクティブです。

関連概念:

318 ページの『キー定義の例』

このトピックの例を使用して、各タイプの QMF パネルに関する完全なファンクション・キー定義の入力方法を確認してください。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。

コマンドとファンクション・キーの結び付け

QMF パネルの各ファンクション・キーは、そのファンクション・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。ファンクション・キーがアクティブになるようにするには、コマンドを識別する行をファンクション・キー表に追加します。

カスタマイズしたファンクション・キーもこのように機能させるには、表に入力する 2 行のうちの 1 行に、下の表に示す値を指定します。

表 55. 2 行のうちの最初の値は、各ファンクション・キー定義に必要です。

| 列 | 値 | 情報 |
|------------|-------------------------|---|
| PANEL | カスタマイズする QMF パネルの ID | <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。例えば、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 を使用して行った変更は、そのクラス ID によりカスタマイズされる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 | カスタマイズするファンクション・キーの番号 | 例えば、F5 の定義を変更する場合は、5 をこの列に入力します。 |
| 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。例えば、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、新規のファンクション・キー初めて使用する前に、それぞれの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 がファンクション・キーに関連付けられたコマンドを取り出すときに、ディスプレイ装置へ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ことも確認します。

ファンクション・キーにラベルを付けて画面上に配置

各 QMF パネルのファンクション・キーは、ファンクション・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ようにするには、ラベルを識別する行をファンクション・キー表に追加します。

この行で、下の表に示されている値がファンクション・キー表の列にあることを確認してください。

表 56. 2 行のうちの 2 番目の値は、各ファンクション・キー定義に必要です

| 列 | 値 | 情報 |
|------------|--|--|
| PANEL | カスタマイズする QMF パネルの ID | これは、定義の最初の行に使用したものと同一 ID です。 |
| ENTRY_TYPE | L | L は、この行がファンクション・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。 |
| NUMBER | フルスクリーン・パネルをカスタマイズしているか、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かによって決まります | <p>フルスクリーン・パネルをカスタマイズしている場合は、これは画面上にキーが表示される行の番号です。例えば、ホーム・パネルでは、F5 は行 1 に表示されるので、この定義の NUMBER 列は値 1 を持ちます。</p> <p>ウィンドウ・パネルまたはヘルプ・パネルをカスタマイズしている場合、NUMBER はファンクション・キーの番号を表しています (コマンドをリンクする表に追加した最初の行の場合と同様)。</p> |
| PF_SETTING | ファンクション・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。例えば、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かどうかにかかわらず、そのパネル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しないキーのデフォルト・キー設定を自動的に割り当てません。例については、318 ページの図 53 を参照してください。</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 319 ページの図 56 および 319 ページの図 57 に示されています。</p> |

新規ファンクション・キー定義のアクティブ化

作成したカスタマイズ済みファンクション・キー定義をユーザーが使用できるようにするには、これらの定義をアクティブ化する必要があります。

手順

1. ユーザー・プロファイルの PFKEYS フィールドを、ファンクション・キー定義表の名前で更新します。

以下の例では、左側のステートメントは英語の QMF ユーザー JONES に表 MY_PFKEYS を、また右側のステートメントはドイツ語の NLF ユーザー SCHMIDT に表 MEIN_PFKY を割り当てます。

重要: Q.PROFILES 表を更新するステートメントの列 TRANSLATION および ENVIRONMENT には、常に値を指定してください。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。

表 57. カスタマイズされたファンクション・キーをユーザーからアクセス可能にする

| 基本 QMF (英語) | ドイツ語 NLF |
|---|---|
| <pre>UPDATE Q.PROFILES SET PFKEYS = 'MY_PFKEYS' WHERE CREATOR='JONES' AND TRANSLATION = 'ENGLISH' AND ENVIRONMENT = 'TSO'</pre> | <pre>UPDATE Q.PROFILES SET PFKEYS = 'MEIN_PFKY' WHERE CREATOR='SCHMIDT' AND TRANSLATION = 'DEUTSCH' AND ENVIRONMENT = 'TSO'</pre> |

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。以下に例を示します。

```
GRANT SELECT ON MY_PFKEYS TO PUBLIC
```

お客様サイトでのファンクション・キーの保守を最小限に抑えるために、表の視点を割り当てることができます。視点に対してのみ SELECT 特権を付与して、ユーザーが使用以外の目的でファンクション・キーにアクセスできないようにします。

ファンクション・キー表の視点の割り当てに関する方針を使用して、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てかどうかを決定してください。

次のタスク

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。

関連タスク:

303 ページの『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許容範囲内に抑えるには、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。

ファンクション・キー表に関する問題の検査と診断

ファンクション・キー表の名前をユーザーの Q.PROFILES 項目に挿入して新しいファンクション・キー定義をアクティブ化すると、新しい定義がテスト可能になります。

手順

新しい定義のテストは、以下の 2 つの方法のいずれかで行うことができます。

- QMF を終了して、新しい QMF セッションを開始します。
- QMF 内で `CONNECT TO servername` コマンドを入力して再接続します。ここで、`servername` は QMF ホーム・パネルに表示されるロケーション名と同じものです。

タスクの結果

QMF への再接続後に新規のファンクション・キー定義が表示されない場合は、QMF トレース出力にエラーまたは警告メッセージが表示されていないかどうかを確認してください。

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大域変数 `DSQAP_PFKEY_TABLE` を検査します。この大域変数に、新たに作成または変更したファンクション・キー表の名前が含まれていない場合は、Q.PROFILES 表のユーザーの行を確認してください。

キ一定義の例

このトピックの例を使用して、各タイプの QMF パネルに関する完全なファンクション・キー定義の入力方法を確認してください。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。

示されている例では、フルスクリーン・パネルおよびウィンドウ・パネルをリストする表からのパネル ID を使用しています。

フルスクリーン・パネルでキーの定義を入力

以下の SQL ステートメントを使用して、ホーム・パネル上の F2 を EDIT TABLE から IMPORT に変更します。ホーム・パネルをパネル ID HOME で識別し、ユーザーが F2 を押したときに実行するコマンドをカスタマイズするため、番号 2 (表示された最初のステートメント) 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=Help 2=Import 3=End 4=Show 5=Chart 6=Query')
```

図 53. ホーム・パネルで QMF コマンドのファンクション・キーを変更

QMF ホーム・パネルはここで、次に示すように、Import を表示します。

```
Type command on command line or use PF keys. For help, press PF1 or type HELP.
-----
1=Help      2=Import    3=End      4=Show     5=Chart    6=Query
7=Retrieve  8=Edit Table 9=Form     10=Proc    11=Profile 12=Report
OK, cursor positioned.
COMMAND ==>>
```

図 54. ファンクション・キー表に定義を追加後、カスタマイズされたファンクション・キーが画面に表示される

2 番目の INSERT ステートメントの PF_SETTING 列に、ホーム・パネルの上側のキー行に表示される内容を正確に入力します。各キーがカスタマイズされていない場合でも、同様に行ってください。例えば、2 番目のステートメントの PF_SETTING 列に語 Import のみを指定すると、ホーム・パネルは以下のようになります。

```
Type command on command line or use PF keys. For help, press PF1 or type HELP.
-----
Import
7=Retrieve  8=Edit Table 9=Form     10=Proc    11=Profile 12=Report
OK, cursor positioned.
COMMAND ==>>
```

図 55. ファンクション・キー表に新しいキー定義を挿入したときに最初のキー行を指定しないと、その行は欠落する

ウィンドウ・パネルのキーの定義の入力

以下の SQL ステートメントは、F3 キーを指示照会の「表」パネルに追加します。このファンクション・キーは、CANCEL コマンドを実行し、CancelMe というラベルを持っています。

```
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 56. 指示照会の Specify パネルのファンクション・キーの変更

ヘルプ・パネルまたはプロンプト・パネルのキー定義の入力

以下の SQL ステートメントは、F13 キーをすべてのヘルプ・パネルに追加します。このファンクション・キーは、CANCEL コマンドを実行し、CancelMe というラベルを持っています。

```
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 57. 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。

関連資料:

307 ページの『フルスクリーン・パネル ID』
QMF の英語の基本製品のフルスクリーン・パネル ID

308 ページの『ウィンドウ・パネル ID』
ファンクション・キー表にファンクション・キー定義を入力する際に使用するウィンドウ・パネル ID。

第 16 章 QMF 書式のためのカスタム編集出口ルーチン

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。デフォルト編集コードがお客様のサイトの報告書編集の要件に合わない場合は、独自の編集コードを作成することができます。

FORM.MAIN および FORM.COLUMNS パネルの EDIT フィールドにある編集コードを使用すると、報告書データをさまざまな方法でフォーマット設定できます。例えば、給与データを戻す列では 10 進数編集コードを使用できます。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。

編集出口ルーチンおよび QMF

QMF と編集出口ルーチンは連携し、定義した編集コードを使用してデータをフォーマット設定します。

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

次のイメージは、ユーザー編集ルーチンと QMF との連携方法を示しています。

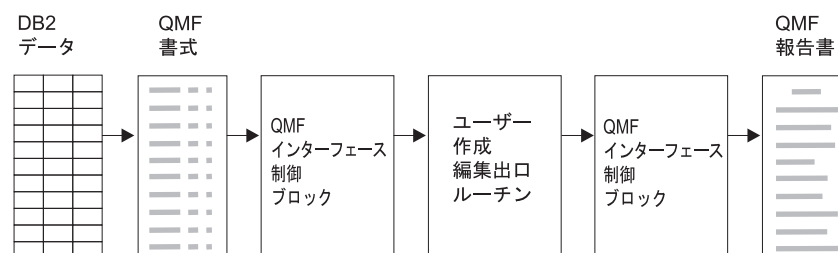


図 58. ユーザー編集ルーチンと QMF for TSO/CICS との連携方法

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ース制御ブロックに渡します。これらの特性は、制御ブロックの特定のフィールドにあります。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。

出口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口ルーチンを呼び出し、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口ルーチンとの情報の受け渡しが行われます。その中には、形式設定されるデータと形式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、編集ルーチンは以下のパラメーターを受け取ります。

- インターフェース制御ブロックからの制御情報
- ECSINPT の値。すなわち、形式設定される入力域からのデータ
- 形式設定結果の出力域を含む ECSRSLT

ECSRSLEN には出力域の長さが含まれます。フォーマット設定の結果が出力域に収まらない場合、出力域は列の折り返しを行うことができないため、その結果は切り捨てられます。

重要: ECSRSLEN フィールドの値で示されているよりも多く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、エラー「DSQ60439: USER EDIT PROGRAM MEMORY OVERWRITE」が出されます。この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。

- QMF 書式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、列の幅を広げます。
- 編集出口プログラムをコード化して、ECSRSLEN フィールドの値を確認して、QMF に渡される結果をユーザーのプログラムで埋め込むかまたは切り捨てるかどうかを決定するようにします。

ECSINPT、ECSRSLT、および ECSRSLEN は、インターフェース制御ブロックのフィールドです。

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 の終了時に出口ルーチンが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。

関連概念:

325 ページの『入出力域を特徴付けるフィールド』

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。出力フィールドには、フォーマット設定した結果を入れる場所に関する情報が含まれています。

関連資料:

323 ページの『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、ユーザー出口ルーチンとの間で情報の受け渡しを行います。

出口ルーチンとの間で受け渡されるフィールド

インターフェース制御ブロックおよび入出力域のこれらのフィールドには、編集出口ルーチンとの間で受け渡される情報が含まれています。

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、ユーザー出口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、ユーザー編集出口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、下の表に示されています。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。

編集出口インターフェース制御構造の言語固有のバージョンは、QMF1210.SDSQUSR n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。

- ハイレベル・アセンブラーの場合は、DSECT DXEECSA を使用する
- PL/I の場合は、コピー・ファイル DXEECSF を使用する
- COBOL の場合は、コピーブック DXEECSB を使用する

表 58. QMF 編集出口インターフェース制御ブロックのフィールド、DXEECS

| 名前 | 説明 |
|----------|---|
| ECSDECPT | QMF プロファイルの DECOPT オプションにより決められた現行の小数点記号 (ピリオドまたはコンマ) を含んでいる。 |
| ECSECODE | ユーザー編集コードが入っている。 |
| ECSERRET | 呼び出し時点では、ゼロが入っている。エラーを記録するには、これをゼロ以外の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。 番号 エラー 99101 認識できない編集コード 99102 編集コードに不適切な入力データ・タイプ 99103 フォーマット設定される項目に無効な入力値 99104 フォーマット設定される項目が短過ぎる 99105 ECSRSLT の結果用にスペースが不十分 (結果が割り当てスペースに対し幅が広すぎる) 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、一般的なヘルプ・パネルと共に、一般的なエラー・メッセージが表示される。 |
| ECSFREQ | フォーマット設定の呼び出しでは E に、終了呼び出しでは T になる。 |
| ECSINDTA | 形式設定される値に関する情報を含む。 |
| 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。 |

表 58. QMF 編集出口インターフェース制御ブロックのフィールド、DXECS (続き)

| 名前 | 説明 |
|----------|---|
| ECSINNUL | フォーマット設定される値が NULL の場合は、N になる。 |
| 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。 |
| ECSINTYP | 形式設定される値のデータ・タイプの内部データベース表記。ここに示されている各データ・タイプのコードは、SQLDA の SQLTYPE フィールドに表示される値と同じです。 384 DATE データ・タイプ 388 TIME データ・タイプ 392 TIMESTAMP データ・タイプ 448 VARCHAR データ・タイプ 452 CHAR データ・タイプ 456 LONG VARCHAR データ・タイプ 464 VARGRAPHIC データ・タイプ 468 GRAPHIC データ・タイプ 472 LONG VARGRAPHIC データ・タイプ 480 FLOAT データ・タイプ 484 DECIMAL データ・タイプ 492 BIGINT データ・タイプ 496 INTEGER データ・タイプ 500 SMALLINT データ・タイプ 908 VARBINARY データ・タイプ 912 BINARY データ・タイプ 940 拡張 FLOAT 996 DECFLOAT データ・タイプ (長形式および拡張形式の両方) 2448 TIMESTAMP WITH TIME ZONE データ・タイプ |
| ECSNAME | 制御ブロックの名前である DXECS が入っている。記憶域ダンプで、目印としての役割を果たす。 |
| ECSRQMF | T に設定されると、終了呼び出しが要求される。 |
| ECSRSDTA | 形式設定された結果に関する情報を含む。 |
| ECSRSLEN | バイトで表した出力域の長さが入っている (この値は、QMF 書式の WIDTH 列から取られる)。ECSRSLEN フィールドの値で示されているよりも多く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、エラー「DSQ60439: USER EDIT PROGRAM MEMORY OVERWRITE」が出されます。 |
| ECSTHSEP | QMF プロファイルの 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)。 |

表 58. QMF 編集出口インターフェース制御ブロックのフィールド、DXECS (続き)

| 名前 | 説明 |
|----------|---|
| ECSUSERS | ある呼び出しから次の呼び出しへそのまま渡される情報を出口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。 |

関連概念:

321 ページの『編集出口ルーチンおよび QMF』

QMF と編集出口ルーチンは連携し、定義した編集コードを使用してデータをフォーマット設定します。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

入出力域を特徴付けるフィールド

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。出力フィールドには、フォーマット設定した結果を入れる場所に関する情報が含まれています。

形式設定される値は ECSINPT フィールドに現れます。どのように表現されるかは、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。例えば、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。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 (例えば、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。例えば、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

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。この結果は、QMF レポートに戻ります。このフィールドの長さ (バイト) は ECSRSLEN フィールドに保管されています。このフィールドは、編集ルーチンが呼び出されるまで常にブランクです。

関連概念:

321 ページの『編集出口ルーチンおよび QMF』

QMF と編集出口ルーチンは連携し、定義した編集コードを使用してデータをフォーマット設定します。

編集コードの選択

独自の編集コードは、XML、BLOB、CLOB、および DBCLOB を除くすべてのタイプのデータ用に作成できます。QMF が実行されているプロセッサおよび出口ルーチンを記述する言語の両方が、10 進浮動小数点命令をサポートしている場合のみ、10 進浮動小数点データ・タイプ用の独自の編集コードを作成できます。

このタスクについて

編集コードは、報告書の特定の列のデータの形式と句読法を QMF に指示する文字セットです。

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 フィールドに UPPER または STRING 値がある場合は、QMF はディスプレイ装置から入力されるすべての入力データを大文字に変換します。編集ルーチンが大/小文字混合の編集コードを受け入れるように作成されている場合は、CASE が MIXED に設定されているときに編集コードを入力してください。

手順

- ルーチンで何をするかを決定し、データを形式設定するルーチンに関連付ける xxxx または Vxxxxのいずれかの編集コードを選択します。
 - U コードについては、編集ルーチンに渡されたデータは、ユーザー編集コードが使用されるフィールドが式の結果でない限り、ソース・データの内部データベース表記になります。式によるすべての結果は、拡張浮動小数点データとして編集ルーチンに戻されます。

- 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行うために使用できます。U および V は大文字でなければなりません。

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。

使用するプログラミング言語が特定のデータ・タイプをサポートしない場合は、V 編集コードを使用してそれらの値を文字に変換します。数値データを含む V コードの場合、QMF はデータを文字フォーマットに変換してから、ユーザー編集ルーチンを呼び出します。変換された数値の長さは、次の表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 59. 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 | 6 |
| 整数 | 10 |
| 大整数 | 19 |
| Decimal | 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) |
| 長形式 10 進浮動小数点 | 17 |
| 拡張形式 10 進浮動小数点 | 31 |

- xxxx を 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 NULL 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
```

2 バイト文字セット・データおよび編集ルーチン

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編集ルーチンを作成する必要がある場合は、そのルーチンがデータを正しく取り扱うようにしなければなりません。

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。

Uxxxx または Vxxxx のいずれかの編集コードを 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。

DBCS データ、および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。

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 ストリングを囲んでいる S0 および SI 文字を探す必要があります。ECSINPT 内に S0 または SI 文字がない場合、このストリングには DBCS データはありません。例えば、ECSINPT に次のようなストリングが入っているとします。

```
ccccSodededededededededeSiccSodededededeSi
```

ここで、c、d、および e は、任意のバイトを表し、S0 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。日本語の文字セットの場合、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。

漢字列からのデータ

フォーマット設定されるデータが漢字データ・タイプの列からのものである場合は、ECSINPT のテキストは、このデータの前にシフト文字が 1 つ付き、さらに後ろにもシフト文字が 1 つ付きます。いずれのシフト文字も 1 バイトです。DBCS ディスプレイ装置では、シフト文字は DBCS 文字ストリングの始まりと終わりを示しています。

S0 は、DBCS ストリング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。S0 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。

編集ルーチンは必ず正しい結果を戻すこと

ECSRSLT フィールドに編集結果を戻す際には未使用バイトの末尾ブランクを付ける必要があります。また、ユーザーのディスプレイ装置にそれらを読み取る機能があることを確認する必要があります。

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 S0 と SI 文字が付いていなければならないことを意味しています。

ECSRSLT フィールドのオーバーフロー

ECSRSLT フィールドがオーバーフローしないように気を付けてください。このフィールドの長さは ECSRSLEN フィールドにあります。結果が入りきらない場合は、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。多くのディスプレイ装置で、以下のような規則が適用されるものがあります。

- 報告書が画面に表示される場合、ユーザーの結果に埋め込まれた SI および S0 文字もディスプレイ上に表示されます。
- SI および S0 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、S0 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。

適正な DBCS 文字はすべて、ECSRSLT フィールドに戻されます。

日付、時刻、タイム・スタンプ・データ、および編集ルーチン

DATE、TIME、TIMESTAMP、または TIMESTAMP WITH TIME ZONE データを形式設定する編集出口ルーチンを作成している場合は、各データ・タイプに対してデフォルトの形式を使用する必要があります。TIMESTAMP または TIMESTAMP WITH TIME ZONE データの形式設定が不要な場合は、日時データに対して DB2 提供の出口ルーチンを使用することもできます。

日付、時間、およびタイム・スタンプ情報で必要な形式

編集ルーチンは、他のデータ・タイプの列のデータを形式設定できるのと同じように、日時列のデータを形式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。

ストリングの要求される形式については、次の表に説明があります。

表 60. DATE、TIME、および TIMESTAMP データのフォーマット設定

| データ・タイプ | ストリングの形式 |
|---------------|---|
| DATE データ | <p><i>yyyy-mm-dd</i>。以下に各部の説明を示します。</p> <p>yyyy 年を指定。常に、4桁の数字。</p> <p>mm 月を指定 (01 は 1 月、12 は 12 月)。常に、2桁の数字で、先行ゼロが可能。</p> <p>dd 日を指定。常に、2桁の数字で、先行ゼロが可能。</p> <p>ダッシュ (-) は実際のダッシュを意味します。</p> <p>例えば、2006-03-12 は、2006 年 3 月 12 日です。</p> |
| TIME データ | <p><i>hh.mm.ss</i>。以下に各部の説明を示します。</p> <p>hh 時刻を指定 (00 から 23 の 24 時間表示による)。常に、2桁の数字で、先行ゼロが可能。</p> <p>mm 分を指定。常に、2桁の数字で、先行ゼロが可能。</p> <p>ss 秒を指定。常に、2桁の数字で、先行ゼロが可能。</p> <p>ピリオドは実際のピリオドを意味します。</p> <p>例えば、13.08.36 は、米国で一般に使用されている表記法では、1:08 P.M. 36 秒となります。</p> |
| TIMESTAMP データ | <p><i>yyyy-mm-dd-hh.mm.ss.nnnnnnnnnnnn</i>。以下に各部の説明を示します。</p> <p>yyyy-mm-dd DATE データの場合と同様に日付を指定。</p> <p>hh.mm.ss TIME データの場合と同様に時刻を指定。</p> <p>nnnnnnnnnnnn 小数秒の数を表す 0 桁から 12 桁の数を指定します。</p> <p>例えば、2010-09-30-13.08.36.123456654321 は、米国で一般に使用されている表記法では、2010 年 9 月 30 日の 1:08 P.M. 36.123456654321 秒となります。</p> |

表 60. 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 WITH TIME ZONE データ | <p><i>yyyy-mm-dd-hh-mm-ss-nnnnnnnnnnnnzth:tm</i></p> <p><i>yyyy-mm-dd</i> DATE データの場合と同様に日付を指定。</p> <p><i>hh.mm.ss</i> TIME データの場合と同様に時刻を指定。</p> <p><i>nnnnnnnnnnnn</i> 小数秒の数を表す 0 桁から 12 桁の数を指定します。</p> <p><i>z</i> 以前はグリニッジ標準時 (GMT) として知られていた協定世界時 (UTC) に相対する時間帯オフセットを示す正 (+) または負 (-) 符号。</p> <p><i>th</i> 時間帯時間数を表す 2 桁の値。</p> <p><i>tm</i> 時間帯分数を表す 2 桁の値。</p> <p>形式の時間帯部分の有効な範囲は、-24:00 から +24:00 です。UTC を指定するには、-0:00 または +0:00 で時間帯を指定するか、時間帯オフセットとその符号を大文字の Z で置き換えます。</p> <p>例えば、2010-09-30-13.08.36.123456654321-08:00 は、米国のカリフォルニア州サンホセでは、2010 年 9 月 30 日の 1:08 P.M. 36.123456654321 秒を示します。タイム・スタンプ 2010-09-30-13.08.36.123456654321Z は、UTC が有効になっている場合は常に、午後 1 時 8 分および 36.123456654321 秒の時刻を示します。</p> |

TSO での日時データ用 DB2 出口

ユーザーは、QMF for TSO で報告書を作成する場合、適切な QMF 編集コード (日付の場合は TDL、時刻の場合は TTL) を指定することによって、DATE または TIME のデータにローカル形式を指定できます。QMF は、DB2 に付属の適切な出口を呼び出して形式設定を行います (DSNXVDTX は日付を、DSNXVTMX は時刻を形式設定します)。

DB2 によって出荷されるこれらの出口はスタブです。これらのスタブは、ローカル形式が定義されていない場合に使用されることを意味し、形式設定を行うものではありません。これらが適切に動作するようにするには、出口のローカル・コピーと置き換える必要があります。

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、出口のローカル・コピーを QMF で使用可能にします。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・ライブラリーが初めて検索されると、QMF は、代わりに DB2 に付属の 2 つのスタブをロードして使用します。以下の STEPLIB ステートメントの例では、形式設定ルーチンはライブラリー XYZ.FORMAT にあり、DB2 プログラム・ライブラリーは DSN1110.SDSNLOAD です。

```

//STEPLIB DD DSN=ISP.SISPLLOAD,DISP=SHR
//          DD DSN=QMF1210.SDSQLOAD,DISP=SHR
//          DD DSN=XYZ.FORMAT,DISP=SHR          (local formatting library)
//          DD DSN=DSN1110.DSNLOAD,DISP=SHR     (DB2 program library)
//          DD DSN=GDDM.OSPID.SADMMOD,DISP=SHR

```

図 59. 編集ルーチンの使用可能化

TTL および TDL 編集コードにより処理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。

関連情報:

 DB2 資料

DB2 出口を使用して日時ルーチンを指定する方法に関する情報を検索します。

プログラミング言語の編集ルーチン

編集コードによって記述されたデータを複数の言語のいずれかでフォーマット設定するための編集出口ルーチンを作成することができます。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作成する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。

さまざまなバージョンのサンプル編集出口ルーチンが QMF1210.SDSQSAP n ライブラリー (n はご使用の各国語を表す 1 文字の ID) にあります。このサンプル・ルーチンの名前は、以下で示すように、プログラミング言語と環境によって異なります。

表 61. プログラミング言語と環境ごとのサンプル編集出口ルーチンの名前

| 言語 | TSO、ISPF、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 |

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

編集ルーチンをハイレベル・アセンブラーで作成

TSO、ISPF、ネイティブ z/OS、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。

TSO、ISPF、およびネイティブ z/OS用のアセンブラー編集ルーチンの作成

TSO、ISPF、およびネイティブ z/OS 用の編集ルーチンをアセンブラーで作成するには、言語特有の詳細について理解する必要があります。

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

インターフェース制御構造 DXEECS 用のサンプル・アセンブラー DSECT は、DXEECSA として QMF と出荷され、QMF1210.SDSQUSR n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。インターフェース制御構造により、編集インターフェース DSQUEDIT の入力フィールドが定義されます。

サンプル・プログラム (DSQUXDTA) を使用している場合は、この DSECT を含む COPY ステートメントはすでにサンプル内に含まれています。サンプル・プログラムで開始する代わりに独自のルーチンを作成している場合は、使用しているプログラムにこの COPY ステートメントが含まれていることを確実にしてください。

インターフェース制御ブロックは、ユーザーの編集コードを含み、ソース・データおよび編集結果のターゲット・ロケーションを識別し、さらにユーザー編集ルーチンが使用するスクラッチパッド域を提供します。制御ブロックは、ユーザー編集ルーチンへの呼び出しの間も存続しています。スクラッチパッド域は、出口ルーチンの最初の呼び出しの後には、QMF により変更されることはありません。

- 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。

アセンブラー用のサンプル編集プログラム DSQUXDTA は、QMF1210.SDSQSAP n ライブラリー (n はご使用の各国語を表す 1 文字の ID) 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。このプログラム例で開始する場合は、ユーザーのプライベート・プログラム・ソース・ライブラリーにコピーし、名前を DSQUXDT に変更してください。

下図は、TSO、ISPF、またはネイティブ 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を示しています。

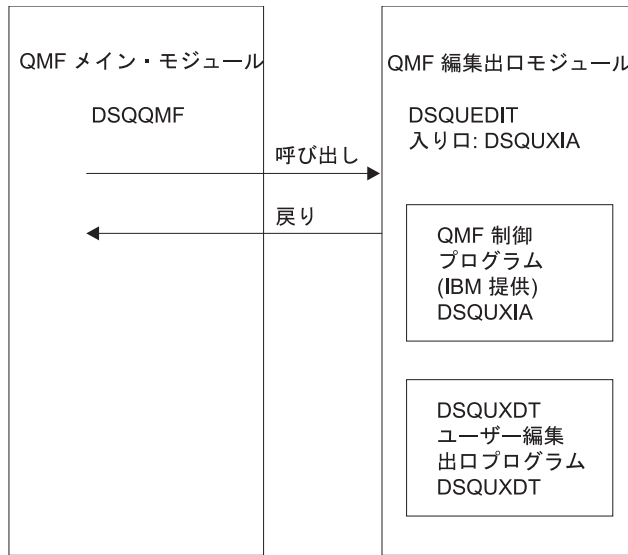


図 60. TSO、ISPF、またはネイティブ z/OS のアセンブラー編集出口ルーチンのプログラム構造

プログラムを呼び出し、QMF に制御を戻すためのステートメント

ユーザー編集プログラムは、TSO、ISPF、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に入力するときは、以下の条件があります。

- 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。

下図は、これらの条件の例を示しています。

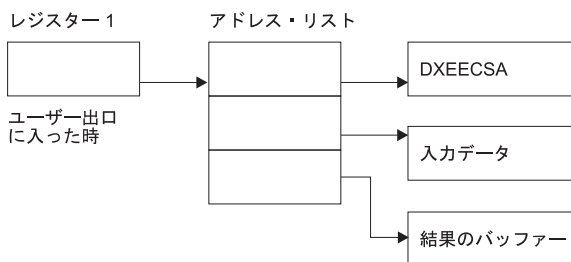


図 61. アセンブラーの編集出口プログラムに入る際に存在する条件

標準的な方法に従って、レジスターを呼び出し時の値にリストアし、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 10、9、および 8 に設定されます。

```

ECSPTR    EQU R10
          L      ECSPTR,0(R1)
          USING  DXEECS,ECSPTR
ECSINPTP  EQU R9
  
```

```

          L      ECSINPT,4(R1)
        USING   ECSINPT,ECSINPT
ECSRSLTP EQU R8
          L      ECSRSLTP,8(R1)
        USING   ECSRSLT,ECSRSLTP

```

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。

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。

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

ユーザーのプログラムをアセンブルおよびリンク・エディットする際には、アセンブラ用の 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。DXEECSA は、QMF1210.SDSQUSR n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。

QMF モジュール・ライブラリー QMF1210.SDSQLOAD にある制御モジュール DSQUXIA をユーザー編集プログラム (DSQUXDT) に組み込むことによって、新しい QMF 編集出口モジュールを作成します。DSQUXIA は、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 ビットのアドレッシング・モードを推奨します。

以下のステートメントは、TSO、ISPF、およびネイティブ z/OS のジョブをアセンブルおよびリンク・エディットするための例を示しています。このジョブを実行する前に、SYSLMOD ステートメント内で参照されているロード・ライブラリーを作成してください。そのライブラリー内に、ユーザー編集出口モジュールが常駐します。

```

//sampasm    JOB
//STEP1      EXEC PROC=ASMACL
/* Provide access to QMF edit macro DXEECSA
//C.SYSLIB   DD DSN=QMF1210.SDSQUSRE,DISP=SHR
//C.SYSIN    DD *
            .
            .
            .
            Your program or copy of QMF sample DSQUXDTA
            .
            .
/*
/* Provide access to QMF interface module
//L.QMFLOAD  DD DSN=QMF1210.SDSQLOAD,DISP=SHR
//L.SYSLMOD  DD DSN=USER.LOADLIB,DISP=SHR
//L.SYSIN    DD *
            INCLUDE QMFLOAD(DSQUXIA)
            ENTRY DSQUXIA
            MODE  AMODE(31) RMODE(31)
            NAME  DSQUEDIT(R)
/*

```

図 62. TSO、ISPF、またはネイティブ z/OS のアセンブラー編集出口ルーチンをアセンブルおよびリンク・エディットするためのサンプル・ステートメント

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

関連資料:

323 ページの『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、ユーザー出口ルーチンとの間で情報の受け渡しを行います。

アセンブラー編集ルーチンを CICS 用に作成

CICS 用の編集ルーチンをアセンブラーで作成するには、言語特有の詳細について理解する必要があります。

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。

- QMF とユーザー編集インターフェースとの間のインターフェース制御ブロック

インターフェース制御ブロック DXEECS 用のアセンブラー DSECT である DXEECSA が QMF に付属しており、QMF1210.SDSQUSR n ライブラリー (n はご使用の各国語を表す 1 文字の ID) にあります。

提供の CICS 用のサンプル・プログラム (DSQUXCTA) を使用している場合は、この DSECT を含む COPY ステートメントはすでにサンプル内に含まれています。提供のサンプル編集プログラムで開始する代わりに独自のプログラムを作成している場合は、使用しているプログラムに COPY ステートメントを必ず含めてください。

インターフェース制御ブロックは、ユーザーの編集コードを含み、ソース・データおよび編集結果のターゲット・ロケーションを識別し、さらにユーザー編集ルーチンが使用するスクラッチパッド域を提供します。制御ブロックは、ユーザー編集ルーチンへの呼び出しの間も存続しています。スクラッチパッド域は、出口ルーチンの最初の呼び出しの後には、QMF により変更されることはありません。

- 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 DFHEAIO として CICS と共に出荷。
- ユーザーの編集出口プログラム。名前は DSQUECIC。

提供の アセンブラー 内の編集プログラム DSQUXCTA は、QMF1210.SDSQSAP n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。このプログラム例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、ユーザーのプライベート・プログラム・ソース・ライブラリーにコピーし、名前を DSQUECIC に変更してください。

CICS 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以下に示します。

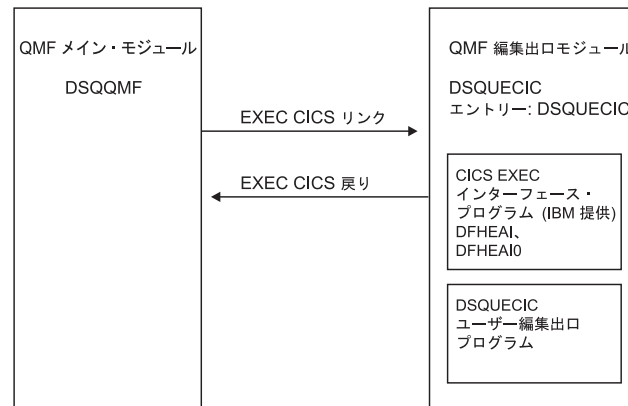


図 63. CICS のアセンブラー編集出口ルーチンのプログラム構造

プログラムを呼び出し、QMF に制御を戻すためのステートメント

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に入力するときは、以下の条件があります。

- レジスター 1 には、CICS に付属のマクロ 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。

プログラムの流れを以下に示します。

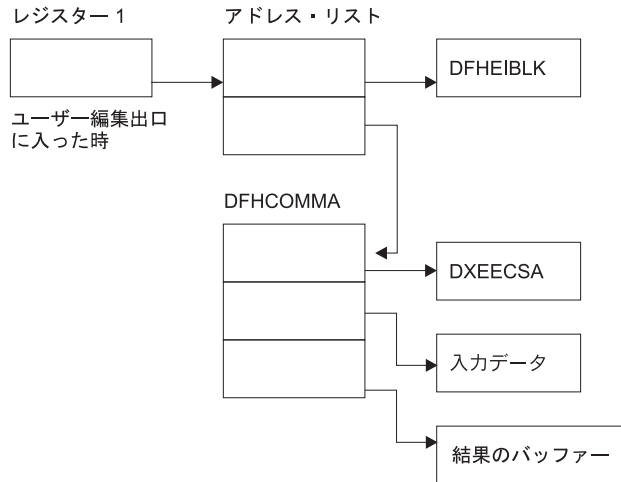


図 64. CICS アセンブラー編集ルーチンと QMF との対話方法

- レジスター 13 には、CICS に付属のマクロ DFHEISTG で記述されているように、標準の CICS 作業用記憶域のアドレスが入っています。

標準の CICS RETURN コマンド、例えば、EXEC CICS RETURN を使用して、QMF に制御を戻します。

プログラムの変換、アセンブルおよびリンク・エディット

アセンブラー用の CICS 変換プログラムを使用して、ユーザーのプログラムを変換する必要があります。ユーザーのプログラムを変換する際、通常、CICS は、アドレス可能度を設定する標準の CICS プロログ (DFHEIENT) を提供し、レジスターを標準の CICS 作業用記憶域に保存し、さらに標準の CICS エピログ (DFHEIRET) を提供します。

アセンブリー中、QMF 編集出口インターフェース制御ブロック DXEECSA は、QMF1210.SDSQUSR n ライブラリー内にあり (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)、CICS マクロ・ライブラリーが使用可能でなければなりません。

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 (DSQUECIC) 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。

以下は、CICS のジョブを変換、アセンブル、およびリンク・エディットするためのステートメントの例です。このジョブを実行する前に、SYSLMOD ステートメント内で参照されているロード・ライブラリーを作成してください。そのライブラリー内に、ユーザー編集出口モジュールが常駐します。

```

//SAMPASM JOB ...
/* Add a parameter PROGLIB to procedure DFHEITAL
/*      PROGLIB=&PROGLIB,
//TRNCOMLK EXEC PROC=DFHEITAL,PROGLIB=QMF1210.SDSQLOAD'
//TRN.SYSIN DD *
.
.
.
Your program or modified copy of QMF sample DSQUXCTA
.
.
.
/*
/* Provide access to QMF edit macro DXEECSA
//ASM.SYSLIB DD DSN=QMF1210.SDSQUSRE,DISP=SHR
//LKED.SYSLMOD DD DSN=USER.LOADLIB,DISP=SHR
//LKED.SYSIN DD *
INCLUDE SYSLIB(DFHEAI)
INCLUDE SYSLIB(DFHEAI0)
ORDER DFHEAI,DFHEAI0
ENTRY DSQUEECIC
MODE AMODE(31) RMODE(31)
NAME DSQUEECIC(R)
/*

```

図 65. CICS のアセンブラー編集出口ルーチンを反感、アSEMBル、およびリンク・エディットするステートメントの例

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

編集ルーチンを PL/I で作成

QMF 編集ルーチンを PL/I で作成できます。

TSO、ISPF、またはネイティブ z/OS の CEEPIPI Language Environment インターフェースおよび PL/I 編集ルーチン

CEEPIPI 言語処理環境インターフェースを使用するための指示に従って、TSO、ISPF、またはネイティブ z/OS 内で実行する QMF 編集ルーチンを作成します。

このインターフェースは、次の部分で構成されます。

- インターフェース制御構造 DXEECS

この制御構造は、QMF と一緒に DXEECS として出荷されます。制御構造は、すべての呼び出しで編集出口プログラムに渡されます。ここでは、QMF と編集出口ルーチンとの間の状況および通信情報が含まれます。また、形式設定されるデータと形式設定された結果が保管されるポインターに関する情報も含まれません。

DXEECS 制御構造の詳細については、QMF1210.SDSQUSR n ライブラリーにあるコピー・ファイル DXEECSF を参照してください (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。

- 制御プログラム DSQUXILE。QMF と共に出荷

DSQUXILE は、QMF 編集出口モジュール DSQUEDIT へのメイン・エントリー・ポイントでなければなりません。DSQUXILE は、QMF モジュール・ライブラリー QMF1210.SDSQLOAD 内にあります。DSQUXILE 制御プログラムは、CEEPIPI プログラムを呼び出して Language Environment[®] を初期化し、QMF セッションの終了時に Language Environment を強制終了します。DSQUXILE 制御プログラムは、CEEPIPI プログラムを呼び出して DSQUXDT プログラムの実行を指定することによってユーザー出口プログラム DSQUXDT を呼び出します。

- ユーザーの編集出口プログラム

PL/I 用の編集出口プログラム例 DSQUXDTP は、QMF1210.SDSQSAP n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。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、ユーザーのプライベート・プログラム・ソース・ライブラリーにコピーし、名前を DSQUXDT に変更してください。

- 言語処理環境事前初期設定サービス・プログラム。名前は CEEPIPI。

下の図は、CEEPIPI Language Environment インターフェースを使用する 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。

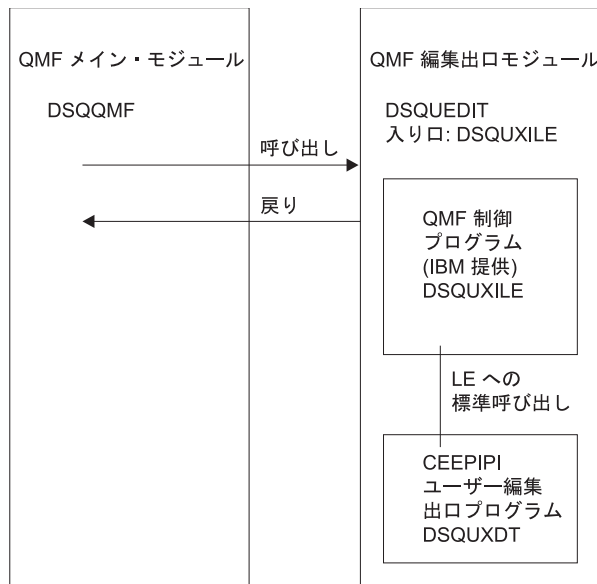


図 66. CEEPIPI Language Environment インターフェースを使用して作成された PL/I 編集出口ルーチンのプログラム構造

プログラムを呼び出し、QMF に制御を戻すためのステートメント

QMF は、CEEPIPI プログラムを呼び出して DSQUXDT の実行を指定することによって、編集出口ルーチンを PL/I プログラムとして呼び出します。以下のパラメーターが、示された順序で提供されます。

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

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 のコンパイル

REENTRANT をプロシージャ・オプションとして指定することによって編集出口プログラムをコンパイルします。PROCEDURE ステートメントの適切な構文については、『プログラムを呼び出し、QMF に制御を戻すためのステートメント』を参照してください。

コンパイル時には、QMF1210.SDSQUSR n ライブラリー (n はご使用の各国語を表す 1 文字の ID) 内にある QMF 編集出口インターフェース制御ブロック DXEECSF が SYSLIB 割り振りで使用可能でなければなりません。

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 QMF1210.SDSQLOAD にある制御モジュール (DSQUXILE) をユーザー編集プログラム (DSQUXDT) に組み込むことにより、新しい QMF 編集出口モジュール (DSQUEDIT) を作成します。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 ビットのアドレッシング・モードを推奨します。

以下のステートメントは、TSO、ISPF、またはネイティブ z/OS のジョブをアセンブルおよびリンク・エディットするための例を示しています。編集ルーチンが 10 進数浮動小数点データを形式設定する場合は、ARCH(7) および FLOAT(DFP) コンパイラー・オプションを追加する必要があります。このジョブを実行する前に、SYSLMOD ステートメント内で参照されているロード・ライブラリーを作成してください。そのライブラリー内に、ユーザー編集出口モジュールが常駐します。

```

//samPLI JOB
//STEP1 EXEC PROC=IBMZCPL
//* Provide access to QMF edit macro DXEECS
//PLI.SYSLIB DD DSN=QMF1210.SDSQUSRE,DISP=SHR
//PLI.SYSIN DD *
.
.
.
Your PL/I edit exit program DSQUXDT
or copy of sample program DSQUXDTP
that has been renamed to DSQUXDT
.
.
.
If your edit routine formats decimal
floating-point data, add ARCH(7) and
FLOAT(DFP) compiler options here.
/*
/* Provide access to QMF interface module DSQUXILE
//LKED.QMFLOAD DD DSN=QMF1210.SDSQLOAD,DISP=SHR
//LKED.SYSLMOD DD DSN=USER.LOADLIB,DISP=SHR
//LKED.SYSIN DD *
        INCLUDE QMFLOAD(DSQUXILE)
        ENTRY DSQUXILE
        MODE AMODE(31) RMODE(ANY)
        NAME DSQUEDIT(R)
/*

```

図 67. CEEPIPI Language Environment インターフェースを使用して作成された PL/I ルーチン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

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

関連資料:

323 ページの『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、ユーザー出口ルーチンとの間で情報の受け渡しを行います。

PL/I 編集ルーチンを CICS 用に作成

CICS 用の編集ルーチンを PL/I で作成するには、言語特有の詳細について理解する必要があります。

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。

- PL/I データ構造

このデータ構造は、DXEECSとして QMF と出荷され、QMF1210.SDSQUSR n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。この構造をユーザーのプログラムに組み込みます。

データ構造は、すべての呼び出しで編集出口プログラムに渡されます。ここには、QMF と編集出口ルーチンとの間の状況および通信情報が含まれます。また、形式設定されるデータと形式設定された結果が保管されるポインターに関する情報も含まれます。

- CICS コマンド・レベル・インターフェース

QMF は、CICS コマンド・レベル・インターフェースを使用して、QMF 編集出口モジュール DSQUECIC を呼び出します。完了した編集出口モジュールは、EXEC CICS RETURN ステートメントを発行して、制御を QMF に戻す必要があります。

- ユーザーの編集出口プログラム。名前は DSQUECIC。

QMF がインストールされていて、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と共にインストールされ、CICS 内の提供の PL/I 用の編集プログラム例は DSQUXCTP であり、QMF1210.SDSQSAP n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。

このプログラム例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、ユーザーのプライベート・プログラム・ソース・ライブラリーにコピーし、名前を DSQUECIC に変更してください。

QMF がインストールされているときには、CICS プログラム定義リソースはアセンブラーに対して構成されます。PL/I 編集出口プログラムを使用するためには、CICS プログラム定義を PL/I に再構成する必要があります。

以下に、CICS の PL/I 編集出口ルーチンのプログラム構造を示します。

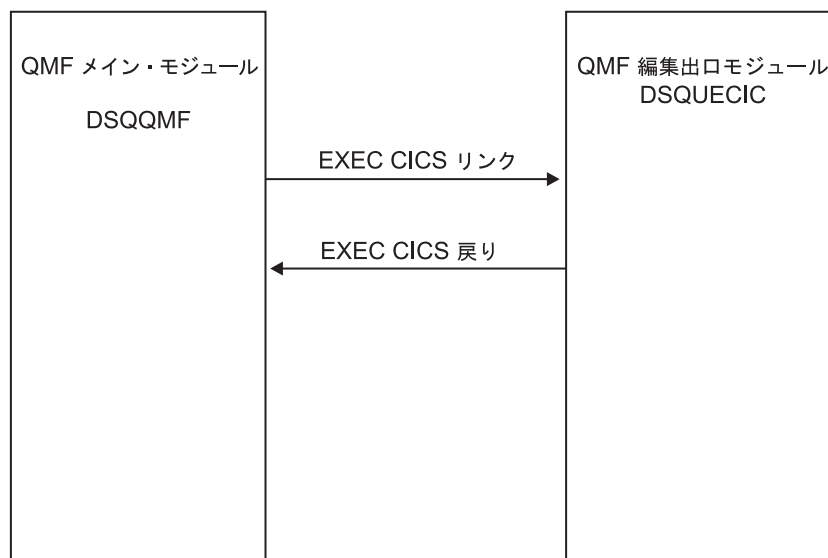


図 68. CICS の PL/I 編集出口ルーチンのプログラム構造

プログラムを呼び出し、QMF に制御を戻すためのステートメント

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。

QMF は、次に示されているように、CICS 連絡域 DFHCOMMAREA で、ユーザー編集ルーチン制御ブロック (DXEECS) と入力データおよび出力データにアドレスを渡します。

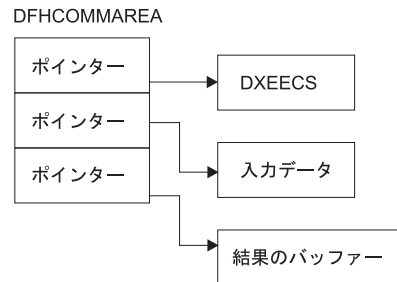


図 69. CICS 連絡域 DFHCOMMAREA の処理方法

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 を指すパラメーターを提供します。

```
DSQUECIC:
  PROCEDURE(DFHCOMP) OPTIONS(REENTRANT,MAIN);
```

以下で示すように、PL/I プログラムで DFHCOMMAREA の独自の記述も指定します。

```
/******  
/* CICS DFHCOMMAREA DESCRIPTION OF EDIT EXIT PARAMETERS */  
/******  
DECLARE  
  DFHCOMP PTR;  
DECLARE  
  1 DFHCOMM BASED(DFHCOMP),  
    02 DFHCOMM_ECSPTR PTR,  
    02 DFHCOMM_INPTR PTR,  
    02 DFHCOMM_OUTPTR PTR;
```

図 70. CICS 連絡域 DFHCOMMAREA の編集出口パラメーターの説明

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指定するには、次で示すように、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   */
ECSINPTP = DFHCOMM_INPTR    /* ADDRESS OF INPUT DATA   */
ECSRSLTP = DFHCOMM_OUTPTR   /* ADDRESS OF RESULT AREA   */
```

図 71. DXEECS 制御ブロックへのアドレス可能度の指定

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、コンパイルおよびリンク・エディット

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、CICS 環境制御ブロック (EIB) の入力パラメーターおよびデータ構造定義を提供します。

QMF 編集出口インターフェース制御ブロック DXEECS は、QMF1210.SDSQUSR n ライブラリー (n はご使用の各国語を表す 1 文字の ID) にあります。この制御ブロックは、コンパイル中にマクロ・ライブラリー内で使用可能でなければなりません。

PL/I コンパイラー・オプション PP(CICS)、SYSTEM(CICS)、および RENT。

実行モード (RMODE) および 31 ビットのアドレス・モード (AMODE) を持つ新しい QMF 編集出口モジュール DSQUECIC を作成します。

以下は、CICS のジョブを変換、コンパイル、およびリンク・エディットするためのステートメントの例です。編集ルーチンが 10 進数浮動小数点データを形式設定する場合は、ARCH(7) および FLOAT(DFP) コンパイラー・オプションを追加する必要があります。このジョブを実行する前に、SYSLMOD ステートメント内で参照されているロード・ライブラリーを作成してください。そのライブラリー内に、ユーザー編集出口モジュールが常駐します。

```

//SAMPLI    JOB    .....
//COMLK     EXEC  PROC=IBMZCPL
//PLI.SYSLIB DD   DSN=QMF1210.SDSQSRE,DISP=SHR
//PLI.SYSIN DD   *
.
.
.
Your program or modified copy of QMF sample DSQUXCTP
that has been renamed to DSQUECIC
.
.
.
Include compiler options: PP(CICS),SYSTEM(CICS) and RENT.
If your edit routine formats decimal
floating-point data, add ARCH(7) and
FLOAT(DFP) compiler options.
.
.
.
/*
//LKED.SYSLMOD DD DSN=USER.LOADLIB,DISP=SHR
//LKED.SYSIN DD   *
MODE AMODE(31),RMODE(31)
NAME DSQUECIC(R)
/*

```

図 72. CICS の編集出口を PL/I で変換、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

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

編集ルーチンを COBOL で作成

QMF 編集ルーチンを COBOL で作成できます。

TSO、ISPF、またはネイティブ z/OS の IGZERRE インターフェースおよび COBOL 編集ルーチン

IGZERRE インターフェースを使用するための指示に従って、TSO、ISPF、またはネイティブ z/OS 内で実行する COBOL 編集ルーチンを作成します。

このインターフェースは、次の部分で構成されます。

- インターフェース制御構造 DXEECS

この制御構造は、QMF と一緒に出荷されます。これをユーザーのプログラムに組み込みます。

制御構造は、すべての呼び出しで編集出口プログラムに渡されます。ここには、QMF と編集出口ルーチンとの間の状況および通信情報が含まれます。また、形式設定されるデータと形式設定された結果が保管されるポインターに関する情報も含まれます。DXEECS 制御構造の詳細については、QMF1210.SDSQUSRn ラ

ライブラリーにあるコピーブック DXEECS を参照してください (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。

- 制御プログラム DSQUXIC

この制御プログラムは、QMF と一緒に出荷されます。これは、QMF 編集出口モジュール DSQUEDIT へのメイン・エントリー・ポイントでなければなりません。DSQUXIC は、QMF モジュール・ライブラリー QMF1210.SDSQLOAD 内にあります。DSQUXIC 制御プログラムは、IGZERRE プログラムを呼び出して COBOL 環境を初期化し、QMF セッションが終了したときに COBOL 環境を強制終了します。DSQUXIC 制御プログラムは、ユーザー出口プログラム DSQUXDT を COBOL プログラムとして呼び出します。

- ユーザーの編集出口プログラム。名前は DSQUXDT。

COBOL 内の編集出口プログラム DSQUXDT は、QMF1210.SDSQSAP n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。このプログラム例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、ユーザーのプライベート・プログラム・ソース・ライブラリーにコピーし、名前を DSQUXDT に変更してください。

次の図は、COBOL 編集出口ルーチンのプログラム構造を示します。

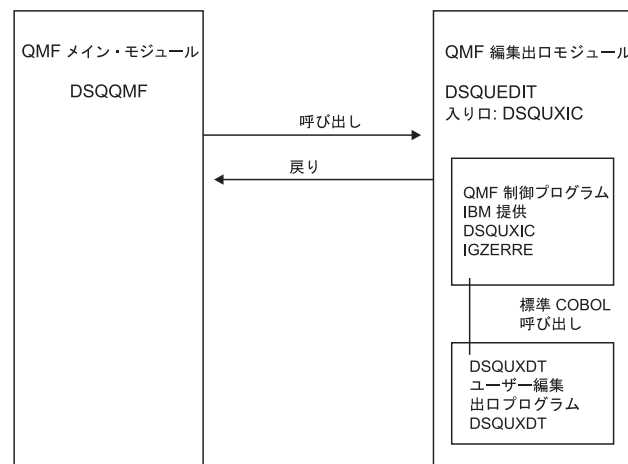


図 73. COBOL 編集出口ルーチンのプログラム構造

プログラムを呼び出し、QMF に制御を戻すためのステートメント

QMF は、標準の COBOL CALL ステートメントを使用して、ユーザーの編集出口ルーチンを COBOL プログラムとして呼び出します。以下のパラメーターが、示された順序で提供されます。

```
PROCEDURE DIVISION  
  USING DXEECS, ECSINPT, ECSRSLT.
```

標準 GOBACK ステートメントを使用して制御を QMF に返します。

プログラムのコンパイルおよびリンク・エディット

プログラムを LIB、RENT、QUOTE、および NODYNAM オプションでコンパイルします。ユーザー編集ルーチン制御ブロック DXEEECSC は、引用符をリテラル区切り文字として使用します。IBM より配布される DXEEECSC 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。

コンパイル時には、QMF1210.SDSQUSR n ライブラリー (n はご使用の各国語を表す 1 文字の ID) 内にある QMF 編集出口インターフェース制御ブロック DXEEECSC が SYSLIB 割り振りで使用可能でなければなりません。

DSQUXDT のコンパイル後、結果のロード・モジュールを QMF1210.SDSQLOAD ライブラリーに入れます。

制御モジュール (DSQUXIC) をユーザー編集出口プログラム (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 は、QMF モジュール・ライブラリー QMF1210.SDSQLOAD 内にあります。DSQUXIC は、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 ビットのアドレッシング・モードを推奨します。

以下のステートメントは、TSO、ISPF、またはネイティブ z/OS で実行されるアプリケーションをコンパイルおよびリンク・エディットするための例を示しています。このジョブを実行する前に、SYSLMOD ステートメント内で参照されているロード・ライブラリーを作成してください。そのライブラリー内に、ユーザー編集出口モジュールが常駐します。

```

//samCOBOL JOB
//STEP1 EXEC PROC=IGYWCL,PARM='LIB,RENT,NODYNAM,QUOTE'
/* Provide access to QMF edit macro DXEECS
//COBOL.SYSLIB DD DSN=QMF1210.SDSQSRE,DISP=SHR
//COBOL.SYSIN DD *
.
.
.
Your COBOL edit exit program DSQUXDT or a copy of
sample program DSQUXDTC that has been renamed to
DSQUXDT
.
.
.
/*
/* Provide access to QMF interface module DSQUXIC
//LKED.QMFLOAD DD DSN=QMF1210.SDSQLOAD,DISP=SHR
//LKED.SYSLMOD DD DSN=USER.LOADLIB,DISP=SHR
//LKED.SYSIN DD *
INCLUDE QMFLOAD(DSQUXIC)
INCLUDE SYSLIB(IGZERRE)
ENTRY DSQUXIC
MODE AMODE(31) RMODE(ANY)
NAME DSQUEDIT(R)
/*

```

図 74. IGZERRE インターフェースを使用して COBOL で作成された編集出口をコンパイルし、リンク・エディットするステートメントの例

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

TSO、ISPF、またはネイティブ z/OS の CEEPIPI Language Environment インターフェースおよび COBOL 編集ルーチン

CEEPIPI Language Environment インターフェースを使用するための指示に従って、TSO、ISPF、またはネイティブ z/OS で実行する QMF 編集ルーチンを COBOL で作成します。

このインターフェースは、QMF for CICS で実行する編集出口プログラムには使用できません。

COBOL での TSO、ISPF、またはネイティブ z/OS 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構造 DXEECS

この制御構造は、QMF と一緒に DXEECSC として出荷されます。この構造をユーザーのプログラムに組み込みます。この制御構造の情報はすべての呼び出しで編集出口プログラムに渡されます。ここには、QMF と編集出口ルーチンとの間の状況および通信情報が含まれます。また、形式設定されるデータと形式設定された結果が保管されるポインターに関する情報も含まれます。

DXEECS インターフェース制御構造について詳しくは、QMF1210.SDSQUSR n ライブラリー (n はご使用の各国語を表す 1 文字の ID) 内にあるコピーブック DXEECS を参照してください。

- 制御プログラム。IBM によって提供され、DSQUXILE として QMF と共に出荷。

QMF 制御プログラム DSQUXILE は、IBM によって提供され、QMF 編集出口モジュール DSQUEDIT へのメイン・エントリー・ポイントでなければなりません。DSQUXILE は、QMF モジュール・ライブラリー QMF1210.SDSQLOAD 内にあります。DSQUXILE 制御プログラムは、CEEPIPI プログラムを呼び出して Language Environment を初期化し、QMF セッションの終了時に Language Environment を強制終了します。DSQUXILE 制御プログラムは、CEEPIPI プログラムを呼び出して DSQUXDT プログラムの実行を指定することによってユーザー出口プログラム DSQUXDT を呼び出します。

- ユーザーの編集出口プログラム。名前は DSQUXDT。

COBOL 内の編集出口プログラム DSQUXDTC は、QMF1210.SDSQSAP n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。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、ユーザーのプライベート・プログラム・ソース・ライブラリーにコピーし、名前を DSQUXDT に変更してください。

- 言語処理環境事前初期設定サービス・プログラム。名前は CEEPIPI。

下の図は、CEEPIPI インターフェースを使用する COBOL 編集出口ルーチンのプログラム構造を示しています。

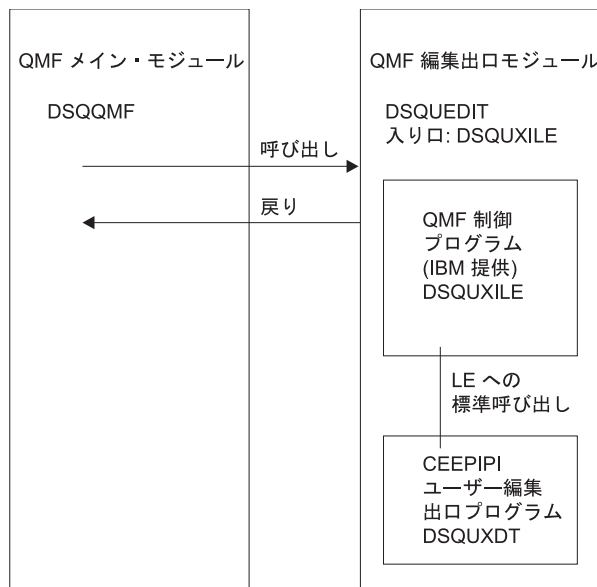


図 75. CEEPIPI インターフェースを使用する COBOL 編集出口ルーチンのプログラム構造

QMF と共に出荷されている編集制御ブロック (DXEECS) およびサンプル COBOL プログラム (DSQUXCTC) では、リテラルを区切るために引用符 (") を使

用します。サイトまたはプログラムで、代わりにアポストロフィ (') を使用している場合は、DXEECS を変更するか、あるいは構造をユーザーのプログラムにコピーし、引用符をアポストロフィに変更する必要があります。

プログラムを呼び出し、QMF に制御を戻すためのステートメント

QMF は、CEEPIPI プログラムを呼び出して DSQUXDT の実行を指定することによって、編集出口ルーチンを COBOL プログラムとして呼び出します。以下のパラメーターが、示された順序で提供されます。

```
PROCEDURE DIVISION
    USING DXEECS, ECSINPT, ECSRSLT.
```

標準の GOBACK ステートメントを使用して COBOL 編集出口プログラム DSQUXDT から制御を QMF に戻します。

DSQUXDT のコンパイル

コンパイル時には、QMF1210.SDSQUSR n ライブラリー (n はご使用の各国語を表す 1 文字の ID) 内にある QMF 編集出口インターフェース制御ブロック DXEECS が SYSLIB 割り振りで使用可能でなければなりません。

プログラムを LIB、RENT、QUOTE、および NODYNAM オプションでコンパイルします。DXEECS は引用符をリテラル区切り文字として使用します。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。

プログラムのリンク・エディット

制御モジュール DSQUXILE をユーザー編集出口プログラム (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。DSQUXILE は、QMF モジュール・ライブラリー (QMF1210.SDSQLOAD) 内にあります。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・モードを推奨します。

以下のステートメントは、TSO、ISPF、またはネイティブ z/OS のジョブをコンパイルおよびリンク・エディットするための例を示しています。このジョブを実行する前に、SYSLMOD ステートメント内で参照されているロード・ライブラリーを作成してください。そのライブラリー内に、ユーザー編集出口モジュールが常駐します。

```

//samCOBOL JOB
//STEP1 EXEC PROC=IGYWCL,PARM='LIB,RENT,NODYNAM,QUOTE'
//* Provide access to QMF edit macro DXEECS
//COBOL.SYSLIB DD DSN=QMF1210.SDSQSRE,DISP=SHR
//COBOL.SYSIN DD *
.
.
.
Your COBOL edit exit program DSQUXDT
.
.
.
/*
/* Provide access to QMF interface module DSQUXILE
//LKED.QMFLOAD DD DSN=QMF1210.SDSQLOAD,DISP=SHR
/* Provide access to the LE program library
//LKED.SYSLIB DD ...
// DD DSN=SYS1.SCEELKED,DISP=SHR
//LKED.SYSLMOD DD DSN=USER.LOADLIB,DISP=SHR
//LKED.SYSIN DD *
INCLUDE QMFLOAD(DSQUXILE)
ENTRY DSQUXILE
MODE AMODE(31) RMODE(ANY)
NAME DSQUEDIT(R)
/*

```

図 76. CEEPIPI インターフェースを使用して COBOL で作成された編集出口ルーチンをコンパイルし、リンク・エディットするステートメントの例

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

関連資料:

323 ページの『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、ユーザー出口ルーチンとの間で情報の受け渡しを行います。

COBOL 編集ルーチンを CICS 用に作成

CICS 用の編集ルーチンを COBOL で作成するには、言語特有の詳細について理解する必要があります。

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

この制御ブロックの詳細については、QMF と一緒に出荷され、QMF1210.SDSQUSR n ライブラリー内にあるコピーブック DXEECS を参照してください (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。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。

- CICS コマンド・レベル・インターフェース

QMF は、CICS コマンド・レベル・インターフェースを使用して、QMF 編集出口モジュール DSQUECIC を呼び出します。完了した編集出口モジュールは、EXEC CICS RETURN ステートメントを発行して、制御を QMF に戻す必要があります。

- ユーザーの編集出口プログラム。名前は DSQUECIC でなければなりません。

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・リソース管理表で、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。

IBM 提供の COBOL 内の編集プログラム DSQUXCTC は、QMF1210.SDSQSAP n ライブラリー内にあります (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)。このプログラム例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には、ユーザーのプライベート・プログラム・ソース・ライブラリーにコピーし、名前を DSQUXCT に変更してください。

下図は、CICS の COBOL 編集出口ルーチンの構造を示しています。

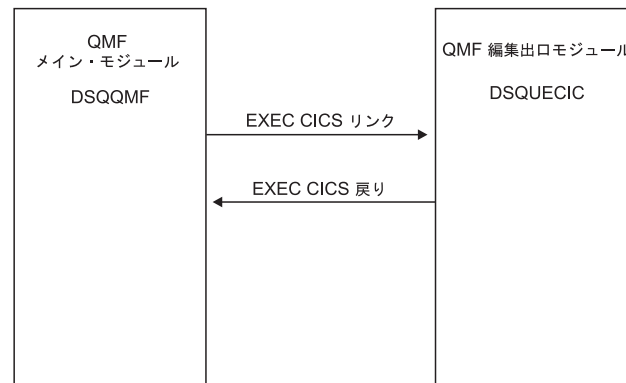


図 77. CICS の COBOL 編集出口ルーチンのプログラム構造

プログラムを呼び出し、QMF に制御を戻すためのステートメント

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。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 DXECS です。これはユーザーの編集コードを含み、ユーザー編集ルーチンが使用するためのスクラッチパッド域を用意します。制御ブロックは、ユーザー編集ルーチンへの呼び出しの間も存続しています。スクラッチパッド域は、出口ルーチンの最初の呼び出しの後には、QMF により変更されることはありません。

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 (DXECS、入力デ

ータ、および出力データ) にアドレスが渡されます。

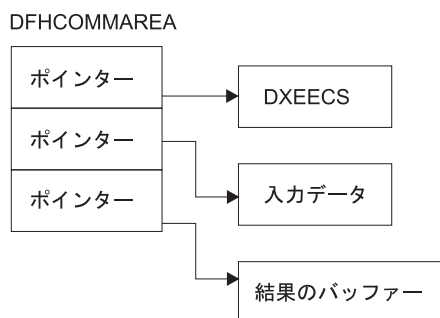


図 78. DFHCOMMAREA を使用して CICS で編集出口ルーチンのアドレス可能性を指定

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.
```

前出の図に示すように、QMF は、CICS 連絡域 DFHCOMMAREA で、ユーザー編集ルーチンの制御ブロックである DXEECS、入力データ、および出力データにアドレスを渡します。次のように、COBOL プログラム・リンケージ・セクションで DFHCOMMAREA のユーザー独自の記述を提供してください。用意します。

```
LINKAGE SECTION.
```

```
01 DFHCOMMAREA.  
02 ECSADR POINTER.  
02 ECSINADR POINTER.  
02 ECSRLADR POINTER.
```

ユーザー編集ルーチンの制御ブロック DXEECS、入力データ域 ECSINPT、および結果のデータ域 ECSRSLT にアドレス可能性を与えるには、次の例のように、これらのデータ域のアドレスを DFHCOMMAREA にある値に設定します。

```
SETUP SECTION.
```

```
SET ADDRESS OF DXEECS TO ECSADR.  
SET ADDRESS OF ECSINPT TO ECSINADR.  
SET ADDRESS OF ECSRSLT TO ECSRLADR.
```

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、コンパイルおよびリンク・エディット

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。353 ページの『プログラムを呼び出し、QMF に制

御を戻すためのステートメント』のリンケージ・セクションの例で指定した構造を提供することにより、標準の CICS 連絡域 DFHCOMMAREA を置き換えます。

QMF 編集出口インターフェース制御ブロック DXEEESC は、QMF1210.SDSQUSR n ライブラリー内にあり (ここで、 n は、ユーザーの使用する各国語を表す 1 文字の ID です)、コンパイル中、マクロ・ライブラリー内で使用可能である必要があります。

COBOL コンパイラー・オプション RENT、RES、および NODYNAM、さらにランタイム・オプション NOSTAE および NORTEREUS を指定します。

DXEEESC は引用符をリテラル区切り文字として使用します。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。

CICS によって提供される EXEC CICS インターフェースをユーザー編集出口プログラム (DSQUXCTC) に組み込むことにより、新しい QMF 編集出口モジュール DSQUEECIC を作成します。モジュール DSQUEECIC は、31 ビット・アドレッシング・モードで実行可能でなければなりません。

以下は、CICS のジョブを変換、コンパイル、およびリンク・エディットするためのステートメントの例です。このジョブを実行する前に、SYSLMOD ステートメント内で参照されているロード・ライブラリーを作成してください。そのライブラリー内に、ユーザー編集出口モジュールが常駐します。

```
//SAMCOBOL JOB ...
/* Add a parameter PROGLIB to procedure DFHYITVL
/*      PROGLIB=&PROGLIB,
//TRNCOMLK EXEC PROC=DFHYITVL,PROGLIB=QMF1210.SDSQLOAD',
//      PARM.TRN='QUOTE',
//      PARM.COB='RENT,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=QMF1210.SDSQSRE,DISP=SHR
//COB.SYSLMOD DD DSN=USER.LOADLIB,DISP=SHR
//LKED.SYSIN DD *
ENTRY DSQUEECIC
MODE AMODE(31) RMODE(31)
NAME DSQUEECIC(R)      /*
```

図 79. CICS の COBOL 編集出口プログラムを変換、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

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

第 17 章 QMF リソースの制御

管理プログラム出口ルーチンにより、ユーザーのサイトでエンド・ユーザーのアクティビティーを制限し、システム・リソースの使用を制御できます。

注: このトピックには、汎用プログラミング・インターフェースとそれに関連する指針としての情報が含まれます。

いくつかの管理ツールが QMF で使用できます。

- QMF for TSO/CICS に付属の管理機能を使用できます。
- QMF High Performance Option/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および制御を行うことができます。HPO/マネージャーは、QMF セッション・アクティビティー用のリアルタイム・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を提供します。HPO/マネージャーは、QMF 管理プログラムを置き換えます。
- DB2 リソース限定機能を使用して、QMF の操作を管理することができます。

QMF for TSO/CICS に付属のデフォルト管理プログラム出口ルーチン

このデフォルト管理プログラム・ルーチンは、デフォルトのリソース限界を提供します。例えば、ユーザーがデータベースから取り出すことができる行数や、QMF コマンドの実行に費やされる時間を制限できます。

TSO、ISPF、およびネイティブ z/OS で使用するために提供されている管理プログラム出口ルーチン (DSQUEGV1) は、データベースから検索できる行数、および QMF コマンドの実行に費やされるプロセッサ時間を制御するオプションを提供します。CICS 用に提供される管理プログラム出口ルーチン (DSQUEGV3) は、データベースから検索できる行数を制御します。

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。

以下のタイプの情報が、リソースを制御するために管理プログラムで使用されます。

- 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、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。これらの制御ブロックは、デフォルトの管理プログラム出口ルーチンを使用している場合は更新できません。デフォルトの管理プログラム出口ルーチンを変更する方法、または独自のルーチンを作成する方法に

については、379 ページの『デフォルトの管理プログラム出口ルーチンの変更または独自のルーチンの作成』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、デフォルト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、管理プログラム出口ルーチンは 362 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す。取り出し行数にデフォルト値を使用していない場合は、この警告をアクティブ化することもできます。

デフォルトの管理プログラム出口ルーチンは、QMF に制御を戻すときに行数のカウントをリセットし、管理プログラムの呼び出しごとに行数が累積されないようにしています。

関連タスク:

364 ページの『独自のリソース限界の定義』

データベースから取り出される行数に独自のリソース限界を設定するために、リソース・コントロール表にリソース・グループを追加することができます。

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、リソース制御情報が含まれています。

Q.RESOURCE_TABLE の各行には、以下の値が含まれています。

- リソース・グループの名前 (RESOURCE_GROUP)。これは、同じ方法でアクティビティーが管理される 1 人以上のユーザ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対して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。

下の表に、Q.RESOURCE_TABLE の各列についての説明があります。この表には、索引 Q.RESOURCE_INDEX があります。キー付きの列は、RESOURCE_GROUP と RESOURCE_OPTION です。

表 62. Q.RESOURCE_TABLE 表の構造

| 列名 | データ・タイプ と長さ | NULL の使 用 | 機能 / 値 |
|-----------------|----------------|--------------|--|
| RESOURCE_GROUP | CHAR(16) | 不可 | リソース・グループの名前を含む。 Q.PROFILES のユーザー行の RESOURCE_GROUP フィールド を更新し、そのユーザーの管理をアクティブ化す る。 ユーザーの QMF セッションに使用中のリソー ス・グループは、DSQAP_RESOURCE_GRP 大域変 数に記録されます。 |
| RESOURCE_OPTION | CHAR(16) | 不可 | モニターするリソースのユーザー独自の名前。 |
| 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。例えば、データベースの検索行数は整 数値のリソース。 |
| FLOATVAL | FLOAT | 可 | 浮動小数点値のリソース・オプションのリソース限 界を表す。FLOATVAL はデフォルトの管理プロ グラムについては NULL 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。例えば、DAY_OF_WEEK リソース・オプショ ンを確認し、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、デフォ ルトの管理プログラムの場合は、コメント列として使 用。 |

以下の 2 つの表は、IBM から出荷された状態の Q.RESOURCE_TABLE のデフォルト行を示しています。デフォルト表には、SYSTEM という名前の事前定義されたリソース・グループが含まれています。

下の表に、すべての環境に共通の行を示します。

表 63. デフォルト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： これらのオプションはすべての環境に共通です。

| RESOURCE_GROUP | RESOURCE_OPTION | INTVAL | FLOATVAL | CHARVAL |
|----------------|-----------------|--------|----------|-------------------------------|
| SYSTEM | SCOPE | - | - | 管理プログラムがアクティブであるかどうかを示す |
| SYSTEM | ROWPROMPT | 25000 | - | 25,000 行をフェッチした後にユーザーにプロンプトする |
| SYSTEM | ROWLIMIT | 100000 | - | 100,000 行後にアクティビティをキャンセルする |

また、下の表には、TSO、ISPF、およびネイティブ z/OS バッチ環境に限定して適用される Q.RESOURCE_TABLE の行を示します。

表 64. TSO、ISPF、またはネイティブ z/OS バッチ環境のみで使用可能な追加のデフォルト・リソース・オプション

| RESOURCE_GROUP | RESOURCE_OPTION | INTVAL | FLOATVAL | CHARVAL |
|----------------|-----------------|--------|----------|---------------------------|
| SYSTEM | TIMECHECK | 900 | - | 時間チェックの間の 15 分間 |
| SYSTEM | TIMEPROMPT | 360 | - | 6 分の CPU 時間後にユーザーにプロンプトする |
| SYSTEM | TIMELIMIT | 1440 | - | 24 分の CPU 時間の後にキャンセルする |

上の 2 つの表に示されているリソース・オプションには、以下の意味があります。

SCOPE

管理のアクティブ化に使用されます。

- Q.RESOURCE_TABLE の INTVAL 列内の値ゼロは、特定のリソース・グループに対する管理をアクティブ化します。
- NULL (デフォルト) を含む SCOPE のゼロ以外の値は、リソース・グループに対する管理を非アクティブ化します。

ROWPROMPT = 25000

データベースが 25,000 行以上検索されると、ユーザーに警告を与えます。

ROWLIMIT = 100000

ユーザーが警告を受けたとき継続することを決定した場合、管理プログラム 出口ルーチンは、100,000 行検索された時点でデータ検索アクティビティを取り消します (検索は FETCH の場合のみ)。

ROWLIMIT はバッファー・サイズによって異なります。したがって、バッファーに 100,000 で割りきれない行数が入っている場合は、100,000 行以上検索することもあり得ます。

TSO、ISPF、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。

TIMECHECK = 900

TIMECHECK は実時間の秒数で指定されます。つまり、値 900 は時間の検査、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。

TIMEPROMPT = 360

TIMEPROMPT はプロセッサ時間の秒で指定されます。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。したがって、値 360 はプロセッサ時間が 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。

TIMELIMIT = 1440

TIMELIMIT はプロセッサ時間の秒で指定されます。したがって、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 1440 秒 (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。

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は、Q.PROFILES 表内のユーザーの行からの情報を、Q.RESOURCE_TABLE 内の値と比較します。

IBM は、Q.RESOURCE_VIEW と呼ばれる Q.RESOURCE_TABLE の視点を提供しています。この視点には、Q.RESOURCE_TABLE の 5 つの列すべてが含まれています。QMF が管理プログラム出口ルーチン呼び出すたびに、QMF はユーザーが所属するリソース・グループに従って、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。リソース・グループを判別するために、QMF は Q.PROFILES 表のユーザー行の RESOURCE_GROUP 列の値を検査して、Q.RESOURCE_VIEW で一致する値を検査します。管理プログラム出口ルーチンは、このリソース情報を使用してユーザーがいつリソースの限界に達するかを判別するのに役立っています。

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。これらの制御ブロックは、デフォルトの管理プログラム出口ルーチンを使用している場合は更新できません。

QMF は、QMF セッション内のさまざまな時点で管理プログラム出口ルーチンに対して機能呼び出しを行います。

関連概念:

370 ページの『QMF が管理プログラムを呼び出す時点』

管理プログラム出口ルーチンに対する機能呼び出しは、特定のタイプの QMF アクティビティに先行するか、もしくはその後続くかのいずれかです。

関連タスク:

379 ページの『デフォルトの管理プログラム出口ルーチンの変更または独自のルーチンの作成』

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、デフォルト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。

デフォルト管理プログラム出口によるリソース制限

TSO、ISPF、およびネイティブ z/OS で使用するために提供されている管理プログラム出口ルーチン (DSQUEGV1) は、データベースから検索できる行数、および QMF コマンドの実行に費やされるプロセッサ時間を制御するオプションを提供します。CICS 用に提供される管理プログラム出口ルーチン (DSQUEGV3) は、データベースから検索できる行数を制御します。

デフォルトのリソース限界

デフォルト管理プログラムは、データベースから返される行数をモニターおよび制御します。さらに、TSO、ISPF、およびネイティブ z/OS で実行されている管理プログラムは、QMF コマンドの実行に費やされた時間の値を持っています。

デフォルトの管理プログラム出口ルーチンには、出荷時に、以下のような 2 つの行数の定義値が事前に設定されています。

- リソースの限界が近づいていることをユーザーに警告する行数

プロンプト・パネルは、検索された行数が 25,000 に達した時点で、図 80 に示すメッセージをユーザーに表示し、警告します。

```
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：図の左上隅のパネル ID 内の文字 *n* は、136 ページの表 27 に記載されている 1 文字の言語 ID を表します。

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ーは中断されません。DB2 は引き続き、行をフェッチし、プロセッサ時間を使います。

- QMF コマンドが取り消される行数

デフォルトの QMF 管理プログラムは、100,000 行が検索されるとデータ検索を取り消します (図 80 のメッセージに応答してユーザーが Enter キーを押した場合)。管理プログラムがデータ検索を取り消すと、図 81 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。例えば、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、図 82 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 行および 100,000 行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 コマンドの実行に費やされる時間について、さらに次の 2 つの事前定義値 (時間制限値と時間プロンプト値) があります。

- 時間プロンプト・パネルは、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、363 ページの図 83 に示すメッセージをユーザーに表示し、警告し

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：図の左上隅のパネル ID 内の文字 n は、136 ページの表 27 に記載されている 1 文字の言語 ID を表します。

- 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

データベースからの検索行数が 25,000 に達するとユーザーに警告を与え、100,000 に達すると QMF アクティビティーを取り消すように、管理プログラム出口ルーチンをセットアップすることができます。

このタスクについて

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、357 ページの『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、364 ページの『独自のリソース限界の定義』の手順を参照してください。

手順

1. SQL 照会パネルで図 84 に示したステートメントを入力し、RUN QUERY コマンドを発行して Q.RESOURCE_VIEW 表を更新します。

```
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. DSQSPRID パラメーターの TSOID 値を使用して QMF を開始しない限り、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。

重要: TRANSLATION 列には必ず値を指定します。指定しないと、Q.PROFILES 表で意図した行以外の行も変更してしまう場合があります。

例えば、364 ページの表 65 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。

DSQSPRID の値に TSOID を指定して QMF を開始した場合、リソース・グループ名はユーザーの TSO ユーザー ID になります。

ユーザーの QMF セッションに使用中のリソース・グループは、DSQAP_RESOURCE_GRP 大域変数に記録されます。

表 65. デフォルトのリソース限界を使用するためのユーザー・リソース・グループの更新

| 基本 QMF (英語) | ドイツ語 NLF |
|--|--|
| UPDATE Q.PROFILES SET RESOURCE_GROUP = 'SYSTEM' WHERE CREATOR='JONES' AND TRANSLATION='ENGLISH' | UPDATE Q.PROFILES SET RESOURCE_GROUP = 'SYSTEM' WHERE CREATOR='SCHMIDT' AND TRANSLATION='DEUTSCH' |

次のタスク

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、QMF 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別のセッションを開始して新規のリソース・グループをアクティブ化することにより可能です。

独自のリソース限界の定義

データベースから取り出される行数に独自のリソース限界を設定するために、リソース・コントロール表にリソース・グループを追加することができます。

このタスクについて

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。管理プログラムは、行数が 10,000 に達したときに GROUP1 のユーザーにプロンプトを出し、行数が 15,000 に達したときにユーザーのアクティビティーを取り消します。また、TSO、ISPF、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。

手順

リソース・コントロール表にリソース・グループを追加するには、以下の手順に従ってください。

1. リソース限界に近づいていることを示す警告をユーザーが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 に進んでください。

```
INSERT INTO Q.RESOURCE_VIEW (RESOURCE_GROUP,RESOURCE_OPTION,INTVAL)
VALUES('GROUP1','ROWPROMPT',10000)
```

図 85. 10,000 行検索済みのプロンプトの構成

2. 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。 次の例では、15,000 行での取り消しを設定します。

```
INSERT INTO Q.RESOURCE_VIEW (RESOURCE_GROUP,RESOURCE_OPTION,INTVAL)
VALUES('GROUP1','ROWLIMIT',15000)
```

図 86. 15,000 行の検索で QMF のアクティビティーを取り消す構成

3. リソース限界に近づいたことを示す警告をユーザーが受けるまでのプロセッサ経過時間を設定します。プロセッサ経過時間は、TSO、ISPF、およびネイティブ z/OS バッチ環境にのみ適用されます。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、4 の手順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 87. TSO、ISPF、およびネイティブ z/OS バッチでプロセッサ時間が 300 秒 (5 分) 経過後に発行されるプロンプトの構成

4. 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 88. TSO、ISPF、およびネイティブ z/OS バッチでプロセッサ時間が 1000 秒 (16 分を超過して) 経過後に QMF アクティビティーを取り消す構成

5. 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 89. TSO、ISPF、またはネイティブ z/OS バッチで約 13 分ごとに QMF がユーザーのアクティビティーとリソース限界を比較する構成

6. GROUP1 リソース・グループの管理をオンにします。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 90. GROUP1 リソース・グループに対する管理プログラムの開始

7. ユーザー JONES を英語版 QMF 環境内のリソース・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 91. Q.PROFILES 表のユーザー・リソース・グループの更新

同様のステートメントを使用して NLF 環境のユーザー・プロファイルを更新しますが、TRANSLATION には、QMF が NLF に使用する名前を表す値を使用してください。QMF が NLF に使用する名前については、136 ページの表 27 を参照してください。

次のタスク

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。

関連タスク:

379 ページの『デフォルトの管理プログラム出口ルーチンの変更または独自のルーチンの作成』

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、デフォルト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。

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

独自のリソース・コントロール表の作成

独自の表を作成するか、Q.RESOURCE_TABLE を名前変更することができます。

このタスクについて

Q.RESOURCE_VIEW が、自分が作成した表で定義されている視点であり、この表に Q.RESOURCE_TABLE 表の列構造が含まれている場合は、作成した表に追加の列を組み込むことができます。

重要: QMF for TSO/CICS を実行している場合は、視点の除去時に 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を除去して再作成する際には、QMF の外部で作業を行う必要があります。

手順

独自のリソース表を作成するには、以下の手順に従ってください。これらの SQL ステートメント例によって、MY_RESOURCES という表が作成されます。ステートメント内で、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。

1. Q.RESOURCE_VIEW はこの表で定義されているため、次のようなステートメントを使用してデータベースから Q.RESOURCE_TABLE を消去します。

```
DROP TABLE Q.RESOURCE_TABLE
```
2. 表を再作成します。TSO では、DSQTSSN1 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 DSQTSSN1
```

図 92. 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

3. QMF に付属の表スペースを使用しない場合は、独自の表スペースを作成する必要があります。QMF アプリケーション・プランを明示的に再バインドする場合は、そのプランに対する BIND 特権も必要になります。
4. Q.RESOURCE_VIEW を新しい表 MY_RESOURCES 上の視点として再定義します。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。

次の例は、Q.RESOURCE_VIEW を新しい表 MY_RESOURCES 上の視点として再定義する方法を示しています。例の表および列の名前を、独自の表および列の名前に置き換えてください。

```
CREATE VIEW Q.RESOURCE_VIEW
(RESOURCE_GROUP, RESOURCE_OPTION, INTVAL, FLOATVAL, CHARVAL)
AS SELECT GROUPNAME, CONSTRAINT, INTEGER, FLOAT_VALUE, CHARACTER
FROM MY_RESOURCES
```

5. Q.RESOURCE_VIEW での SELECT 特権を PUBLIC に付与します。
6. 新しい視点をテストします。視点のテストは、SPUFI を使用して行うことができます。最後に、QMF アプリケーション・プランを再バインドします。

関連概念:

358 ページの『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、リソース制御情報が含まれていません。

関連情報:

 DB2 資料

SQL コマンドに必要な権限に関する情報を検索します。

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メンバー・コンポーネントの名前は、環境および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、管理プログラム出口ルーチン・コンポーネントの名前と、各コンポーネントが果たす目的を示します。下の表に示されている名前の中の n 文字を、使用中の NLF と一致する 1 文字の言語 ID と置き換えてください。コンポーネント名の中で、1 は TSO、ISPF、およびネイティブ z/OS を表し、3 は CICS を表していません。

表 66. デフォルト管理プログラムのコンポーネント

| 環境 | 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 | DSQU n GV1 | QMF1210.SDSQLOAD | TSO、ISPF、およびネイティブ z/OS 用のロード・モジュール |
| | DSQU n GV1 | QMF1210.SDSQUSR n | TSO、ISPF、およびネイティブ z/OS 用の管理プログラム出口ルーチンのソース・コード |
| | DXEU n GV1 | QMF1210.SDSQUSR n | TSO、ISPF、およびネイティブ z/OS 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 |
| CICS | DSQU n GV3 | QMF1210.SDSQLOAD | CICS 用のロード・モジュール |
| | DSQU n GV3 | QMF1210.SDSQUSR n | CICS 用の管理プログラム出口ルーチンのソース・コード |
| | DXEU n GV3 | QMF1210.SDSQUSR n | 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 |
| | DXEU n GM | QMF1210.SDSQUSR n | CICS での管理プログラムのプロンプトの BMS マップ |
| 全環境 | DXEGOVA | QMF1210.SDSQUSR n | DXEGOVA 制御ブロックの DSECT |
| | DXEXCBA | QMF1210.SDSQUSR n | DXEXCBA 制御ブロックの DSECT QMF バージョン 12.1 では、この制御ブロックには 32 KB より大きい SQL 照会を管理するための 3 つの新規フィールドがあります。 |

NLF を使用している場合、各言語環境に異なるバージョンのモジュール DSQU n GV x 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。例えば、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、ISPF、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、ISPF、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。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

QMF と管理プログラムとの対話方法

QMF と管理プログラム出口ルーチンとの対話方法は、環境によって異なります。

ユーザーのセッション開始時に、QMF は管理プログラムをユーザーの仮想記憶域に入れます。TSO、ISPF、およびネイティブ z/OS では、QMF は LOAD コマンドを発行します。CICS では、QMF は EXEC CICS LOAD コマンドを発行して、管理プログラムをユーザーの仮想記憶域に入れます。

ロード・モジュール・ライブラリー QMF1210.SDSQLOAD は、ユーザーの STEPLIB データ・セットに連結されたライブラリーに入っていることを前提としています。

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。

下図は、TSO、ISPF、またはネイティブ z/OS での管理プログラム出口ルーチンのプログラム構造を示しています。

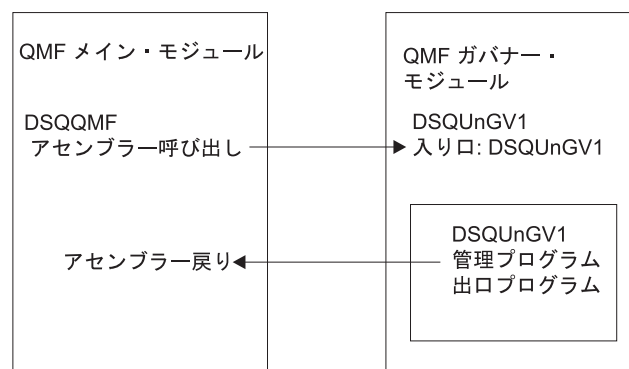


図 93. TSO、ISPF、またはネイティブ z/OS での QMF と管理プログラム出口との対話方法

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 に付属のプロログおよびエピログ・マクロ DFHEIENT および DFHEIRET
- 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 DSQU n GV3。

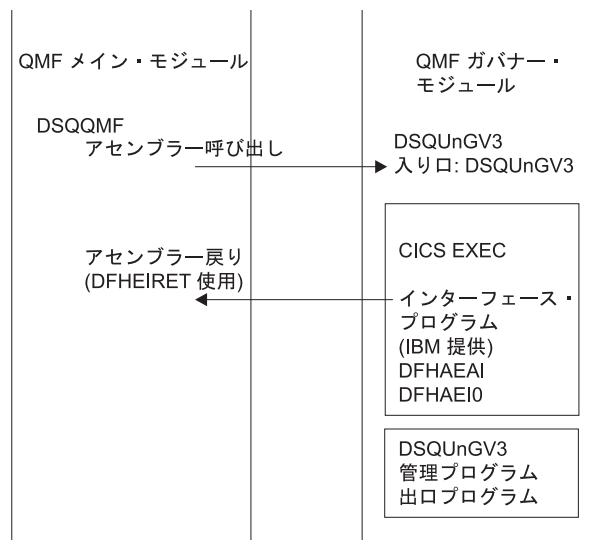


図 94. CICS 提供の環境での QMF と管理プログラム出口との対話方法

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。

CICS では、管理プログラム出口ルーチンへの入り口点は 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。

ロードが失敗した場合、あるいはモジュールが 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ログに記録し、管理プログラム出口を使用不可にし、さらに管理プログラムなしでセッションを継続します。

QMF が管理プログラム出口ルーチン呼び出す方法と時期

管理プログラム出口ルーチンを作成または変更する際には、QMF が出口ルーチン呼び出す時点での、ルーチンに対するアドレス可能度を設定する必要があります。

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 アクティビティに先行するか、もしくはその後続くかのいずれかです。

下の図に、管理プログラムがリソースの使用を制限する方法の概要を示します。

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 ステータス

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述べています。

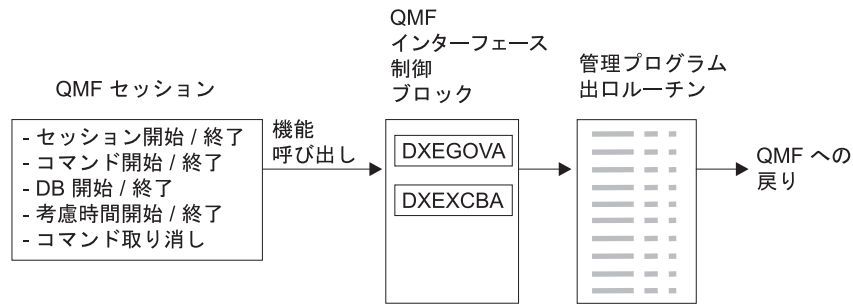


図 95. 管理プログラム出口ルーチンと QMF for TSO/CICS との連携方法

機能呼び出しは、以下の時点で行われます。

表 67. QMF が管理プログラム出口ルーチンに対して機能呼び出しを行う時点：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。

| QMF アクティビティー | 管理プログラム出口ルーチンに対する機能呼び出しの説明 |
|--------------------|--|
| 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、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リソース情報がそのユーザーが QMF の使用を許可されていないことを示している場合です。 |
| コマンド実行の前後 | QMF は、すべてのコマンドの実行前と実行後に管理プログラムを呼び出します。コマンド完了に関する呼び出しの前に、コマンド開始に関する呼び出しが複数存在する可能性があります。例えば、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。 |

表 67. QMF が管理プログラム出口ルーチンに対して機能呼び出しを行う時点 (続き): 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 アクティビティに先行するか、もしくはその後続くかのいずれかです。

| QMF アクティビティ | 管理プログラム出口ルーチンに対する機能呼び出しの説明 |
|--------------------------|---|
| データベース・アクティビティの開始前および終了後 | <p>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・アクティビティの完了時にも管理プログラムを呼び出します。</p> <p>QMF がデータを検索する場合、最小サイズ 4 KB のバッファに可能な最大行数を納めます。QMF は、最初の行を検索してバッファに入れるときに 1 回、バッファ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。</p> <p>また、SQL、QBE、または指示照会が RUN QUERY コマンドを使用して実行依頼されたとき、あるいは QMF が QMF コマンドで必要な SQL ステートメント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。例えば、SAVE DATA コマンドは DELETE、CREATE、および INSERT ステートメント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 DATA オブジェクトがあった場合、DATA 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。</p> <p>以下の QMF コマンドは、常にデータベース・アクティビティを強制します。</p> <ul style="list-style-type: none"> • DISPLAY (オブジェクトが表または視点の場合) • EDIT TABLE • ERASE TABLE • EXPORT TABLE • IMPORT TABLE • PRINT (オブジェクトが表または視点の場合) • RUN (照会の場合) • SAVE DATA (1 つ以上の暗黙 CREATE TABLE ステートメントが生成) • 報告書が表示されるときに、データのフェッチをもたらすスクロール・コマンド • データ検索操作 (フェッチ操作) |

表 67. QMF が管理プログラム出口ルーチンに対して機能呼び出しを行う時点 (続き): 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 アクティビティに先行するか、もしくはその後続くかのいずれかです。

| QMF アクティビティ | 管理プログラム出口ルーチンに対する機能呼び出しの説明 |
|-------------|---|
| ユーザー選択の前後 | <p>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 が待機に費やす時間は考慮時間と呼ばれます。</p> <p>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。</p> <p>以下のいずれのアクティビティが行われた場合に考慮時間をとる必要が生じます。</p> <ul style="list-style-type: none"> • 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。例えば、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。例えば、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 | <p>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。</p> |

関連概念:

445 ページの『QMF パフォーマンスの管理』

QMF パフォーマンスをモニターして、一般的なパフォーマンス問題に対する何らかの解決方法または回避方法を見つけることができます。

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 DSQUnGV1 (TSO、ISPF、またはネイティブ z/OS の場合) または DSQUnGV3 (CICS の場合) のアドレスへ分岐することにより、管理プログラム出口ルーチンを呼び出します。

管理プログラムの呼び出し時、QMF は常に DSQUnGV1 (TSO、ISPF、またはネイティブ z/OS の場合) または DSQUnGV3 (CICS の場合) という名前の入り口点に分岐します。n は 1 文字の各国語 ID です。このため、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。代わりに、GOVFNCT 制御ブロック・フィールドを使用します。その値は、出口のタイプを識別する正の整数です。機能呼

び出しの各タイプには、GOVFNCT フィールドの特別の値があります。これらの値は、376 ページの図 99 に示されています。

TSO、ISPF、またはネイティブ z/OS での DSQUEGV1 入り口点への分岐

QMF は、入り口点 DSQUEGV1 のアドレスへ分岐することにより、TSO、ISPF、またはネイティブ z/OS で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 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 DXEXCBA 制御ブロックのアドレスであり、もう 1 つは DXEGOVA 制御ブロックのアドレスです。

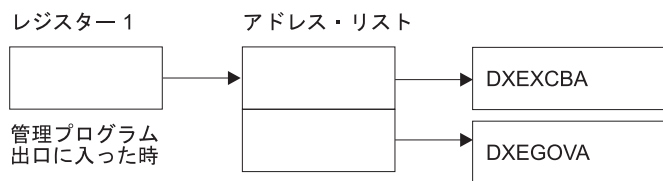


図 96. TSO、ISPF、またはネイティブ z/OS での管理プログラム出口ルーチン呼び出し時のレジスタ 1 の内容

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

管理プログラムが呼び出された後、次のサンプル・コードのような、TSO、ISPF、またはネイティブ z/OS のデフォルト管理プログラムからのコードで、管理プログラムが開始されることがあります。

```

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 97. TSO、ISPF、またはネイティブ z/OS での管理プログラム・セッションの開始時点を示すサンプル・コード

上のサンプル内のコードは最初に、仮想記憶域のダンプで目印としての役割を果たす定数ブロックの周辺に分岐します。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 および DXEGOVA 制御ブロックへのポインター、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、デフォルト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。

管理プログラムは、呼び出しの処理後、標準の方法で制御を QMF に返します。以下のデフォルト管理プログラムのサンプル・コードに示すように、標準のエピローグおよびプロローグを使用する必要があります。

```

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

```

図 98. TSO、ISPF、またはネイティブ z/OS でデフォルトの管理プログラム出口ルーチンが QMF に制御を戻す方法

CICS での DSQU_nGV3 入り口点への分岐

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 に付属のマクロ 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。下の図に、管理プログラムの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。

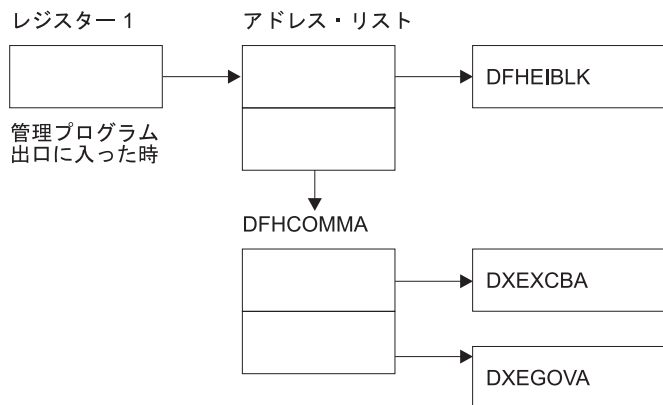


図 99. CICS での管理プログラム出口ルーチン呼び出し時のレジスター 1 の内容

- 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 作業用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 (例えば、CICS HANDLE CONDITION) を使用する場合は、注意してください。CICS HANDLE CONDITION ステートメントを使用する必要がある場合は、EXEC CICS PUSH および EXEC CICS POP ステートメントを使用して既存の処理条件を保管し、リストアします。

以下のような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      FENTRY          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
*
FENTRY  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 100. 管理プログラム・セッションの開始時点でのサンプル・コード (CICS の場合)

上のサンプル内のコードは最初に、仮想記憶域のダンプで目印としての役割を果たす定数ブロックの周辺に分岐します。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 および DXEGOVA 制御ブロックへのポインター、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、デフォルト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。このコマンドは、QMF リソースを解放しないで QMF セッションを終了します。

管理プログラムは以下のようなコードで終了します。

```

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

```

図 101. CICS での管理プログラムの終了コード

関連概念:

380 ページの『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。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

機能呼び出しのアドレス可能度の確立

QMF が管理プログラムを呼び出すときには、常に DSQU n GV1 (TSO、ISPF、またはネイティブ z/OS) 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 GOVFUNCT フィールドを使用してください。

IBM 提供の管理プログラム出口ルーチンの GOVFUNCT は、機能呼び出しのタイプを識別する文字値を持っています。この文字値は、順に 1 から 10 の 1 バイトの 2 進整数の値を持ちます。例えば、QMF セッションの開始に関する機能呼び出しでは、GOVFUNCT の値は GOVINIT で、これは X'1' の数値と等価です。

機能呼び出しの各タイプのための文字値および数値の両方を、次の図に示します。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 DATABASE |
| GOVEDBAS | EQU | 7 | ----- | END DATABASE |
| GOVSACTV | EQU | 8 | ----- | SUSPEND QMF ACTIVITY |
| GOVRACTV | EQU | 9 | ----- | RESUME QMF ACTIVITY |
| GOVABEND | EQU | 10 | ----- | QMF ABEND OPERATION |

図 102. DXEGOVA 制御ブロックの GOVFUNCT 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、GOVFUNCT の値をブランチ表を使用して 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、ブランチ表を使用してブランチすべきアドレスを見つけます。

次のコード例は、デフォルト管理プログラムのブランチ・アドレスを特定しています。

| | | | |
|---------|---------|------------------|-----------------------------------|
| XR | R07,R07 | ZERO REGISTER 7 | |
| | IC | R07,GOVFNCT | 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 103. 機能呼び出しのタイプの識別と該当するアドレスへのブランチ

デフォルトの管理プログラム出口ルーチンの変更または独自のルーチンの作成

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、デフォルト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。

始める前に

- 368 ページの『管理プログラム出口ルーチンのプログラム・コンポーネント』をお読みください。
- 369 ページの『QMF と管理プログラムとの対話方法』をお読みください。
- QMF が出口ルーチンを呼び出す時点での、ルーチンのアドレス可能性を確立します。370 ページの『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』で、この手順について説明しています。

このタスクについて

リソース制御情報を管理プログラム出口ルーチンに渡すために使用される制御ブロックのうちの 1 つである DXEXCBA は、32 KB よりも大きい SQL 照会の管理に対応するように QMF バージョン 12.1 で変更されました。この機能を活用する予定があり、QMF の旧リリースでデフォルトの管理プログラムを変更した場合、またはユーザー作成の管理プログラム・ルーチンを既に使用している場合は、この機能を提供する制御ブロック・フィールドを利用するようにルーチンを変更する必要があります。

関連概念:

385 ページの『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) とを組み合わせ、リソース限界を超過した時点を判断し、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。

リソース制御情報を管理プログラム出口に渡す

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。

作業を始める前に、管理プログラムで使用可能な形式で独自のリソース限界を定義する方法を知っておく必要があります。

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、QMF1210.SDSQSUSRE 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。

QMF バージョン 12.1 では、32 KB よりも大きい SQL 照会の管理を提供するように、DXEXCBA 制御ブロックに 3 つの新規フィールド (XCBQRYPT、XCBQRYP2、および XCBQRYL2) が追加されました。旧リリースでデフォルト管理プログラム出口ルーチンを変更した場合、または独自の出口を作成した場合、このリリースとの互換性を確保し、この機能を利用するためには、いくつかの移行手順が必要です。

関連概念:

115 ページの『QMF バージョン 12 リリース 1 と旧リリースとの前方互換性』旧リリースで作成したほとんどのオブジェクトは、変更なしで QMF バージョン 12.1 で使用できます。

関連タスク:

364 ページの『独自のリソース限界の定義』データベースから取り出される行数に独自のリソース限界を設定するために、リソース・コントロール表にリソース・グループを追加することができます。

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。

下表は、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。例えば、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、すべての QMF 環境に適用されます。したがって、制御ブロックに示される一部の情報は、ユーザーが使用中の環境の QMF には適用されない場合があります。

表 68. 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 | 説明 |
|----------|---------|--|
| 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。可能な値は、以下のとおり。 <ul style="list-style-type: none"> • GOVINIT (セッションの初期化); GOVTERM (セッション終了) • GOVSCMD (開始コマンド); GOVECMD (終了コマンド) • GOVCONN (接続コマンド) • GOVSDBAS (データベース検索操作の開始); GOVEDBAS (データベース検索操作の終了) • GOVSACTV (ユーザー考慮時間による QMF アクティビティーの延期); GOVRACTV (QMF アクティビティーの再開) • GOVABEND (異常終了の開始) |
| 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 つのブロックは、QMF リソース・コントロール表内のユーザーのリソース・グループの行の 1 つ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 フィールドを使用し、追加行にアクセスする。 |


```

*
*          CHANGE ACTIVITY: NA
*
*          CHANGE DATE: NA
*
*****
*
DXEGOVA  DSECT
          DS    0D
GOVNAME  DS    CL8      -- CONTROL BLOCK IDENTIFICATION
          SPACE
GOVEXCTL DS    XL72     -- EXIT CONTROL
          ORG    GOVEXCTL
GOVFUNCT DS    XL1     ----- FUNCTION CODE
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
GOVPAD10 DS    CL7     ----- RESERVED FIELD
          SPACE
GOVCADDR DS    A      ---- ADDR TO BRANCH TO FOR CANCELLATION
          SPACE
GOVOROWS DS    F      ---- NUMBER OF OPTION ROWS RETRIEVED
          SPACE
GOVSQLRC DS    F      ---- RESOURCE TABLE SQL RETURN CODE
          SPACE
GOVSQLCA DS    A      ---- ADDRESS OF SQLCA FOR ERROR CONDITION
          SPACE
GOVGROUP DS    CL16    ---- GROUP NAME
GOVPAD20 DS    CL32    ---- RESERVED FIELD
          SPACE
GOVUCTL  DS    XL304   -- USER CONTROL AREA
          ORG    GOVUCTL
GOVUSERS DS    CL2048  ----- USER SCRATCH PAD AREA
GOVPAD30 DS    CL48    ----- RESERVED FIELD
          SPACE
          DS    0D
GOVRESC  DS    10XL128 -- RESOURCE CONTROL TABLE
          ORG    GOVRESC
GOVRESCT DSECT      -- RESOURCE CONTROL TABLE MAPPING
          DS    0D
GOVOPTN  DS    CL16    ----- RESOURCE OPTION
GOVNULLI DS    H      ----- INTEGER NULL INDICATOR
GOVPAD40 DS    CL2     ----- RESERVED FIELD
GOVINTVL DS    F      ----- INTEGER OPTION REPRESENTATION
GOVNULLF DS    H      ----- FLOATING POINT NULL INDICATOR
GOVPAD50 DS    CL6     ----- RESERVED FIELD
GOVFLOAT DS    D      ----- FLOATING POINT OPTION REPRESENTATION
GOVNULLC DS    H      ----- CHARACTER NULL INDICATOR
GOVCHLEN DS    H      ----- LENGTH OF THE CHARACTER OPTION
GOVCHAR  DS    CL80    ----- CHARACTER OPTION REPRESENTATION
GOVNEXTR DS    A      ----- POINTER TO NEXT RESOURCE CONTROL ROW

```

図 104. DXEGOVA 制御ブロック

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 制御ブロックの GOVRESCT 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 GOVRESCT 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。例えば、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。

デフォルト管理プログラムのコードの以下のセクションでは、リソース制御情報のブロックを処理します。このコードでは、GOVRESCT 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,GOVRESCT     GET ADDRESS OF 1ST RESOURCE ROW
      USING GOVRESCT,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

      . . .
      . . .
      . . .

GOVRESCT DS  10XL128      -- RESOURCE CONTROL TABLE
      ORG  GOVRESCT
GOVRESCT DSECT           -- DSECT FOR RESOURCE ROW
      . . .
      . . .
      . . .
GOVNEXTR DS  A           -- POINTER TO NEXT RESOURCE ROW
      . . .
      . . .
      . . .

```

図 105. 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) とを組み合わせ、リソース限界を超過した時点を判断し、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。

例えば、ユーザー JONES が INSERT または UPDATE ステートメントを発行できないように、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 XCBQRYPT または XCBQRYPT2 フィールドに INSERT または UPDATE ステートメント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。

QMF バージョン 12.1 では、32 KB よりも大きい SQL 照会の管理を提供するように、DXEXCBA 制御ブロックに 3 つの新規フィールド (XCBQRYPT、XCBQRYPT2、および XCBQRYL2) が追加されました。旧リリースで

デフォルト管理プログラム出口ルーチンを変更した場合、または独自の出口を作成した場合、このリリースとの互換性を確保し、この機能を利用するためには、いくつかの移行手順が必要です。

下表は、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、ユーザーが使用中の環境に適用されない場合があります。

表 69. 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 | 説明 |
|----------|---------|--|
| XCBACTIV | CL1 | データベース・アクティビティーの現行タイプを示している。行が現行 DATA 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。値は以下のとおりです。 <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 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。 |
| XCBAIACT | CL1 | 現行のコマンドが対話式に実行しているか否かを示す。 <ol style="list-style-type: none"> 1 対話式 0 非対話式 (バッチ) 対話式コマンドは、プロンプトと状況パネルを表示する。このフィールドは、コマンド開始の任意の機能呼び出しで値を変更する。コマンドが完了するとゼロにリセットされる。 |
| XCBAUTH | CL8 | ユーザーの SQL 許可 ID を含んでいる。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、XCBAUTHX を参照してください。 |
| XCBAUTHX | CL128 | ユーザーの SQL 許可 ID を含んでいる。 |
| XCBCAN | CL1 | (ユーザーまたは管理プログラムのいずれかによって) 現行コマンドの取り消しが要求されたことを示す。取り消しが要求された場合は、このフィールドは 1 に設定される。ゼロは、取り消しの要求がなかったことを示す。取り消しが要求された時点で、値が変更される。このフィールドは、コマンド終了の機能呼び出しの前にゼロにリセットされる。 |
| XCBCLOC | CL18 | 現行ロケーションの名前が含まれる。 |
| XCBCMDL | F | 実行するコマンドを含むストリングの長さが含まれる。これは、XCBCMDP フィールドにより指されたストリングである。XCBCMDL が値を変更すると、このフィールドの値が変わる。 |

表 69. 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 | 説明 |
|--------------------|---------|---|
| XCBCMDP | A | 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時点で 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。 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。 |
| XCBCVERB | CL18 | 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。 |
| XCBDBMG | CL1 | データベース・マネージャーを識別する。この値は、DB2 (z/OS 版) の場合は 2 に設定される。 |
| XCBEMODE | CL1 | QMF セッションの現行モードを示している。 1 対話式 2 非対話式 (バッチまたはサーバー) この値はセッション中は変更されない。 |
| XCBERRET | F | 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。 |
| 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。 |
| 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。 |
| XCBKPARM | CL1 | DSQSDBCS プログラム・パラメーターの設定方法を管理プログラムに伝える。この値はセッション中、変わらない。取り得る値は、0 がラテン文字、1 が 2 バイト文字セット (DBCS) データ。 |
| 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、メッセージをトレース・データに記録するか、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。この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は変更されない。このフィールドは、CICS には適用されない。 |

表 69. 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 | 説明 |
|----------|---------|--|
| XCBQCE | F | データベースから戻された、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、データベース状況パネルに『相対費用』見積もりとして現れる。この費用見積もりがパネルに表示されないようにするには、DSQDC_COST_EST 大域変数を 0 に設定する。 XCBQCE フィールドの値は、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。 |
| 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 の現行リリースを識別する。この値は QMFV12R1.0 であり、セッション中では変更されない。 |
| XCBQRYL2 | F | フィールド XCBQRYP2 によってアドレス指定される SQL 照会の長さを含む。 |

表 69. 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 | 説明 |
|---------|---------|---|
| 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。照会の長さは最初のハーフワード。</p> <p>QMF バージョン 12.1 で初めて管理プログラム出口ルーチンを作成している場合は、このフィールドの代わりにフィールド XCBQRYP2 および XCBQRYL2 を使用して SQL 照会を指し、照会の長さを保管するようにする。この理由は、これらのフィールドが任意のサイズの照会をサポートするため。</p> <p>XCBQRYP によって参照される照会が 32 KB より大きい場合、照会は 32 KB で切り捨てられ、照会が長すぎることを示すようにフィールド XCBQRYP2 が設定される。この場合、フィールド XCBQRYP2 は照会のアドレスを指定し、フィールド XCBQRYL2 は照会の長さを与える。</p> <p>DSQEC_SQLQRYSZ_2M 大域変数は、32 KB より大きい SQL 照会をサポートされるかどうかを制御します。</p> <p>管理プログラムは、データベース・アクティビティーの開始を呼び出す時に (データ検索の前)、照会を検査し、アクティビティー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 DATA 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。DB2 for z/OS のシステム表または QMF コントロール表についての照会に対しては何も情報が提供されない。</p> <p>DISPLAY TABLE EDIT TABLE</p> <p>ERASE TABLE EXPORT TABLE</p> <p>IMPORT TABLE PRINT TABLE</p> <p>RUN QUERY SAVE DATA</p> |

表 69. 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 | 説明 |
|----------|---------|--|
| XCBQRYP2 | 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。このフィールドは、長さに関係なくすべての照会を収容する。新しい管理プログラム出口ルーチンを作成している場合は、XCBQRYP の代わりにこのフィールドを使用する。フィールド XCBQRYL2 が、XCBQRYP2 フィールドによってアドレス指定された照会の長さを与える。</p> <p>管理プログラムは、データベース・アクティビティーの開始を呼び出す時に (データ検索の前)、照会を検査し、アクティビティー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 DATA 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。DB2 (z/OS 版) のシステム表または QMF コントロール表についての照会に対しては何も情報が提供されない。</p> <p>DISPLAY TABLE EDIT TABLE</p> <p>ERASE TABLE EXPORT TABLE</p> <p>IMPORT TABLE PRINT TABLE</p> <p>RUN QUERY SAVE DATA</p> |
| XCBQRYPT | CL1 | <p>照会が 32 KB より大きいかどうかを示す。大きい場合は、フィールド XCBQRYP によってアドレス指定される照会が切り捨てられる。可能な値は、以下のとおり。</p> <p>0 フィールド XCBQRYP によってアドレス指定された照会は切り捨てられていない (照会は 32 KB 以下)。</p> <p>1 フィールド XCBQRYP によってアドレス指定された照会は切り捨てられた (照会は 32 KB より大きい)。</p> <p>XCBQRYPT が 1 に設定されている場合、XCBQRYL2 は照会の長さを保管し、XCBQRYP2 は照会のコピーのアドレスを含む。</p> <p>DSQEC_SQLQRYSZ_2M 大域変数は、32 KB より大きい SQL 照会をサポートされるかどうかを制御します。</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 | <p>QMF のリリース・レベルを識別する。QMF バージョン 12 リリース 1 の場合、これは 19 である。この値はセッション中では変更されない。</p> |
| XCBRGRP | CL16 | <p>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では変更されない。</p> |

表 69. 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 | 説明 |
|----------|---------|--|
| XCBROWSF | F | DATA 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。 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。例えば、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。 |
| XCBSYST | CL1 | 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、ISPF、またはネイティブ z/OS を示す 3 に設定されている。可能な値は、以下のとおり。 3 TSO、ISPF、またはネイティブ z/OS バッチ 5 CICS |
| XCBTRACE | CL1 | ユーザー出口アクティビティ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)。 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。 |
| XCBUSER | CL8 | 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、ブランクになっている。 |
| XCBUSERS | CL2048 | ユーザーがある呼び出しから次の呼び出しまで管理プログラムに保存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。 |

DXEXCBA 制御ブロックの構造を次に示します。

```

*****
*
*      CONTROL BLOCK NAME: DXEXCBA
*
*      FUNCTION:
*
*      THIS IS THE INTERFACE CONTROL BLOCK BETWEEN QMF AND
*      EXIT ROUTINES.
*
*      STATUS: Version 11 Release 2 Level 0
*
*      INNER CONTROL BLOCKS: NONE
*
*****
*
DXEXCBA DSECT
        DS      0D
XCBNAME DS      CL8      -- CONTROL BLOCK IDENTIFICATION
        SPACE
XCBEXCTL DS      XL190   -- EXIT CONTROL
        ORG      XCBEXCTL
XCBAUTH  DS      CL8      ----- AUTHORIZATION ID
XCBUSER  DS      CL8      ----- USER ID
    
```

| | | | | |
|----------|-------|------|-------|------------------------------------|
| XCBPLAN | DS | CL8 | ----- | PLAN ID |
| | SPACE | | | |
| XCBQMF | DS | CL10 | ----- | CURRENT VERSION/RELEASE |
| | SPACE | | | |
| XCBRELN | DS | CL2 | ----- | QMF RELEASE LEVEL |
| | SPACE | | | |
| XCBTRACE | DS | CL1 | ----- | QMF EXIT TRACE LEVEL |
| XCBTOFF | EQU | C'0' | ----- | NO TRACING |
| XCBTPART | EQU | C'1' | ----- | PARTIAL TRACING |
| XCBTFULL | EQU | C'2' | ----- | FULL TRACING |
| | SPACE | | | |
| XCBSYST | DS | CL1 | ----- | OPERATING SYSTEM |
| XCBSYSTX | EQU | C'3' | ----- | MVS/ESA or XA (TSO,APPC, native) |
| XCBSYSTY | EQU | C'5' | ----- | CICS (MVS) |
| | SPACE | | | |
| XCBPAD10 | DS | CL4 | ----- | RESERVED FIELD |
| | SPACE | | | |
| XCBNLANG | DS | CL1 | ----- | CURRENT NATIONAL LANGUAGE |
| | SPACE | | | |
| | | | | |
| XCBKPARM | DS | CL1 | ----- | SETTING OF K PARAMETER |
| XCBKPARN | EQU | C'0' | ----- | LATIN |
| XCBKPARY | EQU | C'1' | ----- | DBCS |
| | SPACE | | | |
| XCBDBMG | DS | CL1 | ----- | DATA BASE MANAGER |
| XCBDBMGS | EQU | C'1' | ----- | DB2 FOR SQL/DS |
| XCBDBMGD | EQU | C'2' | ----- | DB2 FOR OS/390 |
| XCBDBMGW | EQU | C'3' | ----- | WORKSTATION DB2 |
| | SPACE | | | |
| XCBEMODE | DS | CL1 | ----- | CURRENT EXECUTION MODE |
| XCBIACTV | EQU | C'1' | ----- | INTERACTIVE MODE |
| XCBBATCH | EQU | C'2' | ----- | BATCH MODE |
| | SPACE | | | |
| XCBIAICT | DS | CL1 | ----- | CURRENT INTERACT MODE |
| XCBIAICY | EQU | C'1' | ----- | INTERACTIVE EXECUTION |
| XCBIAICN | EQU | C'0' | ----- | NOT INTERACTIVE EXECUTION |
| | SPACE | | | |
| XCBINCI | DS | CL1 | ----- | CURRENT COMMAND INTERFACE STATE |
| XCBINCIY | EQU | C'1' | ----- | COMMAND INTERFACE ACTIVE |
| XCBINCIN | EQU | C'0' | ----- | COMMAND INTERFACE NOT ACTIVE |
| | SPACE | | | |
| XCBINPRC | DS | CL1 | ----- | PROCEDURE OR LIST CMD EXEC STATE |
| XCBPRCY | EQU | C'1' | ----- | RUNNING A PROCEDURE OR LIST CMD |
| XCBPRCN | EQU | C'0' | ----- | NOT RUNNING PROCEDURE OR LIST CMD |
| | SPACE | | | |
| XCBCVERB | DS | CL18 | ----- | CURRENT COMMAND VERB |
| | SPACE | | | |
| XCBCAN | DS | CL1 | ----- | CANCEL CURRENT COMMAND INDICATOR |
| XCBCANN | EQU | C'0' | ----- | NO CANCELLATION |
| XCBCANY | EQU | C'1' | ----- | CANCELLATION IN PROGRESS |
| | SPACE | | | |
| XCBACTIV | DS | CL1 | ----- | TYPE OF DATA BASE ACTIVITY |
| XCBOPEN | EQU | C'1' | ----- | OPEN |
| XCBFETCH | EQU | C'2' | ----- | FETCH |
| XCBPREP | EQU | C'3' | ----- | PREPARE |
| XCBDESCR | EQU | C'4' | ----- | DESCRIBE |
| XCBCLOSE | EQU | C'5' | ----- | CLOSE |
| XCBEXEC | EQU | C'6' | ----- | EXECUTE |
| XCBEXECI | EQU | C'7' | ----- | EXECUTE IMMEDIATE |
| XCBPAD20 | DS | CL9 | ----- | RESERVED FIELD |
| | SPACE | | | |
| XCBRGRP | DS | CL16 | ----- | RESOURCE GROUP NAME |
| XCBPAD30 | DS | CL22 | ----- | RESERVED FIELD |
| | SPACE | | | |
| XCBCMDP | DS | A | ----- | POINTER TO ORIGINAL COMMAND STRING |
| * | | | ----- | WILL NOT CONTAIN PROMPT VALUES |
| | SPACE | | | |

| | | | |
|----------|-------|---------|--|
| XCBCMDL | DS | F | ----- ORIGINAL COMMAND STRING LENGTH |
| | SPACE | | |
| XCBQCE | DS | F | ----- QUERY COST ESTIMATE VALUE |
| | SPACE | | |
| XCBROWSF | DS | F | ----- DATA BASE ROWS FETCHED FROM SOURCE |
| * | | | ----- SET BY QMF; EXIT MAY RESET |
| | SPACE | | |
| XCBQERR | DS | CL1 | ----- QMF ERROR INDICATOR |
| XCBQERRN | EQU | C'0' | ----- NO QMF ERROR DETECTED |
| XCBQERRY | EQU | C'1' | ----- QMF ERROR DETECTED |
| XCBCLOC | DS | CL18 | ----- CURRENT LOCATION NAME |
| XCBQRYPT | DS | CL1 | ----- XCBQRYPT TRUNCATION SWITCH |
| XCBQRYTN | EQU | C'0' | ----- XCBQRYPT IS NOT TRUNCATED |
| XCBQRYTY | EQU | C'1' | ----- XCBQRYPT IS TRUNCATED |
| XCBPAD32 | DS | CL32 | ----- RESERVED FIELD |
| XCBQRYP2 | DS | A | ----- POINTER TO SQL QUERY |
| XCBQRYL2 | DS | F | ----- LENGTH OF QUERY IN XCBQRYP2 |
| | SPACE | | |
| XCBQRYP | DS | A | ----- POINTER TO SQL QUERY |
| * | | | ----- QUERY LENGTH IS FIRST HALFWORD |
| * | | | ----- QUERY TRUNCATED IF GREATER 32K |
| | SPACE | | |
| XCBUCTL | DS | XL432 | -- USER CONTROL AREA |
| | ORG | XCBUCTL | |
| XCBERRET | DS | F | ----- EXIT ERROR RETURN CODE |
| XCBMGTX | DS | CL78 | ----- EXIT ERROR MESSAGE TEXT |
| XCBMSGNO | DS | CL8 | ----- ISPF MESSAGE NUMBER |
| XCBPANEL | DS | CL8 | ----- ISPF MESSAGE HELP PANEL |
| XCBLOGM | DS | CL1 | ----- LOG MESSAGE INDICATOR |
| XCBLOGMN | EQU | C'0' | ----- QMF SHOULD NOT LOG MESSAGE |
| XCBLOGMY | EQU | C'1' | ----- QMF SHOULD LOG MESSAGE |
| XCBREFR | DS | CL1 | ----- REFRESH SCREEN INDICATOR |
| XCBREFRN | EQU | C'0' | ----- QMF DOES NOT HAVE TO REFRESH SCR |
| XCBREFRY | EQU | C'1' | ----- QMF SHOULD REFRESH SCREEN |
| XCBPAD50 | DS | CL28 | ----- RESERVED FIELD |
| | SPACE | | |
| XCBUSERS | DS | CL2048 | -- USER SCRATCH PAD AREA |
| XCBPAD60 | DS | CL48 | ----- RESERVED FIELD |
| XCBAUTHL | DS | H | -- LENGTH OF AUTHORIZATION ID |
| XCBAUTHX | DS | CL128 | -- AUTHORIZATION ID EXTENDED |
| XCBPAD70 | DS | CL50 | ----- RESERVED FIELD |

図 106. DXEXCBA 制御ブロック

関連概念:

115 ページの『QMF バージョン 12 リリース 1 と旧リリースとの前方互換性』旧リリースで作成したほとんどのオブジェクトは、変更なしで QMF バージョン 12.1 で使用できます。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

454 ページの『トレース機能』

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。

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)。

デフォルト管理プログラムはここに示したコードを使用して、DXEGOVA 制御ブロック内のスクラッチパッド域 GOVUSERS をアドレス指定します。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 107. 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

取り消されたアクティビティーに対するメッセージ

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度を確立します。

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 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 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、XCBMGTXT はブランクを含み、また XCBERRET は 2 進数ゼロを含みます。XCBERRET の値は、画面に表示されるメッセージを決定します。

- デフォルト・メッセージ「OK, command canceled」というメッセージを使用する場合は、XCBERRET の値をゼロのままにします。
- A governor exit cancel occurred with return code xxxxx というメッセージを使用する場合は、XCBERRET にゼロ以外の値を使用します。この値が、xxxxx の代わりにメッセージに表示されます。

QMF 初期化が管理プログラム出口により取り消された場合は、XCBMGTXT および XCBERRET のメッセージが、画面上ではなくユーザーのトレース・データ内に表示されます。

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についてトレース・データに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、デフォルト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 メッセージ定義はより長いメッセージ・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 を設定し、XCBMGTXT および XCBPANEL はブランクのままにしておきます。メッセージ定義でヘルプ・パネルが指定されなかった場合、ユーザーはメッセージ・ヘルプは受け取りません。

メッセージ定義内の長メッセージ仕様を上書きするには、新規のメッセージ・テキストを XCBMGTXT 内に入れます。パネル仕様を上書きするには、新規のパネル ID を XCBPANEL 内に入れます。パネル ID を XCBPANEL に入れると、メッセージ定義でパネルの指定が無くても、メッセージ・ヘルプの準備もします。

関連する ISPF メッセージ定義が存在しない場合は、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 XCBMGTXT に入れ、ヘルプ・パネル ID があれば XCBPANEL に入れます。この場合、XCBPANEL をブランクのままにしておく、ユーザーにはメッセージ・ヘルプは提供されません。

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。

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 U トレース・オプションに使用し、トレース出力を入手します。

デフォルト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で実行している場合、メッセージは画面に表示されません。

関連タスク:

454 ページの『トレース機能』

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。

関連情報:

 IBM Publications Center

ISPF LOG サービスに関する情報を検索します。

管理プログラム出口ルーチンの変換、アセンブル、リンク・エディット

デフォルト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。

TSO、ISPF、およびネイティブ z/OS バッチ用管理プログラム出口ルーチンのアセンブルおよびリンク・エディット

独自のルーチンを作成するか、デフォルト・ルーチンのコピーを変更する場合は、管理プログラム出口ルーチンをアセンブルし、リンク・エディットする必要があります。

手順

TSO、ISPF、およびネイティブ z/OS バッチの管理プログラム出口ルーチンをアセンブルしてリンク・エディットするには、以下の手順に従ってください。

1. 管理プログラム出口ルーチンをアセンブルします。

QMF が管理プログラムのコードでサポートしているのはアセンブラ言語のみです。デフォルト管理プログラムはアセンブラでコーディングされており、コードはハイレベル・アセンブラ用に作成されています。このコードは、QMF1210.SDSQSURE ライブラリーの特定のメンバーを印刷すると見ることができます。

2. 管理プログラム出口ルーチンをリンク・エディットします。

- a. 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。例えば、QMF のロード・モジュールを含む QMF1210.SDSQLOAD を使用できます。このライブラリーは、STEPLIB の連結の一部でも構いません。
- b. モジュール DSQUnGV1 の名前を指定します。ここで、*n* は、使用している各国語を示す 1 文字の ID です。この名前は、デフォルト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 QMF1210.SDSQLOAD ライブラリーに入れると、デフォルトのモジュールと置き換わります。

デフォルト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 QMF1210.SDSQLOAD よりも前にくるようにしてください。

QMF1210.SDSQLOAD より前にない場合、QMF は、ユーザー独自のモジュールではなくデフォルト・モジュールを呼び出します。

- c. この新規モジュールの入り口点は、必ず、DSQU n GV1 でなければなりません。ソース・コードが、DSQU n GV1 ラベルの CSECT ステートメントで始まっている場合は、ほかに何もすることはありません。ソース・コードが DSQU n GV1 ラベルで始まっていない場合は、アセンブラー・コードの END ステートメントにエン트리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트리名を付けてください。デフォルト管理プログラム (DSQUEGV1) は、以下に示すように 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)
```

タスクの結果

独自のルーチンは、31 ビット (以下を参照) 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(24)
  NAME DSQUEGV1(R)
```

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

CICS での管理プログラム出口ルーチンの変換、アセンブル、リンク・エディット

独自のルーチンを作成するか、IBM 提供のルーチンのコピーを変更する場合は、CICS で管理プログラム出口ルーチンをアセンブルし、リンク・エディットする必要があります。

手順

CICS の管理プログラム出口ルーチンをアセンブルしてリンク・エディットするには、以下の手順に従ってください。

1. アセンブラー用の CICS 変換プログラムを使用して、プログラムを変換します。

ユーザーのプログラムを変換する際、CICS はアドレス可能性を確立する標準の CICS プロログ (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。標準のプロログは、標準の CICS エピログ (DFHEIRET) も備えています。

2. プログラムをアセンブルします。

QMF で管理プログラム出口ルーチン用にサポートされている言語はアセンブラーのみです。デフォルト管理プログラムは、この言語でコーディングされていません。コードはハイレベル・アセンブラー用に作成されています。このコードは、QMF1210.SDSQUSRE ライブラリーの特定のメンバーを印刷すると見ることができます。

3. 管理プログラムをリンク・エディットします。

a. 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。例えば、QMF のロード・モジュールを含む QMF1210.SDSQLOAD を使用できます。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。

b. モジュール DSQU n GV3 の名前を指定します。ここで、 n は、使用している各国語を示す 1 文字の ID です。この名前は、デフォルト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 QMF1210.SDSQLOAD ライブラリーに入れると、デフォルトのモジュールと置き換わります。

デフォルト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 内の別のライブラリーに入れることができます。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 QMF1210.SDSQLOAD よりも前にくるようにしてください。QMF1210.SDSQLOAD より前にない場合、QMF は、ユーザー独自のモジュールではなくデフォルト・モジュールを呼び出します。

c. この管理プログラム・モジュールの入り口点は、必ず DSQU n GV3 でなければなりません。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タ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ー名を付けてください。

d. CICS コマンド・インターフェース制御モジュール 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、次に示すように、管理プログラムは AMODE(31) および RMODE(31) 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(31)
  NAME DSQUEGV3(R)
```

ユーザー・アクティビティの取り消し

ユーザー・アクティビティを取り消すには、QMF 取り消しサービスおよび QMF 割り込み機能を含め、複数のオプションがあります。

QMF 取り消しサービス

QMF には、QMF コマンドまたはデータベース・アクティビティの取り消しに使用できる取り消しサービスが用意されています。QMF 取り消しサービスが呼び出されると、QMF は、QMF コマンドまたはデータベース・アクティビティによって検査される内部取り消しスイッチをオンにします。QMF がアクティビティを取り消し可能である場合は、取り消しメッセージが表示されます。QMF アクティビティが完了しようとしている場合、取り消し要求は無視され、アクティビティは完了することができます。

アクティビティを取り消すために管理プログラムが QMF 取り消しサービスを呼び出すようにするには、DXEGOVA 制御ブロックの GOVCADDR フィールドに現れるアドレスに分岐します。図 108 に、QMF 取り消しサービスに対してアドレス可能性を設定するステートメントを示します。管理プログラムから制御が返されたときに QMF 取り消しサービスが管理プログラムの状態を復元できるように、レジスター 13 が管理プログラムの保存域を指していることを確認してください。

```
L R15,GOVCADDR
      BALR R14,R15
```

図 108. 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 が指すポイント (この場合は BALR コマンドに続くコマンド) に制御を返します。レジスター 15 には戻りコードが含まれています。図 109 に戻りコードを示します。

```
0 - QMF accepted the cancel request.
100 - QMF is not active.
```

図 109. 取り消しサービス戻りコード

以下の管理プログラム機能を、QMF 取り消しサービスを使用して取り消すことができます。

| | |
|-------------------|-------------------|
| 管理プログラム機能コード * | 実行されているアクティビティ |
| GOVSCMD (3) | QMF コマンドの開始 |
| GOVSDBAS (6) | データベース・アクティビティの開始 |

* 管理プログラム機能コードは DXEGOVA 制御ブロック内にあります。

非同期タイマー・ルーチン

タイマー・ルーチンが DXEGOVA 制御ブロックの GOVCADDR フィールドで指定された QMF 取り消しサービスのアドレスにアクセスできる場合、取り消しサービスはタイマー・ルーチンから呼び出すことができます。

重要: アイドルの QMF セッションを取り消すためにタイマー・ルーチンを使用することはできません。QMF がユーザー入力を待機している場合、タイマー・ルーチンは、ユーザーが画面に入力するまで、または QMF アテンション・ハンドラーが ATTN キーによってアクティブ化されるまで完了できません。

z/OS サービス

QMF 管理プログラム出口ルーチンは、ABEND などの z/OS サービスを使用して QMF セッションを終了することができます。

アイドル QMF セッションの取り消し

アイドルで、入力を待機している QMF セッションを取り消すには、SYS1.PARMLIB の z/OS システム管理機能パラメーター (SMFPRMxx) メンバーの Job Wait Time (JWT) オプションを使用できます。詳しくは、z/OS 情報で Time Limit Exit に関する情報を参照してください。

アイドル QMF セッションが保持するロックの解放

QMF セッションがアイドルである間にロックを保持しているデータベース・スレッドを取り消すには、DB2 for z/OS 情報でアイドル・アクティブ・スレッドのタイムアウトに関する情報を参照してください。

QMF 割り込み機能

QMF 割り込み機能は、実行に時間がかかりすぎているコマンドの取り消しを試行するために使用できます。詳細については、452 ページの『診断情報を取り込むための割り込みの作成』を参照してください。

DB2 リソース限定機能の使用

DB2 は、独自の管理プログラムであるリソース限定機能を備えています。DB2 リソース限定機能は、QMF for TSO または QMF High Performance Optionのいずれかの管理機能とともに使用できます。DB2 管理プログラムは独立して使用することも可能です。

DB2 リソース限定機能を使用すると、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。また、この機能を使用して、動的に実行される SELECT、INSERT、UPDATE、および DELETE の各ステートメントのために費やされるプロセッサ時間をモニターできます。

関連情報:

 [DB2 資料](#)

DB2 リソース限定機能を使用したリソース使用効率の向上に関する情報を検索します。

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。ただし、DB2 管理プログラムのモニター対象とモニター方法は、QMF 管理プログラムとは異なります。

QMF および DB2 管理プログラムは以下の点で異なります。

- DB2 管理プログラムのモニターは、動的な SELECT、INSERT、UPDATE、および DELETE ステートメントに限定されます。例えば、CREATE または DROP ステートメント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、行のフェッチ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、例えば、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、行のフェッチから、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、行のフェッチを行っても、フェッチ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を提供しません。当該 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。

DB2 管理プログラムでモニター可能な QMF コマンド

DB2 管理プログラムを使用して QMF コマンドをモニターすることができます。

下の表の各行は、DB2 管理プログラムで個別にモニター可能な QMF コマンドのグループを表しています。表の 3 列目にある各パッケージの説明を読んで、ユーザーの QMF 構成に応じて管理する最適なパッケージを判断してください。

説明で DB2 for z/OS 以外のシステムに言及しているパッケージ名は、QMF CONNECT コマンド、または 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーでの SQL ステートメントの管理に使用されます。DB2 リソース限定機能を使用してリモート・サーバーで SQL ステートメントを管理するには、追加の構成手順で管理をセットアップする必要があります。

表 70. DB2 管理プログラムで個別にモニター可能な QMF コマンドのグループ

| QMF コマンド | 管理されるステートメントのタイプ | 管理する QMF パッケージ名 |
|--|----------------------------|---|
| <ul style="list-style-type: none"> RUN QUERY (QBE P. 照会、指示照会、および SQL SELECT 照会) DISPLAY TABLE (CONFIRM=YES) EXPORT TABLE PRINT TABLE 報告書をスクロールするコマンド (BOTTOM、TOP、FORWARD、BACKWARD、RIGHT、および LEFT) | SELECT | <ul style="list-style-type: none"> DSQJFSUS - 単一行フェッチ操作を制御する。MR プログラム・パラメーターが NO に設定された状態で DB2 for z/OS データベース、DB2 for iSeries データベース、または DB2 for Linux, UNIX, and Windows データベースで QMF が開始された場合に適用されます。 DSQJFSUM - 複数行フェッチ操作を制御する。QMF が、MR プログラム・パラメーターを YES に設定されて DB2 for z/OS データベースで開始された場合に適用されます。 DSQJFSUV - DB2 (VSE および VM 版) データベースからの行のフェッチを制御する |
| <ul style="list-style-type: none"> SAVE DATA IMPORT TABLE | INSERT | <ul style="list-style-type: none"> DSQJSDT7 - 単一行挿入操作を制御する。QMF が、MR プログラム・パラメーターを NO に設定されて DB2 for z/OS、DB2 for iSeries、または DB2 for Linux, UNIX, and Windows データベースで開始された場合に適用されます。 単一行挿入は、XML データまたは LOB データを含む表を保存するために SAVE DATA コマンドが発行された場合にも使用されます。 DSQJSDT8 - 複数行挿入操作を制御する。QMF が、MR プログラム・パラメーターを YES に設定されて DB2 for z/OS データベースで開始された場合に適用されます。 |
| RUN QUERY (QBE I、U、または D. 照会、または INSERT、UPDATE、または DELETE ステートメントを含む SQL 照会) | INSERT UPDATE DELETE | <ul style="list-style-type: none"> DSQJESQL - DB2 (VSE および VM 版) データベース向け DSQJESQM - その他のすべてのデータベース・タイプ向け |
| EDIT TABLE (MODE=ADD) | INSERT | DSQJTSQL |
| EDIT TABLE (MODE=CHANGE, SAVE=IMMEDIATE) | UPDATE DELETE | DSQJHSQL |
| EDIT TABLE (MODE=CHANGE, SAVE=END) | UPDATE DELETE | DSQJNSQL (DB2 (VSE および VM 版) のみ) |

関連概念:

403 ページの『QMF コマンドの DB2 管理の構成』

特定のリソース限定表 (RLST) に行を追加することで、ローカル DB2 for z/OS データベースに送信される QMF コマンドの管理を構成します。

211 ページの『複数行のフェッチと挿入のサポートの使用可能化』

DSQSMRFI パラメーター。データベースがフェッチおよび挿入の際に処理する行が複数行か、単一行かを制御します。

関連情報:

DB2 資料

DB2 リソース限定機能を使用したリモート・サーバーでの SQL ステートメントの管理に関する情報を検索します。

DB2 管理プログラムにより照会が取り消された場合の QMF の対応方法

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て、SQL エラー・コード -905 を戻し、照会が取り消されたことを QMF に知らせます。

QMF が取り消しをどのように処理するかは、DB2 管理プログラムが QMF セッションのどの時点で照会を取り消したかによって異なります。

QMF の初期化中

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 はトレース・データに説明のレコードを書き込みます。

例えば、QMF ユーザーの特定の時間内の QMF の使用を制限したいとします。この時間内に、制限対象のユーザーに最大プロセッサ時間ゼロを割り当てて、QMF ホーム・パネルが表示される前に QMF セッションが取り消されるようにできます。

QMF の初期化後

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。例えば、DB2 管理プログラムが INSERT ステートメントを取り消すとします。INSERT ステートメントの結果として変更されたデータベースはロールバックされ、QMF はエラー・メッセージを発行します。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、行フェッチが取り消しにつながる可能性があります。取り消しが生じると、データ・オブジェクトは未完了のままになります。DB2 がカーソルをクローズしてしまうため、DATA オブジェクトが完了できず、RESET DATA コマンドが発行される必要があります。

QMF コマンドの DB2 管理の構成

特定のリソース限定表 (RLST) に行を追加することで、ローカル DB2 for z/OS データベースに送信される QMF コマンドの管理を構成します。

QMF がインストールされている DB2 for z/OS サブシステムは始動時に、特定の RLST に関連付けられます。RLST は、RLST が関連付けられているサブシステム内のすべてのユーザー (QMF ユーザーを含む) についての入力を DB2 管理プログラムに提供します。

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。例えば、特定の期間用の 1 つの RLST で、その期間中にユーザーが QMF セッションを開始できないようにできます。この場合、QMF を開始しようとして

も、QMF の初期化中にセッションは終了してしまい、トレース・データにメッセージが現れます。別の RLST を使用してサイト固有の規則や制限を処理したり、個々のユーザーまたはユーザー・グループの管理を行ったりすることができます。例えば、数人の個人ユーザー用の行、およびその他のすべての人に適用される行を追加できます。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。

ローカルの DB2 for z/OS データベースに送信される QMF コマンドの管理を構成するには、管理対象にする 401 ページの『DB2 管理プログラムでモニター可能な QMF コマンド』内の各パッケージについて 1 行を DB2 RLST に追加します。RLST 内で更新する列は、QMF コマンドを予測的に管理するか、または対応的に管理するかによって異なります。パッケージについて予測型管理および対応型管理の両方が必要な場合は、QMF パッケージ名に複数行を関連付けることができます。

DB2 リソース限定機能を使用してリモート・サーバーで SQL ステートメントを管理するには、追加の構成手順が必要です。

予測型管理のための RLST の更新方法

予測型管理を行うには、RLST 表の以下の列を更新します。

表 71. QMF コマンドを予測的に管理するために DB2 for z/OS RLST 表に追加する値

| RLST 表の列 | 値 |
|----------------|--|
| RLFPKG | 管理対象の QMF コマンドに関連付けられたパッケージの名前 パッケージ名は 401 ページの『DB2 管理プログラムでモニター可能な QMF コマンド』に記載されています。 |
| RLFFUNC | 7 |
| RLFASUERR | ユーザー定義 |
| RLFASUWARN | ユーザー定義 |
| RLF_CATEGORY_B | ユーザー定義 |

対応型管理のための RLST の更新方法

対応型管理を行うには、RLST 表の以下の列を更新します。

表 72. QMF コマンドを対応的に管理するために DB2 for z/OS RLST 表に追加する値

| RLST 列 | 値 |
|---------|---|
| RLFPKG | 管理対象の QMF コマンドに関連付けられたパッケージの名前。 パッケージ名は 401 ページの『DB2 管理プログラムでモニター可能な QMF コマンド』に記載されています。 |
| RLFFUNC | 2 |

表 72. QMF コマンドを対応的に管理するために DB2 for z/OS RLST 表に追加する値 (続き)

| RLST 列 | 値 |
|---------|--|
| ASUTIME | RLFPKG 内のパッケージに関連付けられた QMF コマンドの完了に割り振られる最大プロセッサ時間 |

重要: RLST 表の RLFCOLLN 列は更新しないでください。QMF コレクション ID (Q) により QMF 機能を管理しようとする、特定の状況で QMF の内部処理が停止するか、QMF が適切に機能しなくなる可能性があります。

関連情報:

 [DB2 資料](#)

DB2 リソースを使用したリモート・サーバーでの SQL ステートメントの管理に関する情報、RLST の作成または更新に関する情報、および予測型管理と対応型管理に関する情報を検索します。

第 18 章 バッチ・モードでの 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。したがって、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。

代わりに、管理者およびユーザーは、プロシージャーおよびアプリケーションをバッチ・モードで作成、実行できます。この場合、プロシージャーおよびアプリケーションはユーザーのセッションから独立して実行されるため、ユーザーは引き続きコマンドを発行することができるようになります。

QMF を非対話型で実行する必要があるため、QMF for TSO を使用している場合は、QMF を DB2 for z/OS ストアード・プロシージャーとして始動することを考慮する必要があります。このインターフェースを使用すると、QMF for Workstation などの DB2 for z/OS ストアード・プロシージャーを呼び出すことができるプログラムであればどれも、QMF for TSO をストアード・プロシージャーとして始動し、出力を結果セットで受け取ることができます。また、ストアード・プロシージャーを実行して、リモート DB2 クライアントからバッチ・ジョブを開始することもできます。

NLF を使用している場合、複数言語サイトのユーザーは、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。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

z/OS での QMF のバッチ・プログラムとしての実行

QMF バッチ・モードは、TSO、ISPF、およびネイティブ z/OS で使用できます。ISPF および TSO の場合、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。

バッチ・モードで作業するには、適切な権限が必要です。また、セキュリティー上の影響、デバッグ・プロシージャー、およびバッチ・ジョブの送信方法を理解する必要があります。

バッチ・モードで操作する権限

バッチ・ジョブを実行するには、特定の QMF および DB2 権限が必要です。

これらの権限を、必要としているユーザーに付与するには、最初に、ジョブの実行で使用するログオン ID と DB2 1 次許可 ID を調べます。

- サイトで RACF を使用している場合、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。
- サイトで RACF を使用していない場合、ログオン ID および 1 次許可 ID は、バッチ・ジョブを実行する JCL で決定されます。

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、プロシーチ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。

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、その権限をバッチ・モードで使用することもできます。

- ログオン ID に対応する許可 ID が実行プロシーチャーを持っている、もしくはそのプロシーチャーを共有している。

プロシーチ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。

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る場合は、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。

QMF バッチ・ジョブを実行するための JCL

管理者またはユーザーは、ステートメントを実行する JCL ファイルを作成する必要があります。

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。

- 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
```

示されているステートメントは、アカウント情報もユーザーの名前も含んでいないため、すべてのサイトに適しているわけではありません。ここで示しているオペランドでは、次の指定を行っています。

- ログオン ID は LMN です。
- ログオン・パスワードは ABC です。
- 端末メッセージがジョブ終了時にユーザー LMN に送信されます。

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、その他のオペランドを指定できます。

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

ジョブ・ステップで EXEC ステートメントを使用すれば、次のように、バッチ・モードで 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 つ以上の SYSOUT DD ステートメントにオペランド HOLD=YES を追加すれば、その出力は 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、以下のように 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・セットに対して、次に示すような DD ステートメントを指定します。

```
//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 に含めるには、次に示すような DD ステートメントを指定します。

```
//SYSTSIN DD *  
EXEC CLISTA  
PROFILE PREFIX(LMN)  
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSMODE=B,DSQSRUN=LMN.PROCA)  
:  
/*
```

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うことができます。
- 2 番目のステートメントは、ユーザーの DS 名接頭部に LMN を設定します。

- 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 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。

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 ステートメントは、サイトのセットアップによっては、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 名接頭部をリストア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 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 QMF に対するユーザー識別

RACF を使用していないサイトの場合、QMF は、ユーザーのログオン ID がユーザーの DS 名接頭部と等しいと見なします。この接頭部が NULL の場合、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 のインストール・パラメーター 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。

- 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。例えば、ログオン ID LMN のもとで実行するプロシージャが、次の 2 つのコマンドを発行するとします。

```
EXPORT QUERY TO 'LMN.QUERYX.QUERY'  
IMPORT QUERY FROM QUERYX
```

EXPORT コマンドは、ログオン ID (LMN) を、エクスポートされたオブジェクトを含むデータ・セットの名前の最初の修飾子に使用します。次に、IMPORT コマンドがこのデータ・セットの内容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。

これらの場合、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。

関連概念:

187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すことができるプログラム・パラメーターについて説明します。

197 ページの『TSO で QMF プロファイル・キーとして使用する ID の指定』
QMF for TSO の各ユーザーに固有のプロファイルを作成することを選択する場合は、表の CREATOR 列の値はすべて TSO ログオン ID にするか、または DB2 1 次許可 ID にすることができます。QMF でユーザーの TSO ログオン ID を使用してユーザーを認証することを選択する場合は、QMF の開始時にこの選択を指定する必要があります。

QMF バッチのネイティブ z/OS での実行

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。

以下の例のような JCL を使用できます。

```

/*****/ 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=QMF1210.SDSQLOAD,DISP=SHR 00305000
// DD DSN=DSN1110.SDSNEXIT,DISP=SHR 00306000
// DD DSN=DSN1110.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=QMF1210.SDSQMAPE,DISP=SHR 00312000
//* 00313000
//* 00314000
//* Data sets used by QMF 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 110. 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、次のことを覚えておいてください。

- 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 での QMF の実行時には作動しません。
- デフォルトのユーザー ID 接尾部は使用不可。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 プロシージャ) は使用できない。TSO 以外のアドレス・スペースで、REXX を使用して QMF を実行する場合は、413 ページの図 111に示したように、IRXJCL を使用する必要がある。

ここに示す REXX プログラムは、QMF 呼び出し可能インターフェースを使用して、QMF を開始し、バッチ・モードで QMF コマンドを実行します。

```

//QMF BATCH JOB REGION=8M,
// MSGCLASS=H, TIME=(2,30), USER=&SYSUID, NOTIFY=&SYSUID, CLASS=A
//ROBQMF1 EXEC PGM=IRXJCL
//STEPLIB DD DSN=DSN1110.DB2A.SDSNLOAD, DISP=SHR
//          DD DSN=DSN1110.DB2A.SDSNEXIT, DISP=SHR
//          DD DSN=QMFDEV.QMF1210.SDSQLOAD, DISP=SHR
//ADMGGMAP DD DSN=QMFDEV.QMF1210.SDSQMAPE, DISP=SHR
//SYSEXEC DD DSN=ROBIN.QMF1210.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,(10,20),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
CALL DSQCIX "EXIT"
SAY DSQ_MESSAGE_ID DSQ_MESSAGE_TEXT
EXIT DSQ_RETURN_CODE
/*

```

図 111. バッチ・モードで QMF を始動および実行する REXX プログラム

これらの JCL サンプルは、予備ファイルを割り振ります。予備ファイルのファイルを割り振る代わりに、DSQSPTYP パラメーターを使用して、データを拡張記憶域に予備書き込みすることもできます。

関連タスク:

201 ページの『拡張仮想記憶域への報告書データの書き込み (TSO のみ)』

QMF for TSO では、QMF が実行されているシステムで利用可能な拡張記憶域が非常に限られている場合を除き、予備データ用の拡張記憶域が使用されます。

TSO での QMF バッチの実行

TSO SUBMIT コマンドを使用して、バッチ・ジョブを z/OS に送信します。

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。

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 コマンドを発行するには、適切な TSO 権限が必要です。その権限を付与するのは、TSO 管理タスクです。

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 セッション中は、QMF の TSO コマンドを使用する
- TSO READY モードの場合、またはジョブの JCL をカスタマイズする CLIST 内の場合

カスタマイズは、呼び出される CLIST に値が渡されるパラメーターに基づいて行うことができます。

以下に、プロシージャの実行中に発生する可能性があるエラーを示します。

- プロシージャの終了
- 非コミットの DB2 リカバリー単位のバックアウト

ジョブの JOB ステートメントで、ジョブの終了時にユーザーにメッセージを送信するように指定できます。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、ユーザーはジョブ出力にエラーがないか調べることができます。適切な JCL を使用すれば、この出力は、ユーザーが印刷したり、エディターで調べることができるデータ・セットに経路指定することができます。これらのデータ・セットの 1 つに確認メッセージおよびエラー・メッセージのレコードを保管でき、また必要に応じて、実行した QMF コマンドのレコードを保管できます。

QMF BATCH コマンドを使用した ISPF での QMF バッチの実行

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 を使用している場合、バッチ・アプリケーション用に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。

関連タスク:

299 ページの『コマンド同義語のカスタマイズ』

デフォルトのコマンド同義語が要件に合わない場合は、独自の同義語を作成することができます。コマンド同義語が作成された後は、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。

ISPF でバッチ・アプリケーションを使用する権限

バッチ照会/プロシージャ・アプリケーションは、ユーザーのバッチ・ジョブ用のプロシージャおよび 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、ジョブを実行依頼できません。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・プロシージャー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、プロシージャー、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。
- 照会またはプロシージャーの結果を新しい表に保存する場合は、ユーザーによる表の作成を可能にする必要があります。

関連概念:

253 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』サイトの要件によって、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。

ISPF でのバッチ・アプリケーションの使用

バッチ照会/プロシージャー・アプリケーションは、共用プロシージャーを実行することによって開始するため、QMF ユーザーは誰でも使用できます。

ユーザーはバッチ照会/プロシージャー・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ーを持っている必要があります、さらに必要に応じて、報告書をフォーマット設定するための書式も持っていなければなりません。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。

ユーザーが該当するフィールドに入力し、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。

- ヘルプファンクション・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。
- 終了ファンクション・キーを押して、アプリケーションを終了する。

(ファンクション・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 を使用している場合、BATCH に対して変換されたコマンド同義語を発行して、照会またはプロシージャーをバッチ・モードで実行します。例えば、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 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。ここで n は、1 文字の言語 ID です。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

ISPF でのバッチ・アプリケーションの開始

バッチ・アプリケーションを使用する利点は BATCH コマンド・プロンプト・パネルにあります。ユーザーが JCL または QMF プロシージャに関する知識を持っている必要がないためです。ユーザーはパネルでジョブが何をすべきかの概要を指示するだけで、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。

このタスクについて

バッチ照会/プロシージャ・アプリケーションは、ユーザーが QMF 下で作業中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・ジョブが SELECT 照会であると想定した場合、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 DATA オブジェクトの保存
- 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

手順

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

タスクの結果

このコマンドによって、以下に示す QMF BATCH コマンド・プロンプト・パネルが表示されます。


```

                                QMF Batch Command Prompt
Enter name of Query or Procedure ==>
Use Object from work area?      ==> NO
Is Object QUERY or PROC        ==> QUERY
PROC arguments ==>
Enter Name of FORM to use      ==>
Use FORM from work area?      ==> NO
Enter Name of QMF Batch PROC ==>
Enter Name of Data to Save    ==>
Enter Name of Report Dataset ==>
Is Dataset New?               ==>
Enter Optional Volser         ==>
Enter Report Width            ==> 133
Should Report be Printed?     ==> YES
Enter Output Class            ==> A
Log Messages and Commands?    ==> YES
Enter Userids and Nodes to Send Report:
  Userid(1) ==>                Node ==>
  Userid(2) ==>                Node ==>
Enter Database SUBSYSTEM ==>      Enter Database PLANID ==>
Enter TSO Logon Password ==>      ( Press ENTER Key to Process Request )
Please enter a name for the batch ITEM.

```

図 112. QMF BATCH コマンド・プロンプト・パネル

バッチ・コマンド・プロンプト・パネルのフィールド

バッチ・プロンプト・パネルのフィールドは、バッチ・ジョブで何を実行するかを記述します。

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。QMFは、これらの必須フィールドについて、必要に応じて、プロンプトを表示して値を要求します。下の表に、必須フィールドについての説明があります。

表 73. QMF BATCH アプリケーションに必要な入力フィールド

| フィールド・テキスト | 説明 |
|----------------------------------|--|
| Enter name of Query or Procedure | 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、この名前で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。 |
| Is Object QUERY or PROC | バッチで実行するオブジェクトのタイプ。QUERY または PROC のどちらかでなければならない。 |

表 73. QMF BATCH アプリケーションに必要な入力フィールド (続き)

| フィールド・テキスト | 説明 |
|------------------------------|---|
| Enter Name of QMF Batch PROC | 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、各照会ごとに Enter name of Query or Procedure 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 コマンドが含まれる。このプロシージャからは、Is Object QUERY or PROC フィールドで指定された照会またはプロシージャが実行される。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

下の表に、パネル上の残りのオプションの入力フィールドについての説明があります。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。YES/NO フィールドの値を削除すると、QMF から入力を要求するプロンプトが表示されます。

表 74. QMF 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・テキスト | 説明 |
|----------------------------|---|
| Use Object from work area? | 現在一時記憶域にバッチ照会またはプロシージャが存在する場合は、このフィールドに 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。 |
| PROC arguments | Enter name of Query or Procedure 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数。 |
| Enter Name of FORM to use | 書式を使用してバッチ照会を実行するには、このフィールドに書式の名前を指定する必要がある。使用する書式によって、 <ul style="list-style-type: none"> デフォルトの書式の場合、フィールドを空のままにしておく。 データベースにある場合、書式はこの名前でも保管される。書式が誰か他の人に所有され、共用されている場合は、名前を修飾する必要がある。 現行の書式の場合、保管できる名前を指定する。書式はそれぞれの許可 ID で保管されるため、名前は修飾しないで指定する必要がある。 <p>この書式は、プロファイルの設定として CONFIRM=NO を用いて保管される。</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> |

表 74. QMF 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・テキスト | 説明 |
|------------------------------|---|
| Use FORM from work area? | 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保管される。書式がデータベースにある場合は、NO を入力する。このフィールドのデフォルト値は NO。 |
| Enter Database SUBSYSTEM | QMF が使用する DB2 サブシステムの名前を入力する。このフィールドの値は、プログラム・パラメーター DSQSSUBS の値と同じ。 |
| Enter Database PLANID | QMF アプリケーション・プランの名前を入力する。このフィールドの値は、プログラム・パラメーター DSQSPLAN の値と同じ。デフォルトは QMF12 です。 |
| Enter TSO Logon Password | ログオン・パスワードを入力する。これは画面には表示されない。 |
| Log Messages and Commands? | 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 レベルでのトレースを希望しない場合は、NO を指定する。バッチ・プロシージャ内に生成される SET PROFILE (TRACE=NONE コマンドを超えたところから、トレースは続行されなくなる。 |
| Enter Name of Data to Save | 照会またはプロシージャを実行した結果生成されたデータを保存する場合は、このフィールドに値を指定する必要がある。DATA オブジェクトは、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。 |
| Enter Name of 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 JCL ステートメントによって z/OS 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、@、#、または \$ 以外の特殊文字が入っていなければ、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。 |
| Is Dataset New? | 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。 |
| Enter Optional Volser | オプション: Is Dataset New? に YES を入力した場合、このフィールドに値を入力する。します。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。 |

表 74. QMF 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・テキスト | 説明 |
|---|--|
| Enter Report Width | 「Is Dataset New?」フィールドに YES と入力した場合は、このフィールドへの入力が必要。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、デフォルト値 133 を使用する。 |
| Should Report Be Printed? | このフィールドには YES または NO を指定する必要がある。YES はジョブを印刷することを示し、NO はジョブを印刷しないことを示す。 |
| Enter Output Class | ジョブからの印刷出力の出力クラスを入力。印刷出力には、以下のものが含まれる。 <ul style="list-style-type: none"> システム・メッセージ 印刷する場合、報告書 (DSQPRINT) DSQDEBUG トレース・データ・セット 生成された場合、異常終了ダンプ (DSQDUMP) インストール先システムでその準備がされている場合は、ディスプレイ装置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できる。 |
| Enter Userids and Nodes to Send Report: | 結果の報告書を他のユーザーに送信する場合、これらのフィールドにユーザー ID とノードを入力する。このフィールドを使用する場合、報告書出力用のデータ・セットの名前を Enter Name of Report Dataset フィールドに指定する必要がある。 <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。その情報が NAMES.TEXTLIST データ・セットにある場合は、ユーザーの「Node」フィールドへの入力は不要。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。

- QMF1210.SDSQCLTE ライブラリー内の CLIST 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。

- QMF1210.SDSQMLBE ライブラリーのメンバー DSQBE00、DSQBE01、および DSQBE02 内の ISPF メッセージ定義

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。

- 次のような各種の目的を持つ、QMF1210.SDSQPLBE ライブラリー内の各種の ISPF パネル定義。
 - 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。
- QMF1210.SDSQSLBE ライブラリー内の特定のファイルのカスタマイズ例。
 - DSQABB1J はバッチ・ジョブのサンプル JCL を提供します。これは、バッチ・モードで照会を実行するサンプル・プロシージャです。
 - DSQABB1P と DSQABB1S は、サンプルの QMF プロシージャです。これらは、ジョブに対して JCL をサブミットするプロシージャの例を提供します。

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。

DSQABB1J サンプル・ファイル内には、バッチ・モードで 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、デフォルト名 DSN を持っているものと見なされます。QMF を DB2 サブシステム内で異なる名前で行う場合は、ISPSTART コマンドの PARM オペランドに DSQSSUBS=ssid を追加します (ここで、ssid 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。
- 追加ロジックを追加して、サイトの規則を強制する。

例えば、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。

- アプリケーションで生成された JCL を変更して、サイトのニーズに合わせる。

次のようなことが可能です。

- JOB ステートメントへのアカウント情報の追加。
- SYSTSIN データ・セットの ISPSTART ステートメントにある QMF アプリケーション・プランの名前の変更。バージョン 12.1 のデフォルトの QMF プラン名は QMF12 です。

次のような、追加の変更も必要です。

- 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が `userid.DSQ1EBFT.PROC` という名前のデータ・セットを保守することはできません (ここで、`userid` は、ユーザーの TSO ログオン ID)。このようなデータ・セットが存在すると、QMF バッチ・アプリケーションが正しく動作しないことがあります。

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。

この例では、すべてのユーザーが同じ PROFILE PREFIX ステートメントを持つようにバッチ・アプリケーションを変更し、すべてのユーザーが固有のユーザー ID を持つと仮定します。&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。

以下に示すように、DSQABB1S SKELETON ファイルに 3 つの変更を行う必要があります。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 z/OS 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 113. DSQABB1S SKELETON ファイルの変更

以下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
:
:
/*****/ 00244000
/* IMPORT AND RUN FILE TAILORED SKELETON */ 00245000
/*****/ 00246000
ISPEXEC SELECT PGM(DSQCCI) + 00247000
/* PARM( &IMPORT &PROC &FROM DSQ1EBFT (&MEMBER = DSQABB1S ) 00248000
PARM( &IMPORT &PROC &FROM &SYSUID..DSQ1EBFT (&MEMBER = DSQABB1S ) 00248000
IF &LASTCC ^= 0 THEN + 00249000
:
:
CLEANUP: FREE FILE(ISPFILE) DELETE 00274000
DONE: SET &ZPLACE = &SAVEPLC 00275000
SET &ZPFCTL = &SAVEPFC 00276000
SET &ZPF01 = &STR(&SAVEPF01) 00277000
SET &ZPF13 = &STR(&SAVEPF13) 00278000
SET &ZPF03 = &STR(&SAVEPF03) 00279000
SET &ZPF15 = &STR(&SAVEPF15) 00280000
SET &ZPF10 = &STR(&SAVEPF10) 00281000
SET &ZPF22 = &STR(&SAVEPF22) 00282000
SET &ZPF11 = &STR(&SAVEPF11) 00283000
SET &ZPF23 = &STR(&SAVEPF23) 00284000
ISPEXEC VPUT (ZPLACE ZPFCTL ZPF01 ZPF13) PROFILE 00285000
ISPEXEC VPUT (ZPF03 ZPF15 ZPF10 ZPF22 ZPF11 ZPF23) PROFILE 00286000
/* DELETE DSQABB.&PROC 00287000
DELETE &SYSUID..DSQABB.&PROC 00287000
EXIT CODE(&RCDE) 00288000

```

図 114. DSQABB12 CLIST の変更

ISPF または TSO のフォアグラウンドでの QMF バッチ・ジョブの開始

QMF をフォアグラウンドのバッチ・モードで開始するには、QMF の通常の開始方法をすべて使用できます。

例えば、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
```

このステートメントで、*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=auth_ID.proc_name)
```

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 が終了すると、TSO READY モードに戻ります。TSO コンソール、ログ、およびリスト・ファイルに対して、永続後処理パラメーターを指定する前にプロシージャが終了すると、CLIST の終了前に、ISPF 後処理プロンプト・パネルが表示される場合があります。このパネルが表示されないようにするには、これらのファイルに対して永続後処理パラメーターを指定しておきます。それぞれに対して、D (「削除」の指定) の値を指定すればたいの場合、十分です。この後処理の指定方法が分からない場合は、ISPF の専門家に尋ねるか、ISPF ヘルプを使用してください。

関連概念:

157 ページの『第 9 章 QMF の開始』

QMF は、z/OS からのみ開始できます。QMF は、TSO、ISPF の下にバッチ・ジョブとして、あるいは CICS の下にセットアップすることができます。

バッチ・モードのプロシージャまたはアプリケーションのデバッグ

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。

メッセージとコマンドを最高詳細レベルでトレースする L2 トレースは、バッチ・モードで実行されるプロシージャのデフォルトです。トレース設定を変更するには、プロシージャに SET PROFILE コマンドを追加します。例えば、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 または L2 トレースのいずれの場合も、ログはトレース・データ・セット (デフォルトで DSQDEBUG) 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・メッセージがトレース・データに書き込まれます。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。

リモート DB2 クライアントからの QMF バッチ・ジョブの開始

QMF ストアード・プロシージャ DSQQMFSP がインストールされた DB2 データベースに接続されているすべての DB2 クライアントで、TSO バッチ・ジョブ用に QMF を開始することができます。バッチ・ジョブは、QMF ストアード・プロシージャを実行する z/OS システムから開始されます。

始める前に

リモート DB2 クライアントからバッチ・ジョブを開始する前に、QMF ストアード・プロシージャ・インターフェースおよびバッチモードの権限を設定する必要があります。また、始める前に、以下の情報を参照してください。

- 143 ページの『QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ)』
- 407 ページの『バッチ・モードで操作する権限』
- 408 ページの『QMF バッチ・ジョブを実行するための JCL』
- 424 ページの『バッチ・モードのプロシージャまたはアプリケーションのデバッグ』

手順

リモート・クライアントから QMF バッチ・ジョブを開始するには、以下のようになります。

ストアード・プロシージャを呼び出す SQL CALL ステートメントを発行して、z/OS バッチ・ジョブを開始します。CALL ステートメントの最初のパラメーターとして、QMF ストアード・プロシージャによって実行される QMF プロシージャの名前を指定します。

次に例を示します。

```
CALL Q.DSQQMFSP('W397754.LAUNCH_STAFF_REPORT','L2',' ',' ','')
```

例

次の QMF REXX プロシージャ例 (W397754.LAUNCH_STAFF_REPORT) では、スタッフ報告書バッチ・ジョブが開始されます。

```
/* REXX - Launch Staff Report Batch Job. */
/* This procedure uses the REXX FtpApi to launch the batch */
/* job. */
```

```

/*****/
/* Initialize Rexx program variables */
hostname = "STLMVS1" /* z/OS host name to receive the batch job */
uid      = "W397754" /* z/OS user id that is used to login to ftp */
uidpw    = "xxxxxxx" /* user id pw used to login to ftp */
pcode = 0 /* Initialize return code */
TRACEID = 'PAZ' /* Set FTP trace flag */

/* Create REXX FTP environment */
ftp_rc = ftpapi('fcai.', 'create', TRACEID)
if ftp_rc < 0 then
do
pcode = -10
exit pcode
end

/* Connect to host. See hostname variable for name of host */
ftp_rc = FtpApi('fcai.', 'init', '-w 300 '||hostname)
if ftp_rc < 0 then exit -21

/* Logon user to FTP session */
cmd = "USER " uid
ftp_rc = FtpApi('fcai.', 'scmd', cmd, 'W')
if ftp_rc < 0 then exit -22

cmd = "PASS " uidpw
ftp_rc = FtpApi('fcai.', 'scmd', cmd, 'W')
if ftp_rc < 0 then exit -23

/* Set JES to receive JCL file */
cmd = "QUOTE SITE FILETYPE=JES"
ftp_rc = FtpApi('fcai.', 'scmd', cmd, 'W')
if ftp_rc < 0 then exit -24

/* Send JCL file to JES for execution */
cmd = "PUT 'W397754.STAFF.REPORT.JCL'"
ftp_rc = FtpApi('fcai.', 'scmd', cmd, 'W')
if ftp_rc < 0 then exit -25

/* Quit the FTP session */
cmd = "QUIT"
ftp_rc = FtpApi('fcai.', 'scmd', cmd, 'W')
if ftp_rc < 0 then exit -26

/* Terminate te REXX FTP Environment */
ftp_rc = ftpapi('fcai.', 'term')
if ftp_rc < 0 then exit -27

exit pcode

```

QMF の CICS でのバッチ・トランザクションとしての実行

QMF バッチ・トランザクションは、端末から実行することも、端末を使用せずに実行するトランザクションとして実行することもできます。

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 プロシージャを実行してください。このプロシージャは非対話式に実行できます。

CICS では、QMF は会話型トランザクションとして対話式に実行されます。

CICS での端末からのバッチ・アクティビティの実行

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。

例えば、ここに示したプロシージャを実行して、CICS 補助記憶域に報告書を作成することができます。(QMF は、QMF 線形手順内の「--」で始まる行を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 (QUEUE=STATRPT1,QUEUETYPE=TS)
--
-- End of procedure
```

図 115. CICS 補助記憶域に報告書を作成

このプロシージャをバッチ・モードで実行するには、ここで説明した QMF トランザクションを実行します。

```
QMFE M=B,I=STATRPT1_PROC
```

QMF は、画面には何も表示せずにこのトランザクションを実行します。プロシージャが正常終了すると、報告書が CICS 記憶域キュー STATRPT1 に作成されます。この後、CICS に付属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
```

CICS での端末を使用しないバッチ・アクティビティの実行

バッチ・ジョブを実行している端末がロックされているときにプロシージャを実行するには、EXEC CICS START コマンドを使用します。

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。次の例では、QMF プロシージャ STATRPT1_PROC を、端末を使用せずにバッチ・モードで実行します。

```
EXEC CICS START TRANSID(QMFE) FROM(M=B,I=STATRPT1_PROC)
```

このトランザクションが完了すると、CICS に付属のトランザクション CEBR を使用して CICS 記憶域キュー STATRPT1 を表示することができます。

CICS でのプロシージャのデバッグ

QMF には、メッセージ、コマンド、QMF 関数をトレースする機能があります。この機能は、QMF プロシージャをバッチ・モードで実行すると問題が生じるような場合に役立ちます。

QMF がバッチ・モードで実行している場合、メッセージとコマンドは自動的にトレースされます。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。例えば、STATRPT1_PROC プロシージャ

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=B,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。QMF は、トレース項目を書き込んでいるキュー名に対して、CICS ENQ コマンドを発行します。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 トレース出力を CICS 補助記憶域に経路指定する必要がある場合は、フル・コンポーネント・レベルは設定しないでください。一時記憶域が短時間で満杯になる可能性があるからです。メッセージ・レベル・トレース以外のトレースの場合、トレース出力を保持するための一時データ・キューを使用してください。

終了戻りコード

QMF の終了戻りコードは、以下のとおりです。

- 0 正常終了
- 8 異常終了

関連概念:

457 ページの『トレース出力の適切な詳細レベルの設定』
すべての QMF 機能を詳細にトレースすることができ、また個々の QMF 機能をトレースすることができます。

第 19 章 問題のトラブルシューティングと診断

QMF の使用中にユーザーに発生する問題を切り分けて解決するために、トラブルシューティング情報およびサポート情報を使用することができます。

QMF サービスの適用

QMF に問題が発生した場合は常に、まず QMF インストール済み環境のサービス・レベルが最新であることを確認してください。

サービスの現行レベルを判別し、IBM ソフトウェア・サポート Web サイトで、QMF とその前提条件製品の最新 PTF を検索します。また、QMF の予防保守計画 (PSP) バケットを要求してください。PSP バケットには、既知の問題に関連する一般ヒント、APAR、および文書変更が含まれています。QMF バージョン 12 リリース 1 の PSP アップグレード名は QMFC10 です。PSP アップグレード QMFC10 の有効なサブセット名を以下に示します。

表 75. PSP アップグレード QMFC10 に含まれる QMF PSP サブセット名

| PSP サブセット名 | 適用対象 |
|------------|------------------------|
| HHPCC10 | QMF HPO |
| HQDC110 | QMF Data Service |
| HSQCC10 | 英語 (QMF 基本製品) |
| JSQCC1Q | QMF for z/OS OTC |
| JSQCC1C | QMF Classic Edition |
| JSQCC1E | QMF Enterprise Edition |
| JSQCC51 | U/C 英語 |
| JSQCC55 | デンマーク語 |
| JSQCC5G | カナダ・フランス語 |
| JSQCC56 | フランス語 |
| JSQCC57 | ドイツ語 |
| JSQCC58 | イタリア語 |
| JSQCC59 | 日本語 |
| JSQCC5A | 韓国語 |
| JSQCC5B | ブラジル・ポルトガル語 |
| JSQCC5C | スペイン語 |
| JSQCC5D | スウェーデン語 |
| JSQCC5E | スイス・フランス語 |
| JSQCC5F | スイス・ドイツ語 |
| JYQCC10 | QMF アプリケーション |

関連概念:

45 ページの『リクエスター (DB2 for z/OS) データベースのインストール前提条件』


スタンドアロン・データベースまたはリクエスター・データベースとして機能する DB2 for z/OS データベースに QMF をインストールするには、ハードウェアおよびソフトウェアの要件を満たす必要があります。

関連タスク:

461 ページの『QMF サービス・レベルの判別』

ターゲット・ゾーンおよび配布ゾーンに対する SMP/E 報告書の実行は常に、サービス・レベルを判別するための最適な方法です。ただし、QMF サービス・レベルを他の方法で判別することもできます。

関連情報:

 IBM ソフトウェア・サポート Web サイト
QMF の最新 PTF とその前提条件製品を検索します。

一般的な問題の訂正

より詳細な診断を試行する前に、一般的な問題および可能性がある解決法のリストを確認してください。

初期化時に発生する可能性があるエラー

一般に、QMF 初期化中または接続プロセス中に発生するエラーで QMF は停止しません。ただし、これらのタイプのエラーは、通常は QMF トレース・データに記録されます。

トレース・データの場所は、ご使用の環境によって異なります。

- TSO、ISPF、およびネイティブ z/OS では、トレース・データは DSQDEBUG に保管される。
- CICS では、QMF セッションの開始時に DSQSDBQT および DSQSDBQN プログラム・パラメーターを使用してキューの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・キューに保管されます。

トレース・データを見るのに加え、画面のメッセージを確認してください。メッセージが何も表示されておらず、QMF for TSO を始動しようとしている場合は、TSO コマンド (PROFILE MSGID WTPMSG) を発行して、QMF を再始動します。このコマンドは、メッセージを TSO 内の画面に記録します。

トレース出力をレビューし、メッセージが画面に発行された後で、次のトピックを呼んで問題を判別してください。これらのエラーがいずれも問題でないと思われる場合は、問題の判別とその原因の診断に役立つ、追加の診断援助機能を検討してください。

関連概念:

449 ページの『診断援助機能の使用』

多様な診断援助プログラムを使用して問題を診断します。

関連タスク:

454 ページの『トレース機能』

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。

QMF が現行のロード・モジュールを使用していないときに発生する問題

QMF を初期化するときまたはリモート・データベースに接続するときの予期しない動作、およびエラーまたは警告メッセージは、QMF が 1 つ以上の旧リリースからのモジュールを使用して初期化しているときに発生する可能性があります。

手順

QMF が正しいロード・モジュールで初期化されているかどうかを確認するための次の手順に従ってください。

1. コマンド SHOW GLOBALS を発行して DSQAO_QMF_VER_RLS 変数の値を確認します。QMF バージョン 12 リリース 1 でのこの変数の値は、QMFV12R1.0 です。

変数値が予想した値と違う場合は、初期化処理中に、QMF が旧リリースのロード・モジュールにアクセスしていることが考えられます。SDSQLOAD ライブラリーをどのように割り振りしたかによって、QMF が初期化時にアクセスするロード・モジュールは、予期しないデータ・セット、リンク・リスト、または LPA になる場合があります。

DSQAO_QMF_VER_RLS 変数の値が QMFV12R1.0 であっても、DSQUOPTS モジュールがロードされていると、初期化プロセスは以下のエラーによって中断されることがあります: 438 ページの『DSQ90579』。

2. QMF がアクセスするロード・モジュールのレベルを決定します。この手順を完了する方法は、QMF を始動できるかどうかによって異なります。

- QMF を始動できる場合:

- a. プログラムを DSQSDEBUG パラメーターに対して値 ALL で始動します。
- b. トレースを実行した後に、結果トレース出力をレビューします。QMF バージョン 12.1 モジュールは、QMF トレース内の目印 V12R1.00 に関連付けられています。以下に例を示します。

```
-----
***** 09/12/22 10.25.52 ***** -----
DSQCPARM:Applied Service Level (1)
34347370 DSQCABND V12R1.00 09/09/16 13:36 WIM11328
343476C8 DSQCADJM V12R1.00 09/09/16 13:36
345F50C8 DSQCAVAR V12R1.00 09/09/16 13:36 WIM11266
3464D8D0 DSQCBDS V12R1.00 09/09/16 13:36 WIM7688
34347E88 DSQCBMPA V12R1.00 09/09/16 13:36
34349290 DSQCBOR V12R1.00 09/09/16 13:36 WIM11215
346B69D8 DSQCB64C V12R1.00 09/09/16 13:36 WIM11525
```

- QMF を始動できない場合:

結果ダンプ出力 (SYSUDUMP、DSQUDUMP、または SYSMDUMP 出力) のバージョンおよびリリース情報を検査します。すべての QMF モジュールに関連付けられている目印が同じであることを確認します。QMF バージョン 12.1 モジュールは、上の例に示すように、目印 V12R1.00 に関連付けられています。

3. QMF 初期化処理が、現行のリリースの代わりに 1 つ以上の旧リリースからのロード・モジュールを使用していることを検出した場合は、ロード・モジュールがアクセスされる場所を決定します。

- QMF for TSO を ISPF 下で実行している場合は、ISPF ISRDDN 診断ユーティリティを使用して QMF ロード・モジュールのファイル割り振り情報を取得できます。ISRDDN は、TSO ユーザーのアドレス・スペース内の割り振り DD 名のリストを提供するユーティリティです。また、この機能を使用して QMF ファイル割り振りを表示する方法は、これらの割り振りがあるように行われているかによって異なります。例えば、QMF ファイル割り振りが ISPF を通じて行われている場合は、QMF からコマンド (TSO ISRDDN) を発行できます。このコマンドは、現行のデータ・セット割り振りのリストを返します。リストを参照して、すべてのデータ・セットが正しい場所からアクセスされていることを確認します。データ・セットの割り振り例については、以下の情報を参照してください。

- 64 ページの『TSO でのリクエスター・インストールのカスタマイズ』
- 73 ページの『CICS でのリクエスター・インストールのカスタマイズ』

QMF バージョン 12.1 データ・セットのデフォルトのデータ・セット接頭部は、QMF1210 です。ご使用の QMF バージョン 12.1 ロード・モジュールのデータ・セット接頭部は、これとは異なる場合があります。

QMF バージョン 12.1 ライブラリーを指していない割り振りを検出した場合は、これらの割り振りを正しい場所を指すように変更します。

- ISRDDN ユーティリティを使用して、特定のモジュールの割り振り情報を取得でき、必要に応じてオプションでモジュールを参照することができます。例えば、次のコマンドを発行して DSQQMF モジュールがロードされた場所を判別し、オプションでモジュールを参照することができます。TSO ISRDDN MEMBER DSQQMF

DSQQMF モジュールが予測された場所からアクセスされ、予期しているバージョンおよびリリース ID を含んでいることを確認してください。例えば、次の出力は、DSQQMF モジュールが、DD 名 DSQLLIB を介してアクセスされた QMFCOM.QMF1210.SDSQLOAD データ・セットからロードされたことを示します。ご使用の QMF データ・セットの特定の名称は、異なります。

選択した QMF バージョン 12.1 データ・セット接頭部がバージョンおよびリリース情報を含んでいない場合は、この情報を判別する支援として次のコマンドを使用して DSQQMF ロード・モジュールを参照します。TSO ISRDDN BROWSE DSQQMF

QMF バージョン 12.1 については、下図に示すように、目印が V12R1.00 と一致することを確認します。モジュールの日付、時刻、およびサービス・レ

ベルは、適用された PTF によって異なります。

```
BROWSE    DSQQMF JPA Start:34484B48 Size:0038BE10    Line 00000000 Col 001 080
***** Top of Data *****
+0 (34484B48) 47F0F122 2EC4E2D8 C3C9D5E3 4040E5F1 *    DSQCINT V1 *
+10 (34484B58) F0D9F14B F0F04040 404040F0 F961F0F9 * 1R1.00 12/12 *
+20 (34484B68) 61F1F640 F1F37AF3 F740E6C9 D4F1F1F5 * /16 13:37 WIM125 *
+30 (34484B78) F9F840D3 89838595 A2858440 D481A385 * 33 Licensed Mate *
+40 (34484B88) 99898193 A2406040 D7999697 8599A3A8 * rials - Property *
+50 (34484B98) 40968640 C9C2D440 F5F6F3F5 60C4C2F2 * of IBM 5615-DB2 *
+60 (34484BA8) 6B40F5F6 F0F560C4 C2F2404D C35D40C3 * , 5697-P43 (C) C *
+70 (34484BB8) 9697A899 898788A3 40C9C2D4 40C39699 * opyright IBM Cor *
+80 (34484BC8) 974B40F1 F9F8F26B 40F2F0F1 F04B40C1 * p. 1982, 2013. A *
+90 (34484BD8) 939340D9 898788A3 A240D985 A28599A5 * ll Rights Reserv *
+A0 (34484BE8) 85844B40 E4E240C7 96A58599 95948595 * ed. US Governmen *
+B0 (34484BF8) A340E4A2 8599A240 D985A2A3 998983A3 * t Users Restrict *
+C0 (34484C08) 858440D9 898788A3 A2406040 E4A2856B * ed Rights - Use, *
+D0 (34484C18) 4084A497 93898381 A3899695 40969940 * duplication or *
+E0 (34484C28) 8489A283 9396A2A4 99854099 85A2A399 * disclosure restr *
+F0 (34484C38) 8983A385 844082A8 40C7E2C1 40C1C4D7 * icted by GSA ADP *
+100 (34484C48) 40E28388 8584A493 8540C396 95A39981 * Schedule Contra *
+110 (34484C58) 83A340A6 89A38840 C9C2D440 C3969997 * ct with IBM Corp *
+120 (34484C68) 4B4090EC D00C187F 41807FFF 41908FFF * -----*
Command ==> Scroll ==> PAGE
```

図 116. ISRDDN ユーティリティーを使用してバージョンおよびリリース方法を取得するために DSQQMF モジュールを参照する

関連タスク:

454 ページの『トレース機能』

QMF には、ユーザーのセッション中に QMF のアクティビティー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。

異常終了

初期化時に、特定の異常終了が発生することがあります。異常終了を診断する場合、ネイティブ z/OS、TSO、または CICS の診断機能を使用することが必要になることがあります。

初期化時の一般的な異常終了について詳しくは、以下のいずれかのセクションを参照してください。

- 434 ページの『0C1』
- 434 ページの『0C4』
- 434 ページの『0C7』
- 434 ページの『ASRA』
- 436 ページの『AEY9』
- 436 ページの『AZTS』
- 436 ページの『G050』

0C1

0C1 異常終了が発生した場合、問題は QMF 初期化処理が、現行のリリースの代わりに 1 つ以上の旧リリースからのロード・モジュールを使用していることが原因である可能性があります。この状況が問題かどうかを判断するには、431 ページの『QMF が現行のロード・モジュールを使用していないときに発生する問題』を参照してください。

0C4

通常はパネルを表示する QMF コマンドへの応答で 0C4 異常終了が発生した場合は、次の手順を実行して問題のトラブルシューティングを行ってください。

1. DSQ1nPNL インストール・ジョブを実行したことを確認します。n は 1 文字の各国語 ID です。このジョブは、VSAM パネル・ライブラリー DSQPNLn を QMF バージョン 12.1 の製品パネルおよびヘルプ・パネルで設定します。このジョブの詳細については、50 ページの『VSAM パネル・ライブラリーのデータの設定』または QMF プログラム・ディレクトリーを参照してください。ご使用の QMF リリースおよび各国語用の正しいプログラム・ディレクトリーを使用していることを確認してください。
2. DSQ1nPNL ジョブが正常に実行された後でも問題が続く場合は、以下の DD ステートメントのファイル割り振りを確認してください。

- QMF for TSO:

DD ステートメント ADMGGMAP の割り振りを確認して、QMF1210.SDSQMAPn を指していることを確実にします。

- QMF for CICS:

QMF が常駐する CICS 領域の始動 JCL 内の ADMF DD ステートメントが GDDM ADMF データ・セットを正しく指していることを確認します。DSQ1nADM インストール・ジョブは QMF バージョン 12.1 GDDM マップをこのデータ・セットに書き込みます。

3. ファイル割り振りが正しい場合は、追加のトラブルシューティング提案について 431 ページの『QMF が現行のロード・モジュールを使用していないときに発生する問題』を参照してください。異常終了は、QMF バージョン 12.1 が誤って 1 つ以上の旧リリースからのモジュールを使用して初期化されたときに様々な場所で発生する可能性があります。

0C7

0C7 異常終了が発生した場合、問題は、QMF 初期化処理が、現行のリリースの代わりに 1 つ以上の旧リリースからのロード・モジュールを使用していることが原因の可能性があります。この状況が問題かどうかを判断するには、431 ページの『QMF が現行のロード・モジュールを使用していないときに発生する問題』を参照してください。

ASRA

ASRA 異常終了は、QMF for CICS を実行している場合に発生する可能性があります。このエラーを受け取った場合は、問題の可能性のある以下の領域を確認してください。

- QMF 始動時
 - すべての GDDM リンク・エディットが正常に実行されたかを確認します。
 - GDDM IVP が CICS 領域内で正常に実行されたかを確認します。
 - QMF プログラム・ディレクトリー内に文書化されている QMF SMP/E APPLY の手順が正常に完了していることを確認します。これらの手順は、QMF ロード・モジュールを CICS ロード・モジュールにリンクします。ご使用の QMF リリースおよび各国語用の正しいプログラム・ディレクトリーを使用していることを確認してください。
 - 領域が QMF バージョン 12 リリース 1 ロード・ライブラリーおよびマップ・グループを割り振りしていることを確認します。インストールおよびカスタマイズ処理のこれらの手順の詳細については次のトピックのいずれかを参照してください。
 - 78 ページの『CICS 始動ジョブ・ストリームの更新』
 - 76 ページの『GDDM ADMF データ・セットへの QMF GDDM マップのロード』
- 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 に対して正常にインストールおよび調整されているかを確認します。GDDM が CICS と同じ CSI ゾーンに配置されているかを確認します。
- 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、CSD エントリー GDDM モジュール ADMASPLC 用に存在しているかを確認します。
- モジュール DSQQMFE、CSECT DSQIELI において:

CSD エントリーが DB2 for z/OS 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します。
- モジュール DSQCBST、CSECT DSQCMCVP において:

QMF サービスの適用後、QMF コードがライブラリー・ルックアサイドにある場合に z/OS LLA REFRESH が行われたかを確認します。
- 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」ファンクション・キーを押したときに発生します。DSQPNLn の CSD ファイル・エントリーが RECFM=V を持っていることを確認します (ここで、n は、1 文字の各国語 ID です)。
- HELP または RUN コマンドの発行時:

QMF データ・セット DSQPNLn (ヘルプ・テキストおよび他の画面テキストを含む) が、正常に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ませんでした。問題を訂正するには、以下のようになります。

 - CSD エントリーが正しく定義されていることを確認します。

- DSQPNL n 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ことを確認します。DD ステートメントについては、78 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。

また、DSQPNL n データ・セットに関連付けられているコンソール・エラー・メッセージを探します。

n を、使用している NLF を表す 1 文字の言語 ID に置き換えます。

AEY9

DB2 for z/OS 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

GDDM が IOSYNCH=YES を指定して実行されているか確認する。

G050

CICS に合わせてカスタマイズした 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ことを確認します。

関連概念:

464 ページの『環境に固有の診断の使用』


TSO、z/OS、または CICS の診断機能を用いて、異常終了を診断する必要があることがあります

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

関連情報:

 DB2 照会報告書作成プログラム・ライブラリー

QMF バージョン 12.1 プログラム・ディレクトリーに関する情報を参照してください。

QMF メッセージ

QMF 初期化時にエラーが発生した場合は、エラー・メッセージを確認し、適切な処置を行うことによって復旧できます。

DSQI0026

このメッセージは通常、始動時に出されます。QMF を CICS の下で実行している場合、QMF n トランザクションがクリア画面から入力されていることを確認します。

DSQ10297

サブシステム ID が無効。

このエラーは、ISPF 始動時に、呼び出し可能インターフェースを使用している場合、または QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始した場合に発生する可能性があります。DSQSSUBS パラメーター (短い名前は S) が、QMF が開始されるサブシステムを参照していることを確認します。

QMF を DB2 for z/OS ストアード・プロシージャーとして実行している場合、DSQSSUBS パラメーターは Q.DSQQMFSP プロシージャーがインストールされているサブシステムを参照していなければなりません。QMF の開始について詳しくは、以下のトピックのいずれかを参照してください。

- 95 ページの『ISPF を使用した QMF の開始』
- 143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』 および 160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』
- DB2 QMF アプリケーション開発の手引き。QMF を呼び出し可能インターフェースでの開始の詳細を提供します。

DSQ10344

コード化文字セット ID (CCSID) には、さまざまな処理ステージと交換ステージを通して文字の意味とレンダリングを割り当てて保存するために必要なすべての情報が含まれています。この情報には常に、少なくとも 1 つのコード・ページが含まれています。また、CCSID には、さまざまなコード・ポイントを処理する方法を管理する関連コード化スキームも含まれています。

初期化時に、QMF は GDDM アプリケーション・コード・ページ (APPCPG パラメーターによって指定される) を 2 つの異なる CCSID 値と比較します。

- CURRENT APPLICATION ENCODING SCHEME 特殊レジスター

データベースのコード化スキームは、DB2 for z/OS の特殊レジスターで指定されます。QMF を SBCS 言語内で実行している場合、GDDM アプリケーション・コード・ページはこの特殊レジスターの値と一致していなければなりません。

- SYSIBM.SYSTEM_EBCDIC_CCSID セッション変数

このセッション変数は、DB2 for z/OS 上の EBCDIC データに使用される CCSID を指定します。QMF を SBCS 言語内で実行している場合、GDDM アプリケーション・コード・ページはこのセッション変数の値と一致していなければなりません。

GDDM アプリケーション・コード・ページが上記の 2 つの CCSID データベースの 1 つと一致しない場合は、DB2 から GDDM に表示のために送信されるデータと表示内容が異なることを示すメッセージ DSQ10344 が警告として出されます。潜在的な表示の問題を訂正するには、APPCPG 外部デフォルトを変更することによって GDDM CCSID を変更する必要があります。詳しくは、以下のいずれかのトピックを参照してください。

- 64 ページの『GDDM 外部デフォルトのカスタマイズ』 for TSO
- 75 ページの『GDDM 外部デフォルトのカスタマイズ』 for CICS

詳しくは、IBM Publications Centerの GDDM 資料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 for z/OS リソース・コントロール表 (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。例えば、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、次のようにエントリーをコーディングしてください。

```
DSNCRCT TYPE=ENTRY, TXID=QMFE, PLAN=QMF12, AUTH=DEPT1
```

この例では、許可 ID は DEPT1 であり、プラン ID は QMF12 です。

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 ニックネーム定義ファイルにエラーがあります。

QMF IVP には、照会を印刷するための手順が組み込まれています。サイトで GDDM ニックネームを使用する場合、IVP プロシージャの PRINT QUERY ステートメントを PRINT QUERY (PRINTER = *gddmnickname* に変更してください。GDDM プリンター・ニックネームを作成するプロシージャは、273 ページの『第 13 章 印刷および図表作成機能のセットアップ』に提供されています。

DSQ90579

このエラーは、DB2 QMF for z/OS 下で QMF for TSO または QMF for CICS を開始しようとしているが、DSQUOPTS 出口ルーチン内の OTC_LICENSE 変数が TERMS_ACCEPTED に設定されていることを示しています。この問題を訂正するには、以下のようにします。

1. DSQUOPTS 出口ルーチン内の OTC_LICENSE 変数の値を NOT_USED に変更します。
2. 176 ページの『DSQUOPTS ルーチンによる大域変数の設定』で説明されているように、ルーチンをアセンブルおよびリンク・エディットします。
3. QMF を開始する初期化プロシージャに割り振られた出口ライブラリーに、変更した DSQUOPTS ロード・モジュールを入れます。サンプル・ジョブ DSQ1UOPT を使用して DSQUOPTS 出口ルーチンをアセンブルし、リンク・エディットすると、変更された DSQUOPTS ロード・モジュールがデフォルトの QMF の出口ライブラリー QMF1210.SDSQEXIT に保管されます。

DSQUOPTS 出口ルーチンについて詳しくは、175 ページの『QMF 開始時の大域変数および QMF セッション動作の初期化』を参照してください。

DSQ50405

QMF を開始するときの次のエラー・メッセージは、QMF が旧リリースからのロード・モジュール・ライブラリー (SDSQLOAD) で初期化されていることを示します。

```
Number of REXX program parameters returned should be &V1, not &V2.
```

数値は変数の代わりにメッセージ内に表示されます。この問題は、リンク・リストまたは LPA 内に旧リリースのロード・モジュールがある場合、QMF バージョン 12.1 SDSQLOAD ライブラリーを正しく割り当てたときにも発生する可能性があります。このメッセージは、QMFが QMF1210.SDSQEXCE(DSQSCMDE) のバージョン 12.1 コピーにアクセスしているが、SDSQLOAD ロード・ライブラリーの以前のコピーにアクセスしていることを示しています。この問題を診断および訂正するために行う手順については、431 ページの『QMF が現行のロード・モジュールを使用していないときに発生する問題』を参照してください。

SQL コード -805 に関連する QMF エラー・メッセージ

以下の QMF エラーは SQL コード -805 に関連しており、初期化中に発生する可能性があります。

- DSQ10205
- DSQ11205
- DSQ12105
- DSQ13005
- DSQ14152
- DSQ14153
- DSQ14154
- DSQ15805
- DSQ16805
- DSQ17805
- DSQ30805
- DSQ31805
- DSQ35805
- DSQ36805

SQL コード -805 から返されるすべてのトークンを記録してください。この SQL コードについて詳しくは、DB2 資料 を参照してください。トラブルシューティングのヒントについては、次のページを参照してください: <http://www.ibm.com/support/docview.wss?uid=swg21567609>

SQL コード -551 に関連する QMF エラー・メッセージ

初期化中に、QMF は、QMF を開始するユーザーの許可 ID に、Q.PROFILES コントロール表に対する INSERT 特権または DELETE 特権があるかどうかを検査します。許可 ID がこれらの特権のいずれかを持っている場合は、QMF はそのユーザーが QMF 管理者であるとみなします。このユーザーが QMF 管理者ではない場合、INSERT 特権、DELETE 特権、またはその両方に -551 SQL コードが発行されます。この SQL コードは、このような状況下で予想されるものです。これはエラーではないため、メッセージがエンド・ユーザーには表示されないか、または QMF トレース・データ・セット (DSQDEBUG) またはエラー・ログ表 (Q.ERROR_LOG) に取り込まれていません。このメッセージは QMF 初期化を停止したり、QMF 操作中に問題を引き起こしたりしません。ただし、何らかの理由により -551 SQL コ

ードを抑制する必要がある場合は、175 ページの『QMF 開始時の大域変数および QMF セッション動作の初期化』のプロシージャーに従って QMF 管理者権限検査を使用不可にする必要があります。

以下の QMF メッセージは、SQL コード -551 に関連しています。

- DSQ10951
- DSQ11851
- DSQ12751
- DSQ15551
- DSQ16551
- DSQ17551
- DSQ30551
- DSQ31551
- DSQ35551
- DSQ36551

QMF 管理者権限について詳しくは、18 ページの『QMF 管理に必要な権限』を参照してください。

QMF の開始に関する問題

QMF の開始に関する問題のトラブルシューティング・プロセスを使用して、問題の原因を判別し、解決方法を見つけます。

QMF を開始できない場合は、次の方法で問題をトラブルシューティングしてください。

- QMF の開始による問題の発生範囲がすべてのユーザーか、特定の個人またはグループのみかを判別する。
- 表示画面にメッセージが示されているかどうかを確認し、さらにトレース・データに表示されたメッセージについての説明を「DB2 QMF メッセージおよびコード」で探す。
- 画面に何も表示されていず、トレース・データにも何もない場合は、ISQL に入って `SELECT * FROM Q.ERROR_LOG` ステートメントを発行し、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ます。
- 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・メッセージが表示された場合は、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 QMF の開始時に 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 製品が正常に動作しているかどうかを確認します。

- QMF 初期化プロセスが現行のリリースのロード・モジュールにアクセスしていることを確認します。初期化の問題は、QMF が 1 つ以上の旧リリースからのモジュールを使用して初期化している場合に発生します。詳細については、431 ページの『QMF が現行のロード・モジュールを使用していないときに発生する問題』を参照してください。

正しくないバージョンの DSQUOPTS 出口ルーチンが初期化時にロードされた場合、または OTC_LICENSE 変数の値が、開始しようとしている QMF のセッションには適切でない場合は、エラーが発生することがあります。詳細については、438 ページの『DSQ90579』を参照してください。

コマンド同義語、ファンクション・キー、およびリソース・コントロール表に関する問題

コマンド同義語、ファンクション・キー、およびリソース・コントロール表に問題がある場合は、ユーザーの特権または表の構造およびデータを確認することによって復旧できます。

コマンド同義語、ファンクション・キー、およびリソース・コントロール表について、以下の点を確認してください。

- ユーザーが関連する表に対して 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、299 ページの『コマンド同義語のカスタマイズ』示されている。
 - ファンクション・キー表の構造については、307 ページの『QMF ファンクション・キーのカスタマイズ』に示されている。
- 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、301 ページの『コマンド同義語定義を表に入力』参照。
 - 有効なファンクション・キー定義については、314 ページの『ユーザーのファンクション・キー定義を表に入力』参照。
- 表のすべての行が固有であること。
- ユーザーの QMF プロファイルの SYNONYMS フィールドおよび PFKEYS フィールドに、コマンド同義語定義およびファンクション・キー定義が保管されている表の名前が含まれていること。現在使用されている表の名前を確認するには、以下の大域変数のいずれかを参照してください。
 - DSQAP_SYNONYM_TBL 大域変数には、現行 QMF セッションに使用されているコマンド同義語表の名前が含まれています。
 - DSQAP_PFKEY_TABLE 大域変数には、現在使用されているファンクション・キー表の名前が含まれています。

QMF の開始後の警告メッセージ

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。

QMF を開始した後に出される警告メッセージは、TSO と CICS の両方に同じ許可 ID を使用したことが原因であると考えられます。ユーザーが TSO および CICS の両方に同じデータベース許可 ID を持っていて、同じ QMF プロファイルを両方の環境に使用している場合は、QMF が警告を出す場合があります。例えば、Q.COMMAND_SYNONYMS 表内の環境固有コマンドに対する無効なエントリーについての警告が出される場合があります。両方の環境用のコマンドを含む QMF コ

マンド同義語表をセットアップできますが、ユーザーは、そのときに実行中の QMF 内の環境に固有ではないコマンドを発行した場合にエラーを受け取る場合があります。この問題が原因である警告メッセージを除去するには、各環境に固有の QMF プロファイルを割り振ります。

関連概念:

454 ページの『トレース・データ・セットの割り振り』
QMF の実行方法に応じて、DSQUDUMP データ・セットまたは DSQDEBUG データ・セットが割り振られていることを確認します。

正しくない出力

場合によっては、問題が生じてもエラー・メッセージが表示されないことがあります。このようなエラーの最も一般的なタイプは、出力の誤りです。

例えば、QMF ホーム・パネルがバージョン 12 リリース 1 を読み取らず、代わりに別のリリースを指している場合が考えられます。このエラーが発生している場合は、次のいずれかの方法で訂正してください。

- QMF for TSO を使用している場合は、DD ステートメント ADMGGMAP の割り振りを確認して、QMF1210.SDSQMAP n を指していることを確実にします。
- QMF for CICS を使用している場合は、QMF が常駐する CICS 領域の始動 JCL 内の ADMF DD ステートメントが GDDM ADMF データ・セットを正しく指していることを確認します。DSQ1 n ADM インストール・ジョブは QMF バージョン 12.1 GDDM マップをこのデータ・セットに書き込みます。

予期しない動作は、QMF が現行のリリースと 1 つ以上の旧リリースからのロード・モジュールの混合セットを使用しているときにも発生します。

注: 表示された報告書や印刷された報告書で見られる誤った出力は、不適切な順序の照会が原因である場合があります。表示された報告書または印刷された報告書を完了するために十分な記憶域が QMF になく、データの再取得のために複数回のデータベースへのアクセスが必要である場合、すぐに報告書データを同じ順序にしなければなりません。BREAK や GROUP のような取扱コードは、報告書の処理のためにはデータが同じ順序であることを必要とします。これは、報告書に関係している SQL 照会に ORDER BY ステートメントを指定することによって行います。詳しくは、BREAK 編集コードと GROUP 編集コードについて調べてください。

関連タスク:

431 ページの『QMF が現行のロード・モジュールを使用していないときに発生する問題』

QMF を初期化するときまたはリモート・データベースに接続するときの予期しない動作、およびエラーまたは警告メッセージは、QMF が 1 つ以上の旧リリースからのモジュールを使用して初期化しているときに発生する可能性があります。

印刷の問題

印刷エラーの原因は、QMF または GDDM のいずれかにある可能性があります。

GDDM エラー

印刷中に GDDM エラーが発生した場合、QMF は GDDM プリンター・ニックネームを含むメッセージを表示します。

QMF は次の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。このトピックでは、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。

GDDM 印刷エラーは、ADM という文字から始まります。

表 76. 一般的な GDDM エラー

| GDDM エラー・コード | メッセージ | 説明および可能な解決方法 |
|--------------|---|--|
| DSQ50623 | GDDM error. ADM0307 E FILE 'ADMPRINT.REQU—QUEUE' NOT FOUND. Severity 8. Function DSOPEN. *** CMD=PRINT | 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 | 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 | ニックネーム定義が正しくありません。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。 |
| DSQ50631 | GDDM error. ADM0904 E ALPHANUMERIC FIELDS ARE NOT SUPPORTED FOR THIS DEVICE. Severity 8. Function ASDFLD. *** CMD=PRINT | ユーザーが印刷しようとしている出力は、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は無効です。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。 |
| 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 外部デフォルト・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。 |

関連情報:

 IBM Publications Center

GDDM エラー、有効な装置トークン、およびプリンターのファミリー別出力に関する説明を GDDM 資料で検索します。

QMF エラー

印刷エラーの発生元が QMF である場合、その多くは復旧が可能です。

下の表にある情報が、印刷時に発生する可能性のあるエラーを解決するために役立ちます。

表 77. QMF 印刷エラーとそれらに適した解決方法

| 症状 | 意味 | 処理 |
|---|---|--|
| コマンド行またはファンクション・キーから PRINT コマンドを発行すると、次のようなメッセージが表示される。 GDDM printer nickname is required for PRINTER. | 印刷しようとしているオブジェクトにはプリンター名が必要になるが、プロファイルにデフォルト・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・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、プロシージャー、プロファイル、SQL または QBE 照会については、PRINTER=' ' を指定して GDDM の印刷をう回する。 • その他のオブジェクトについては、装置トークンを指定しない PROCOPT=((PRINTCTL,0)) を使用する。 |
| 報告書、表、プロシージャー、プロファイル、SQL または QBE 照会の印刷時に、次のメッセージが表示される。 File DSQPRINT did not open. | プロファイルにデフォルト・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。 | PRINT コマンドを出す前に、DSQPRINT を割り振る。 |

TSO の下で QMF を使用していて、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。FREE DDNAME(DSQPRINT)

表示エラー

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 つ以上の列に (文字データではなく) 2 進データが入っています。これらの制御文字を処理するには、HEX 関数または編集コードを使用します。

HEX 関数を使用する方法

HEX 関数は、引数に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、引数を 16 進表記で表した値となります。例えば、関数の引数が 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。例えば、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
```

16 進および 2 進データの編集コード

QMF に用意されている編集コードによって、QMF は文字の列に 2 進データを表示することができます。編集コードとは、X または XW (16 進データの場合) と、B、BW、C、または CW (2 進データの場合) です。

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。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。

関連概念:

321 ページの『第 16 章 QMF 書式のためのカスタム編集出口ルーチン』
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。デフォルト編集コードがお客様のサイトの報告書編集の要件に合わない場合は、独自の編集コードを作成することができます。

記憶域関連問題の解決

エラー・メッセージ DSQ14352、DSQ14362、DSQ14484、または DSQ14504 (要求を完了するには、2 GB 境界より上の記憶域が不足しています) が出された場合は、117 ページの『2 GB 境界より上の記憶域の要件に関する変更』を参照してください。QMF バージョン 11 以降、QMF は、XML データ・タイプと LOB データ・タイプに 2 GB 境界より上の記憶域を使用します。

QMF パフォーマンスの管理

QMF パフォーマンスをモニターして、一般的なパフォーマンス問題に対する何らかの解決方法または回避方法を見つけることができます。

動的ステートメントの EXPLAIN 情報の取り込み

適格な動的 SQL 照会の EXPLAIN 情報を取り込むことで、これらの照会のパフォーマンスをモニターすることができます。EXPLAIN データには、SQL ステートメントを処理するために使用されるアクセス・パスに関する情報が含まれています。

始める前に

以下の前提条件が必要です。

- DB2 for z/OS がバージョン 10 新機能モード以降である
- PLAN_TABLE 表および DSN_STATEMENT_CACHE_TABLE 表が DB2 for z/OS サーバー上に存在する

このタスクについて

動的 SQL ステートメントの EXPLAIN 機能の動作は、CURRENT EXPLAIN MODE 特殊レジスターによって制御されます。QMF では、DSQEC_EXPL_MODE 大域変数を介してこの特殊レジスターを設定できます。この大域変数は、RUN QUERY コマンドによって発行されるステートメントにのみ影響します。

手順

動的 SQL ステートメントの EXPLAIN 情報を取り込むには、以下のようになります。

DSQEC_EXPL_MODE 大域変数を YES または EXPLAIN に設定します。

値 YES の場合、EXPLAIN 可能な動的 SQL ステートメントは正常に実行されます。情報は、各ステートメントが準備および実行された後で EXPLAIN 表に取り込まれます。

値 EXPLAIN の場合、EXPLAIN 可能な動的 SQL ステートメントは実行されませんが、アプリケーションで各ステートメントが準備された後、EXPLAIN 表に情報が取り込まれます。ステートメントが実際に正常に実行されることに依存するアプリケーションは、この大域変数が EXPLAIN に設定されている場合に実行すると失敗します。単純なアプリケーション・ロジックを使用するアプリケーションでのみ、このオプションを使用してください。

QMF 照会を照会加速に適格にする

DB2 for z/OS データベースで使用可能な照会アクセラレーターを使用すると、特定の種類の照会のパフォーマンスを大幅に向上させることができます。QMF 照会は、DB2 資料にリストされている照会加速の前提条件および条件を満たしている場合に、照会アクセラレーターでの実行に適格となることができます。

このタスクについて

一般に、QMF 照会は、照会加速の適格性要求を本質的に満たしています。例えば、QMF RUN QUERY コマンドは、ユーザー発行の SQL ステートメントを動的に実行します。ただし、QMF 照会が照会加速に必ず適格になるようにするには、以下の手順を実行する必要があります。

手順

QMF 照会が照会加速に適格になるようにするには、以下のようにします。

- CURRENT QUERY ACCELERATION 特殊レジスターが、照会加速に適切な値に設定されていることを確認します。

CURRENT QUERY ACCELERATION 特殊レジスターの値は、次の照会を実行することで判別できます。

```
SELECT CURRENT QUERY ACCELERATION
FROM SYSIBM.SYSDUMMY1
```

CURRENT QUERY ACCELERATION 特殊レジスターの初期値は、DB2 QUERY_ACCELERATION サブシステム・パラメーターの値によって決まります。QMF SQL 照会で SET CURRENT QUERY ACCELERATION ステートメントを発行して、QUERY_ACCELERATION サブシステム・パラメーターの現行値をオーバーライドすることができます。

- 照会アクセラレーターの機能によっては、QMF DSQSMRFI (MR) プログラム・パラメーターが重要になる場合があります。一部の照会アクセラレーターでは、行セット・カーソルを使用して実行される照会はサポートされません。DSQSMRFI (MR) プログラム・パラメーターを YES に設定して QMF が開始された場合、QMF は行セット・カーソルを使用します。

一部の照会アクセラレーターでは、行セット・カーソルを使用すると、照会が照会加速に不適格になります。DSQSMRFI (MR) パラメーターの現行値を確認するには、SHOW GLOBALS コマンドを発行して、DSQAO_DSQSMRFI 変数の値を確認してください。DSQAO_DSQSMRFI 大域変数の値 0 は、DSQSMRFI プログラム・パラメーターに NO が指定されたことを意味します。

DSQAO_DSQSMRFI 大域変数の値 1 は、DSQSMRFI プログラム・パラメーターに YES が指定されたことを意味します。

記憶域の問題の解決

ユーザーは、操作で必要なすべての行を取得するために十分な仮想記憶域が使用可能でないときに、照会の実行または報告書の形式設定にパフォーマンスの低下に気付くことがあります。1 つ以上の領域で記憶域を増大することにより、この問題から復旧することができます。

ユーザーの報告書記憶域の増加

記憶域関連のエラーまたはパフォーマンス問題が発生している場合、ユーザーの仮想記憶域または予備データ構成を調整することができます。

- 報告書用仮想記憶域の調整

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、パフォーマンスが低くなるという経験をすることがあります。例えば、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よ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。パフォーマンスを最大にするためには、ユーザーが必要とすると思われる報告書のうち、最大の報告書に対応できる値を使用してください。ただし、DSQSRSTG パラメーターに非常に高い値を設定するとパフォーマンスを低下させる危険性があることに注意してください。

- 報告書用予備記憶域の定義

QMF for TSO で、アクティブな記憶域でこれ以上必要がないデータを保管して報告書パフォーマンスを向上するための拡張仮想記憶域の使用を指定することができます。

QMF for TSO/CICS の両方のファイルにデータを予備書き込み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。予備ファイルがあっても、このファイルが全データを保持できるほどの大きさでない場合は、不完全なデータの状態としてユーザー側に示されることがあります。CICS では、予備ファイルには各行 4 KB 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。

予備ファイルを使用している場合、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。この場合は、入出力異常終了が発生します。QMF は、DCB SYNAD 出口のエラー処理ルーチンを提供し、DCB 異常終了出口を使用して提供された情報を使用してこれらの I/O 異常終了からリカバリーします。次に、QMF は、予備ファイルの使用を終了してデータベースからデータを再びフェッチします。

重要: DCB 異常終了 (B37 異常終了) をインターセプトする z/OS ツールを使用している場合、QMF 予備ファイルをそのような操作から除外することを確認します。除外しない場合、QMF が予備ファイルを適切に管理できなくなり、予期しない結果が発生するだけでなく、エラーのトレースおよび診断が困難になります。

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 ステートメントによって、この記憶域グループにボリュームを追加します。

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。これが発生した場合は、CICS 領域のサイズがユーザーのニーズを満たすのに十分な大きさであることを確認します。インストールした追加の製品に必要な記憶域について考慮してください。

関連概念:

50 ページの『記憶域のアドレッシング要件』

SMP/E を使用して配布メディアから QMF バージョン 12.1 ライブラリーをコピーするために必要な記憶域の容量については、ご使用の QMF バージョンおよび各国語に該当する QMF プログラム・ディレクトリーを参照してください。

関連タスク:

201 ページの『拡張仮想記憶域への報告書データの書き込み (TSO のみ)』

QMF for TSO では、QMF が実行されているシステムで利用可能な拡張記憶域が非常に限られている場合を除き、予備データ用の拡張記憶域が使用されます。

関連情報:

 DB2 資料

DB2 ALTER STOGROUP ステートメントに関する情報を検索します。

リソース競合の問題の解決

SELECT 照会が、必要な行が他の操作によって更新されているときにタイムアウトになります。DB2 for z/OS 上でアプリケーションの並行性を増加してリソース競合を制限するには、QMF DSQEC_CON_ACC_RES 大域変数をサイトのニーズに従って設定することができます。

この大域変数の値は、DB2 for z/OS バージョン 10 および DB2 for z/OS バージョン 9 SKIP LOCKED DATA オプションで使用可能な並行アクセス解決をサポートします。

DSQEC_BUFFER_SIZE 大域変数設定による QMF パフォーマンスの向上

DSQEC_BUFFER_SIZE 大域変数の値を大きくすると、QMF パフォーマンスは向上します。

このタスクについて

デフォルトで、単一行と複数行の両方のフェッチ操作においてデータ行を保守するために、QMF は 4K バッファを使用します。複数行フェッチ操作では、データ行の長さが 2056 バイトを超えている場合、フェッチごとに 1 行だけが標準バッファに収容されます。結果として、長いデータ行に対する複数行フェッチ操作が、単一行フェッチより速く行われることはありません。

この動作を変更するために、管理者は DSQEC_BUFFER_SIZE 大域変数の値を大きくすることができます。バッファ・サイズが大きくなると、QMF はフェッチごとにより多くのデータ行を取り出すことができるようになり、データベース・トラフィックの削減によってパフォーマンスが大きく向上します。

診断援助機能の使用

多様な診断援助プログラムを使用して問題を診断します。

関連概念:

430 ページの『一般的な問題の訂正』

より詳細な診断を試行する前に、一般的な問題および可能性がある解決法のリストを確認してください。

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。

すべての QMF メッセージには DSQnnnnn という形式のメッセージ番号 (nnnnn は 5 桁の数字) が付いています。

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、IBM ソフトウェア・サポートでは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、次のように大域変数 DSQDC_SHOW_PANID を 1 に設定しておいてください。

```
SET GLOBAL (DSQDC_SHOW_PANID=1
```

関連資料:

 DB2 QMF メッセージおよびコード

実行可能な問題の解決方法をメッセージ番号またはコードで検索します。

エラー・メッセージを出した QMF 機能の判別

QMF メッセージ番号を使用して、メッセージを出した QMF コンポーネントを判別することができます。コンポーネントが分かると、問題を特定の機能に切り分けることができます。

QMF 機能とそれに関連するメッセージ番号の範囲を、次の表に示します。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。

下表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

表 78. 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 |

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、またはその他何らかのソフトウェア・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。

システム・エラーが発生しました。 コマンドは実行されていない可能性があります。

メッセージに関する詳細を表示するには、ヘルプ・キーを押してください。

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。

関連資料:

 DB2 QMF メッセージおよびコード

実行可能な問題の解決方法をメッセージ番号またはコードで検索します。

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 戻りコードに関連付けられることがあります。

例えば、QMF メッセージ DSQ10422 を受け取ったものとして、このメッセージは、SQL 戻りコード -30060 に関連付けられます。この戻りコードのテキストは以下のとおりです。

RDB AUTHORIZATION FAILURE

SQL 戻りコードに関連付けられているすべての QMF メッセージのオンライン・ヘルプは、SQLCA (SQL Communicates Area) のコンテンツを含みます。SQLCA 出力を参照するには、QMFメッセージ領域にメッセージが表示されている間にヘルプ・キーを押して、ヘルプ・テキストの最後までスクロールします。SQL 戻りコードについてさらに詳細が必要な場合は、DB2 資料を参照してください。

複数の SQL ステートメントを実行する SQL 照会は、複数の SQL 戻りコード (正または負) が発生する可能性があります。これらの場合では、照会が完了したときに表示される QMF メッセージ (および関連する SQL コード) は、照会内の最後のステートメントにのみ関連付けられます。QMF トレース出力を表示して、すべての SQL 戻りコードおよび照会内のすべてのステートメントに関連付けられている SQLCA 情報を参照することができます。

DSQDC_POS_SQLCODE 大域変数を使用して、QMF が正の SQL 戻りコードを処理する方法を制御することができます。次の値が、この変数で有効です。

- 2 ユーザーが正の SQL コードに関連付けられているオンライン・ヘルプを表示できるようにします。
- 1 SQL コードと関連付けられているメッセージを DSQDEBUG に記録します。
- 0 SQL コードと関連付けられているメッセージは記録されず、ヘルプ・テキストは提供されません。

関連タスク:

454 ページの『トレース機能』

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。

関連情報:

 DB2 資料

SQL 戻りコードに関する情報を検索します。

診断情報を取り込むための割り込みの作成

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できます。

このタスクについて

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 DSQDEBUD データ・セットへ書き込ませることができます。

TSO では、QMF コマンドが実行されていないときでも 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。

手順

QMF を割り込みするには、次の手順を行います。

1. 問題を診断している対象のユーザー ID のもとで QMF にログオンします。
2. 問題を再作成します。
3. 問題に関する情報の収集を開始する時点で、割り込みを作成します。

大部分のシステム構成では、ATTN キー、あるいはリセット・キーと PA1 キーを一緒に (または、端末エミュレーション・セッションで同等のキーの組み合わせ) 押すことによって、アテンション割り込みを作成できます。これらのキーの組み合わせが機能しない場合は、現在のシステム構成に合わせて適切な情報を参照して、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 117. QMF 割り込みハンドラー・プロンプト 1

4. 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 (または端末エミュレーション・セッション内で同等のキーの組み合わせ) 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 118. QMF 割り込みハンドラー・プロンプト 2

5. 以下のいずれかのアクションを実行します。

- 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すことができるとは限りません。
- 診断情報を取り込むには、DEBUG を入力します。次の出力例は、DEBUG を入力したことへの応答に含まれています。

```

-- 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 119. 割り込みプロンプトに対する応答として DEBUG を入力することによって取り込まれた診断情報

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、制御が CSECT DSQEINPT にあったことを示しています。制御は CSECT の DSQDSUPV、DSQDSUPX、DSQEADAP、DSQEMAIN を順番に通じ、この CSECT に達しています。

6. 必要なトレース・オプションを入力します。以下のいずれかのオプションを入力することで、割り込みプロンプト・パネルに応答します。

- 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すことができるとは限りません。
- 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。

トレース機能

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。

トレース・データ・セットの割り振り

QMF の実行方法に応じて、DSQUDUMP データ・セットまたは DSQDEBUG データ・セットが割り振られていることを確認します。

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、ISPF、またはネイティブ z/OS のトレース情報

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 DSQUDUMP 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 QMF と共に提供されている TSO ログオン・プロシージャー例に割り振られています。

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(DSQUDUMP) SYSOUT(A) USING(DUMP)
ALLOC DDNAME(SYSUDUMP) SYSOUT(A)
```

図 120. TSO 用のデータ・セットの割り振り

TSO を DB2 for z/OS ストアード・プロシージャーとして開始する場合、トレース出力用のデータ・セットおよびその他の機能は QMF が実行する WLM アドレス・スペースを開始する JCL 内に割り振られている必要があります。

- CICS のトレース情報

トレースは DSQDEBUG データ・セットに記録されます。このデータ・セットは、CICS 始動 JCL において割り振ってください。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・データ・セットは、区画外データ・セットとして記述されます。これは CICS インストール・ジョブ QMF1210.SDSQSAPE(DSQ1ECSD) が実行されたときに定義されます。次の図は定義情報を示します。

```
DEFINE TDQUEUE(DSQD)
DESCRIPTION(QMF Trace Info) GROUP(QMF1210E)
TYPE(EXTRA) BLOCKSIZE(6050) DDNAME(DSQDEBUG)
RECORDFORMAT(VARIABLE) TYPEFILE(OUTPUT) RECORDSIZE(121)
BLOCKFORMAT(BLOCKED)
```

図 121. 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、DSQSDBQN パラメーターで指定した CICS キュー ID (デフォルトは DSQD) を使用してトレース・データ・セットをクローズしなければなりません。トレース・データ・セットをクローズするには、CICS に付属のトランザクション CEMT を使用します。以下に例を示します。

```
CEMT I TDQUEUE(DSQD)
```

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、トレース・データ・セットをクローズするために使用したのと同じ CEMT トランザクションを使用して再びオープンします。

関連タスク:

143 ページの『QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

トレース機能の開始

ユーザーにトレース・オプションを指定して SET PROFILE コマンドを入力させるか、またはユーザーのプロファイルの TRACE フィールドを更新することにより、QMF トレースを開始します。QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始するときには、トレース・オプションはインターフェースを開始した CALL ステートメントに渡されます。

始める前に

454 ページの『トレース・データ・セットの割り振り』で説明しているように、DD 名 DSQDEBUG を持つデータ・セットが割り振られていることを確認します。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、トレース用にのみ使用されるものです。

手順

1. 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。

値 ALL を DSQSDEBUG プログラム・パラメーターに指定すると、QMF の初期化中に発生する可能性があるプログラム障害を含むすべての QMF アクティビティが最高のレベルの詳細までトレースされます。

出力が一時記憶域キューの最大サイズを超える場合は、一時データ・キューを使用します。

2. トレース・オプションを指定します。 QMF セッション中、トレース・オプションの一部のセットは DSQSDEBUG パラメーターに値 NONE を指定しない限り、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*)

前述のコマンドで、*value* は、ALL またはトレース出力での QMF 機能とその詳細レベルを示す文字列です。

- 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。例えば、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。

関連概念:

457 ページの『トレース出力の適切な詳細レベルの設定』

すべての QMF 機能を詳細にトレースすることができ、また個々の QMF 機能をトレースすることができます。

460 ページの『QMF トレース・データの表示』

DSQDEBUB 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る必要があります。

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

トレース出力の適切な詳細レベルの設定

すべての QMF 機能を詳細にトレースすることができ、また個々の QMF 機能をトレースすることができます。

すべての QMF 機能を最も詳細なレベルでトレースするには、Q.PROFILES 表の TRACE 列に値 ALL を使用します。このオプションの結果が大きすぎる場合は、DSQEC_TRACE_LIMIT 大域変数を設定して、トレース出力量を特定のサイズに制限することができます。

個々の QMF 機能をトレースするには、Q.PROFILES 表の TRACE 列に異なる文字ストリングを使用します。トレースする QMF 機能には文字を、各機能のトレース詳細レベルには数値を使用します。以下のように、各文字と数値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)

機能レベルでのトレースを使用する場合、文字ストリングで指定した機能のみがトレースされます。下の表に、各 QMF 機能に使用する文字を示します。

表 79. 機能レベル・トレース ID

| トレース ID | トレースされた 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、大域変数リスト、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ー) |
| R | 記憶管理機能 |
| U | ユーザー編集出口ルーチンや管理プログラム出口ルーチンなどの、ユーザー出口 |

例えば、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 プロファイル内の TRACE オプションの値は、QMF を DB2 for z/OS ストアード・プロシージャーとして開始すると無視されます。ただし、SET PROFILE (TRACE コマンドを *object-name* パラメーターで指定される初期プロシージャー内に含めて、L2-destination パラメーターが DSQDEDEBUG に設定されている限り、ストアード・プロシージャー・セッションの期間にトレースの詳細のレベルを変更できます。SET PROFILE コマンドを発行してトレース設定を変更するときに、メッセージおよびコマンドのトレースを継続する場合は、場合は、初期プロシージャー内で SET PROFILE (TRACE コマンドに指定されているオプション間に L2 があることを確認してください。

QMF セッション中にユーザーのアクティビティ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のレコードが書き込まれます。L2 を使用すると、すべての L1 レコードの他に、QMF コマンドの実行を記述する追加のレコードが書き込まれます。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。次の RUN QUERY コマンドの例は、ユーザーが表に存在しない列を指定したため、失敗します。

```

-----
***** 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 122. L2 トレース・コードを使用したユーザーのコマンドおよびメッセージのトレース

DSQDEDEBUG 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されます。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。例えば、前の例に示されたメッセージは、Column &V1 is not in table &V2 として DB2 QMF メッセージおよびコードに記載されています。この例の下半分は、メッセージの &V1 の値が DATE で、&V2 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・メッセージに関連するヘルプ・パネルの定義にも現れることがあります。必要に応じて、トレース・データの変数値を HELP 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。

関連タスク:

143 ページの『QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

165 ページの『QMF for TSO の QMF for Workstation からの開始とロジックを持つプロシージャの実行』

この例では、QMF for Workstation から、QMF for TSO をストアード・プロシージャとして開始します。続いて、デフォルト値を設定するロジックを持つプロシージャを実行し、大域変数値を検索して、照会を実行します。

個々の QMF モジュールのトレース

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。

重要: モジュール・レベルでのトレースは、IBM ソフトウェア・サポートのガイドンスのもとでのみ実行してください。

例えば、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、DSQUTRAC モジュールを使用してトレースしたいモジュールをリストします。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

ある特定のモジュールまたはモジュールのセットのみトレースするには、DSQEC_TRACE_MODULE 大域変数を使用する必要があります。この変数には、トレースするモジュールの名前を指定します (6 つまでの名前をコンマで区切って指定できます)。その後、次のコマンドを実行します。SET PROFILE (TRACE=ALL。これで、DSQEC_TRACE_MODULE に指定されているモジュールだけのトレースが開始されます。コマンド行で SET GLOBAL コマンドによってモジュールを指定する場合は、モジュール名を単一引用符で囲む必要があります。

QMF プロファイル内の TRACE オプションの値は、QMF を DB2 for z/OS ストアード・プロシージャとして開始すると無視されます。ただし、SET PROFILE (TRACE コマンドを *object-name* パラメーターで指定される初期プロシージャ内に含めて、*L2-destination* パラメーターが DSQDEBUG に設定されている限り、ストアード・プロシージャ・セッションの期間にトレースの詳細のレベルを変更でき

ます。SET PROFILE コマンドを発行してトレース設定を変更するときに、メッセージおよびコマンドのトレースを継続する場合は、場合は、初期プロシージャー内で SET PROFILE (TRACE コマンドに指定されているオプション間に L2 があることを確認してください。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

165 ページの『QMF for TSO の QMF for Workstation からの開始とロジックを持つプロシージャーの実行』

この例では、QMF for Workstation から、QMF for TSO をストアード・プロシージャーとして開始します。続いて、デフォルト値を設定するロジックを持つプロシージャーを実行し、大域変数値を検索して、照会を実行します。

QMF トレース・データの表示

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る必要があります。

QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして実行している場合は、印刷または表示のためにデータ・セットを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・データが長くなることがあります。

• TSO でのトレース・データの印刷または表示:

DSQDEBUG データ・セットは、TSO ログオン・プロシージャーを介して割り振ることができます。ただし、必要に応じて、DSQDEBUG を PRINT ファイルとして定義する次の例のようなステートメントを発行することによって、ファイルを印刷用に再割り振りすることができます。

```
FREE FILE(DSQDEBUG)
ATTR DEBUG RECFM( F B A) LRECL(121)
ALLOC DDNAME(DSQDEBUG) SYSOUT(A) USING(DEBUG)
```

これらのステートメントは、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を含むトレース・ファイルを割り振ります。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。

重要: DSQDEBUG 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。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 データ・セットを割り振る (または再割り振りする) こともできます。次のステートメント例は、最初のバイトが ANSI 紙送り制御文字の 81 文字固定長レコードを含むトレ

ス・ファイルを割り振ります。トレース情報は、ANSI 制御文字を含まない 80 文字が 1 行にフォーマット設定されます。

```
FREE FILE(DSQDEBUG)
ATTR DEBUG RECFM( F B A) LRECL(81)
ALLOC DDNAME(DSQDEBUG) DSNNAME(DEBUG.LIST) NEW KEEP
```

- CICS でのトレース・データの印刷または表示:

トレースは DSQDEBUG データ・セットに記録されます。このデータ・セットは、CICS 始動 JCL において割り振ってください。

QMF の初期化中に警告またはシステム・エラーが発生した場合、QMF のトレース・データ・セットを調べてエラーの原因を理解する必要があります。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。

関連概念:

454 ページの『トレース・データ・セットの割り振り』

QMF の実行方法に応じて、DSQUUDUMP データ・セットまたは DSQDEBUG データ・セットが割り振られていることを確認します。

関連タスク:

143 ページの『QMF ストアード・プロシージャ・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

QMF サービス・レベルの判別

ターゲット・ゾーンおよび配布ゾーンに対する SMP/E 報告書の実行は常に、サービス・レベルを判別するための最適な方法です。ただし、QMF サービス・レベルを他の方法で判別することもできます。

手順

QMF サービス・レベルは以下の方法で判別できます。

- SHOW コマンドを発行する

SHOW コマンドの SERVICE オプションは、指定したモジュールに関するサービス情報をメッセージ内に返します。例えば、SHOW SERVICE.DSQCSID0 は次のようなメッセージを返します。

```
OK, DSQCSID0 V11R2.00 09/09/16 13:39 WIM11299
```

- トレース・データを調べてサービス・レベルを入手する

トレース出力は、QMF インストール済み環境のサービス・レベルを入手するために役立ちます。サービス・レベル情報は、異常終了の発生時に自動的に表示されます。

異常終了以外の問題が発生している場合は、以下の方法で QMF サービス・レベルを入手できます。

1. トレースに値 ALL を指定することによって、トレースに最高のレベルの詳細を設定します。 QMF を呼び出すときに DSQSDEBUG パラメーターにこの値を設定するか、または次のコマンドを使用して Q.PROFILES 表に値を設定するために使用できます。 SET PROFILE (T=ALL
2. コマンド SET PROFILE (T=NONE を使用して、トレースを値 NONE にリセットします。
3. QMF を終了します。
4. DSQDEBUEG ファイルを調べます。

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 1 つ以上の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 ***
V10R1.00 10/10/25 12:00 PNnnnnn
```

APAR PNnnnnn は、サービスが適用された最新の APAR です。ここで、nnnnn 文字は数字を表します。

PNnnnnn の代わりにブランク値が表示されるか、WIMnnnnn と表示される場合は、モジュールが「基本レベル」であるか、サービスが適用されなかったか、または使用可能なサービスがありません。

- サービス・レベルを判別するためのバッチ・ジョブを実行する

QMF サービス・レベルを判別するには、以下のようなバッチ・ジョブを送信する方法もあります。

```
//ROBINK JOB (&SYSUID,090,T48,G249),ROBIN,
// NOTIFY=ROBIN,TIME=2,MSGCLASS=H,USER=&SYSUID
//AMBSTEP EXEC PGM=AMBLIST
//SYSPRINT DD SYSOUT=*
//DDDD1 DD DSN=QMFDEV.QMF1210.SDSQLOAD,DISP=SHR
//SYSLIB DD DUMMY
//SYSIN DD *
LISTIDR DDN=DDDD1,MODLIB
```

このジョブは、ロード・モジュール内の各 CSECT に適用された最新の PTF を示す報告書を生成します。

次のタスク

DSQPNLx ランタイム・パネル・ファイルに適切なレベルのサービスが適用されていることを確認してください。 x は言語 ID です。 PTF の適用後、同等のサービス更新が見落とされ、ランタイム・パネル・ファイルに適用されていない場合があります。 これにより、予測不能な動作が発生するおそれがあります。

DSQPNLx ランタイム・パネル・ファイルのサービス・レベルを確認するには、QMF ホーム・パネルを表示し、PF1 を押し、「QMF について」を選択します。 DSQPNLx のサービス・レベルが示されます。

DSQPNLx のサービス・レベルが最新でない場合は、最新 PTF の適用後に QMF1210.SDSQSAPE(DSQ1xNPL) を実行したかどうかをシステム管理者にお問い合わせください。

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。

このタスクについて

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 DSQSDEBUG がこのトレースを制御します。

手順

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

関連タスク:

221 ページの『トレース・オプションの設定』

QMF にはトレース機能が用意されており、QMF およびユーザーのアクティビティの追跡に役立ちます。

エラー・ログ報告書の検査

Q.ERROR_LOG 表は、リソースおよびその他の問題に関する情報をログに記録する QMF コントロール表です。

表の構造を以下に示します。

表 80. Q.ERROR_LOG 表の構造

| 列名 | データ・タイプと長さ | NULL の使用 | 機能 / 値 |
|-----------|---|----------|---|
| DATESTAMP | CHAR(8) | いいえ | エラーの発生した日付。形式は <i>yyyymmdd</i> です。 |
| TIMESTAMP | CHAR(5) | いいえ | エラーが発生した時刻。形式は <i>hh:mm</i> です。 <i>hh</i> は時、 <i>mm</i> は分です。 |
| USERID | VARCHAR(128) 例外: DB2 (VSE および VM 版) データベースに接続している場合、USERID 列のデータ・タイプおよび長さは CHAR(8) です。 | いいえ | QMF for TSO で、この列はエラーが発生しているユーザーの許可 ID または TSO ログオン ID を含みます。 QMF for CICS で、これはエラーが発生しているユーザーの CICS 端末 ID です。 QMF for TSO を DB2 for z/OS プロシージャとして実行しているときに発生したエラーの場合、この列は QMF が実行される WLM 管理のアドレス・スペースを開始したユーザーの許可 ID を含みます。 |
| 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。 |

表 80. Q.ERROR_LOG 表の構造 (続き)

| 列名 | データ・タイプと長さ | NULL の使用 | 機能 / 値 |
|---------|--------------|----------|--|
| 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を示されます。

問題の診断の参考にするために、次の例に示されているように、Q.ERROR_LOG 表を照会してエラーに関する情報を入手することができます。

```
SELECT TIMESTAMP, MSG_NO, MSGTEXT
FROM Q.ERROR_LOG
WHERE USERID = 'id'
AND DATESTAMP = 'date'
AND TIMESTAMP BETWEEN 'time1' AND 'time2'
ORDER BY TIMESTAMP, MSG_NO, MSGTEXT
```

図 123. 問題情報を得るためのエラー・ログの照会

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

関連タスク:

143 ページの『QMF ストアード・プロシージャー・インターフェースのインストール (TSO のみ)』

QMF バージョン 12.1 では、QMF for TSO を DB2 for z/OS ストアード・プロシージャーとして開始できます。ユーザーは、タスクを実行するために QMF for TSO にログオンする必要はありません。

環境に固有の診断の使用

TSO、z/OS、または CICS 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。以下に例

を示します。

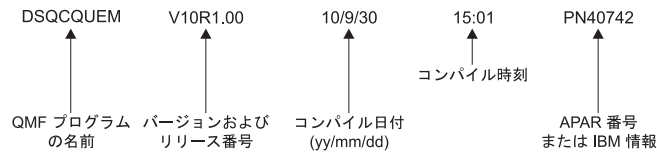


図 124. QMF プログラムを識別するスタンプの例

TSO における問題の診断

TSO での異常終了を診断するには、z/OS インフォメーション・センターの手順を使用することが必要になるか、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出口ルーチン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。

- ユーザー編集出口ルーチンによる問題の場合は、DXEECS、入力域、および結果域が出力に追加されます。
- 管理プログラムによる問題の場合は、DXEXCBA および DXEGOV が出力に追加されます。

CICS における問題の診断

QMF での異常終了を診断するには、CICS 資料に記載されている手順を使用する必要が生じることがあります。

別のプログラムが原因で QMF が異常終了した可能性があるため、トランザクションの CICS ダンプ内の必要な情報の多くは、CICS 資料にあるこれらの手順で見つけることができます。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。この情報で説明されている QMF 診断機能にエラーの原因についての十分な詳細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ちます。

- 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

時には、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に、CICS がユーザーのディスプレイ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。以下の例は、使用できる診断表示属性を示しています。この定義を使用すると、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 125. CICS 診断表示用の TYPETERM 指定

IBM への問題報告

技術情報などの自助オプションを使用しても回答や解決方法が見つからない場合は、IBM ソフトウェア・サポートにお問い合わせください。IBM ソフトウェア・サポートでは、製品の問題に関する支援の提供、よくある質問に対する回答、および製品の問題を解決するためのユーザー支援を行っています。

問題を IBM に報告する前に、IBM ソフトウェア・サポート Web サイトを調べて、その問題が既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 Software Support 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。

既に報告されている問題の検索

多くの場合、IBM 知識ベースを検索することにより、問題の解決方法を見つけることができます。

IBM ソフトウェア・サポート Web サイトを検索するには、問題を示す検索語で 1 つのストリングを作成します。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で始まります。

表 81. 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 | B5B |
| カナダ・フランス語 | B5G |
| デンマーク語 | B55 |
| 英語 | B10 |
| フランス語 | B56 |
| ドイツ語 | B57 |
| イタリア語 | B58 |
| 日本語 | B59 |
| 韓国語 | B5A |
| スペイン語 | B5C |
| スウェーデン語 | B5D |
| スイス・フランス語 | B5E |
| スイス・ドイツ語 | B5F |
| 大文字英語 | B51 |

IBM ソフトウェア・サポートでの作業

問題の診断がうまく進まず、使用可能な診断援助機能すべてを使用した場合は、IBM ソフトウェア・サポート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、ソフトウェア・サポート担当者から問題についてさらに詳細な情報を要求される場合があります。例えば、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ります場合があります。多くの場合、このような種類の情報はトレース機能を使用して得ることができます。

関連タスク:

454 ページの『トレース機能』

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。

第 3 部 付録

付録 A. 旧リリースでの変更の要約

QMF バージョン 12 リリース 1 は、バージョン 8.1 NFM、バージョン 9.1 NFM、バージョン 10.1、バージョン 11.1、およびバージョン 11.2 からの移行をサポートします。旧リリースの機能拡張が、QMF の移行計画に影響を与えることがあります。

移行元のリリースに関連するトピックを選択して、後続のリリースについてのトピックを読んでください。

関連概念:

113 ページの『第 6 章 移行およびフォールバックの考慮事項』

QMF バージョン 12.1 のインストール後またはバージョン 11 への移行後に、サイト固有の条件下で新しいインストール済み環境を十分にテストするために、QMF の旧リリースのオブジェクトにアクセスし、それらのオブジェクトを使用することが必要になる場合があります。

バージョン 11.2 での変更点

QMF バージョン 11 リリース 2 に対する機能強化は以下のとおりです。

拡張 QMF エディター

QMF Enhanced Editor 機能により、QMF SQL の照会および手順を作成し、まったく新しいレベルへと編集することが可能になります。SQL 照会に関しては、QMF Enhanced Editor は SQL 構文、予約語、関数、およびデータ・タイプのカスタマイズ可能な強調表示およびフォーマット設定を提供します。括弧検査も使用可能です。

照会支援機能は、表名の提案、列名とデータ・タイプの情報、および提案列値情報を提供します。プレビュー・ペインで SQL 結果を素早く確認できます。

QMF Enhanced Editor は、QMF 線形手順およびロジックを持つ手順に使用できます。照会オブジェクトと同様の強調表示も使用可能です。

QMF Enhanced Editor は、QMF コマンド行から、EDITOR キーワードを「EE」に設定した QMF EDIT コマンドを入力することによってアクセスします。新しい DSQEC_EDITOR 大域変数は、EDITOR キーワード指定のカスタマイズ可能なオーバーライドを提供します。

QMF フォルダーによる編成

QMF 照会、手順、書式、および分析を、フォルダーと呼ばれるグループに編成することが可能になりました。

LIST コマンド、SAVE コマンド、および ERASE コマンドが FOLDER キーワードを指定して作動するように更新されました。これにより、より明確な方法で QMF オブジェクトを表示および操作できるようになりました。SAVE コマンドを使用すれば、最初のオブジェクトを保管することによって自動的に新規フォルダーを作成

できます。RENAME コマンドは、FOLDER オブジェクトを名前変更するように、また QMF オブジェクトの名前変更時に FOLDER 参照を名前変更するように更新されました。

新しい DSQEC_CURR_FOLDER 大域変数を使用すると、自分専用の FOLDER キーワードの省略時値を定義することができます。

TSO 機能強化のための QMF 分析

3 つの新しいモデル、ウィルコクソン符号順位検定、マン・ホイットニー U 検定、および F 検定が、QMF Analytics for TSO フィーチャーの統計モデリング機能に追加されました。

解析的分析で使用する列の選択機能が、拡張データ・タイプのターゲティングおよび情報によって改善されました。列はモデル要件に基づいて示されます。モデルへの列入力を容易にするために、最大や最小などの値のサンプルを見ることができます。

ユーザー定義マッピング機能が QMF Analytics for TSO に追加されました。OpenGIS WKT マップ定義が DB2 表またはエクスポートされたデータ・セットで使用できます。これらは QMF Analytics for TSO によって読み取ることができ、ユーザー固有のマップをフォーマット設定します。アフリカ、北アメリカ、南アメリカ、およびドイツのマップが事前定義マップの既存ライブラリーに追加されました。

分析の保管が更新され、既存の分析オブジェクトのリストが可能になりました。

SQL 機能強化

SQL 照会の機能強化には、以下の変更点が含まれます。

- 以下の特殊レジスターを、QMF SQL 照会内から設定できるようになりました。

CURRENT APPLICATION COMPATIBILITY

動的 SQL の互換性レベル・サポートを指定します。

CURRENT GET_ACCEL_ARCHIVE

アクセラレーター・サーバーでアーカイブされた表を照会が参照する場合に、動的 SQL 照会でアーカイブ・データを使用するかどうかを指定します。

- 新しい DSQSFISO プログラム・パラメーター・オプションを使用すると、CHAR などのスカラー日時関数はデータベースの実際の日時形式を反映し、QMF の ISO フォーマット・プリコンパイラー・オプション (旧リリースで使用されていた形式) は反映しません。新しい DSQAO_DATE_FORMAT および DSQAO_TIME_FORMAT 大域変数は、データベースで使用される省略時の日時形式をユーザーに通知します。新しい DSQAO_DSQSFISO 大域変数には、DSQSFISO プログラム・パラメーター設定の値が含まれています。

QMF コマンドとコマンド動作

QMF コマンドおよびコマンド動作に対する機能強化には、以下の変更点が含まれます。

- 従来は QMF SAVE DATA コマンドを使用して行われていたデータベース表の保管プロセスが改善されました。これにより、パフォーマンスが向上し、記憶域の所要量が削減されました。RUN QUERY コマンドの新しい TABLE キーワードを使用すると、戻ってデータ・オブジェクトを完了する必要なしに、SAVE DATA を行うことが可能です。TABLE キーワードを指定した RUN QUERY は、ユーザーに報告書を返すことなくデータの検索と行の挿入の両方を行うために、完全にデータベース内で作動します。
- SPACE キーワードが、SAVE DATA、IMPORT TABLE、および RUN QUERY の各コマンドに追加されました (TABLE キーワードの指定時)。これにより、通常はプロファイルで設定する SPACE 指定をオーバーライドすることが可能になりました。ユーザーのプロファイルを更新することなく、特定のデータベースおよび表スペースをデータ保管のターゲットとすることができます。
- ACCELERATOR キーワードが、SAVE DATA、IMPORT TABLE、および RUN QUERY の各コマンドに追加されました (TABLE キーワードの指定時)。これにより、DB2 アクセラレーター専用表の作成が可能になりました。新しい DSQEC_SAV_ALLOWED 大域変数は、アクセラレーター、従来の表スペース、またはその両方を使用できるかどうかを制御します。ACCELERATOR キーワードの省略時値を提供する新しい DSQEC_SAV_ACCELNM 大域変数が追加されました。
- DB2 表または QMF オブジェクトを単一ステップで名前変更する機能を提供する、新しいコマンド RENAME が追加されました。
- QMF EDIT コマンドが更新されました。保管された照会および手順がデータベースから直接ロードされるようになり、照会および手順が事前に一時記憶域にある必要はなくなりました。
- 新しい DSQCP_RMV_BLANKS 大域変数を使用すると、QMF EDIT TABLE コマンドの使用時に VARCHAR および VARGRAPHIC の各列で末尾空白を保持することができます。関連する末尾空白を含む列は、QMF 表編集機能で編集後に保持されます。
- QMF 報告書の移出が、制御情報なしで行うことができるようになりました。EXPORT REPORT コマンドを、新しい値「TEXT」に設定された DATAFORMAT キーワードと併用して、フォーマット済み QMF 報告書を含む事前割り振りデータ・セットを取得することができます。報告書出力はテキスト添付に適しており、読みやすいものになります。

省略時の書式オーバーライド

以下の新しい大域変数を使用して、QMF の省略時書式編集コードをオーバーライドすることができます。

- DSQDC_EC_CHAR
- DSQDC_EC_NUM
- DSQDC_EC_DEC
- DSQDC_EC_DATE
- DSQDC_EC_TIME

省略時書式は、カスタマイズされた保管済みの書式を使用することなく、大域変数設定によって制御可能です。編集コード変更のみが存在する、専用化された書式の量を削減できます。

QUERY BY EXAMPLE (QBE) 機能拡張

QBE インターフェースが、128 文字の所有者名および表名と 30 バイトの列名をサポートするように更新されました。QBE オブジェクト用の DRAW、IMPORT、および EXPORT の各コマンドも、所有者名、表名、および列名の増加した長さをサポートするように更新されました。

アプリケーション・プログラミング・サポート

QMF ガバナー・インターフェース内のセッション取り消しトピックに関する資料が改善されました。

QMF の管理および保守容易性

DB2 QMF のこのリリースには、以下の保守容易性および管理の機能強化が含まれています。

- DSQEC_SAV_ALLOWED 大域変数により、管理者は、SAVE DATA、IMPORT TABLE、または RUN QUERY から TABLE の各コマンドの実装を制御することができます。この大域変数は、データを従来の表スペースに保管する、データをアクセラレーターに保管する、またはコマンドの使用をブロックする、のいずれが可能であるかを制御します。この大域変数によって、省略時値を指定するか、またはキーワード値で省略時値をオーバーライドすることができます。
- DSQEC_SPAC_OVERRIDE 大域変数により、管理者は、SPACE パラメーターを使用して SAVE DATA、IMPORT TABLE、または RUN QUERY から TABLE の各コマンドで Q.PROFILES.SPACE 値をオーバーライド可能かどうかを制御できます。
- QMF の手順および照会用の記憶域サイズが最適化され、QMF カタログ内の記憶域に最小スペース量が使用されるようになりました。
- 省略時のトレース状態に対して、改善された省略時トレース診断が作成されるようになりました。省略時トレースでの QMF リリースおよびデータベース接続の検出が容易になります。QMF トレース・サイズ制御が、新しい DSQEC_TRACE_LIMIT 大域変数によって追加されました。
- 新しい「SHOW SERVICE」コマンドは、サービス・レベル情報を表示します。
- DSQPNLE パネル・サービス・レベル情報を「QMF について」パネルから素早く判別できます。このパネルにはメイン・ヘルプ・システムからアクセスします。

これらの機能強化について詳しくは、QMF バージョン 11 リリース 2 の情報を参照してください (IBM Knowledge Center: <http://ibm.com/support/knowledgecenter/SS9UMF/welcome.html>)。

DSQ0EQ24 の削除

QMF.SDSQSAPE メンバー DSQ0EQ24 に入っていた QMF サンプル照会 SAMPLE_SYNONYM が、製品から削除されました。

バージョン 11.1 での変更点

QMF バージョン 11 リリース 1 に対する機能強化は以下のとおりです。

QMF Analytics for TSO

QMF Analytics for TSO を使用すれば、QMF for TSO から返される照会結果を分析することができます。QMF Analytics for TSO は QMF for TSO 内からのみ開始でき、統計分析、予測関数、および追加の図表タイプを、使いやすいメニュー方式のインターフェースからすべて提供します。

QMF Analytics for TSO では、「保存」ファンクション・キーを押すことで、図表または統計分析仕様を ANALYTIC オブジェクトとして保存することができます。その後、RUN QUERY コマンドで ANALYTICS オプションを指定することにより、保存された仕様をそのコマンドで使用できます。また、LIST、DISPLAY、および ERASE の各コマンドを使用して、ANALYTIC オブジェクトをリスト、表示、および消去することもできます。

改善された報告書記憶域管理

報告書記憶域管理に対する機能強化には、以下の変更点が含まれます。

- 記憶域割り振りの新たな最大量

DSQSBSTG パラメーターで指定可能な最大記憶域が 2GB に拡大されました。DBSQSBTG パラメーターには、固定値の代わりにパーセンテージも指定できるようになりました。このパーセンテージによって、報告書に使用する使用可能な記憶域の最大パーセンテージを指定します。

- 記憶域の動的処理

旧リリースの QMF では、報告書に関する記憶域要件は、最初の報告書が返された後で、セッション中に 1 回のみ見積もられていました。今回のリリースでは、QMF はセッション全体を通して、新しい報告書が要求されたときに、動的に記憶域要件を見積もり、管理します。

- 記憶域設定を決定するための新しい大域変数

4 つの新しい大域変数 DSQAO_DSQSBSTG、DSQAO_DSQSRSTG、DSQAO_DSQSPILL、DSQAO_DSQSPTYP には、QMF 初期化に使用された記憶域管理パラメーターの値が含まれています。これらの大域変数に保管された情報は、管理者が記憶域管理およびトラブルシューティングを行う際に役立ちます。

ラージ・データ・タイプの機能強化

ラージ・データ・タイプに対する機能強化には、以下の変更点が含まれます。

- LOB データ拡張

– QMF では、DISPLAY コマンドおよび SELECT 照会に対して最大 2 GB の CLOB データおよび BLOB データと、最大 1 GB の DBCLOB データが返されるようになりました。これらのデータ・タイプの全データをその最大まで表示するには、QMF 書式での列幅を最大幅に設定してください。CLOB データおよび BLOB データの最大幅は 32767、DBCLOB データの最大幅は 16383 です。次に、データ・タイプに適した折り返し編集コード (CW、BW、または GW) を使用します。

- QMF の IMPORT、EXPORT、および SAVE DATA の各コマンドを使用して、LOB データをインポート、エクスポート、および保存できるようになりました。
- LOB データの保存および検索の設定を、以下の新しい大域変数によって制御できるようになりました。
 - DSQEC_LOB_RETRV を使用して、QMF コマンド (DISPLAY TABLE、EXPORT TABLE、EXPORT DATA、SAVE DATA、PRINT TABLE、PRINT REPORT、および RUN QUERY) による LOB データの取得方法および表示方法を制御できます。
 - DSQEC_LOB_SAVE を使用して、ユーザーが SAVE DATA コマンドまたは IMPORT TABLE コマンドによって LOB データを保存できるかどうかを制御できます。
 - DSQEC_LOB_COLMAX を使用して、取得する LOB データの最大サイズ (キロバイト単位、最大 2 GB) を指定できます。DSQEC_LOB_COLMAX を有効にするには、大域変数 DSQEC_LOB_RETRV を 1 または 3 に設定する必要があります。DSQEC_LOB_COLMAX で指定された最大より大きい LOB 列を含むデータをユーザーが処理しようとする、エラーが出され、報告書データは表示されません。
- QMF では、分散作業単位環境内のリモート・ロケーションにある LOB データにアクセスするために、表および視点の 3 部構成の名前がサポートされるようになりました。3 部構成の名前を使用できるようにするには、DSQEC_LOB_RETRV 大域変数を 2 または 3 に設定する必要があります。
- XML データ拡張
 - 旧バージョンの QMF では、XML データ形式のデータに対する IMPORT コマンドの発行時に、XML 文書全体がバッファ記憶域に読み込まれていました。QMF バージョン 11 では、一度に 1 行をバッファ記憶域に読み込むことができるようになりました。この機能強化により、記憶域の使用量が削減され、インポートされたデータのサイズに応じて XML データ形式のデータの処理効率が向上しました。
 - QMF では、DB2 for Linux, UNIX, and Windows バージョン 9.5 以降、または DB2 for iSeries バージョン 7 以降への接続時に、XML データを使用する DISPLAY、EXPORT、IMPORT、SAVE、SELECT、および RUN の各操作がサポートされるようになりました。
- LOB データおよび XML データのフェッチのための拡張記憶域サポート

旧リリースでは、LOB データおよび XML データのフェッチに 2 GB 境界より下の記憶域が使用されていました。今回のリリースでは、QMF がこのデータ・タイプでサポートする最大サイズが増加しているため、2 GB 境界より下の記憶域が大量に使用される可能性があります。代わりに、LOB データおよび XML データがフェッチされると、必要になるまで 2 GB 境界より上の記憶域に保管されるようになりました。必要になった時点で、データは 2 GB 境界より下の記憶域に移動され、不要になると 2 GB 境界より下の記憶域から削除されます。

テンポラル・データの幅広いサポート

QMF バージョン 10 には、SQL 照会を使用したテンポラル・データの選択、作成、および更新のためのサポートが組み込まれています。テンポラル・データは、以下のコマンドおよびオブジェクトでもサポートされるようになりました。

指示照会

「指示照会」の「指定」パネルに、テンポラル表またはテンポラル視点の照会に時間枠指定を追加するために使用できる、「時間枠」オプションが加わりました。

LIST コマンド

LIST コマンドで、テンポラル表と履歴表の両方がリストされるようになりました。「表リスト」パネルで DESCRIBE を使用すると、表のサブタイプを含め、表に関する情報が表示されます。「履歴表」が有効なサブタイプになりました。DB2 for Linux, UNIX, and Windows では、履歴表は「表」サブタイプで表示されます。

EXPORT コマンド

テンポラル表は EXPORT TABLE コマンドではデータベースから直接エクスポートできませんが、EXPORT DATA コマンドを使用すると、テンポラル・データを QMF 一時記憶域からエクスポートすることができます。エクスポートされたテンポラル・データは、履歴データとは関連がありません。

IMPORT コマンドおよび SAVE コマンド

ACTION=APPEND または ACTION=REPLACE を指定した IMPORT コマンドおよび SAVE コマンドを使用して、既存のテンポラル・データを付加または置換することができます。ただし、これらのコマンドを使用して新しいテンポラル表を作成することはできません。新しい非テンポラル表は、「IMPORT TABLE *newtable*」および「SAVE DATA AS *newtable*」によって作成されます。

SQL 照会の機能強化

SQL 照会の機能強化には、以下の変更点が含まれます。

- 作業単位を制御するために、複数の SQL ステートメントを含む照会に、COMMIT ステートメントを組み込むことができるようになりました。確認プロンプトを有効にするためにユーザーの QMF プロファイルで CONFIRM オプションが YES に設定されていると、照会内の各 COMMIT ステートメントに対してプロンプトが表示されます。
- 複数のステートメントを含む SQL 照会に、1 つの SELECT ステートメントを組み込むことができるようになりました。旧リリースでは、SELECT ステートメントを含む SQL 照会には他のステートメントを入れることはできませんでした。
- 確認プロンプトが有効になっている場合は、組み込まれた INSERT、DELETE、MERGE、または UPDATE の各ステートメントによってデータベースを変更する SELECT ステートメントに対してプロンプトが表示されます。このタイプのデータベース変更についての確認プロンプトは、このタイプ

の SELECT ステートメントをサポートするすべてのデータベースでサポートされています。確認プロンプトは、以前はどの SELECT ステートメントについても表示されませんでした。

- QMF の今回のリリースでは、DB2 特殊レジスターの追加サポートが提供されています。
 - 新しい大域変数 DSQEC_EXPL_MODE を使用して、CURRENT EXPLAIN MODE 特殊レジスターを設定できます。この特殊レジスターは、RUN QUERY コマンドによって実行される動的 SQL ステートメントの EXPLAIN 動作を制御します。
 - 以下の特殊レジスターを QMF SQL 照会内から設定できます。

CURRENT MAINTAINED TABLE TYPES FOR OPTIMIZATION

動的 SQL 照会の最適化プロセス中に、最適化が有効になっているもののマテリアライズ照会表が考慮されるかを制御します。

CURRENT QUERY ACCELERATION

照会に対して DB2 オプティマイザーによる照会加速を考慮するかどうかを指定します。

CURRENT TEMPORAL BUSINESS_TIME

アプリケーション期間テンポラル表に対する参照用に、デフォルト BUSINESS_TIME 期間指定で使用される TIMESTAMP(12) 値を指定します。

CURRENT TEMPORAL SYSTEM_TIME

システム期間テンポラル表に対する参照用に、デフォルト SYSTEM_TIME 期間指定で使用される TIMESTAMP(12) 値を指定します。

- 新しい大域変数 DSQAO_DSQSMRFI を使用して、DSQSMRFI プログラム・パラメーターの値を検査できます。
- QMF では、NULL 可能 ROWID SQLTYPE 905 を扱うようになりました。これは、LEFT OUTER JOIN が照会に含まれている場合に、ある状況下で DB2 から返されるものです。
- QMF ユーザーは、DISTINCT データ・タイプを参照するストアード・プロシージャを呼び出すことができるようになりました。
- QMF のこのリリースでは、新しい DSQEC_KEEP_THREAD 大域変数が 1 に設定されている場合、SET ステートメントによる DB2 大域変数の設定がサポートされています。DB2 大域変数は、SQL ステートメントによってアクセスおよび変更が可能な名前付きメモリー変数です。DB2 大域変数は、特定のアプリケーション・コンテキストに関連付けられ、そのアプリケーション・スコープに固有な値を含んでいます。大域変数は作成後、その定義がさまざまな DB2 接続の間で共有されます。そのため、大域変数の有効範囲は、DB2 特殊レジスターのそれと似ています。ただし、各接続は、各大域変数の独自のインスタンスを維持しているため、変数の内容は同じ接続内の SQL ステートメント間でのみ共有されます。
- ROWLIMIT 処理は QMF ではなく DB2 によって行われるようになりました。この変更により、ユーザーが複雑な照会のために ROWLIMIT を指定した RUN

QUERY を実行する際に、QMF でのパフォーマンスが向上しています。この機能強化では特に、照会内に ORDER BY 節を持つ大きな結果セットのパフォーマンスが向上します。

QMF コマンドとコマンド動作

QMF コマンドおよびコマンド動作に対する機能強化には、以下の変更点が含まれます。

- EXPORT 操作に対する機能強化
 - 今回のリリースでは、CSV (コンマ区切り値) 形式でデータをエクスポートする機能が追加されています。エクスポートされたデータは、その後ご使用のワークステーションにダウンロードできます。このワークステーションでは、CSV 形式のデータを Microsoft Excel などのアプリケーションで使用することができます。
 - 旧リリースでは、SQL 照会とプロシージャの各オブジェクトを、LRECL が 79 バイトのデータ・セットにエクスポートする必要がありました。これらのオブジェクトは、LRECL が 79 バイトから 32,760 バイトの新規および既存のデータ・セットにエクスポートできるようになり、既存の DFSMS DATA クラスを使用できます。新しい大域変数 DSQEC_DSLRECL1 を使用して、SQL 照会または QMF プロシージャの新規データ・セットへのエクスポート時に使用するレコード・サイズを設定できます。
 - QMF では、EXPORT コマンドの結果を変更するために使用するデータ・セットの 1 次トラック数および 2 次トラック数を設定できます。旧リリースでは、PS および PDS のデータ・セットに指定できるトラック数は最大 65,535 でした。この制限が 16,777,215 トラックまで緩和されました。この割り振りは、QMF 大域変数 DSQEC_DSALLOC_PRI の設定によって制御されます。また、このサポートにより、QMF は拡張アドレス・ボリューム (EAV) を使用して作動することが可能になります。
- QMF カタログの Q.OBJECT_DIRECTORY 表でのオブジェクトの LAST_USED タイム・スタンプを設定するための制御の増加

DSQEC_LAST_RUN 大域変数に、タイム・スタンプの更新を RUN コマンドのみに限定するオプションが追加されました。この変更をサポートするために、DSQUOPTS 初期化ルーチンも更新されました。

- DPRE コマンド同義語の改善

DPRE は、QMF for TSO での印刷プレビューを提供するコマンド同義語です。旧リリースでは、同義語は QMF プロシージャを実行し、これにより同義語の発行時に一時記憶域にあったプロシージャ・オブジェクトはすべて置き換えられていました。同義語定義および関連付けられたコードは、REXX exec として実行されるようになりました。これにより、最新のプロシージャ・オブジェクトが一時記憶域に残ります。

- 新しい SHOW ANALYTICS コマンド

このコマンドは、新しい QMF Analytics for TSO 製品へのアクセスを提供します。475 ページの『QMF Analytics for TSO』に詳細が記載されています。

アプリケーション・プログラミング・サポート

旧リリースでは、QMF は固定小数点または 10 進数の桁あふれを起こす可能性のある演算での制御を得るために、z/OS プログラム・マスクを設定していました。QMF がプログラム・マスクを設定したままにすると、他のアプリケーションで問題が発生するため、以下のサービスまたはコマンドの呼び出し前に、現行マスクは保存され、ゼロに設定されるようになりました。

- QMF 管理プログラム出口ルーチン
- REXX を必要とするロジックを持つ書式またはプロシージャのための REXX インターフェース
- EDIT PROC コマンドおよび EDIT QUERY コマンド
- TSO コマンド、および ISPF を必要とするコマンド

また、プログラムが QMF を呼び出すと、QMF は呼び出し側のプログラム・マスクを保存し、QMF で使用するためにプログラム・マスクを初期化します。QMF が制御を呼び出し側に返すと、呼び出し側のプログラム・マスクが復元されます。

呼び出し可能インターフェース・アプリケーションの機能強化

REXX で作成されたアプリケーションを除く、呼び出し可能インターフェースを使用するアプリケーションで、トレース情報を追加するために新しい TRACE コマンドが使用できるようになりました。

大域変数管理の機能強化

QMF 大域変数管理に対する機能強化には、以下の変更点が含まれます。

- 初期大域変数設定の制御

管理者は、新しい Q.GLOBAL_VARS 表に大域変数用の初期値を設定し、それらの値をユーザーが更新可能かどうかを指定できます。この表は、QMF インストール中に空でインストールされ、QMF 初期化時に読み取られます。

- セッション間での大域変数値の保存

新しい大域変数 DSQEC_USERGLV_SAV は、ユーザー大域変数および「DSQ」で始まる大域変数の値をセッション間で保存するかどうかを制御します。保存する値は Q.GLOBAL_VARS 表に格納され、ユーザー ID に関連付けられます。この大域変数について詳しくは、QMF 解説書を参照してください。

- ユーザー入力の保存

新しい大域変数 DSQEC_SESSGLV_SAV は、特定のパネルの特定のデータ入力フィールドのユーザー入力を、QMF セッション内で、および QMF セッション間で保存するかどうかを制御します。この大域変数について詳しくは、QMF 解説書を参照してください。ユーザー入力は、DXY 接頭部の付いた新しい大域変数を使用して保存されます。

EAV DASD のサポート

z/OS バージョン 1 リリース 11 では、拡張形式の順次データ・セットおよび VSAM データ・セット用に拡張アドレス・ボリューム (EAV) サポートが導入され

ました。QMF バージョン 11 は、これらのボリュームをサポートするために拡張されました。

スレッド管理

新しい QMF 大域変数 DSQEC_KEEP_THREAD は、QMF セッションが終了するまで、QMF で使用する DB2 スレッドを保持する機能を提供します。旧リリースの QMF では、QMF で使用された各 DB2 スレッドは、各照会の終了時に削除されていました。

バージョン 10.1 での変更点

QMF バージョン 10.1 の機能拡張は以下のとおりです。

QMF for TSO へのストアード・プロシージャ・インターフェース

新しいインターフェースでは、QMF for TSO を DB2 for z/OS ストアード・プロシージャとして開始することができます。この新しい機能によって、DB2 for z/OS ストアード・プロシージャを呼び出すことのできるすべてのソフトウェア・プログラムは QMF for TSO を開始できるようになり、最大 20 の結果セットで報告書を受け取ることができます。

照会の機能拡張

照会の機能拡張には次のものが含まれます。

- 単一 SQL 照会における複数 SQL ステートメントのサポート

各 SQL 照会には、データベース保守に関連する複数のステートメント (UPDATE、INSERT、ALTER、CREATE、DROP、EXPLAIN など) を含むことができるようになりました。SELECT または CALL など、結果を返すステートメントは、CREATE PROCEDURE ステートメントと同様に、1 つの SQL 照会につき 1 つのステートメントに制限されたままです。

新しい大域変数 DSQEC_RUN_MQ を使用すると、複数の SQL ステートメントを許可するかどうかを設定できます。複数のステートメントが許可される場合、その照会内のすべてのステートメントは単一の作業単位として発行されます。照会内の 1 つのステートメントが失敗すると、失敗したステートメントの前のステートメントによって行われたデータベース変更はロール・バックされ、後続のステートメントは実行されません。SET などのいくつかのステートメントは、QMF セッションの各局面に適用されるため、失敗が発生した場合でもロール・バックされません。

置換変数の値および確認プロンプトに対する応答は、照会内のすべての SQL ステートメントに適用されます。

- 長さが 32 KB を超える QMF SQL 照会の実行のサポート

RUN QUERY コマンドで実行可能な SQL 照会の最大サイズは、DB2 for z/OS に送信される SQL 照会の場合は 2 MB に、DB2 for iSeries または DB2 for Linux, UNIX, and Windows に送信される SQL 照会の場合は 65 KB に増加さ

れました。この機能は、DSQEC_SQLQRYSZ_2M 大域変数を設定することによってアクティブ化できます。DB2 (VSE および VM 版) に送信される SQL 照会は 8 KB に制限されたままです。

QMF では、指示照会および QBE 照会の場合、引き続き 32 KB の照会サイズがサポートされます。これは、RUN QUERY コマンドの送信先となるデータベースでこのサイズの SQL ステートメントがサポートされていない場合を除きます。

報告書のパフォーマンスおよびリソース制御の改善

報告書のパフォーマンスおよびリソース制御の機能拡張には次のものが含まれます。

- QMF for TSO での予備データの 64 ビット仮想記憶域

新しい 64 ビット仮想記憶域では、アクティブな記憶域で不要になった QMF for TSO データを扱います。この新しい記憶域によって TSO にスピル・ファイルを割り振る必要がなくなり、XML などの大量の記憶域を必要とするデータ・タイプに対する幅広いサポートが可能となりました。

新しい TSO プログラム・パラメーター DSQSPTYP は、このサポートをアクティブにします。新しい大域変数 DSQEC_EXTND_STG を使用すると、QMF で取得する拡張記憶域の量を指定できます。

- QMF 報告書におけるデータ行の最大長の拡大

旧リリースの QMF for TSO/CICS では、QMF 報告書で返される 1 データ行の長さには 32 KB の制限がありました。新しい大域変数 DSQEC_TWO_GB_ROW では、以下の制限付きで最大 2 GB の行の長さのサポートを設定できます。

- レコードの最大サイズがページ・サイズより大きい表を作成することはできません。DB2 は、4 KB、8 KB、16 KB、または 32 KB のサイズのページ内にレコードを保存します。したがって、1 つの表からのみデータを取り出す場合、行の長さの制限は 32 KB のままです。
- 表に LOB データを持つ列が含まれている場合、LOB 列のデータは表示および印刷の操作のときだけでなく、照会結果でも 32 KB で切り捨てられます。ただし、LOB データを含む行の長さは最大 2 GB のままです。

データを QMF、IXF、または HTML 形式でエクスポートまたはインポートする場合、行の最大長は 32 KB のままです。しかし、XML 形式を使用して文字データをエクスポートまたはインポートすることができ、この形式であれば、最大 2 GB のレコード長をサポートします。

- ユーザー・コマンドによって生成されたデータベース活動の独立管理

QMF パッケージが再構築され、エンド・ユーザー・コマンドによるデータベース・アクティビティを処理するモジュールが、QMF 内部の SQL によるデータベース・アクティビティを処理するモジュールから分離されました。この新しい構造化を使用すると、QMF コマンドの下記グループを DB2 リソース限定機能による個々の管理ごとに分離することができます。

- RUN QUERY (任意のタイプの SELECT 照会)、DISPLAY TABLE (CONFIRM オプションが YES の場合)、EXPORT TABLE、PRINT TABLE、BOTTOM、TOP、FORWARD、BACKWARD、RIGHT、および LEFT
- SAVE DATA および IMPORT TABLE
- RUN QUERY (照会に INSERT、UPDATE、または DELETE ステートメントを含む場合)
- EDIT TABLE (ADD モード)
- EDIT TABLE (SAVE パラメーターが IMMEDIATE に設定されている場合の CHANGE モード)
- EDIT TABLE (SAVE パラメーターが END に設定されている場合の CHANGE モード)
- ERASE
- QMF 内から同時アクセス解決オプションを設定する機能

この機能拡張によって DSQEC_CON_ACC_RES 大域変数が追加されました。この変数を使用して、QMF がアクセスしようとしているデータに対するロックを解決する方法を指定できます。新しい DB2 バージョン 10 バインド・オプションの CONCURRENTACCESSRESOLUTION の値が、既存の SKIP LOCKED DATA オプションと同様に、この大域変数を介してサポートされます。

QMF コマンドおよび SQL ステートメントのサポートに対する機能拡張

QMF コマンドおよび SQL ステートメントのサポートの機能拡張には、次のものがあります。

- DISPLAY TABLE コマンドのオプションの確認プロンプト

QMF で、DISPLAY TABLE コマンド用の CONFIRM パラメーターが提供されるようになりました。CONFIRM が YES に設定されていて、コマンドを完了するための見積もりリソースが DB2 リソース限定機能で定義されている割り振り済みリソースを超えると、QMF はコマンドを取り消すかどうかを決定するようユーザーに求めるパネルを表示します。

- SAVE および IMPORT コマンドの暗黙的キャストのサポート

SAVE または IMPORT コマンドの送信先となるデータベースで暗黙的キャストのサポートが提供されている場合、QMF では、保存またはインポートするデータ内の列のデータ・タイプと長さが、置換または追加する既存の表内の列と同じである必要はなくなりました。

- QMF LIST コマンドの OWNER パラメーターのデフォルト (オプション)

この機能拡張によって新しい大域変数 DSQEC_LIST_OWNER が追加されました。この変数を使用すると、LIST コマンドの OWNER パラメーターにデフォルトのデータベース許可 ID を指定でき、コマンドが発行されるたびにこのパラメーターを指定する必要がなくなります。この変数には任意の有効なデータベース許可 ID を設定するか、ALL を指定して、所有者に関係なくアクセスが許可されているすべてのオブジェクトをリストすることができます。変数に何も値が

指定されていない場合、以前の QMF リリースと同様に、デフォルトでは OWNER パラメーターは現在のデータベース許可 ID になります。

- オブジェクトの「最終使用」タイム・スタンプを更新するコマンドを制限する機能

以前のリリースでは、QMF カタログの Q.OBJECT_DIRECTORY 表の LAST_USED 列のタイム・スタンプは、以下のどのコマンドを発行した場合もオブジェクトが参照されるたびに更新されていました。

- CONVERT
- DISPLAY
- EXPORT
- IMPORT
- LAYOUT
- PRINT
- RUN
- SAVE

このタイム・スタンプは、LIST コマンドによって生成される QMF データベース・オブジェクト・リストの「最終使用日」フィールドに表示されます。この機能によって新しい大域変数 DSQEC_LAST_RUN が提供されます。これを使用するとタイム・スタンプの更新動作をデフォルトで以前のリリースと同じにするか、または RUN、SAVE、および IMPORT コマンドによってのみタイム・スタンプが更新されるように制限できます。

- SQL CALL ステートメントの追加パラメーター

この機能拡張によって、QMF では SQL CALL ステートメントで指定できるパラメーター数が 32 から 63 に増加しました。

- DB2 特殊レジスターの追加サポート

この機能拡張によって、QMF SQL 照会で SET ステートメントを使用して DB2 for z/OS の CURRENT SCHEMA および CURRENT REFRESH AGE の特殊レジスターをそれぞれ設定できるようになりました。

幅広いデータ・タイプのサポート

DB2 データ型に対する QMF サポートの機能拡張には次のものがあります。

- XML データのサポートの拡大

XML データを含む行は最大 2 GB までデータを保持できるようになりました。この新しい制限は、XML 形式でデータをインポートまたはエクスポートする場合にもサポートされます。

- 10 進浮動小数点データのサポートの拡大

QMF は、10 進浮動小数点 (DECFLOAT) データ・タイプに対して完全サポートを提供するよう拡張されました。ただし、QMF を実行中のプロセッサで 10 進浮動小数点命令がサポートされていることが前提となります。QMF は、DECFLOAT データ・タイプの長形式および拡張形式をサポートします。QMF を実行中のプロセッサで 10 進浮動小数点の命令をサポートしていない場合、デフォルトではデータに使用される編集コードは M となり、報告書にはメタデ

ータが表示されます。例えば、長形式の値の代わりに DECFLOAT(16) が表示され、拡張形式の値の代わりに DECFLOAT(34) が表示されます。

- **TIMESTAMP** データ・タイプのより高い精度のサポート

この機能拡張によって、TIMESTAMP データ・タイプに対して DB2 for z/OS バージョン 10 で許可されている高い精度がサポートされます。小数秒を表す桁数が 0 から 12 になっています。

- **新しい TIMESTAMP WITH TIME ZONE** データ・タイプのサポート

この機能拡張は、新しい DB2 バージョン 10 のデータ・タイプ **TIMESTAMP WITH TIME ZONE** に対する QMF サポートを提供します。この新しいデータ・タイプは、TIMESTAMP データ・タイプにタイム・ゾーン情報を含めるよう拡張されています。タイム・ゾーンは、現地時間と協定世界時 (UTC) との差異を時間および分で示したものです。この UTC は、以前はグリニッジ標準時 (GMT) と呼ばれていました。

QMF 書式の改善

この機能拡張によって新しい大域変数 **DSQDC_COL_LABELS** が提供されます。これは QMF フォームで示される列見出しをデフォルトで、列に割り当てられているデータベース・ラベルに設定するかまたは選択された表の列名に設定するかどうか制御できます。デフォルトではデータベース・ラベルが使用されます。

診断機能の改善

データベース・マネージャーが発行した SQL コードに関連した QMF エラーに対して、追加情報が提供されるようになりました。このようなエラーに対して以下の機能が提供されます。

- エラーについて説明する QMF メッセージ・ヘルプ・パネルの最後に、SQL 連絡域 (SQLCA) が表示されるようになりました。
- 新しい大域変数 **DSQDC_POS_SQLCODE** を使用すると、さまざまな詳細レベルで、正の SQL コードの通知を設定できます。

バージョン 9.1 での変更点

QMF バージョン 9.1 の機能拡張は以下のとおりです。

データベース処理の機能拡張

DSQSMRFI プログラム・パラメーターは、DB2 複数行のフェッチおよび挿入処理を利用できます。この変更の以前は、処理するデータ行ごとに SQL FETCH あるいは INSERT ステートメントが必要でした。複数行の処理では、単一の SQL ステートメントで複数のデータ行の処理が可能です。分散データベースの場合は、この改良によりネットワーク・トラフィックが減少します。QMF は、アクセス対象のデータベースでも複数行のフェッチおよび挿入をサポートしている場合に、複数行処理のサポートを使用できます。以下の QMF コマンドは、複数行サポートを使用します。

BOTTOM
FORWARD
DISPLAY TABLE

DPRE
EXPORT DATA
EXPORT TABLE
PRINT REPORT
PRINT TABLE
RUN QUERY
SAVE DATA.

DB2 for z/OS バージョン 9 データ・タイプのサポート

- 大整数

BIGINT は、63 ビット整数を表示できる数値データ型です。BIGINT でサポートされる数値の範囲は、-9223372036854775808 から +9223372036854775807 までです。

- 固定長および可変長バイナリー

QMF は、バージョン 9 以前に既にサポートされている BLOB データ・タイプと互換性のある BINARY および VARBINARY データ・タイプをサポートします。

- XML

リレーショナル表の列に、XML 文書を含めることができます。DB2 の XML 列には、構造上の制限や、長さの規定はありません。バージョン 9 では、QMF for TSO/CICS には、XML 文書のインスタンスごとに 32 KB のサイズ制限があります。XML 文書の文字データは、UTF-8 形式でエンコードされています。QMF は、XML データをエクスポートする場合、この形式を維持します。XML データは、表示または印刷される報告書での表示用に文字データに変換されます。

QMF では、XML データ・タイプ対応のデータベース・リリースに接続している場合にのみ、XML データを使用した操作がサポートされます。

- 10 進数浮動小数点

QMF が実行しているプロセッサが 10 進数浮動小数点指示をサポートしている場合は、10 進数浮動小数点データ・タイプは QMF バージョン 9 内で QMF for WebSphere および QMF for TSO/CICS で完全にサポートされます。プロセッサが 10 進数浮動小数点をサポートしていない場合は、QMF for TSO/CICS Version 9 は、10 進数浮動小数点データを含む表を表示するときに、メタデータ (M) 編集コードを使用できるようにして、10 進数浮動小数点データへのアクセスを許容します。

インポート/エクスポートの機能拡張

- XML データをインポートおよびエクスポートする機能

QMF には、DB2 表を HTML 表にマップするサンプル・スタイル・シートが用意されています。XML データを z/OS UNIX パスに直接エクスポートしたり、z/OS UNIX パスから XML データをインポートしたりすることができます。

- UNIX パス名を使用してデータをエクスポートおよびインポートする機能

このフィーチャーには、z/OS UNIX パス名を使用してデータにアクセスし、Web サーバーから XML データをインポートする機能のみならず、HTML 形式の QMF 報告書をエクスポートする機能、あるいは XML データを Web サーバーにエクスポートする機能もあります。UNIX パスは、階層ファイル・システム (HFS) のファイル名、あるいは、ネットワーク・ファイル・システム (NFS) のファイル名にマップすることができます。

- PDSE データ・セットを使用して QMF オブジェクトをエクスポートおよびインポートする機能

この機能拡張により、PDSE データ・セットを使用して、QMF オブジェクトをエクスポートおよびインポートすることができます。また、QMF オブジェクトを新規データ・セットにエクスポートするときに、PDSE データ・セットを新規に作成することもできるようになりました。PDSE データ・セットの割り振りは、DSQEC_PO 大域変数によって制御されます。

- データ・セット・サイズの簡単な指定

この機能拡張により、一連の QMF 大域変数を使用して、DB2 QMF EXPORT コマンドで使用するデータ・セットのサイズを指定できます。この変数は、DSQEC_DSALLOC_PRI、DSQEC_DSALLOC_SEC、および DSQEC_DSALLOC_DIR です。

セキュリティの機能拡張

QMF CONNECT コマンドは、RACF 大/小文字混合パスワードの使用をサポートします。

診断機能

この機能拡張は、DB2 for z/OS バージョン 9 での新しい SQL コード用の診断サポートを提供します。

バージョン 8.1 での変更点

QMF バージョン 8.1 の機能拡張は以下のとおりです。

インストール

インストール・プロセスは、QMF バージョン 8.1 で改善され、実行するポスト SMP/E 作業が少なくなっています。

データベース操作

- 許可 ID、現行 SQL ID、および表名で、128 文字までの長さがサポートされます。列名は、30 文字までの長さにするすることができます。サポートは、データベースで許される長さに基づいています。

サポートには、QMF ダイアログ・パネルにおけるより大きなデータ入力フィールドおよび名前の表示が含まれます。

- CALL ステートメントは、DB2 ストアード・プロシージャを実行するために、SQL 照会パネルから発行することができます。出力パラメーターは、ユーザーによって定義された QMF 置換変数内に入れられます。戻される結果セットは、照会から戻される結果セットのすべてのフィーチャーを持っています。

- 新しい大域変数 DSQEC_SP_RS_NUM は、複数の結果セット戻す DB2 ストアード・プロシージャの特定の結果セットを指定できるようにします。
- BLOB、CLOB、および DBCLOB データ・タイプのための完全に統合されたサポート。LOB データは、一時記憶域またはデータベースから、QMF 報告書に表示したり、印刷することができますが、32 KB で切り捨てられます。

M 編集コードは、デフォルトですべての LOB 列に割り当てられます。この編集コードは、実際のデータの代わりに、関連付けられている列メタデータを表示します。データ・タイプによっては、他の編集コードも使用できます。

- z/OS バージョン 8 SQL コードに対する複数の新しい DB2 UDB のサポート。
- QMF バージョン 8.1 では、DB2 UDB (z/OS 版) データベースのみがリクエスターとして機能できます。また、QMF ランタイム・ライブラリーは、z/OS 用のみに提供されており、VM または VSE 用はありません。これは、QMF は DB2 UDB (z/OS 版) データベース内でのみ始動できることを意味します。DB2 UDB (z/OS 版) データベース上で稼働している QMF バージョン 8.1 は、z/OS が稼働していないサーバーに、QMF バージョン 8.1 がインストールされていれば引き続き接続できます。

QMF コマンド

- 新しい PRINT コマンド・オプションにより、印刷装置名が指定されない場合に、報告書出力書式内の紙送り制御文字の抑止を可能にします。

DSQEC_CC 大域変数は、PRINT コマンドでの CC (紙送り制御) キーワードの設定に柔軟性を提供します。

- QMF バージョン 8.1 では、SET GLOBAL および GET GLOBAL コマンドで指定される大域変数のサイズが 55 文字から 2000 文字に拡張されました。

バージョン 7.2 での変更点

QMF バージョン 7.2 の機能拡張は以下のとおりです。

インストール

- DB2 UDB (OS/390® 版) および (z/OS 版) 環境には、ユーザー出口ライブラリー QMF720.SDSQEXIT が含まれていました。このライブラリーは、DSQUOPTS ルーチンのユーザー変更バージョン、ローカル・ユーザー日時出口 (DB2 提供の出口)、QMF ユーザー編集コード・ルーチン、および QMF 呼び出し可能インターフェース・プログラムを保管することができます。
- OS/390 および VM 環境は、DSQ1BVW インストール・ジョブが含まれました。これはすべてのデフォルト QMF 視点をサポートされている DB2 データベース上に作成または再作成します。QMF LIST コマンドが、PUBLIC に特権を付与する必要なしに、1 次または 2 次許可 ID の

いずれかに許可されるすべての表および視点をリストできるようにする拡張 LIST コマンド機能に対してインストール・ジョブも提供されています。

データベース操作

- QMF バージョン 7.2 は、VM、VSE、および z/OS 版の QMF がアプリケーション・サーバーとして DB2 (Linux/390 版) に接続するための機能が導入されました。
- 複数の新しい DB2 UDB (OS/390 版) および (z/OS 版) SQL コードに対するサポートが追加されました。

QMF 初期化

- DSQAO_QMFADM 大域変数が導入されました。この変数は、QMF を開始したユーザーが QMF 管理者かどうかを示します。大域変数 DSQEC_DISABLEADM は、このチェックをオフにすることができます。
- 大域変数 DSQEC_SHARE および DSQEC_DISABLEADM のデフォルト値をオーバーライドするために DSQUOPTS 出口ルーチンが使用可能になりました。

アプリケーション・サポート

@IF REXX 関数が FORM.CALC、FORM.CONDITIONS、および FORM.COLUMNS 定義で使用可能になり、改善された NULL、オーバーフロー、および未定義の処理によるデータ表示の条件付き変更の拡張機能が提供されました。このスプレッドシートに類似した機能が QMF 報告書の作成者に提供され、入力値に基づいて報告書の出力を拡張制御します。

QMF コマンドおよびオブジェクトの操作

- QMF LIST TABLES コマンドが、DB2 の 2 次許可 ID によって定義された権限グループを扱います。
- QMF CONNECT コマンドにサポートが提供され、3 つの QMF プラットフォーム (VM、VSE および OS/390) 間の DB2 (OS/390 版) ユーザー ID およびパスワードの統一サポートが使用可能になりました。
- QUERY、PROC、FORM、および PROFILE オブジェクトでの変更を破棄する前に、ユーザーに対してプロンプトが出されます。この機能は、以下の大域変数で提供されます。
 - DSQEC_PRO_QUERY
 - DSQEC_PRO_PROC
 - DSQEC_PRO_FORM
 - DSQEC_PRO_PROF
 - DSQEC_PRO_ENABLE
- SET GLOBAL コマンドでは、プロシージャ作成者が大域変数の値を別の大域変数の値に設定することができるようにします。

バージョン 7.1 での変更点

QMF バージョン 7.1 の機能拡張は以下のとおりです。

QMF バージョン 7.1 では、以下の機能拡張が提供されます。

データベース操作

- サーバー製品の DB2 ファミリー全体への分散アクセスが、以下に対する追加のサポートと共にこのリリースに導入されました。
 - DB2 (AS/400 版) サーバー、バージョン 4.4
 - DB2 (VSE 版) アプリケーション・リクエスター
- クロスプラットフォーム DRDA パッケージ・バインディングは、VM、VSE および OS/390 の QMF に対する追加のインストールおよび保守オプションを提供しました。
- 以下のデータ・タイプに対するサポート:
 - ROWID データ・タイプ

QMF バージョン 7.1 は、このデータ・タイプに対する完全に統合されたサポートを提供しました。

- LOB データ・タイプ

QMF バージョン 7.1 は、以下の領域においてこのデータ・タイプに対する制限付きのサポートを提供しました。

- 表エディター
- 指示照会
- DRAW SQL 照会

QMF は、QMF 報告書内の LOB 長さを 32 KB に制限します。新しい SQL 関数は、LOB 列を文字列にキャストします。

QMF コマンドおよびオブジェクトの操作

- 日付・時刻編集コード TD および TT が追加されました。これらのコードは、データベース管理システムの日付および時刻形式の変更に基づいて、報告書書式設定の特性を自動的に調整します。
- 現行コンテキストから値を引き出す QMF コマンドにデフォルトが実装されました。従って、PROC、QUERY、または FORM オブジェクトとの作業は、以下のコマンドで簡単になっています。
 - CONVERT
 - EDIT
 - EXPORT
 - PRINT
 - RESET
 - RUN
 - SAVE
- スtringを指定する方法を増やすなど、引用符付きStringを使用できるコマンド・オプションの柔軟性と制御が強化されました。長さも増えました。優れたオプションは次のとおりです。

- SAVE および IMPORT コマンドの COMMENT
- MESSAGE コマンドの TEXT
- CICS コマンドの FROM (トランザクション・データ)
- SHOW コマンドを使用した QMF ホーム・パネルへの直接ナビゲーション。

バージョン 6.1 での変更点

QMF バージョン 6.1 の機能拡張は以下のとおりです。

インストール

ISPF LIBDEF サービスに対するサポートが導入されました。このサービスは、QMF プログラム・ライブラリーを割り振るために使用できます。

データベース操作

- DB2 ユニバーサル・データベース バージョン 5 に対するサポートが導入されました。
- 正式な QMF 管理者権限が導入されました。これは、管理者に任意のユーザーが所有している QMF オブジェクトを表示、保存、消去、インポート、およびエクスポートする無制限の能力を提供します。
- DB2 特殊レジスターに対するサポートが導入されました。

QMF 初期化

ユーザーの現行の許可 ID は、QMF ホーム・パネルに表示されます。

アプリケーション・サポート

言語処理環境を使用したユーザー定義の編集コードを作成するためのサポートが追加されました。

リソース・コントロール

ユーザーは、多くのリソースを使用することが予想される長い実行照会を実行するかまたはキャンセルするかどうかを事前に選択できます。この機能は、DB2 予見ガバナ機能と共に機能する確認パネルによって提供されます。

QMF コマンドおよびオブジェクトの操作

- QMF 書式および報告書に以下の改善が導入されました。
 - ユーロ通貨記号のサポート。
 - QMF 報告書でゼロをブランクに置き換えることによってゼロの抑止を提供する編集コード。
 - 通貨形式設定のための編集コードおよび関連する大域変数は、同じ報告書内で 2 つの異なる通貨記号を使用できるようにします。
 - QMF スクロール移動量にデフォルトを設定する大域変数。
- QMF コマンドに以下の改善が導入されました。
 - ERASE TABLE コマンドが導入されました。これは、データベース別名を削除するために使用できます。
 - QMF がコンテキスト内のオブジェクト名およびタイプの両方を認識できるようにする SAVE コマンド構文への改善。

- 変更モードでの列デフォルトの表エディター・サポートが追加されました。

付録 B. 特定のサポートを必要とする QMF 機能

これらの機能のサポートは、データベースまたは環境によって異なります。

データベース・タイプによって異なる機能

以下の機能のサポートは、データベースによって異なります。

表 82. 特定のデータベース管理システムのサポートを必要とする機能

| サポートされる機能 | DB2 for z/OS | DB2 for Linux, UNIX, and Windows | DB2 for iSeries | DB2 (VSE および VM 版) |
|---|--|---|---|--|
| サポートされる照会ステートメントの長さ | 2 MB* | 65 KB* | 65 KB* | 8 KB |
| SELECT ステートメントの列数 | 750 | 255 | 255 | 255 |
| 単精度浮動小数点数のインポート | X | | | X |
| LIKE ステートメントのある長いフィールド | X | | | X |
| データベース同義語 | | | | X |
| 表または視点のデータベース別名 | X | X | X | |
| 表エディターで使用可能な SAVE=IMMEDIATE オプション (CURSOR HOLD をサポート) | X | X | X | |
| DB2 大域変数の設定 | X | X | | |
| 3 部構成の名前を含む QMF コマンド | 3 部構成の名前を持つコマンドを、このタイプのデータベースから開始できます。 QMF がストアード・プロシージャとして開始されていない限り、これらのコマンドをこのタイプのデータベースに送信することもできます。 | QMF がストアード・プロシージャとして開始されていない限り、3 部構成の名前を持つコマンドをこのタイプのサーバーに送信できます。 | QMF がストアード・プロシージャとして開始されていない限り、3 部構成の名前を持つコマンドをこのタイプのサーバーに送信できます。 | 3 部構成の名前を持つコマンドをこれらのサーバー・タイプに送信することはできません。 |

* 最大 2 MB の SQL 照会のサポートを DB2 for z/OS データベースで、また最大 65 KB の SQL 照会のサポートを DB2 for Linux, UNIX, and Windows データベースでアクティブにするには、照会を実行する前に DSQC_SQLQRYSZ_2M 大域変数を 1 に設定します。

関連概念:

160 ページの『QMF の DB2 for z/OS ストアード・プロシージャーとしての開始』

QMF for TSO のストアード・プロシージャー・インターフェースでは、DB2 for z/OS ストアード・プロシージャーを呼び出すことができる任意のソフトウェア・プログラムを使用して、QMF for TSO を開始することができます。また、このインターフェースでは、事前定義された QMF 照会またはプロシージャーを実行でき、トレース・データの結果セットを含めて最大 21 個の結果セットを受け取ることができます。

CICS で使用できない機能

一部の機能は TSO でのみサポートされています。

以下の機能は CICS では使用できません。

- マルチスレッド・サポートの使用。
- QMF Analytics for TSO の使用。
- QMF Enhanced Editor の使用。
- アクティブ記憶域で不要になった報告書データを収容するための拡張記憶域の使用。代わりにスピル・ファイルを使用する必要があります。
- QMF を DB2 for z/OS ストアード・プロシージャーとして開始する機能。
- インターフェース:
 - コマンド・インターフェース
 - 文書インターフェース
- プログラム・パラメーター:
 - DSQSCMD (QMF 呼び出し可能インターフェースのみ)
 - DSQSMTHD
 - DSQSPLAN
 - DSQSPRID
 - DSQSPTYP
 - DSQSRSTG
 - DSQSSUBS
- コマンド:
 - BATCH (およびその関連アプリケーション)
 - DPRE (およびその関連 REXX exec)
 - EDIT QUERY
 - EDIT PROC
 - ISPF (およびその関連アプリケーション)
 - LAYOUT (およびその関連アプリケーション)
 - 以下の大域変数を参照する SET GLOBAL コマンド:
 - DSQEC_EXTND_STG
 - 以下の大域変数を参照する SHOW GLOBAL コマンド:
 - DSQEC_EXTND_STG

- DSQAO_STO_PROC_INT
- STATE (コマンド・インターフェースが必要)
- マクロ: GETQMF
- 書式機能:
 - REXX を必要とする報告書の計算または式
 - 条件付き書式設定
 - 列定義
 - ローカル定義編集コード TDL および TTL (それぞれ日付と時刻の修飾設定用)
- ロジックを持つプロシージャ (REXX が必要)
- トランザクションを取り消す機能
- リモート・ロケーションでデータを更新する機能 (リモート・ロケーションの表およびビューはすべて、QMF for CICS では読み取り専用)
- 外部変数

付録 C. DB2 に常駐している QMF オブジェクト

これらの QMF オブジェクトは、DB2 for z/OS サブシステム内で QMF バージョン 12.1 を実行する必要があります。必要に応じてリカバリー操作時にこの情報をガイドとして使用することができます。

QMF プラン

以下のプランが DB2 QMF for TSO/CICS に付属しています。

重要: 本書、ジョブ自体、または IBM ソフトウェア・サポートによる指示がない限り、プランおよびインストール・パッケージの QMF バインド・ジョブ内で BIND PACKAGE または BIND PLAN オプションを変更または追加することはできません。

表 83. QMF プラン

| プラン名 | このプランをインストールするジョブのバインド | 注 |
|----------|------------------------|-----------------------------|
| QMF1210 | DSQ1BINR | デフォルトの QMF プラン |
| DSQIN121 | 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

以下のパッケージが QMF に付属しています。

表 84. QMF パッケージ

| パッケージ名 | バインド・ジョブ |
|------------------------|--------------|
| DSQJ* | DSQ1BPKG JCL |
| リモート・サーバーの場合: DSQJ* | DSQ1BPKG JCL |

TSO および CICS での QMF コントロール表および表スペース

QMF に付属のコントロール表は以下のとおりです。

QMF を DB2 for iSeries データベースにインストールした場合、これらのコントロール表は以下の表スペースではなく、コレクション Q にインストールされます。

重要: 必要に応じて、特定の時点の QMF 操作をリカバリーできるようにするために、これらのすべての表を同時にバックアップします。

表 85. QMF コントロール表とそれらが常駐する表スペース

| コントロール表名 | 表スペース | 表スペースのサイズ (1 KB 単位) | 表の内容 | 索引 |
|---|----------|----------------------|---|----------------------|
| Q.PROFILES | DSQTSPRO | 100 - 1 次 20 - 2 次 | QMF セッション中のリソースおよびデータに対する個々のユーザーのアクセスに関する情報を保持する QMF プロファイルが含まれています。 | Q.PROFILEX |
| Q.OBJECT_DIRECTORY | DSQTSCT1 | 200 - 1 次 20 - 2 次 | データベース内の QMF の照会、書式、プロシージャー、フォルダー、および分析オブジェクトすべてに関する一般情報が含まれています。 | Q.OBJECT_DIRECTORYX |
| Q.OBJECT_REMARKS | DSQTSCT2 | 200 - 1 次 20 - 2 次 | 照会、書式、プロシージャー、および分析オブジェクトの作成または置換時に保存されたコメントが含まれています。 | .OBJECT_REMARKSX |
| Q.OBJECT_DATA Q.OBJECT_DATA2 | DSQTSCT3 | 5000 - 1 次 200 - 2 次 | 内部 QMF 形式で表された照会、書式、プロシージャー、フォルダー、および分析オブジェクトが含まれています。 Q.OBJECT_DATA2 は DB2 for Linux, UNIX, and Windows データベース専用です。 | Q.OBJECT_OBJDATA |
| Q.COMMAND_SYNONYMS Q.COMMAND_SYNONYM_n | DSQTSSYN | 100 - 1 次 20 - 2 次 | QMF コマンド同義語に関する情報が含まれています。Q.COMMAND_SYNONYMS 表は、英語の同義語を保管します。各 QMF NLF に Q.COMMAND_SYNONYM_n 表があります。n を使用している各国語の 1 文字の ID に置き換えます。 | Q.COMMAND_SYNONYMNSX |
| Q.RESOURCE_TABLE | DSQTSGOV | 100 - 1 次 20 - 2 次 | 管理プログラム出口ルーチンに渡されるリソース制御情報が含まれています。 | Q.RESOURCE_INDEX |
| Q.ERROR_LOG | DSQTSLOG | 100 - 1 次 20 - 2 次 | システム、リソース、および予期しない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 |
| Q.GLOBAL_VARS | DSQTSGLV | 110 - 1 次 20 - 2 次 | システム定義およびユーザー定義の大域変数およびセッション変数が含まれます。 | Q.GLOBAL_VARSX |

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

40 ページの『インストール・パス G: 3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーの準備』

この手順は、3 部構成の名前を含む QMF コマンドによってアクセスされるリモート・サーバーを準備するために使用します。

QMF 視点

以下の視点が QMF に付属しています。QMF はこれらの視点を、表に示されているプラットフォーム上で使用し、QMF LIST コマンドが発行されたときにオブジェクト・リストを作成します。

表 86. QMF LIST コマンドが発行されたときにオブジェクト・リストを作成するために使用された視点

| 視点名 | 各プラットフォーム上で視点に基づく表 |
|--------------------|--|
| Q.DSQ_GLOBAL_VARS | Q.GLOBAL_VARS (すべてのプラットフォーム用)。この視点は DB2 (VSE および VM 版) では作成されません。 |
| Q.DSQ_RESERVED_DB | <ul style="list-style-type: none"> • SYSIBM.SYSCOLUMNS (DB2 for z/OS 用) • SYSCAT.COLUMNS (DB2 for Linux, UNIX, and Windows 用) • QSYS2.SYSCOLUMNS (DB2 for iSeries 用) |
| Q.DSQ_RESERVED_OBJ | Q.OBJECT_DIRECTORY (すべてのプラットフォーム用) |
| Q.DSQEC_ALIASESL | <ul style="list-style-type: none"> • SYSIBM.SYSTABLES (DB2 for z/OS 用) • SYSCAT.TABLES (DB2 for Linux, UNIX, and Windows 用) • QSYS2.SYSTABLES (DB2 for iSeries 用) |
| Q.DSQEC_COLS_LDB2L | <ul style="list-style-type: none"> • SYSIBM.SYSCOLUMNS および SYSIBM.SYSTABAUTH (DB2 for z/OS 用) • SYSCAT.COLUMNS および SYSCAT.TABAUTH (DB2 for Linux, UNIX, and Windows 用) • QSYS2.SYSCOLUMNS (DB2 for iSeries 用) |
| Q.DSQEC_COLS_RDB2L | SYSIBM.SYSCOLUMNS および SYSIBM.SYSTABAUTH (DB2 for z/OS 用) |
| Q.DSQEC_QMFOBJS | Q.OBJECT_DIRECTORY および Q.OBJECT_REMARKS (すべてのプラットフォーム用) |
| Q.DSQEC_TABS_LDB2L | <ul style="list-style-type: none"> • SYSIBM.SYSTABAUTH および SYSIBM.SYSTABLES (DB2 for z/OS 用) • SYSCAT.TABAUTH および SYSCAT.TABLES (DB2 for Linux, UNIX, and Windows 用) • QSYS2.SYSTABLES (DB2 for iSeries 用) |
| Q.DSQEC_TABS_RDB2L | SYSIBM.SYSTABAUTH および SYSIBM.SYSTABLES (DB2 for z/OS 用) |
| Q.RESOURCE_VIEW | Q.RESOURCE_TABLE (すべてのプラットフォーム用) |

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 を実行することにより、z/OS からのサポートされている DB2 データベース上のすべてのデフォルト QMF コントロール表視点を作成または再作成することができます。このジョブは、既存のジョブを除去し、新しい視点を作成してその視点に必要な特権を付与します。DB2 for z/OS と共に使用可能な拡張 LIST コマンド機能をインストールした場合、ジョブ DSQ1BUDV を実行して、ジ

ジョブ DSQ1BVW を実行した後に視点をインストールします。ジョブ DSQ1BUDV は、DB2 2 次許可 ID で QMF コントロール表視点を使用可能にします。

関連タスク:

147 ページの『拡張 LIST コマンド機能のインストール (z/OS のみ)』
拡張 LIST コマンド機能がインストールされている場合、表と視点に対する特権は PUBLIC の代わりに、ユーザーの 1 次または 2 次許可 ID にのみ付与する必要があります。この許可 ID は、OWNER=ALL オプションを LIST TABLES または LIST ALL コマンドに指定した場合にこのコマンドで生成されるオブジェクト・リストに含まれています。

DB2 for z/OS 記憶域グループ

以下の DB2 for z/OS 記憶域グループ (STOGROUP) オブジェクトが、QMF のインストール時に作成されます。

表 87. QMF 用に作成される DB2 for z/OS 記憶域グループ

| 記憶域グループ名 | 説明 |
|-------------|--|
| DSQ1STBG | QMF サンプル表を保管する |
| DSQSGCTL | QMF コントロール表を保管する |
| DSQSGDEF | SAVE DATA によって作成された表のデフォルトの記憶域グループ |
| DSQSGSY n | QMF 各国語機能 (Q.COMMAND_SYNONYMNSX) 用のコマンド同義語インデックスを保管する それぞれの各国語に別々のスペースがあり、固有の文字 n の代わりに固有の文字で示されます。 |

QMF のインストール後に、これらの記憶域グループをユーザー管理グループとして構成しないでください。

関連タスク:

134 ページの『QMF 各国語機能のインストール』
QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

LOB データを保管するためのスペース

1 つ以上の LOB 列を含む表に対して SAVE DATA コマンドを発行すると、DB2 および QMF によって複数のオブジェクトが作成されます。

QMF では、QMF プロファイルの SPACE フィールドに指定された表スペースに、1 つの基本表と索引が作成されます。

DB2 では、保管されている表内の LOB 列ごとに、補助表スペース、表、および索引が作成されます。

TSO/CICS 用の VSAM クラスター

これらの VSAM クラスターは、QMF に付属しています。

表 88. VSAM クラスター

| クラスター名 | クラスターが必要なオブジェクト |
|---|---------------------|
| 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.DSQTSGLV.I0001.A001 | DSQTSGLV |
| 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 X |
| QMFDSN.DSNDBC.DSQDBCTL.PROFILEX.I0001.A001 | Q.PROFILEX |
| QMFDSN.DSNDBC.DSQDBCTL.COMMANDR.I0001.A001 | Q.COMMAND_SYNONYMSX |
| QMFDSN.DSNDBC.DSQDBCTL.GLOBALRV.I0001.A001 | Q.GLOBAL_VARSX |

QMF サンプル表

QMF には、製品の機能について理解し、それらの機能をテストするために役立つサンプル表が用意されています。これらの表には、架空の電子部品製造メーカー J & H Supply Company の応募者、面接、部品、製品、従業員、供給業者に関するデータが収められています。

表 89. QMF サンプル表

| 表 | 含まれている情報 |
|-------------|-------------------|
| Q.APPLICANT | 新規採用候補者 |
| Q.INTERVIEW | 新規採用の面接 |
| Q.ORG | 会社の組織構造 |
| Q.PARTS | 製品の部品データ |
| Q.PRODUCTS | 会社の製品 |
| Q.PROJECT | 請け負ったプロジェクト (部門別) |
| Q.SALES | 売上および歩合 |
| Q.STAFF | 会社の従業員 |
| Q.SUPPLIER | ベンダー情報 |

QMF Analytics for TSO には、QMF Analytics for TSO 機能について理解するために役立つサンプル表も用意されています。下表に、各サンプル表に含まれるデータのタイプを示します。

表 90. QMF Analytics for TSO サンプル表

| 表 | 含まれている情報 |
|----------------|---|
| Q.CASHFLOW | QMF Analytics for TSO 割引キャッシュ・フロー分析で使用できる、コストおよび収益に関するデータ |
| Q.CLIMATE_10YR | 10 年間にわたる気候に関するデータ |
| Q.CLIMATE_USA | 降雨量および日照量に関するデータを含む、米国の気候に関するデータ |
| Q.WORLDMINFO | J & H Supply Company が事業を展開している地域に関するデータ |

DB2 for z/OS と DB2 for Linux, UNIX, and Windows のどちらでも、サンプル表は DSQ1STBT 表スペースにあります。DB2 for z/OS データベースでは、この表スペースは DSQ1STBB データベースにあります。DB2 for Linux, UNIX, and Windows データベースでは、DSQ1STBT 表スペースは DSQTSAMP データベース・パーティション・グループにあります。

付録 D. 外部エディター

QMF オブジェクトおよび報告書では、外部のエディターを使用できます。

エディターを使用した **QMF** 照会およびプロシージャの変更

QMF 外部のエディターで、EDIT コマンドを使用して、一時記憶域にある QMF SQL 照会またはプロシージャを変更することができます。

始める前に

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。ただし、エディターが ISPF/PDF の場合、CLIST ではなく、ISPF EDIT サービスを使用して直接呼び出されます。

このタスクについて

外部エディターの 1 つとして、ISPF/PDF を使用できます (QMF が ISPF 下で開始されている場合)。EDIT コマンドを発行すると、QMF は QMF 一時記憶域から一時編集転送データ・セットにオブジェクトをエクスポートします。編集セッションが完了すると、一時データ・セットの内容は、一時記憶域にインポートされます。

手順

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。例えば、次のコマンドで QMF は XYZEDIT という名前の CLIST を呼び出して、USERA.XYZDATA.TEXT という名前のデータ・セットを編集します。

```
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 DD 名 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 QMF1210.SDSQCLTE で、すべての QMF ユーザーで使用可能になります。
3. 個々のユーザーに対して、編集するオブジェクトのデータ・セットを割り振り、カタログします。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・サイズ 4029 (BLKSIZE=4029)
4. 各ユーザーの JCL で、そのユーザー用として 3 の手順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 を使用します。

5. EDIT コマンドの指定方法について、ユーザーに示します。 コマンドのフォーマットは、次のとおりです。

```
EDIT yyyy (EDITOR=xxxx)
```

この構文では、yyyy は PROC または 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る CLIST の名前です。

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 1 文字言語 ID) で PDF エディターを実行します。さらに、QMF はファンクション・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。
 - 別のファンクション・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 DSQn 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。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す。次に 2 番目のプログラムが、必要なファンクション・キーを使用して、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なセットアップ要件を使用して、ISPF アプリケーション ID 下で実行されているエディターを呼び出します。

次の 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 "LINIT DATAID(EDT) DDNAME(DSQEDIT)"
Address ISPEXEC "EDIT  DATAID("EDT")"
Address ISPEXEC "LMFREE DATAID("EDT")"
Exit 0
```

図 126. 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 コマンド行から次の QMF EDIT コマンドを発行します。

EDIT QUERY (E=MYEDIT)

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

QMF 報告書のドキュメントへの挿入

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 提供のマクロであり、CICS では使用できません。このマクロを使用すると、QMF の外部で作業しているユーザーは、QMF セッションを開始し、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。

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。

- 各ユーザーが適正な QMF ライブラリーで操作するようにする。

サンプルの TSO ログオン・プロシージャでは、これらのライブラリーに次の形式の名前が付けられています。

QMF1210.SDSQ*

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、ドキュメント・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ができません。

- 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。

NLF を使用する場合、該当する NLF 版のドキュメント・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、QMF1210.SDSQCLTE のメンバーです。
- それ以外のコンポーネントは、QMF1210.SDSQSAPE のメンバーです。

ドキュメント・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。

手順

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。このデフォルト名はマクロに使用されます。

NLF を使用する場合、メイン・マクロはライブラリー QMF1210.SDSQCLT n のメンバー DSQA n D1P です。ここで、 n は、使用中の NLF の 1 文字の言語 ID です。メインの英語マクロと同様に、他のコンポーネントに影響を与えることなく、名前を変更できます。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。例えば、GETQMF n に変更してください。

- 元の (QMF1210.SDSQCLTE) を含むライブラリーに、名前変更したコピーを入れます。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

Q.DSQAED1S プロシージャをデータベースに入れる

Q.DSQAED1S プロシージャは、文書編集インターフェース機能をインストールします。このプロシージャは、文書編集インターフェースのセットアップの一環としてデータベース内に入れます。

このタスクについて

Q.DSQAED1S プロシージャは、QMF1210.SDSQSAPE ライブラリーのメンバー DSQAED1S に入っています。

NLF を使用する場合、QMF1210.SDSQSAP n ライブラリーのメンバー DSQA n D1S の 1 文字の各国語 ID を変更します。

手順

QMF 管理者として、次の QMF コマンドを入力することで、データベースに Q.DSQAED1S を入れることができます。

```
IMPORT PROC DSQAED1S FROM 'QMF1210.SDSQSAPE(DSQAED1S)' (SHARE=YES
```

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

データ・コンポーネントの変更

ユーザーがこのマクロを使用して QMF 報告書を文書に挿入する前に、特定のデータ・コンポーネントを変更する必要があります。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。

データ・コンポーネントは 5 つあり、すべてはライブラリー QMF1210.SDSQSAPE にあります。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含まれません。代わりに、これらはメッセージまたはユーザー報告書に現れる情報を含みます。

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 QMF1210.SDSQSAPE に保持することができます。

この方法を使用する場合、元のコンポーネントの名前を変更し、変更後のコンポーネントに元の名前を付けます。

- 変更後のコンポーネントを新規のライブラリーに入れることができる。

この方法を使用する場合、他のすべてのデータ・コンポーネントを、古いライブラリーから新しいライブラリーにコピーする必要があります。マクロ DSQAED1P を変更する必要があります。

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、QMF1210.SDSQSAPn ライブラリーのメンバー DSQAnD0L の 1 文字の各国語 ID を変更します。

DCF コンポーネント

文書構成プログラム (DCF) は、IBM テキスト処理システムです。サイトが DCF を使用する場合、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。

ユーザーは、ドキュメント・インターフェースに、現行ドキュメントが DCF による書式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。以下のデフォルト・ステートメントが、デフォルト・コンポーネント内にあります。

```
.* 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。次のデフォルト・ステートメントが、デフォルト・コンポーネント内にあります。

```
.* 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。以下のデフォルト・ステートメントが、デフォルト・コンポーネント内にあります。

```
.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。以下のデフォルト・ステートメントが、デフォルト・コンポーネント内にあります。

```
.* QMF Document Interface footing control:
.RE
.* QMF REPORT END
```

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

関連情報:

 IBM Publications Center

文書構成プログラム・コンポーネントに関する情報を DCF 資料で検索します。

CLIST およびマクロの変更

文書編集インターフェースをセットアップするには、いくつかのコンポーネントに対してさらに変更が必要になる場合があります。

これらのコンポーネントはすべて、ライブラリー QMF1210.SDSQCLTE にあります。CLIST またはマクロを変更する場合は、コピー (オリジナルではない) を変更し、それを別のライブラリーに入れてください。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、そこがない場合は、QMF1210.SDSQCLTE のステートメントの前に 1 つを挿入します。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。例えば、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ステートメントのようになります。

```
//SYSPROC DD DSN=SYSUT2.CLIST,DISP=SHR
//          DD DSN=XYZ.NEWCLIST,DISP=SHR
//          DD DSN=QMF1210.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 = QMF1210.SDSQSAP&LANGCHAR
SET &BASELIB = QMF1210.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。

&LANGCHAR は、各 NLF を識別する 1 文字の ID を参照します。

&BASELIB

QMF サンプル・ライブラリー QMF1210.SDSQSAPE を識別します。

&LANGCHAR の値が E の場合、両方の変数はライブラリー

QMF1210.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。この方法で呼び出されると、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。DSQPRINT をデフォルトに復元したくない場合は、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はすべて CLIST コードに行います。例えば、サイトに特定のデータ・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、例えば、ドイツ語アプリケーションとは逆に、ライブラリーに英語のコンポーネントが含まれ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て、必要に応じてファイル名をカスタマイズします。

```
/* Remove the Following "GOTO NOLIBDEF" statement to allocate */
/* ISPF libraries using the ISPF LIBDEF service. */
GOTO NOLIBDEF
/* ALLOCATE QMF ISPF LIBRARIES USING LIBDEF */
```

```

SET PNAME = 'QMF1210.SDSQPLB&LANGCHAR'    /* ISPF Panel Library */
SET MNAME = 'QMF1210.SDSQMLB&LANGCHAR'    /* ISPF Message Library */
SET SNAME = 'QMF1210.SDSQSLB&LANGCHAR'    /* ISPF Skeleton Library */
SET LNAME = 'QMF1210.SDSQLOAD'           /* QMF Modules */
ISPEXEC LIBDEF ISPLLIB 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 */
/* You might or might not need to free libdefs here. */
/* If you do, then remove comments from LIBDEF statements. */
/*****/
/* ISPEXEC LIBDEF ISPLLIB DATASET ID() */
/* ISPEXEC LIBDEF ISPLLIB DATASET ID() */
/* ISPEXEC LIBDEF ISPLLIB 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大きい場合は、SPACE オペランドの変更が必要になることがあります。

- 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 のサブシステムのサブシステムの名前は、DSN でなければならず、QMF のアプリケーション・プランの名前は、QMF12 でなければなりません。設定が異なる場合は、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。以下のステートメントの例は、ABC および QMFXXX という名前のサブシステムおよびアプリケーション・プランに適用されます。

```

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、いくつかの QMF プログラム・パラメーターのデフォルト値を上書きします。

関連概念:

187 ページの『第 10 章 起動時のプログラム・パラメーターおよびプリファレンスの設定』

このトピックでは、QMF を標準インターフェースで開始する際に QMF に渡すこ

とができるプログラム・パラメーターについて説明します。

関連タスク:

134 ページの『QMF 各国語機能のインストール』

QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

付録 E. QMF ユーザー定義関数

QMF ユーザー定義関数についての情報です。

QMF ユーザー定義関数およびプロシージャ

オプションの QMF インストール・フィーチャーのいくつかを使用して、ユーザー定義関数およびユーザー定義プロシージャを作成することができます。

下表に、オプションのインストール・フィーチャーに関連付けられているユーザー定義関数およびユーザー定義プロシージャを示します。ご使用の環境で使用可能なユーザー定義関数およびユーザー定義プロシージャは、インストールされているオプション・フィーチャーによって異なります。

表 91. ユーザー定義関数およびユーザー定義プロシージャ

| 名前 | 特定名 | タイプ | 関数またはプロシージャを定義するオプションのインストール・ジョブ | 使用 |
|------------------|--------------|--------|----------------------------------|---------------------------|
| Q.DSQMFSP | DSQMFSP | プロシージャ | DSQ1BSP | QMF ストアード・プロシージャ・インターフェース |
| Q.DSQABA1E | DSQABA1E | プロシージャ | DSQ1BUDF | 拡張リスト・ビュー処理 |
| Q.DSQABA1E | DSQABA1E_D | 機能 | DSQ1BUDF | 拡張リスト・ビュー処理 |
| Q.APPL_AUTHNAMES | DSQABA1E_F0V | 機能 | DSQ1BUDF | 拡張リスト・ビュー処理 |
| Q.APPL_AUTHNAMES | DSQABA1E_F1V | 機能 | DSQ1BUDF | 拡張リスト・ビュー処理 |

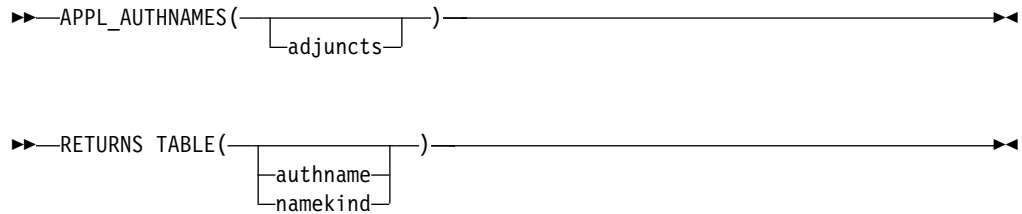
重要: QMF のインストール後に WLM 環境が変わった場合は、ALTER ステートメントを実行して、使用しているすべてのユーザー定義関数またはユーザー定義プロシージャに関する WLM 環境情報を更新する必要があります。以下に例を示します。

```
ALTER PROCEDURE Q.DSQMFSP
WLM ENVIRONMENT new_environment_name;
ALTER FUNCTION Q.DSQABA1E
WLM ENVIRONMENT new_environment_name;
ALTER SPECIFIC FUNCTION Q.DSQABA1E_F0V
WLM ENVIRONMENT new_environment_name;
ALTER SPECIFIC FUNCTION Q.DSQABA1E_F1V
WLM ENVIRONMENT new_environment_name;
ALTER PROCEDURE Q.DSQABA1E
WLM ENVIRONMENT new_environment_name;
```

APPL_AUTHNAMES

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。

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。



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。以下に例を示します。

```
'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。

表 92. APPL_AUTHNAMES 関数から作成された表

| 列名 | データ・タイプ | 説明 |
|----------|----------------|-------------------|
| authname | CHARACTER(128) | 現行プロセスの許可 ID の名前。 |

表 92. APPL_AUTHNAMES 関数から作成された表 (続き)

| 列名 | データ・タイプ | 説明 |
|----------|--------------|--|
| namekind | CHARACTER(1) | <p>AUTHNAME の名前値の種類コード。</p> <p>1 1 次許可 ID またはユーザー名</p> <p>2 2 次許可 ID またはグループ名</p> <p>3 現行許可 ID これは、現行の SQL ID が 1 次 ID でも 2 次 ID でもない場合のみ適用されます。</p> <p>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。</p> |

CALL DSQABA1E

DSQABA1E ストアド・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。

ストアド・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—)————▶▶

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ストリング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ストリング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

DSQABA1E 関数は、IBM ソフトウェア・サポート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。

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—(—)————▶▶

この関数の結果は、データ・タイプ VARCHAR の文字ストリングで、実際の長さは 5,300 バイト以下です。このストリングは、QMF 報告書の列幅を 53、EDIT コードを CW にフォーマット設定する場合に適しています。

付録 F. QMF および GDDM プログラムの CICS への定義の方法

QMF for TSO/CICS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 プログラムの CICS への定義の方法

インストール中に、デフォルト・トランザクション ID QMF_n が QMF に定義されます。トランザクション ID は、システム定義 (CSD) ファイル内に定義されます。

常駐の QMF プログラム

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
- DSQU n GV3
- DSQUECIC

プログラム名およびトランザクション ID 内の変数 n は、QMF を実行している言語に対応する 1 文字の各国語 ID (NLID) を表します。

CICS は、RMODE(31) 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。

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、ユーザーがトランザクションを開始するたびにロードされるからです。ただし、サイト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ができます。

関連概念:

『QMF のインストール時における GDDM 定義のロード方法』
QMF は、QMF 画面の印刷と表示に GDDM サービスを使用します。

関連タスク:

134 ページの『QMF 各国語機能のインストール』
QMF 各国語機能 (NLF) は、特定言語に合うようにカスタマイズされた環境を提供します。一般に、ベースとなる英語製品で使用可能な QMF の機能は、NLF でも使用可能です。

QMF のインストール時における GDDM 定義のロード方法

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、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 のためにカスタマイズする時に、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。

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 ADMF ファイル内に、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。例えば、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 のインストール後に図表機能を追加

CICS 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リストアするのではなく、GDDM-PGF をすべてインストールして CICS のためにカスタマイズしなければなりません。GDDM-PGF をインストールしてカスタマイズした後に、GDDM を CICS のためにカスタマイズ中に 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。図表形式をこれ以上カスタマイズする必要はありません。これらの形式は、QMF のインストール中に定義されています。

関連概念:

45 ページの『リクエスター (DB2 for z/OS) データベースのインストール前提条件』
スタンドアロン・データベースまたはリクエスター・データベースとして機能する DB2 for z/OS データベースに QMF をインストールするには、ハードウェアおよびソフトウェアの要件を満たす必要があります。

関連情報:

 [IBM Publications Center](#)

CICS 用の GDDM のカスタマイズに関する情報を GDDM 資料で検索します。

CICS でのリソース使用を制御するためのトランザクション経路指定

ユーザーのシステムの高速トランザクションを、余分のリソースを消費する可能性のある潜在的な長時間実行の QMF 照会から保護するために、連続領域操作または相互システム間通信を使用して、QMF トランザクションの実行を単一領域に隔離することを検討してください。

1 つの CICS 端末所有領域を定義し、複数のトランザクション ID または動的経路指定出口を使用して、QMF トランザクション要求を別の領域に経路指定します。

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。この資料の他の言語版を IBM から入手できる場合があります。ただし、これを入手するには、本製品または当該言語版製品を所有している必要がある場合があります。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒103-8510

東京都中央区日本橋箱崎町19番21号

日本アイ・ビー・エム株式会社

法務・知的財産

知的財産権ライセンス渉外

IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様自身の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができませんが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。これらのサンプル・プログラムは特定物として現存するままの状態を提供されるものであり、いかなる保証も提供されません。IBM は、お客様の当該サンプル・プログラムの使用から生ずるいかなる損害に対しても一切の責任を負いません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生した創作物には、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年).

このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。

© Copyright IBM Corp. _年を入れる_.

プログラミング・インターフェース情報

この文書は、QMF のサービスを利用するためのプログラミングを行う際に必要となるプログラミング・インターフェースに関する情報を提供します。

商標

IBM、IBM ロゴ、および [ibm.com](http://www.ibm.com)[®] は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

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。

用語集

異常終了 (abend) (abnormal end of task (abend))

実行中に回復機能が解決できなかったエラー状態のために、タスク、ジョブ、またはサブシステムが終了すること。

アドレス・スペース (address space)

コンピューター・プログラムまたはプロセスで使用可能なアドレスの範囲。アドレス・スペースは、物理記憶域、仮想記憶域、またはその両方を指すことがある。

拡張プログラム間通信 (Advanced Program-to-Program Communication)

APPC を参照。

集約関数 (aggregate function)

列中のデータを総計する関数の任意のグループ。この関数は、形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 を使用して要求される。

集約変数 (aggregation variable)

集約関数は、「FORM.BREAK」、
「FORM.CALC」、
「FORM.DETAIL」、または
「FORM.FINAL」パネルを使用して報告書に配置される。その値は、報告書の生成時に、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出力される。

別名 (alias)

表、視点、データベース、またはニックネームを識別するために使用される代替名。別名は、同じ DB2 システム/サブシステムにおいて、またはリモートの DB2 システム/サブシステムにおいて、表、視点、またはデータベースを参照する SQL ステートメント内で使用できる。

APAR (プログラム診断依頼書)

IBM 提供プログラムのサポート対象リリースの問題点に対する修正要求。

許可プログラム機能 (authorized program facility (APF))

z/OS 環境で、制限付き機能の使用を許可されたプログラムの識別を可能にする機能。

アプリケーション・プログラミング・インターフェース (application programming interface (API))

高水準言語で記述されたアプリケーション・プログラムが、オペレーティング・システムまたは他のプログラムの特定のデータや機能を使用できるようにするインターフェース。

アプリケーション (application)

QMF サービスを使用して、特定のビジネス・プロセス (複数の場合もあり) を直接サポートする機能を提供する、1 つ以上のコンピューター・プログラムまたはソフトウェア・コンポーネント。

拡張プログラム間通信機能 (Advanced Program-to-Program Communication (APPC))

相互接続した複数のシステムが互いに通信してプログラムの処理を分担できるようにする、SNA LU 6.2 プロトコルのインプリメンテーション。

アプリケーション計画 (application plan)

バインド処理中に生成される制御構造。QMF バージョン 12.1 アプリケーション・プランのデフォルト名は QMF12 です。

アプリケーション・プログラミング・インターフェース (application programming interface)

API を参照。

アプリケーション・リクエスター (application requester)

DRDA に対応したリモート・リレーションアル・データベース管理システム (RDBMS) への要求の送信元。QMF が開始できる唯一のデータベース・タイプであるため、DB2 for z/OS データベースのみがアプリケーション・リクエスターとして機能できる。

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。アプリケーション・サーバー設置場所のデータベース管理システム (DBMS) が、要求に対応する。QMF for TSO が DB2 for z/OS ストアード・プロシージャーとして実行されている場合、リモート・サーバーとの接続はサポートされない。

引数 (argument)

実行時に関数またはプロシージャー間で渡されるか、または返される値。

許可 ID (authorization ID) (authorization identifier (authorization ID))

1 組の特権を指定する文字ストリングで、権限の検証に使用される。許可 ID は、オブジェクト、個々のユーザー、組織グループ、関数、またはデータベース・ロールを表している。QMF は、QMF 初期化中に、Q.PROFILES テーブルの列に対して、データベース許可 ID を認証するか、またはオプションで QMF TSO ログオン ID を認証する。

プログラム診断依頼書 (Authorized Program Analysis Report)

APAR を参照。

許可プログラム機能 (Authorized program facility)

APF を参照。

補助表 (auxiliary table)

定義されている表の外側の列を保管する表。基本表 も参照。

基本製品 (base product)

QMF のインストール時に設定される QMF の英語バージョン。他の言語環境は、その言語に関連付けられた各国語機能 (NLF) をインストールすることにより、インストール後に設定される。

基本表 (base table)

SQL CREATE TABLE ステートメントにより作成され、永続データを保持する表。

バイナリー・ストリング (binary string)

コード化文字セットに関連付けられていないため変換されることのない、バイトの連

続。例えば、BLOB データ型はバイナリー・ストリングである。CCSID も参照。

バインド (bind)

DBMS プリコンパイラーからの出力を、アクセス・プラン、アプリケーション・プラン、パッケージなどの、使用可能な制御構造に変換すること。

ビット・データ (bit data)

コード化文字セットに関連付けられていないため変換されることのない、CHAR または VARCHAR データ型のデータ。

バッファー・プール (buffer pool)

データ・ページが読み込まれるメモリー領域であり、処理中にデータ・ページの変更が行われて保存される場所でもある。アドレス・スペースも参照。

組み込み関数 (built-in function)

DB2 データベースに不可欠な、厳密に型指定された高性能関数。組み込み関数は、SQL ステートメント内の、その式が有効なあらゆる場所で参照できる。

呼び出し接続機能 (call attachment facility (CAF))

TSO または z/OS バッチで実行されるアプリケーション・プログラムの DB2 for z/OS 接続機能。CAF は DSN コマンド・プロセッサの代替機能であり、実行環境でのより優れた制御を可能にする。

呼び出し接続機能 (call attachment facility)

CAF を参照。

呼び出し可能インターフェース (callable interface)

QMF オブジェクトと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。

カスケード削除 (cascade delete)

DB2 データベース・マネージャーが、削除された親行の下層行をすべて削除することにより参照制約を強制するプロセス。

カタログ (catalog)

表、視点、索引などのオブジェクトの記述を含む表と視点の集合。QMF オブジェクト・カタログ も参照。

コード化文字セット ID (coded character set identifier (CCSID))

特定のコード化スキーム ID のセット、文字セット ID、コード・ページ ID、およびコード化図形文字表現を一意に識別するその他の情報を含む 16 ビットの数値。QMF では、GDDM が提供する表示サービスを使用するため、GDDM アプリケーション・コード・ページは、データベースで使用中の CCSID と一致している必要がある。バイナリー・ストリング も参照。

文字ストリング (character string)

ビット・データ、1 バイト文字、または 1 バイト文字とマルチバイト文字の組み合わせを表す一連のバイト。

チェック制約 (check constraint)

ユーザー定義の制約で、基本表の特定の列に含むことができる値を指定する。制約も参照。

CICS 顧客情報管理システム (Customer Information Control System (CICS))

オンライン・トランザクション処理サービスおよびビジネス・アプリケーションの管理を提供する IBM ライセンス・プログラム。

節 (clause)

SQL では、SELECT 節や WHERE 節など、言語構造のステートメントの特徴的部分。

CM (互換モード) (CM (Compatibility Mode))

QMF バージョン 8.1 および QMF バージョン 9.1 のインストール・モードで、QMF オブジェクト・カタログ内の所有者名およびオブジェクト名をそれぞれ 8 文字と 18 文字に制限する。NFM も参照。

コード・ページ (code page)

GRAPHIC 文字へのコード・ポイントの特定の割り当て。特定のコード・ページ内では、コード・ポイントは 1 つの限定した意味のみを持つことができる。コード・ページは、未定義のコード・ポイントの処理方法も識別する。

コード化文字セット ID (coded character set identifier)

CCSID を参照。

共存 (coexistence)

2 つの QMF リリースが同じ DB2 サブシステムに存在する状態。QMF バージョン 12.1 は、QMF バージョン 9.1 (新機能モード) または QMF バージョン 8.1 (新機能モード) とのみ共存できる。

列 (column)

データベース表の垂直方向のコンポーネント。列は、特定のデータ・タイプ (例えば、文字、10 進数、整数) と名前を持つ。

列関数 (column function)

集約関数 を参照。

列折り返し (column wrapping)

1 つの列内で値が複数の行にまたがる、報告書内の値の書式設定。列折り返しは、XML データの表示を必要とする場合など、列に含まれる値の長さがその列の幅を越えるときに、しばしば使用される。

コマンド・インターフェース (command interface)

QMF コマンドを発行するためのインターフェース。コマンド・インターフェースを使用すれば、QMF のもとで実行中の ISPF ダイアログから QMF コマンドを発行できる。このインターフェースを使用して、QMF は ISPF 変数プールを介してダイアログと通信する。

コマンド同義語 (command synonym)

サイトの定義コマンドの verb または verb / オブジェクトの部分。QMF プロファイルでコマンド同義語を定義してアクティブにすると、ユーザーは、通常の QMF コマンドと同じように、QMF コマンド行で同義語を入力できる。

コマンド同義語表 (command synonym table)

行ごとに 1 つの地域別定義コマンドを保管する表。この表の名前をユーザーのプロファイルに保管することにより、ユーザーにコマンド同義語のセットを割り当てる。

比較演算子 (comparison operator)

SQL の比較式で、2 つの値の関係を指定するために使用される記号。比較演算子には、= (等しい)、<> (等しくない)、< (よ

り小)、 > (より大)、 <= (より小か等しい)、 および >= (より大か等しい) がある。

互換モード

互換モード (CM) を参照。

コミット (commit)

作業単位で行われたデータベースの変更を他のプロセスが認識できるように、ロックを解除することにより作業単位を終了すること。この処理は、データ変更を永続的なものにする。

連結 (concatenation)

2 つの文字または文字列を結合して 1 つの文字列にすること。

接続 (connection)

データ通信で、エンティティー間で情報を伝達するために設定されるアソシエーション。SQL 接続 も参照。QMF for TSO が DB2 for z/OS ストアード・プロシージャーとして実行されている場合、リモート・サーバーとの接続はサポートされない。

定数 (constant)

変化することのない値を指定する言語エレメント。定数は、ストリング定数または数値定数として分類される。

制約 (constraint)

表での挿入、削除、または更新が可能な値を制限する規則。

制御セクション (control section)

CSECT を参照。

コントロール表 (control tables)

QMF が QMF オブジェクトに関する情報の保管、および QMF 演算を管理するために使用する一連の表。QMF オブジェクト・カタログ も参照。

関連参照 (correlated reference)

副照会の外側にある表または視点の列の参照。

関連名 (correlation name)

単一の SQL ステートメント内でオブジェクト (表、視点、表関数の参照、ネストされた表の式、データ変更表の参照など) の直接的な名前として指定され、使用される識別子。関連名は、SQL ステートメント

で、同じ基本表への 2 つの別個の参照を許す場合や、代替名を使用したオブジェクトの表現を許す場合に便利である。

制御セクション (control section (CSECT))

プログラマーによって再配置可能単位として指定されるプログラムの一部。この部分のエレメントは、すべて主記憶域ロケーションにロードされる。

現在場所 (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。接続が確立されると、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 開始パラメーターを使用して示すことができる。QMF for TSO が DB2 for z/OS ストアード・プロシージャーとして実行されている場合、リモート・サーバーとの接続はサポートされない。

現行オブジェクト (current object)

一時記憶域に保持されている QMF オブジェクト。そのため、使用のたびにデータベースの検索を必要とせずに簡単にアクセスできる。QUERY、FORM、PROC、PROFILE、REPORT、DATA、および CHART の 7 つの一時記憶域がある。ユーザーは、SHOW コマンドと DISPLAY コマンドを使用して DATA 領域以外のすべての領域にナビゲートできる。一時記憶域 も参照。

カーソル (cursor)

セットのデータ行をポイントして選択するためにアプリケーション・プログラムにより使用される名前付き制御構造。

顧客情報管理システム (Customer Information Control System)

CICSを参照してください。

データ・タイプ (data type)

各種データの 1 つを特定する種別。SQL でのデータ・タイプは、列、リテラル、ホスト変数、特殊レジスター、パラメーター、および関数と式の結果を表す属性。

データベース (database)

1 つ以上のアプリケーションに対して提供するために、まとめて保管されている関連したデータ項目や独立データ項目の集合。

データベース管理者 (database administrator)
データベースの設計、開発、運用、セキュリティ、メンテナンス、および使用を担当する個人。

データベース管理システム (database management system (DBMS))
DBMS を参照。

データベース・マネージャー (database manager)
効率的なアクセス、保全性、回復、並行性制御、プライバシー、およびセキュリティを目的に、集中制御、データ独立性、複雑な物理構造を提供することによりデータを管理するプログラム。

データベース・サーバー (database server)
データベース・マネージャーを使用して他のソフトウェア・プログラムまたはコンピューターにデータベース・サービスを提供するソフトウェア・プログラム。

DBCS (2 バイト文字セット) (DBCS (double-byte character set))
各文字が 2 バイトで表現される文字セット。これらの文字セットは、一般的に日本語や中国語のような 1 バイトで表現できない記号が多い各国語で使用される。SBCS も参照。

データベース管理システム (database management system (DBMS))
データベースの作成と編成、変更、およびそのデータベースに保管されているデータへのアクセスについて、制御を行うソフトウェア・システム。

宛先管理テーブル (destination control table (DCT))
CICS で使用されている、一時データ宛先をそれぞれ記述する表。このテーブルは、区画外、区画内、および間接宛先ごとの項目を含む。

デフォルト形式 (default form)
保存された書式が RUN QUERY コマンドで指定されない場合に作成される QMF 書式。

デフォルト値 (default value)
他に値が指定されていない場合に使用される、事前に定義されている値、属性、また

はオプション。DB2 表の列データのデフォルト値は、データを変更する SQL ステートメント (INSERT、UPDATE、MERGE など) に DEFAULT キーワードを指定することにより定義できる。

従属行 (dependent row)
親行の親キーの値と一致する外部キーを含む行。外部キーの値は、従属行から親行への参照を表す。

従属表 (dependent table)
オブジェクトに従属する表。例えば、外部キーを持つ表は、対応する主キーを含む表に従属する。

宛先管理テーブル (destination control table)
DCT を参照。

明細ブロック・テキスト (detail block text)
データ内のある行に対応している、報告書の本文中のテキスト。

明細見出しテキスト (detail heading text)
報告書の見出しにあるテキスト。

詳細バリエーション (detail variation)
「FORM.DETAIL」パネルに指定されるデータ形式設定定義。条件に応じて報告書を全体的にまたは部分的に形式設定するために使用できる。

特殊タイプ (distinct type)
共通の表現を組み込みデータ型と共有する、ユーザー定義のデータ型。

分散データ (distributed data)
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に完全な、ローカルからアクセスできるデータベースに見えるが、実際にはデータ通信ネットワークで接続された複数の場所にあるデータベースで構成されているデータベース。

分散リレーショナル・データベース体系 (Distributed Relational Database Architecture™)
DRDA を参照。

分散作業単位 (distributed unit of work)
分散リレーショナル・データベース処理の

形式。ユーザーやアプリケーション・プログラムが 1 つの作業単位内で、複数の場所でデータの読み取りや更新を行える。1 つの作業単位内では、1 つのシステムで実行されている QMF などのアプリケーションが、複数のリモート・データベース・システムに対して、これらのシステムがサポートしている SQL を使用して SQL 要求を送信できる。要求は、3 部構成の表名または視点名を含む QMF コマンドを介して実行される。3 部構成の名前を含む QMF コマンドは、DB2 for VM or VSE データベースに送信したり、QMF for TSO が DB2 for z/OS ストアード・プロシージャーとして開始された場合に使用したりすることはできない。QMF コマンドの 3 部構成の名前は、ラージ・オブジェクト (LOB) データ・タイプを含む表を参照することもできない。

2 バイト文字セット (double-byte character set)
DBCS を参照。

倍精度浮動小数点数 (double-precision floating-point number)

実数に関する 64 ビットの近似的表現。

DRDA (分散リレーショナル・データベース体系) (Distributed Relational Database Architecture (DRDA))

リモート・データへの透過的なアクセスを提供する形式とプロトコルを定義する体系。DRDA は、アプリケーション・リクエスト関数とアプリケーション・サーバー関数の 2 種類の関数を定義する。

環境 (environment)

関数のパフォーマンスをサポートするために使用される論理および物理リソースの名前付きコレクション。

出口ルーチン (exit routine)

別のプログラムから制御を受け取って特定の関数を実行するプログラム。

Extensible Markup Language

XML を参照。

拡張構文 (extended syntax)

呼び出し可能インターフェース・アプリケーションで QMF SET GLOBAL コマンド、QMF GET GLOBAL コマンド、および特定の関数呼び出しに使用される構文。

拡張構文は、アセンブラー、C、COBOL、Fortran、または PL/I 言語で記述された、QMF 呼び出し可能インターフェース・アプリケーションによって使用されるパラメーターを定義する。

フォールバック (fallback)

現行リリースへの移行を試みたか、または完了した後、ソフトウェア・プログラムの前のリリースに戻る処理。

フェッチ (fetch)

データベースまたはファイルから行を検索して QMF DATA オブジェクトを作成する処理。QMF は、DSQSMRFI パラメーターを介した複数行フェッチをサポートしている。

外部キー (foreign key)

リレーショナル・データベースで、別の表の主キーを参照する表内のキー。

図形データ表示管理プログラム (GDDM)

(Graphical Data Display Manager(GDDM))

ディスプレイ・デバイスまたはプリンターに出力するテキストと図形を定義して表示するグラフィック・ソフトウェア。

大域変数 (global variable)

名前付きエンティティ。その値は、QMF セッションの間はデフォルトで保持される。QMF は、大域変数を使用してセッションとデータベース・アクティビティの両方を管理する。大域変数によっては、SET GLOBAL コマンドで設定できるものもあれば、現在の QMF セッションの状態に関する情報を記録するため、SET GLOBAL コマンドでは設定できないものもある。

グラフィック・ストリング (graphic string)

一連の 2 バイト文字セット (DBCS) 文字。

図形データ表示管理プログラム (Graphical Data Display Manager)

GDDM を参照。

ホスト (host)

データ通信構成内の制御または最上位システム。

ハイパーテキスト・マークアップ言語 (hypertext markup language (HTML))

Standard Generalized Markup Language (SGML) に準拠し、主にハイパーテキスト・リンクを含むテキストおよび図形情報のオンライン表示をサポートするために設計されたマークアップ言語。

ハイパーテキスト・マークアップ言語 (hypertext markup language)

HTML を参照。

ICU (対話式図表ユーティリティ) (ICU (Interactive Chart Utility))

プログラマーではなくても図表、グラフ、ダイアグラムを表示、印刷、またはプロット可能にする、IBM の図形データ表示管理プログラム (GDDM) のメニュー方式のコンポーネント。

ID 列 (identity column)

表に挿入される行ごとに数値を自動的に生成する方法を DB2 データベース・マネージャーに提供する列。ID 列は、AS IDENTITY 節を使用して定義される。ID 列は、1 つの表に 1 つのみ定義できる。

索引 (index)

キーの値によって論理的に順序付けされるポインターのセット。索引は、データに対する迅速なアクセスを提供し、表の行についてキー値の固有性を強制できる。

内部結合 (inner join)

結合されている両方の表の一致する行のみを含める結合演算の結果。外部結合 も参照。

インストール検査手順 (installation verification procedure)

IVP を参照。

統合交換形式 (Integrated Exchange Format)

IXF を参照。

対話式図表ユーティリティ (Interactive Chart Utility)

ICU を参照。

対話式システム生産性向上機能 (Interactive System Productivity Facility)

ISPF を参照。

ISPF (対話式システム生産性向上機能) (ISPF (Interactive System Productivity Facility))

フルスクリーン・エディターおよびダイアログ・マネージャーとして機能する IBM ライセンス・プログラム。アプリケーション・プログラムを記述するために使用され、標準スクリーン・パネルの生成や、アプリケーション・プログラマーと端末ユーザー間での対話式対話の方法を提供する。

IVP (インストール検査手順) (IVP (installation verification procedure))

製品が正しくインストールされていることを検証するプロシージャまたはプログラム。

IXF (統合交換形式) (IXF (Integrated Exchange Format))

さまざまなソフトウェア・プロダクト間で表データを転送するためのプロトコル。

ジョブ制御言語 (job control language (JCL))

オペレーティング・システムに対してジョブを識別し、ジョブの要求を記述するコマンド言語。

ジョブ制御言語 (job control language)

JCL を参照。

結合 (join)

一致する列の値に基づいて複数の表からデータを検索できるようにする SQL 関係演算。

キー (key)

表、索引、または参照制約の記述の中で識別される、列または列の順序付きコレクション。同じ列を複数のキーの一部とすることができる。

キーワード (keyword)

プログラミング言語、人工言語、アプリケーション、またはコマンドの定義済みワードの 1 つ。

キーワード・パラメーター (keyword parameter)

キーワードと続く 1 つ以上の値で構成されるパラメーター。定位置パラメーターも参照。

ラージ・オブジェクト (large object)

LOB を参照。

リンク・エディット (**link-edit**)

リンケージ・エディターを使用して、ロード可能なコンピューター・プログラムを作成すること。

リンケージ・エディター (**linkage editor**)

モジュール間での相互参照を解決することによって、また、必要に応じてアドレスを調整することによって、1 つ以上のオブジェクト・モジュールまたはロード・モジュールからロード・モジュールを作成するコンピューター・プログラム。

リテラル (**literal**)

値がその文字自体によって与えられる文字ストリング。例えば、数値定数 7 は値 7 を持ち、文字定数 'CHARACTERS' は値 CHARACTERS を持つ。

線形手順 (**linear procedure**)

QMF コマンドまたはコマンド同義語のシーケンス・セット。複数の演算を一度に実行するために使用できる。ロジックを持つプロシージャー (*procedure with logic*) も参照。

線形構文 (**linear syntax**)

あるプログラムまたはプロシージャーの 1 ステートメントに入力するか、または QMF コマンド行に入力できる QMF のコマンド構文。

ロード・モジュール (**load module**)

主記憶域にロードして実行するのに適した形式のプログラム。

LOB ラージ・オブジェクト (**large object**) (**LOB (large object)**)

サイズが 0 バイトから 2 ギガバイト (マイナス 1 バイト) の範囲のサイズを持つ一連のバイト。バイナリー・ラージ・オブジェクト (BLOB)、文字ラージ・オブジェクト (CLOB、1 バイト文字のみか、または 1 バイトと 2 バイト文字を混合したものを含めることが可能)、および 2 バイト文字ラージ・オブジェクト (DBCLOB) という 3 種類の LOB データ・タイプがある。QMF は、最大 32 KB の LOB 列サイズをサポートする。

ローカル (**local**)

QMF が現在実行されているシステムにイ

ンストールまたは保管されているデータベース、オブジェクト、またはアプリケーションに関する表現。

場所 (**location**)

分散リレーショナル・データベース・システム内の特定のリレーショナル・データベース・サーバー。各ロケーションは固有のロケーション名を持つ。

場所名 (**location name**)

データベース・サーバーの固有の名前。アプリケーションは、ロケーション名を使用して DB2 データベース・サーバーにアクセスする。

ロック (**lock**)

一連のイベントをシリアライズする、またはデータへのアクセスをシリアライズする方法。

ログ (**log**)

システムで発生するイベントを順番に記述するレコードの集合。

LUW Linux、UNIX、および Windows の省略形。

各国語機能 (**National Language Feature**)

NLF を参照。

新機能モード

NFM を参照。

NFM (新機能モード)

QMF バージョン 8.1 と QMF バージョン 9.1 のインストール・モードの 1 つで、QMF オブジェクト・カタログ内の所有者名およびオブジェクト名をデータベースで許可されている最大長に制限するモード。QMF バージョン 12.1 では、所有者名とオブジェクト名もデータベースが許可している長さがにできる。互換モード (CM) も参照。

NLF (各国語機能) (**NLF (National Language Feature)**)

QMF で使用できる複数のオプション・フィーチャーのいずれか。NLF によってユーザーは、特定のネイティブ言語で QMF と対話できる。

オブジェクト (**object**)

名前付きの記憶域スペースであり、スペー

ス、場合によってはデータを記述する特定のセットから成る。オブジェクトは、記憶域内のスペースを占めるすべてのものを指し、ライブラリーやディレクトリーに配置可能で、機密保護機能を設定可能であり、それに対して定義された操作を実行できる。QMF オブジェクト も参照。

外部結合 (outer join)

結合されている両方の表の一致する行を含み、結合されている表の一致しない行のいくつか、またはすべてを保持する結合演算の結果。内部結合 も参照。

パッケージ (package)

プログラムの準備中に作成される制御構造データベース・オブジェクト。静的 SQL ステートメントまたは XQuery 式の実行可能な書式と、動的 SQL ステートメントの実行可能な書式の配置ホルダーの両方を含むことができる。

パネル (panel)

画面上の情報の定様式表示。項目フィールドを含めることができる。

パラメーター (parameter)

関数、コマンド、またはプログラムに渡される値またはリファレンスで、入力または制御のアクションとして機能する。値はユーザーによって、または別のプログラムやプロセスによって提供される。

区画 (partition)

ページ・セットの一部。各区画は、単一の、独立して拡張可能なデータ・セットに対応している。区画は、区画化されたページ・セットの区画数に応じて、最大 1、2、または 4 ギガバイトまで拡張できる。特定のページ・セットの区画は、すべて同じ最大サイズを持つ。

プラン (plan)

アプリケーション・プラン を参照。

定位置パラメーター (positional parameter)

他のパラメーターと相対的な、決まった位置に指定しなければならないパラメーター。キーワード・パラメーター も参照。

精度 (precision)

有効数字の総数を表す数値の属性。

述部 (predicate)

比較演算を記述または暗示する検索条件の要素。

1 次許可 ID (primary authorization ID)

z/OS 用 DB2 に対してアプリケーション・プロセスを識別するために使用される許可 ID。

主キー (primary key)

リレーショナル・データベースで、データベース表の 1 つの行を一意的に識別するキー。

特権 (privilege)

SQL では、GRANT ステートメントを処理することによりユーザーに与えられる機能。

プロシージャ (procedure)

1 つ以上のタスクを実行するために使用されるステートメントまたはコマンドのシーケンス・セット。線形プロシージャとロジックを持つプロシージャ も参照。

ロジックを持つプロシージャ (procedure with logic)

1 つ以上のタスクを実行するステートメントのセット。REXX コメントで開始し、条件付きロジック (REXX を使用)、計算、ビルド・ストリング、および TSO コマンドまたは CICS コマンドを許可するロジックを持つプロシージャ。線形手順 も参照。

プロファイル (profile)

ユーザーのセッションの特性に関する情報が含まれているオブジェクト。

プログラムの一時修正 (program temporary fix)

PTF を参照。

指示照会 (prompted query)

ユーザー提供のパラメーターによって制御されるメニュー方式の照会。

PTF (program temporary fix)

System i[®]、System p、および System z[®] の各製品について、IBM によってテストされた修正。すべてのお客様が利用できる。

QBE (例示照会) (QBE (Query-by-Example))

照会を図形によって作成できるようにする QMF のコンポーネント。

QMF 管理者権限 (QMF administrator authority)

ユーザーに Q.PROFILES 制御表での行の挿入や削除を許可する権限。この権限を持つユーザーは、他のユーザーが所有する QMF 照会、書式、プロシージャで、SAVE、ERASE、IMPORT、EXPORT、および DISPLAY コマンドを実行できる。その所有者にこれらのオブジェクトをすべてのユーザーと共有するように強制する必要がない。QMF は、初期化中に各ユーザー ID の管理者権限をチェックする。このチェックは DSQUOPTS 出口ルーチンか、または別の任意のプログラムに DSQEC_DISABLEADM 変数を設定することで無効にできる。

QMF 管理者 (QMF administrator)

QMF 管理者権限を持つユーザー。

例示照会 (Query-by-Example)

QBE を参照。

QMF オブジェクト (QMF object)

QMF ユーザーがデータを照会、形式化、および表示するために、または QMF とデータベースとの間の対話を管理するために使用するオブジェクト。QMF オブジェクトは、照会および照会結果データ、書式、プロシージャ、報告書、図表、および QMF プロファイルを含む。各 QMF オブジェクトは、オブジェクトを表示するために使用される名前付き一時記憶域を持つ。報告書と図表を除くすべてのオブジェクトをデータベースに保存できる。報告書と図表は、ユーザー要求時に、特定の QMF 書式仕様をデータベースから返された結果データに適用することにより、動的に作成される。一時記憶域 も参照。

QMF オブジェクト・カタログ (QMF object catalog)

QMF 照会、プロシージャ、書式、フォルダー、および分析オブジェクトに関する情報を保管するコントロール表の集合。これらの管理表は、Q.OBJECT_DIRECTORY、Q.OBJECT_DATA、および Q.OBJECT_REMARKS を含む。

修飾子 (qualifier)

QMF オブジェクトについて用いられるときは、オブジェクトの所有者または場所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。例えば、'TCK'、'XYZ'、および 'QUERY' は、すべてデータ・セット名 'TCK.XYZ.QUERY' の修飾子である。

照会 (query)

特定の条件を基準に、データベースの情報を要求すること。例えば、顧客表の残高が 1000 ドルを超える全顧客のリストの要求など。QMF では、照会は Prompted Query、QBE、または SQL 照会パネルから実行された SQL ステートメントも、これらのステートメントが情報を要求しない (SELECT ステートメント) の場合でも参照する。

リソース管理テーブル (resource control table (RCT))

CICS トランザクションと DB2 リソースとの間の関係を定義する DB2 管理テーブル。

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・データベースを編成し、リレーショナル・データベースへのアクセスを提供するハードウェアとソフトウェアの集合。

オンライン・リソース定義 (resource definition online (RDO))

CICS で、CICS を実行しながらユーザーが特定の CICS リソースを対話式に定義できるようにする機能。具体的には、RDO によってユーザーは端末、プログラム、およびトランザクションを対話式に定義できる。

レコード (record)

行またはその他のデータの記憶表現。

レコード長 (record length)

行やその他のデータを表す記憶の長さ。

再入可能 (reentrant)

すべてのデータベース・スレッドに対して 1 つの共有コピーとして記憶域に常駐でき

る実行可能コード。再入可能コードは、自己書き換えコードではなく、スレッドごとに別々の記憶域を提供する。

参照制約 (referential constraint)

指定された外部キーの NULL 以外の値は、親表の主キーの値としても表現される場合のみ有効であるという要件。参照制約は、常に従属ファイルの視点から定義される。

リレーショナル・データベース (relational database)

表のセットとして見なすことができ、データのリレーショナル・モデルに従って取り扱われるデータベース。各データベースは、データの論理構造と物理構造を記述するシステム・カタログ表のセット、データベースに割り振られているパラメーター値を含む構成ファイル、および実行中のトランザクションとアーカイブ可能なトランザクションによるリカバリー・ログを含む。

リレーショナル・データベース管理システム (relational database management)

RDBMS を参照。

リモート (remote)

QMF が現在実行しているシステム以外のシステムにインストールまたは保管されているデータベース、オブジェクト、またはアプリケーションに関する表現。QMF CONNECT コマンドを使用して、リモート・サーバーでオブジェクト (QMF の照会、書式、プロシージャ、フォルダー、および分析オブジェクト) にアクセスできる。リモート・ロケーションから、表または視点のみにアクセスする場合は、3 部構成の表名、または視点名で QMF コマンドを使用することもできる。リモート・アクセスは、QMF for TSO が DB2 for z/OS ストアード・プロシージャとして実行されている場合は許されない。

リモート作業単位 (remote unit of work)

QMF などのアプリケーション・プログラムが、1 つの作業単位内で 1 つのリモート・データベースのデータへアクセスが行える分散リレーショナル・データベース処理の形式。接続は、QMF CONNECT コ

マンドによって確立される。CONNECT コマンドは、QMF for TSO が DB2 for z/OS ストアード・プロシージャとして実行されている場合には使用できない。

リクエスター (requester)

アプリケーション・リクエスター を参照。

リソース (resource)

ロックまたはクレームのオブジェクト。表スペース、索引スペース、データ・パーティション、索引パーティション、またはロジカル・パーティションが挙げられる。

リソース・コントロール表 (resource control table) RCT を参照。

オンライン・リソース定義 (resource definition online)

RDO を参照。

再構造化拡張実行プログラム (Restructured Extended Executor)

REXX を参照。

再構造化拡張実行プログラム (Restructured Extended Executor (REXX))

汎用高水準プログラミング言語。特にパーソナル・コンピューティングのための EXEC プロシージャまたはプログラムに適している。

ロールバック (roll back)

SQL ステートメントによって変更されるデータを、その最後のコミット・ポイントの状態にリストアすること。複数のステートメントを含むが COMMIT ステートメントを含まない照会で障害が発生した場合は、QMF セッションに影響を与えるステートメント (SET など) を除くすべてのステートメントがロールバックされる。1 つ以上の COMMIT ステートメントを含む照会で障害が発生した場合、正常に実行された前回の COMMIT ステートメントより後で行われた更新はすべてロールバックされる。いずれの場合も、障害発生後に照会は終了する。

ルーチン (routine)

プログラム、またはプログラムによって呼び出される一連の命令。通常、ルーチンは用途が広く、頻繁に使用される。

行 (row)

表の水平方向のコンポーネントで、表の列ごとに 1 つ、一連の値で構成されている。

実行時変数 (runtime variable)

プロシージャまたは照会に含まれる変数。その値は、プロシージャまたは照会の実行時にユーザーによって指定される。実行時変数の値は、現行のプロシージャまたは照会でのみ使用できる。大域変数も参照。

1 バイト文字セット (single-byte character set (SBCS))

各文字が 1 バイトコードで表現されるコード化文字セット。最大 256 文字の表現を許容する 1 バイト・コード・ポイント。2 バイト文字セットも参照。

スカラー関数 (scalar function)

オプションで引数を受け付け、呼び出されるたびに単一のスカラー値を返す SQL 関数。スカラー関数は、SQL ステートメントの中で式が有効なあらゆる場所で参照できる。

スクラッチパッド域 (scratchpad area)

プログラムの実行中に、アプリケーション・プログラムからの情報を保持するため、会話型処理で使用される作業領域。

検索条件 (search condition)

表から行を選択するための基準。検索条件は、1 つ以上の述部で構成される。

2 次許可 ID (secondary authorization ID)

z/OS 用の DB2 で、許可出口ルーチンにより 1 次許可 ID に関連付けられる許可 ID。1 次許可 ID も参照。

セグメント表スペース (segmented table space)

セグメントと呼ばれる同じサイズのページ・グループに分割される表スペース。セグメントは、別の表の行が同じセグメントに保管されないように表に割り当てられる。表スペースも参照。

サーバー (server)

アプリケーション・サーバーを参照。

セッション (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 間で行うすべての対話。

シフトイン文字 (shift-in character)

EBCDIC システムで使用される制御文字 (X'0F')。後続のバイトが、SBCS 文字を表すことを示すため使用される。シフトアウト文字も参照。

シフトアウト文字 (shift-out character)

EBCDIC システムで使用される制御文字 (X'0E') で、後続のバイトが、次のシフトイン制御文字まで、DBCS 文字を表すことを示す。シフトイン文字も参照。

1 バイト文字セット (single-byte character set)

SBCS を参照。

単精度浮動小数点数 (single-precision floating-point number)

実数に関する 32 ビットの近似的表現。

SQL (構造化照会言語) (SQL (Structured Query Language))

リレーショナル・データベースにあるデータの定義および操作のための標準化言語。

SQL 許可 ID (SQL authorization ID)

SQLID を参照。

SQL 接続 (SQL connection)

アプリケーション・プロセスとローカルまたはリモート・サーバー、またはデータベース・サーバー間のアソシエーション。リモート作業単位、分散作業単位も参照。

SQL 関数 (SQL function)

SQL ステートメントと SQL PL ステートメントのサブセットを使用することにより完全に実装される関数。

SQL 許可 ID (SQL authorization ID) (SQL ID (SQL authorization ID))

z/OS 用の DB2 で、いくつかの状況で動的 SQL ステートメントの許可のチェックに使用される ID。

SQL 戻りコード (SQL return code)

SQL ステートメントが前回 1 つ以上の警告、またはエラーなしで正常に完了したかどうかを示す SQLSTATE または SQLCODE。

構造化照会言語通信域 (Structured Query Language Communication Area (SQLCA))

アプリケーション・プログラムに、その SQL ステートメントの実行に関する情報、またはデータベース・マネージャーからの要求を提供する一連の変数。エラーが SQL コードに関連する場合、QMF メッセージ・ヘルプ (ヘルプ・キーを押して選択可能)が SQLCA の内容を表示する。

ストアード・プロシージャ (stored procedure)

ホスト言語ステートメントと SQL ステートメントの両方を含む演算を実行するため、SQL CALL ステートメントを使用して呼び出すことができるルーチン。

ストアード・プロシージャ・インターフェース (stored procedure interface)

QMF for TSO へのインターフェース。QMF を DB2 for z/OS ストアード・プロシージャとして開始し、QMF 照会または必要な作業を実行するプロシージャの名前を渡し、トレース出力の結果セットを含む最大 21 個の結果セットを受け取ることができる。QMF for TSO は、DB2 for z/OS ストアード・プロシージャを実行可能なあらゆる製品からこの方法で開始できる。

構造化照会言語 (Structured Query Language)

SQL を参照。

構造化照会言語通信域 (Structured Query Language Communication Area)

SQLCA を参照。

副照会 (subquery)

別の照会の WHERE 節または HAVING 節に表現される SQL 照会一式。

置換変数 (substitution variable)

(1) 値が大域変数または実行時変数のいずれかによって指定される、プロシージャまたは照会内の変数。(2) 値が大域変数によって指定される、QMF 書式の変数。

サブストリング (substring)

文字ストリングの一部。

サブシステム (subsystem)

z/OS 用の DB2 では、リレーショナル・データベース管理システム (RDBMS) の個別インスタンス。

表 (table)

リレーショナル・データベースで、固定数の列で構成されるデータベース・オブジェクト。順序付けられていない行セットの保管に使用される。基本表 も参照。

表スペース (table space)

データベース内の記憶域の論理単位。z/OS 用の DB2 では、表スペースは、ページ・セットであり、1 つ以上の表を含むことができる。DB2 for Linux, UNIX, and Windows では、表スペースはコンテナの集合であり、表のデータ、索引、長フィールド、および LOB の部分は、同じ表スペースに保管することも、別々の表スペースに保管することもできる。

一時記憶域 (temporary storage)

ユーザーが作業中に QMF オブジェクトを一時的に保管するために使用する記憶域。データベースをそれ以上検索せずに、簡単にオブジェクトにアクセスできる。QUERY、DATA、FORM、PROC、REPORT、CHART、または PROFILE の 7 つの一時記憶域がある。照会結果データ (DATA オブジェクト) を除き、これらの記憶域にある QMF オブジェクトは、SHOW コマンドの後に一時記憶域の名前を指定して実行することにより表示できる。DATA 領域の内容は直接表示できないが、ユーザーは、SHOW REPORT または SHOW CHART コマンドを発行して、FORM 領域の現行書式の仕様で書式化された照会データを表示できる。『QMF オブジェクト』、『現行オブジェクト』も参照。

一時記憶域キュー (temporary storage queue)

CICS の、あらゆる順序で読み取りおよび再読み取りが可能なデータ項目のキュー。キューは、タスクにより作成され、同じタスクまたは別のタスクによって削除されるまで存続する。一時データ・キュー も参照。

スレッド (thread)

DB2 の構造であり、アプリケーションの接続を記述し、その進行を追跡し、リソース関数を処理し、さらに DB2 リソースとサービスへのアクセス可能性の範囲を決定

する。ほとんどの DB2 機能は、スレッド構造のもとで実行される。

3 部構成の名前 (three-part name)

表、視点、または別名のフルネーム。場所名、許可 ID、およびオブジェクト名で構成され、それぞれピリオドで区切られている。3 部構成の名前を含む QMF コマンドは、DB2 for z/OS データベースからのみ開始することができ、DB2 (VM/VSE 版) を除くすべてのデータベースに送信できる。QMF for TSO が DB2 for z/OS ストアード・プロシージャーとして開始されている場合、3 部構成の名前を持つ QMF コマンドはサポートされない。

タイム・シェアリング・オプション (Time Sharing Option)

TSOを参照してください。

トレース (trace)

コンピューター・プログラムまたはトランザクションの処理の記録。トレースから収集された情報は、問題とパフォーマンスの評価に使用される。

トランザクション (transaction)

1 つ以上のアプリケーション・プログラムで構成される処理単位。1 つ以上のオブジェクトに影響し、単一の要求によって開始される。

一時データ・キュー (transient data queue)

CICS 記憶域。オブジェクトは、後続の内部処理または外部処理のために、ここに保管される。一時データ・キューも参照。

トリガー (trigger)

単一の基本表または視点に関連付けられたデータベース・オブジェクト。規則を定義する。規則は一連の SQL ステートメントで構成され、関連付けられた基本表または視点で挿入、更新、または削除のデータベース操作が発生すると実行される。

TSO (タイム・シェアリング・オプション) (TSO (Time Sharing Option))

ユーザーがシステムと対話式に作業できるようにする、z/OS オペレーティング・システムの基本エレメント。

2 フェーズ・コミット (two-phase commit)

外部サブシステム内のリカバリー可能リソ

ースをコミットする 2 段階の処理。最初の段階では、データベース・マネージャー・サブシステムがポーリングされ、コミットの準備ができていることが確認される。サブシステムすべてがポジティブに応答した場合、データベース・マネージャーはこれらのサブシステムにコミットするよう指示する。

ユーザー定義関数(user-defined function (UDF))

CREATE FUNCTION ステートメントを使用して DB2 データベース・システムに定義される関数。定義後は、SQL ステートメントで参照できる。UDF は SQL 関数でも外部関数でもよい。

ユニコード (Unicode)

文字エンコード規格。世界の共通語で記述されたテキストに加え、一部の古典的および伝統的テキストの交換、処理、および表示をサポートする。ユニコード規格には、ISO 10646 で規定された 16 ビットの文字セットがある。

リカバリー単位 (UR) (unit of recovery (UR))

整合点と整合点の間にある作業単位内の一連の操作。

作業単位 (UOW) (unit of work (UOW))

アプリケーション・プロセス内のリカバリー可能な一連の操作。アプリケーション・プロセスは任意の時点で単一の作業単位であるが、その活動中には、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わる場合がある。マルチサイト更新操作では、単一の作業単位が複数の回復単位を含む場合がある。複数のステートメントを含むが COMMIT ステートメントを含まない QMF SQL 照会では、すべてのステートメントが単一の作業単位で構成されるため、セッションに影響するステートメント (SET など) を除くすべてのステートメントが障害発生時にロールバックされる。複数のステートメントと 1 つ以上の COMMIT ステートメントを含む QMF SQL 照会では、作業単位は、1 つの COMMIT ステートメントに加えて、照会の開始時点または前回の COMMIT ステートメントにまでさかのぼるすべての先行ステートメントで構成される。障害が発生した場合、正常に実行された前回の

COMMIT ステートメントより後で行われた更新はすべてロールバックされる。

ユーザー定義関数 (user-defined function)

UDF を参照。

視点 (view)

基礎となる一連の表に保管されているデータに基づく論理表。視点によって返されるデータは、基礎表で実行されている SELECT ステートメントによって決定される。

XML (Extensible Markup Language)

Standard Generalized Markup Language(SGML) に基づくマークアップ言語を定義するための標準メタ言語。

z/OS 64 ビットの実記憶を使用する IBM メインフレーム・オペレーティング・システム。

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

QMF アプリケーション・プランのパッケージに対する 225
値、変数 217

アプリケーション・コード・ページ

GDDM との CCSID の非互換性 64, 75

アンパーサンド (&)

コマンド同義語内 297

暗黙作成表スペース 255

移行

各国語機能 (NLF) 142

異常終了

DCB インターセプト、予備ファイルについての考慮事項
447

以前のリリースとの互換性 116

一時記憶域キュー

QMF サービスを使用する印刷 284

一時データ・キュー

出力の経路指定 273

QMF サービスを使用する印刷 284

印刷

エラー 443

ユーザーに使用可能にする 273

GDDM サービスの使用 274

QMF と GEM 273

印刷プレビュー機能 292

印刷報告書のプレビュー 292

インストール

オブジェクトおよびプログラムの移行 113

概要 15

各国語機能 (NLF) 135

計画

参照： インストールの計画

現在のインストール・モードの判断 20

サーバー・データベース・インストール 81

作成されるリソース 17

サポートされるデータベース・リリース 3

終結処置 119

情報ロードマップ

参照： インストール・ロードマップ

ストアード・プロシージャ・インターフェース

(TSO) 143

制約事項 5

前提条件 45

テストおよび検査 93

インストール (続き)

必要な権限 17

前のリリースのモード 4

リモート作業単位に必要なオブジェクト 8

QMF の各種のリリースの共存 5

インストール計画

必須ロケーション 8

インストール検査手順

サーバー・データベース 109

ストアード・プロシージャ・インターフェース 143

リクエスト・データベース 93

インストール後の終結処置 119

基本 QMF (英語) 119

NLF 135

インストールの計画 3

インストールの計画

インストール・モード 4

インストールのロードマップ 15

前提条件 45

分散データ構成

分散作業単位 (distributed unit of work) 10

リモート作業単位 6

QMF の各種のリリースの共存 5

インストールのテスト

サーバー・データベース 109

ストアード・プロシージャ・インターフェース 143

リクエスト・データベース 93

インストールの前提条件

概念 16

権限 17

ソフトウェア 45

データベース・リリース 3

ハードウェア 45

インストールの要件

概念 16

権限 17

ソフトウェア 45

データベース・リリース 3

ハードウェア 45

インストールのロードマップ

一般的なインストール手順 15

サーバー・データベース

概要 27

3 部構成の名前によるアクセス 39

スタンドアロン構成 20

多文化サポート 42

分散作業単位構成 12

リクエスト・データベース (DB2 (z/OS 版))

概要 20

新規 QMF インストール 23

リモート作業単位構成 8

インストーラのロードマップ (続き)
3 部構成の名前によるアクセスのみ 39
CONNECT コマンドと 3 部構成の名前によるアクセス 20,
27
インストーラ・パッケージ
必須ロケーション 8
インストーラ・ロードマップ
概要 15
各国語機能 42
3 部構成の名前によるアクセス 39
DB2 (z/OS 版) リクエスト
概要 20
新規 QMF インストール 23
インターフェース制御ブロック
DXEGOVA 380
DXEXCBA 380
ウィンドウ・パネル
カスタマイズしたファンクション・キーの例 318
ID 308
エラー
初期化 436
参照: 障害追及
オブジェクト
共有 267
コントロール表 (control tables) 259
最終使用日 176, 259
削除 269
内部表現 259
名前
サポートされる長さ 4
名前、コマンド同義語 294
表示 267
保守 259
リスト 267
ウィンドウ ID 308
カスタマイズ 246
オブジェクト所有権の移行 268
オブジェクトの最終使用日 176, 259
オブジェクト・カタログ
DSQDBCTL データベース 18
QMF の各種のリリースの共存 5
QMF リリース間の互換性 4
オブジェクト・リスト
大域変数を使用したカスタマイズ 246
オンライン・ヘルプ
メッセージのサポート
正の SQL コードの処理の設定 451

[カ行]

カーソル固定オプション 244
拡張形式 10 進浮動小数点
参照: DECFLOAT データ・タイプ
拡張浮動小数点、編集ルーチン 326
仮想記憶域
参照: 記憶域

カタログ
QMF オブジェクト
参照: オブジェクト・カタログ
カタログ視点
必須ロケーション 8
各国語フィーチャー (NLF)
移行
プロシージャー 142
インストーラ
プロシージャー 135
ロードマップ 42
ストアード・プロシージャー・インターフェースで指定 160
ストアード・プロシージャー・インターフェースの構成 143
DBCS デバイス要件 45
紙送り制御文字
結果セットからの除去 160
環境
カスタマイズ 226
デフォルトの設定 517
管理
必要な権限 18
表、作成 253
ユーザー・プロファイルおよびオブジェクト 238
管理者権限
参照: QMF 管理者権限
管理プログラム、DB2
参照: リソース限定機能、DB2
管理プログラム出口ルーチン
入り口点 369
機能呼び出し 378
機能呼び出しのタイプ 370
コマンド処理 373
スクラッチパッド域 394
制御情報、保管 394
制御の流れ 379
制御ブロック
DXEGOVA 380
DXEXCBA 385
説明 357
出口ルーチン情報 385
取り消しサービス 399
パフォーマンス 378
プログラム構造 379
リソース・グループの指定 364
リソース・コントロール表 357
CICS 制御ブロック・インターフェース 379
DB2 リソース限定機能との比較
参照: リソース限定機能、DB2
管理プログラム出口ルーチン QMF
リソース制御情報を渡す 380
キーワード、問題の報告 467
記憶域
拡張のためのモジュールの移動 52
障害追及 447
表スペース、サイズの増加 265
編集ルーチンからのデータ 321

- 記憶域 (続き)
 - 報告書操作
 - プログラム・パラメーター 198
 - 予備データ 200, 447
 - 要件
 - 初期化 51
 - 報告書操作 51
 - 領域サイズ 143
- 記憶域関連問題
 - 解決 445
- 記憶域の割り振り
 - 可変 (TSO) 199
 - 固定 (TSO、CICS) 199
 - 動的 (TSO) 200
 - 比率 (TSO) 199
 - 予備データ
 - 概要 200
 - 拡張記憶域 (TSO) 201
 - ファイル (CICS) 206
 - ファイル (TSO) 202
- 規則
 - コマンド同義語 294
 - ファンクション・キーのカスタマイズ 314
- 機能拡張
 - 旧リリースに対する 471
- 機能呼び出し
 - タイプ 370
 - ブランチ・アドレス 378
- 機能レベル・トレース、構成 450
- 基本 QMF コマンド、同義語としての 295
- 旧 QMF リリースへのフォールバック
 - 後方互換性についての考慮事項 113
 - CM リリースにフォールバックするためのプロシージャ
 - サーバー 90
- 行、データベース
 - 検索数の制御 362
- 競合、リソースの
 - 同時アクセスの解決のオプション 449
 - 非コミット読み取りとカーソル固定 244
- 共存
 - 異なるインストール・モード 5
- 強調表示規則 xii
- 強調表示の規則 xii
- 許可
 - エラー 436
 - コマンド同義語 304
 - QMF 管理者権限
 - 参照: QMF 管理者権限
 - QMF へのアクセス 233
- 許可 ID 18
 - 現行アプリケーションの戻りリスト 514
 - 個別ユーザーとグループの登録 225
 - サポートされる長さ 4, 8
 - 修飾名内の位置 10
 - 初期化時の認証 143
 - 初期プロシージャの実行 171
- 許可 ID (続き)
 - 2 次
 - 参照: 2 次許可 ID
 - 2 次許可 ID でのインストール 17, 43, 54, 60, 62, 83, 85, 88, 89, 147
 - Q 17
 - QMF ID の要件 226
 - QMF 管理者権限の確認 175
 - Q.SYSTEM_INI の要件 182
 - RACF グループ名
 - 参照: 2 次許可 ID
 - TSO ログオン ID による認証 197
 - 参照: QMF タスクに必要な権限
 - クラス ID、ファンクション・キーのカスタマイズ 314
 - グラフィックス・プリンター、ニックネームの定義 274
 - 警告メッセージ 436, 441
 - 計算、予備ファイル・サイズの 202
 - 結果セット、報告書を受け取る 160
 - 権限、DB2
 - 配布、概要 270
 - 権限のカスケード 242
 - 言語パラメーター、ストアード・プロシージャ・インターフェース 160
 - コード、SQL
 - 参照: SQL コード
 - コード化文字セット ID (CCSID) の非互換性 64, 75
 - ご意見、IBM への送付 xiv
 - 高性能オプション/マネージャー 357
 - 構文図
 - CALL ステートメント 143
 - 構文図、読み方 xii
 - 互換モード
 - の確認 20, 27
 - QMF
 - 新機能モードとの共存 5
 - 前のリリース 4
 - ご使用条件の条項、QMF VUE 23, 25
 - コマンド
 - トレース 160
 - リソース限定機能を使用する管理 401
 - RUN
 - 同義語定義 295
 - SET PROFILE 236
 - コマンド同義語の省略 301
 - コマンド同義語表
 - 作成 299
 - 視点 303
 - 保守 302
 - コントロール表
 - Q.ERROR_LOG 463
 - コントロール表 (control tables)
 - 再編成する時期 271
 - バージョン 12.1 の前のバージョンとの互換性 4
 - バッファ・プールの切り替え 272
 - 必須ロケーション 8

コントロール表 (control tables) (続き)

- 保守
 - 環境 271
- ユーザー管理のデータ・セットの VSAM クラスター 270
- DSQDBCTL データベース 18
- QMF の各種のリリースの共存 5
- Q.RESOURCE_VIEW 361
- 参照： オブジェクト・カタログ

[サ行]

サーバー・データベース

- インストール
 - インストール・ジョブ・パラメーター 54
 - 最低限のデータベース・バージョン 3
 - 3 部構成の名前によるアクセス 39
 - CONNECT コマンド・アクセス 27
- 構成
 - 概要 5
 - 初期化の接続 6

サービス

- QMF の PSP サブセット名 429

サービス情報 xii

最終使用日、設定の動作 176, 259

作業単位

- 参照： リモート作業単位または分散作業単位

索引

- 再作成 265
- Q.PROFILES 表 226

サブシステム ID

- 指定、リモート作業単位 6

サポート情報 xii

サンプル表

- 削除
 - NLF 135
- 作成
 - NLF 135
 - NLF (QMF Analytics for TSO) 142
- NLF のインストール 135
- QMF Analytics for TSO 132

指示照会

- 印刷 274, 287
- ウィンドウ ID 308
- SQL 特権 238

システム初期化プロシージャ、デフォルト 182

システム・インターフェース、トレース 450

視点

- アクセスの制御 238
- インストール時に作成される 17
- 基本オブジェクト 258
- 作成するための特権 257
- 照会に関する特権 238
- スクリーニング・ツール 257
- 特権の付与 257
- Q.RESOURCE_VIEW、管理プログラム出口 361

自動経路指定、出力の印刷 273

シフト文字 328

終了メッセージ 465

終了呼び出し、編集ルーチン 321

出力域

- 終了の制御 321
- フォーマット設定の制御 321

照会

- インストール時に作成される 17
- 削除 269
- ステートメント長 493
- ストアード・プロシージャ・インターフェースを使用した
 - 実行 143
- 内部記憶フォーマット 263
- 表示 267
- 複数ステートメント
 - 参照： 複数ステートメント照会
- リスト 267

照会アクセラレーター 446

照会加速

- イネープリング 446

障害追及

- インストール検査手順 (IVP)
 - インポート中の DSQ21662 102
- 記憶域関連の問題 447
- 記憶域要件 447
- 診断援助機能 449
- パフォーマンス問題 446
- リソース競合 449
- 3 部構成の名前の障害 211
- SQL コード
 - 551 175

常駐の QMF プログラム 517

商標 523

初期化

- 記憶域要件 51
- 障害追及 436
- 大域変数の値 (DSQUOPTS ルーチン) 175
- 認証プロセス 143, 197, 234
- パフォーマンス 517
- メッセージ番号 450
- QMF 管理者権限の検査
 - プロセス 18
 - 無効にする 175

初期プロシージャ

- 変数値の受け取り 217

書式 321

- 印刷 287
- 印刷プレビュー 292
- ウィンドウ ID 308
- 新規編集コードの作成 326
- 内部記憶フォーマット 263
- 表示 267
- リスト 267

ジョブ制御言語

- 参照： JCL

- 所有権
 - 移行 268
 - QMF の追跡方法 259
- 新機能モード
 - の確認 20, 27
 - DB2 for z/OS
 - QMF インストール要件 3
 - QMF
 - 互換モードとの共存 5
 - 前のリリース 4
- 診断
 - 援助 449
 - 症状 449
 - ダンプ 465
 - トレース機能 454
- 数値
 - データ
 - 単精度浮動小数点のインポート 493
 - 変換、編集ルーチン 326
- スクラッチパッド域
 - 管理プログラム出口ルーチン 394
- スタンドアロン・データベース構成
 - インストール・ロードマップ 20
- ステートメント、照会
 - サポートされる長さ 493
- ストアード・プロシージャ
 - としての QMF for TSO の開始
 - インターフェースのインストール 143
 - 認証 143
 - としての TSO の開始
 - 制約事項 6
 - QMF for TSO の開始
 - 制約事項 10
- 図表
 - 印刷 273, 287
 - 特定のオブジェクト 287
 - 形式 288
- 図表機能、トレース 450
- セッション 226
- 接続、QMF インストール済み環境間で確立
 - 制約事項 6, 10
 - 分散作業単位
 - 参照：分散作業単位 (distributed unit of work)
 - リモート作業単位
 - 参照：リモート作業単位 (remote unit of work)
- 3 部構成の名前
 - 参照：3 部構成の名前
- CONNECT コマンド
 - 参照：CONNECT コマンド
- DRDA 要件 5
- 線形プロシージャ、コマンド同義語内の 296
- ソフトウェアの前提条件 45
- ソフトウェア・サポート 466

[タ行]

- ダイアログ・コマンド処理機能、トレース 450
- 大域変数
 - 印刷 284
 - ウィンドウ ID 308
 - DSQAO_STO_PROC_INT 143
 - DSQDC_POS_SQLCODE 451
 - DSQEC_BUFFER_SIZE 449
 - DSQEC_CC、ストアード・プロシージャ・インターフェース 160
 - DSQEC_CON_ACC_RES 449
 - DSQEC_DISABLEADM 175
 - DSQEC_EXTND_STG 201
 - DSQEC_LAST_RUN 176
 - QMF 開始時の初期化 175
 - QMF の開始
 - Q.GLOBAL_VARS での大域変数の初期化 179
 - Q.GLOBAL_VARS 表
 - 値の設定 179
 - Q.GLOBAL_VARS 表の設定 179
- 大域変数表
 - 構造 180
- 大規模 QMF 照会のための EDSA 制限サイズ (CICS) 80
- 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 465
 - CICS 領域サイズ 447
- 短整数データ、編集ルーチン 326
- 単精度浮動小数点数
 - サポート 493
- ダンプ、診断のための 465
- 注意
 - 特記事項 521
- 長形式 10 進浮動小数点
 - 参照：DECFLOAT データ・タイプ
- 通信、QMF インストール間で確立
 - 参照：接続、QMF インストール済み環境間で確立
- データの共用ロック
 - 参照：データのロック
- データのロック
 - 非コミット読み取りとカーソル固定 244
- データベース
 - インストールの前提条件 45
 - 権限の前提条件 17, 18, 270
- 接続
 - 権限 225
 - 初期、リモート作業単位 6
 - リモート 157
 - 参照：接続、QMF インストール済み環境間で確立
- それぞれ異なる機能 493
- パーティション・グループ
 - DSQTSCTL 18
 - DSQTSOBJ 18
- 低いパフォーマンス 446
- 非コミット読み取りとカーソル固定 244
- フェッチ・プログラム・パラメーター 209, 211

データベース (続き)

DSQDBCTL 18

GDDM との CCSID の非互換性 64, 75

QMF インストールの前提となるリリース 3

データベース・サービス機能、トレース 450

データ・セット

管理

概要 270

割り振り

ストアード・プロシージャ・インターフェース
(TSO) 143

データ・セットの割り振り

ストアード・プロシージャ・インターフェース 143

データ・タイプ

編集出口の変換 326

ディスプレイ装置

サポート 45

GDDM ニックネーム 275

トグル・スイッチ、管理プログラム出口ルーチン 358

特記事項

商標 523

注意 521

特権 236

ストアード・プロシージャ・インターフェース 143

他のユーザーの付与の取り消し 242

データベース・オブジェクト 238

QMF 管理者権限 18

QMF タスクに必要 238

STATS および REORG 270

参照：表特権

トランザクション

データベース

分離レベル 244

MRO および ISC による経路指定要求 519

取り消しサービス、管理プログラム 399

[ナ行]

名前

オブジェクト

サポートされる長さ 4

3 部構成、複数行フェッチ 211

ADMMNICK 指定 278

ニックネーム

印刷時のエラー 443

定義済み 275

複数のプリンターの定義 278

入力域

終了の制御 321

フォーマット設定の制御 321

認証

各国語機能 135

システム初期化プロシージャ (Q.SYSTEM_INI) 182

初期プロシージャ 171, 215

ストアード・プロシージャ・インターフェース (TSO の
み) 143

認証 (続き)

1 次許可 ID と TSO ID 197

2 次許可 ID

インストール、使用 17

LIST コマンドのサポート 147

QMF 管理者権限の検査

参照：QMF 管理者権限

QMF プランおよびパッケージ、アクセスの制御 225

QMF プロファイル

オープン登録 237

更新 236

制限付き登録 233

追加 234

Q.PROFILES の CREATOR 列 226

参照：QMF 管理者権限

ネットワークと QMF の構成 5

[ハ行]

パーティション・グループ、データベース

参照：データベース、パーティション・グループ

ハードウェア前提条件 45

バックアップ

作成

QMF コントロール表のリスト 497

リカバリー対象

サーバー 90

パッケージ、QMF

DB2 リソース限定機能を使用する管理 401

バッチ

照会またはプロシージャの実行 291

予備ファイルの使用 207

リモート DB2 クライアントからの実行 425

バッファ・プール、コントロール表の割り当て 272

パネル

印刷および表示サポート 518

管理プログラム・プロンプト 362

クラス ID 314

ID 307, 308

パフォーマンス

管理 446

障害追及 447, 449

常駐プログラム 517

表索引 253

予備ファイルの使用 207

DSQSIROW、大きい値 209

DSQSIROW、小さい値 209

参照：記憶域

パフォーマンス、DSQEC_BUFFER_SIZE による向上 449

パラメーター

受け渡し 187

編集ルーチンへ渡す 321

低いパフォーマンス 447

非コミット読み取りオプション 244

ビット編集コード 445

非同期処理の印刷 273

- 非同期タイマー・ルーチン 399
- 表
 - アクセスの制御 238
 - 印刷 287
 - インストール時に作成される 17
 - 索引 253
 - ファンクション・キー 307
 - リソース制御、管理プログラム出口 358
 - SAVE DATA 構成 18
- 表エディター
 - 必要な SQL 特権 238
- 表示サービス、トレース 450
- 表スペース
 - 暗黙作成表スペース 254, 255
 - 拡大 265
 - 削除 238
 - デフォルトのスペース、SAVE DATA コマンド 18
 - 表の作成 253
 - 明示 / 暗黙オプション
 - CREATE TABLE ステートメント 256
 - SAVE および IMPORT コマンド 254
 - 明示作成表スペース 254, 255
 - ユーザー・プロファイル内の指定 226
 - 割り当て 254
- 表特権
 - 概要 241
- ファミリー 1 プリンター 275, 278
- ファミリー 2 プリンター 275, 278
- ファミリー 3 プリンター 275, 278
- ファミリー 4 プリンター 275
- ファンクション・キー
 - カスタマイズ
 - アクティブ化の問題 314
 - 新規の定義のアクティブ化 316
 - ファンクション・キー表の更新 314
- 表 313
 - 作成 313
 - 定義の入力 314
- 表の索引 313
- ファンクション・キー表のコメント 313
- フィードバック、IBM への送付 xiv
- フェッチ・プログラム・パラメーター
 - 取り出される報告書の行数 209
 - 複数行フェッチ 211
- フォルダー
 - 内部記憶フォーマット 263
 - リスト 267
- 複数行フェッチ
 - プログラム・パラメーター 211
 - 3 部構成の名前の制限 10
- 複数ステートメント照会
 - 作業単位 5
 - 制約事項 225
- 複数のデータベース・スレッド 212
- 浮動小数点数
 - 単精度のインポート 493
- 浮動小数点データ、編集ルーチン 326
- ブランチ・アドレス、管理プログラム 378
- プリンター
 - 制御キーワード (PRINTCTL) 278
 - ニックネーム (GDDM) 436
 - マルチアドレス 278
 - ANSI サポート
 - グラフィック装置 273
- フルスクリーン・パネル
 - カスタマイズしたファンクション・キーの例 318
 - トレース 450
 - パネル ID 307
- プログラミング・インターフェース情報 522
- プログラム・パラメーター
 - 合計 187
 - ストアード・プロシージャ・インターフェース (TSO) 143
 - トレース
 - DSQSDBQN 222
 - DSQSDBQT 222
 - DSQSDEBUG 221
 - DSQSBSTG 199
 - DSQSDBCS 223
 - DSQSDBLG 160
 - DSQSDBNM 6, 196
 - DSQSIROW 209
 - DSQSMODE 214
 - DSQSMRFI 211
 - DSQSPILL 187, 200
 - DSQSPLAN 197
 - DSQSPRID 197
 - DSQSPTYP 187
 - DSQSRSTG 200
 - DSQSRUN 215
 - DSQSSPQN 206
 - DSQSSUBS 6, 195
- プロシージャ
 - 印刷 287
 - オブジェクトの保守 263
 - コマンド同義語で使用 296
 - ストアード・プロシージャ・インターフェースを使用した
 - 実行 143
 - 内部記憶フォーマット 263
 - 表示 267
 - リスト 267
- プロセッサ時間
 - 限界の設定 357
 - 制御使用 363
- プロファイル
 - 印刷 287
 - 更新 236, 237
 - コマンド同義語 301
 - 削除 233, 238
 - 作成 226, 233
 - 初期化の検索順序 234
 - デフォルト値 226

プロファイル (続き)
デフォルトの設定 233
トレース値
 ストアード・プロシージャ・インターフェース 160
保守 259
ログオン時の認証 143, 234
CASE 設定、カスタマイズしたファンクション・キー 315
SET PROFILE コマンド 236
プロンプト・パネル
 カスタマイズしたファンクション・キーの例 318
 パネル ID 308
分散作業単位
 インストール・ロードマップ 39
 サポートされないデータベース 493
 必須構成 10
 DRDA 要件 5
分散作業単位 (distributed unit of work)
 インストール手順 91
 インストール・ロードマップ 12
 制約事項 10
分散データ・アクセス
 概要 5
 参照： リモート作業単位または分散作業単位
分散リレーショナル・データベース・アクセス
 参照： DRDA (分散リレーショナル・データベース・アクセス)
分析オブジェクト
 内部記憶フォーマット 263
 表示 267
 リスト 267
分離レベル
 カーソル固定 244
 非コミット読み取り 244
 参照： 同時アクセス解決のサポート
ヘルプ・パネル、キーのカスタマイズ 308, 318
変更の要約 471
編集
 コード
 数値データ処理 326
 プロファイルの CASE フィールド 326
 出口インターフェース 321
 終了呼び出し 321
 出力域 321, 323
 制御ブロック・フィールド 323
 入力域 321, 323
 DBCS サポート 328
 ルーチン 321
変数
 初期プロシージャへの値の引き渡し 217
 同義語定義内 297
 &ALL の使用 297
報告書
 印刷 287
 印刷プレビュー 292
 記憶域
 可変割り振り 199

報告書 (続き)
 記憶域 (続き)
 固定割り振り 199
 障害追及 447
 動的割り振り 200
 比率割り振り 199
 予備データ 200
 参照： 記憶域
 結果セットで受け取る (ストアード・プロシージャ・インターフェース) 160
 データ・フォーマット 321
 低いパフォーマンス 447
 2 進データ 445
 Q.ERROR_LOG 表 463
報告書書式設定サービス、トレース 450
報告書操作の 64 ビット記憶域 (TSO) 201
報告書操作の拡張記憶域 (TSO) 201, 447
保護
 管理者権限の検査
 参照： QMF 管理者権限
RACF
 参照： RACF
保守
 オブジェクトの表示 267
 オブジェクトのリスト 267
 オブジェクト用の表スペースの拡大 265
 コマンド同義語表 302
 QMF 管理者権限 18

[マ行]

前の QMF リリースのインストール・モード 4, 5
無効なサブシステム ID 436
メッセージ
 行限界の超過 362
 システム・エラー・メッセージ 451
 状況、ストアード・プロシージャ・インターフェース 160
 トレース
 ストアード・プロシージャ・インターフェース 160
 正の SQL コードのサポート 451
 番号範囲とそれぞれの意味 450
 問題の診断 450
 ユーザー・アクティビティの取り消し、管理プログラム 394
問題
 参照： 障害追及
問題のトレース
 エラー・ログ報告書 463
 管理プログラム
 詳細レベル、設定 385
 取り消された初期化 394, 403
 メッセージ・ロギングの構成 385, 394
機能
 開始 455
 停止 463
 コマンド同義語 299

問題のトレース (続き)

- システム・エラー 451
 - 出力の割り振り
 - CICS 77, 98
 - TSO 66
 - 詳細レベル 457
 - 詳細レベル、設定 187, 221, 226, 450
 - 初期化 430, 431
 - Q.SYSTEM_INI プロシージャ 182
 - 既に報告されている問題の検索 467
 - 正しくないロード・モジュール 431
 - データ
 - データ・セットの割り振り 454
 - 表示 460
 - データ・セットの高位修飾子 (デフォルト) 47
 - トレースの開始 454
 - バッチ・モード・プロシージャ 417, 424, 427
 - ファンクション・キー 313, 317
 - メッセージ番号範囲とそれぞれの意味 450
 - メッセージ・ログ 436
 - 予備ファイル 204, 447
 - 割り込みの作成 452
 - CICS
 - 出力の割り振り 187
 - 診断援助機能 465
 - DSQDEBUG データ・セット 66, 77, 143
 - QMF の開始後の警告メッセージ 441
 - SQL コード、DSQDEBUG への書き込み 451
 - SQLCA 情報、表示 451
 - TSO
 - 診断援助機能 465
 - ストアード・プロシージャ・インターフェース 143, 160
- 問題報告 466

[ヤ行]

- ユーザー
 - オブジェクト 267
 - オブジェクトに関する許可 238
 - 追加 234
 - 認証 234
- ユーザー定義関数 513
- ユーザー編集ルーチン
 - さまざまなコードの処理 325
- ユーザー編集ルーチン内の SQLTYPE のデータ 323
- 呼び出し可能インターフェース
 - 一般エラー 436
- 予備データ
 - 拡張記憶域 (TSO のみ) 201
 - ファイル
 - CICS 206
 - TSO 202
- 予備ファイル
 - 計算例 202
 - サイズの見積もり 202

予備ファイル (続き)

- パフォーマンス問題 207
- DCB 異常終了のインターセプトについての考慮事項 447

[ラ行]

- リクエスター・データベース
 - インストール
 - インストール・ジョブ・パラメーター 54
 - 記憶域要件 50
 - 新規リリースをインストールするジョブ 60
 - 前提条件 45
 - 前のリリースを移行するジョブ 62
 - モジュールの移動 52
 - ロードマップ 20
 - QMF バージョン 12.1 ライブラリーのロード 47
 - TSO および CICS プリファレンス 64
 - 構成
 - 概要 5
- リストの視点
 - 作成 246
 - 2 次許可 ID 147
- リソース
 - グループ 233
 - プロファイル管理 233
- リソース競合
 - 同時アクセスの解決のオプション 449
 - 参照： 競合、リソースの
- リソース限定機能、DB2
 - 管理可能な QMF コマンド 401
 - QMF 管理プログラムとの比較 401
- リソース限定表 (DB2) 403
- リテラル、コマンド同義語の 298
- リモート DB2 クライアント
 - QMF バッチ・ジョブ 425
- リモート作業単位
 - インストール・ロードマップ
 - z/OS 以外のサーバー 27
 - z/OS サーバーのみ 20
- リモート作業単位 (remote unit of work)
 - インストール・ロードマップ 8
 - コマンド同義語表の作成 157
 - サーバーのインストール 83
 - 制約事項 6
 - 必須構成 6
 - 複数行フェッチでの使用 211
 - リモート・データベース接続のカスタマイズ 157
 - DB2 (VSE および VM 版) のデフォルト視点 247
 - DRDA 要件 5
- リモート・データ・アクセス
 - 概要 5
 - 参照： リモート作業単位または分散作業単位
- リリース間の互換性 113
- リンク・エディットの状態メント
 - 管理プログラム出口ルーチン 396, 397

列

照会でサポートされる数 493

ロード・モジュール

パフォーマンス拡張のための移動 52

ロケーション・ウィンドウ ID 308

[ワ行]

割り込み機能

使用 452

[数字]

10 進数データ、編集ルーチン 326

10 進浮動小数点データ・タイプ

参照： DECFLOAT データ・タイプ

16 進編集コード 445

2 次許可 ID

インストール 17

LIST コマンドのサポート 147

2 バイト文字セット (DBCS) サポート

編集コード 327

ラテン文字 327

3 部構成の名前

アクセスされるオブジェクト 11

インストール・パス

ジョブ 40, 91

ロードマップ 39

T3PARTNM パラメーター 81

インストール・ロードマップ 12

制約事項 40

複数行取り出し/挿入 211

VM/VSE プラットフォーム 10

データベース・サポート 493

必須構成 10, 12

リモート・アクセスの管理 401

参照： 分散作業単位

A

ABENDASRA 433

ABEND、QMF インストール時の 433

ADD モード、表エディター

管理 401

ADMADFC デフォルト・モジュール 280

ADMCFORM DD 名 288

ADMMNICK 指定 275

AEY9 異常終了 433

ALLOBJ 権限の前提条件 (iSeries) 17

APAR (プログラム診断依頼書) 429, 464, 466

APPCPG パラメーター、GDDM 64, 75

APPLDATA 列 263

ARCH(7) コンパイラー・オプション、ユーザー編集ルーチン

339, 342

ASCPUT サービス、印刷 274

AZTS 異常終了 433

B

B37 異常終了と予備ファイルについての考慮事項 447

BACKWARD コマンド

管理 401

BIGINT データ・タイプ

幅 202

SQLTYPE、ユーザー編集ルーチン 323

BINARY データ・タイプ

幅 202

SQLTYPE、ユーザー編集ルーチン 323

BLOB データ、ユーザー編集ルーチン 321

BOTTOM コマンド

管理 401

C

C トレース・レコード 450

CALL ステートメント

構文、ストアード・プロシージャ・インターフェース 143

複数ステートメント照会に対する制約事項 225

CASE 列 (Q.PROFILES) 226

CCSID 値用の特殊レジスター 64, 75

CCSID の非互換性 64, 75

CEBR トランザクション 427

CHANGE モード、表エディター

管理 401

CHAR データ・タイプ

幅 202

SQLTYPE、ユーザー編集ルーチン 323

CICS 環境

一時記憶域キュー 284

一時データ・キュー 284

一般エラー 433

管理プログラムへのインターフェース 369

記憶域要件

固定記憶域の設定 199

初期化 51

報告書操作 51

サポートされない機能 494

診断機能 465

大規模 QMF 照会のための EDSA 制限サイズ 80

トランザクション ID 436

トレース機能の使用 454

ENVIRONMENT 値、QMF プロファイル 226

HANDLE CONDITION 373

IMPORT コマンド 238

NLF カスタマイズ 135

NLF コントロール表 135

TYPETERM 項目、QMF 表示 465

CLOB データ、ユーザー編集ルーチン 321

CONFIRM 列 (Q.PROFILES) 226

CONNECT コマンド
 インストール・ロードマップ 8
 エラー 436, 441
 制約事項 6, 10
 必須構成 6
 分散データ・アクセスの概要 5
 参照： リモート作業単位 (remote unit of work)

CONVERT コマンド
 最終使用日、設定の動作 259

CREATE PROCEDURE ステートメント
 複数ステートメント照会に対する制約事項 225

CREATE TABLE ステートメント
 コマンド同義語 299
 ユーザーの表 253
 SAVE DATA に関する特権 238

CREATETS/CREATETAB 特権
 CREATE TABLE ステートメントを実行するための特権 256

CREATOR 列 (Q.PROFILES)
 定義済み 226
 プロファイル初期化の役割 234

CSD ファイル
 パネル・ファイルに合わせたカスタマイズ 433

CURRENT APPLICATION ENCODING SCHEME 特殊レジスター 64, 75

D

D トレース・レコード 450

DATA オブジェクト
 記憶域
 参照： 記憶域の割り振り
 SAVE DATA 構成 18, 238

DATE データ・タイプ
 形式設定用の編集ルーチン 330
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323

DB2
 管理プログラム 357

DB2 管理プログラム
 リソース限定表 403

DB2 特権の除去
 カスケード効果 242
 不完全な取り消し 242

DBA 権限の前提条件 (VM/VSE) 17

DBADM 権限の前提条件 18, 270

DBCLOB データ、ユーザー編集ルーチン 321

DBCS (2 バイト文字セット) サポート
 ストアード・プロシージャ・インターフェース (TSO) 143
 デバイス要件 45
 編集コード 327
 ラテン文字 327
 CCSID 64, 75

DCB 異常終了と予備ファイルについての考慮事項 447

DECFLOAT データ・タイプ
 サポートの要件 45, 321
 幅 202
 ユーザー出口ルーチンのコンパイラー・オプション
 PL/I (CEEPIPI LE インターフェース) 339
 PL/I for CICS 342
 ユーザー編集ルーチンでの変換 326
 SQLTYPE、ユーザー編集ルーチン 323

DECIMAL データ・タイプ
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323

DECOPT 列 (Q.PROFILES) 226

DEVOK キーワード、ADMMNICK 指定 278

DISPLAY TABLE コマンド
 管理 401
 記憶域
 参照： 記憶域の割り振り

DISPLAY コマンド
 最終使用日、設定の動作 259
 必要な SQL 特権 238
 QMF 管理者特権 18

DPRE (印刷された報告書の表示) 同義語 292
 カスタマイズ 293

DRAW コマンドに必要な SQL 特権 238

DRDA (分散リレーショナル・データベース・アクセス)
 リモート・データ・アクセスの要件 5

DSQ で始まる名前
 アンインストール・ジョブ
 DSQ1DEL1 (z/OS でのアプリケーション・プランの解放) 120, 123
 DSQ1DEL2 (z/OS でのデータベース構成体の除去) 123
 DSQ1DEL3 (z/OS でのユーザー管理データ・セットの削除) 123
 DSQ1DELA (旧 QMF リリースの削除、z/OS) 123
 DSQ1EDX1 (iSeries (LUW 版) でのコントロール表/パッケージの除去) 126
 DSQ1EDX2 (iSeries、LUW でのサンプル表の削除) 126
 DSQ1JFPL (z/OS でのアプリケーション・プランの解放) 120
 インストール・ジョブ
 DSQ1BFRM (CICS 図表およびトレース) 77
 DSQ1BINR (QMF アプリケーション・プランのバインド) 60, 62
 DSQ1BLNI (V12.1 コントロール表の作成) 60, 83
 DSQ1BPKG (QMF パッケージのバインド) 43, 60, 62, 83, 85, 88, 89, 91
 DSQ1BSP (ストアード・プロシージャ・インターフェース) 143
 DSQ1BSQL (インストール・プログラムのバインド) 43, 60, 62, 83, 85, 88, 89, 91
 DSQ1BUDF (拡張 LIST コマンドのインストール) 60, 83, 147
 DSQ1BUDV (拡張 LIST コマンド視点) 43, 60, 83, 147

DSQ で始まる名前 (続き)

インストール・ジョブ (続き)

- DSQ1BVW (デフォルト LIST コマンド視点のインストール) 43, 60, 83, 88, 147
- DSQ1CFRM (CICS 図表およびトレース) 77
- DSQ1EADM (GDDM マップ・グループ) 76
- DSQ1EAS4 (iSeries でのサンプル表の作成) 85
- DSQ1ECSD (CICS システム定義) 74
- DSQ1EDJ4 (LUW でのサンプル表の作成) 83, 85
- DSQ1EDSJ (z/OS でのサンプル表の削除) 62, 85
- DSQ1EDX2 (iSeries、LUW でのサンプル表の削除) 85
- DSQ1EGLK (CICS 管理プログラムの準備) 74
- DSQ1EIVQ (QMF Analytics for TSO サンプル表の作成) 83
- DSQ1EIVS (z/OS でのサンプル表の作成) 60, 62, 83, 85
- DSQ1EPNL (パネル・ライブラリーへのデータ追加) 50, 98
- DSQ1ESQD (サンプル照会/プロシージャーの削除) 102
- DSQ1ESQI (サンプル照会/プロシージャーのインストール) 102
- DSQ1nCCS (NLF コマンド同義語の作成) 135
- DSQ1nRTS (新規コマンド同義語の作成) 142
- DSQ1STGJ (記憶域スペースの作成) 60, 83
- DSQ1TBAJ (VCAT 名の作成) 60, 83
- DSQ1UOPT (DSQUOPTS 出口ルーチンのアセンブル、リンク・エディット) 176

キュー

- DSQD 98

ストアード・プロシージャー

- DSQABA1E (拡張 LIST コマンド) 147, 515
- DSQQMFSP (QMF ストアード・プロシージャー・インターフェース) 143

大域変数

- DSQAO_OTC_LICENSE 100
- DSQAO_OTC_LICENSE (QMF VUE ライセンス) 176
- DSQAO_QMF_VER_RLS (現行 QMF リリース) 100
- DSQAO_STO_PROC_INT (ストアード・プロシージャー・インターフェース) 143
- DSQEC_CC (紙送り制御抑止) 160
- DSQEC_DISABLEADM (QMF 管理者権限検査) 176
- DSQEC_ISOLATION (トランザクション分離レベル) 244
- DSQEC_LAST_RUN (オブジェクト・リストでの最終使用日) 176
- DSQEC_SHARE (SHARE パラメーター・デフォルト) 176

データベース

- DSQDBCTL (QMF コントロール表) 18
- DSQDBDEF (SAVE DATA 結果) 18

データベース・パーティション・グループ

- DSQTSCTL (QMF コントロール表) 18
- DSQTSOBJ (SAVE DATA 結果) 18

データ・セット/ライブラリー

- DSQDEBUG (QMF トレース出力) 66
- DSQEDIT (編集転送ファイル) 66

DSQ で始まる名前 (続き)

データ・セット/ライブラリー (続き)

- DSQPLBE (ISPF パネル・ライブラリー) 47
- DSQPNE (QMF パネル・ライブラリー) 47, 50, 66, 98
- DSQPRINT (QMF 印刷出力) 66
- DSQSPILL (予備ファイル) 66
- DSQUCFRM (ユーザー定義の図表形式) 66
- DSQUODUMP (QMF スナップ・ダンプ) 66

出口ルーチン

- DSQUOPTS (大域変数の初期化) 175

パッケージ名

- リリース ID 122

表スペース

- DSQTSCT1 (Q.OBJECT_DIRECTORY 管理テーブル) 54
- DSQTSCT2 (Q.OBJECT_REMARKS 管理テーブル) 54
- DSQTSCT3 (Q.OBJECT_DATA 管理テーブル) 54
- DSQTSDEF (SAVE DATA 結果) 18, 54
- DSQTSGOV (Q.RESOURCE_TABLE 管理テーブル) 54
- DSQTSLOG (Q.ERROR_LOG 管理テーブル) 54
- DSQTSPRO (Q.PROFILES 管理テーブル) 54
- DSQTSRDO (Q.DSQ_RESERVED 管理テーブル) 54
- DSQTSYN (Q.COMMAND_SYNONYMS 管理テーブル) 54

プログラム

- DSQQMFE 95, 97, 158

プログラム・パラメーター

- DSQSBSTG 199
- DSQSBSTG (固定仮想記憶域) 199
- DSQSDBCS (DBCS データの印刷) 223
- DSQSDBG (トレース・ログ宛先、ストアード・プロシージャー・インターフェース) 160
- DSQSDBNM (初期データベース・ロケーション) 6, 196
- DSQSDBQN (CICS のトレース・キュー名) 222
- DSQSDBQT (CICS のトレース・キュー・タイプ) 222
- DSQSDBG (トレース・レベル) 221
- DSQSIROW (報告書の表示前に取り出される行数) 209
- DSQSMODE (バッチと対話式操作) 214
- DSQSMRFI (複数行フェッチ/挿入) 211
- DSQSPILL (報告書操作の予備記憶域) 200
- DSQSPLAN (QMF アプリケーション・プラン ID) 197
- DSQSPRID (QMF プロファイル・キー) 197
- DSQSPTYP (TSO の予備データ用拡張記憶域) 201
- DSQSRSTG (動的仮想記憶域) 200
- DSQSRUN (初期プロシージャー名) 215
- DSQSSPQN (予備データ用の CICS キュー名) 206
- DSQSSUBS (DB2 サブシステム ID) 195
- DSQSSUBS (データベース・サブシステム名) 6

プロシージャー

- DSQOBINI (デフォルトのシステム初期化プロシージャー) 182
- DSQ1EBAT (TSO インストール検査、バッチ) 106
- DSQ1EIVC (CICS インストール検査) 108
- DSQ1EIVP (TSO インストール検査) 105
- TSO ログオン 66

ロード・モジュール

- リスト 52

DSQ で始まる名前 (続き)
 DSQUOPTS ルーチン 176
 execs
 DSQ1DEFS (インストール・パラメーター・デフォルト)
 54
 DSQSCMDn (ストアード・プロシージャー・インターフ
 ェース・パラメーター) 143
 QMF の開始
 DSQUOPTS での大域変数の初期化 176
 DSQ10297 436
 DSQ10493 436
 DSQ1DEFS EXEC
 SECAUTH パラメーター 17
 DSQ1EGLK 433
 DSQAO_STO_PROC_INT 大域変数 143
 DSQDBCTL データベース 18
 DSQDBDEF データベース 18
 DSQDC_POS_SQLCODE 変数 451
 DSQDEBUG データ・セット
 ストアード・プロシージャー・インターフェース (stored
 procedure interface) 160
 ストアード・プロシージャー・インターフェース、割り振り
 143
 正の SQL コードのサポート 451
 CICS で 454
 DSQEC_BUFFER_SIZE 大域変数 449
 DSQEC_CC 大域変数
 ストアード・プロシージャー・インターフェース (stored
 procedure interface) 160
 DSQEC_CON_ACC_RES 大域変数 449
 DSQEC_DISABLEADM 大域変数 175
 DSQEC_EXTND_STG 大域変数 201
 DSQI0026 436
 DSQPNLE
 CSD 定義 433
 DSQQMFE 433
 DSQQMFSP プロシージャー
 参照: ストアード・プロシージャー、QMF for TSO の開
 始
 DSQSCMDn exec
 ストアード・プロシージャー・インターフェース 143
 DSQTSCTL データベース・パーティション・グループ 18
 DSQTSDEF 表スペース 18
 DSQTSOBJ データベース・パーティション・グループ 18
 DSQUCFRM ddname 288
 DSQUECIC 編集プログラム 336, 342, 352
 DSQUEGV1 モジュール、管理プログラム出口 373
 DSQUEGV3 モジュール、管理プログラム出口 373
 DSQUOPTS ユーザー出口
 説明 175
 DXEGOVA 制御ブロック 380
 DXEXCBA 制御ブロック 385
 DXEXCBA 制御ブロックの XCB フィールド
 XCBAUTH 385
 XCBAUTHX 385
 XCBQRYP 385

DXEXCBA 制御ブロックの XCB フィールド (続き)
 XCBQRYP2 385
 XCBQRYPT 385

E

E トレース・レコード 450
 EDIT TABLE コマンド
 必要な SQL 特権 238
 ENQ コマンド
 印刷キュー 284
 ENTRY_TYPE 列 (ファンクション・キー表) 315
 ENVIRONMENT 列 (Q.PROFILES)
 プロファイル初期化の役割 234
 ERASE コマンド
 QMF 管理者特権 18
 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 225
 EXPLAIN
 QMF での使用可能化 446
 EXPORT TABLE コマンド
 管理 401
 SQL 特権 238
 EXPORT コマンド
 最終使用日、設定の動作 259
 QMF 管理者特権 18

F

F トレース・レコード 450
 FLOAT データ・タイプ
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323
 FLOAT(DFP) コンパイラー・オプション、ユーザー編集ルーチ
 ン 339, 342
 FORWARD コマンド
 管理 401
 FSRCE サービス、印刷 274

G

G トレース・レコード 450
 G050 異常終了 433
 GDDM (図形データ表示管理プログラム) 518
 印刷 274
 エラー・メッセージ、印刷時 443
 サポートされているディスプレイ・デバイス 45
 プリンターのニックネーム 436
 ADMMNICK 指定 278
 ADMADFT デフォルト・モジュール 280
 ADMF データ・セットへのマップの追加 135
 APPCPG パラメーター 64, 75
 QMF 始動時の一般エラー 433
 GDDM とデータベース間のコード・ページの非互換性 64, 75

GDDM-PGF 518
GRAPHIC データ・タイプ
幅 202
SQLTYPE、ユーザー編集ルーチン 323

H

HANDLE CONDITION
CICS 373
HEX 関数 445
HPO/ マネージャー 357

I

I トレース・レコード 450
IMPORT TABLE コマンド
管理 401
必要な SQL 特権 238
表の作成 253
IMPORT コマンド
記憶域
参照： 記憶域の割り振り
最終使用日、設定の動作 259
単精度浮動小数点サポート 493
QMF 管理者特権 18
INTEGER データ・タイプ
幅 202
編集ルーチン 323, 326
ISC (システム間連絡) 519
iSeries プラットフォーム
データベースの前提条件 3
必要な権限 17
分散作業単位構成 10
リモート作業単位構成 6
ISPF (対話式システム生産性向上機能)
一般エラー 436
QMF ライブラリーの LIBDEF 割り振り 168

J

JCL
ストアード・プロシージャ・インターフェース、QMF
for TSO 143

L

L2 トレース・オプション
ストアード・プロシージャ・インターフェース (stored
procedure interface) 160
LAYOUT コマンド
最終使用日、設定の動作 259
LEFT コマンド
管理 401
LENGTH 列 (Q.PROFILES) 226

LIKE キーワード
データベースごとのサポートの相違 493

Linux
参照： Linux、UNIX、および Windows プラットフォー
ム

Linux、UNIX、および Windows プラットフォーム
管理で使用可能なパッケージ 401
データベースの前提条件 3
必要な権限 17, 18
分散作業単位構成 10
リモート作業単位構成 6

LIST コマンド
2 次 ID に許可されている表のリスト
許可されているフィーチャーのインストール 147
DB2 のアップグレード後 43
ALL キーワード 267

LOB データ
記憶オブジェクト 500
LONG VARCHAR データ・タイプ
幅 202
SQLTYPE、ユーザー編集ルーチン 323
LONG VARGRAPHIC データ・タイプ
幅 202
SQLTYPE、ユーザー編集ルーチン 323

LUW
参照： Linux、UNIX、および Windows プラットフォー
ム

M

MODEL 列 259
MODEL 列 (Q.PROFILES) 226
MRO (複数領域操作) 519

N

NLF (各国語機能)
コマンド同義語 296
リリース番号 467
参照： 各国語フィーチャー (NLF)
NLF のリリース番号 467
NUMBER 列 (ファンクション・キー表) 315
NUMTCB パラメーター、ストアード・プロシージャ・イン
ターフェース 143

O

OBJECT 列 (同義語表) 295
OBJECTLEVEL 列、QMF コントロール表 259
OTC_LICENSE 変数 (QMF VUE のご使用条件) 176
OWNER 列、QMF コントロール表 259

P

- P トレース・レコード 450
- PANEL 列 (機能キー表) 314
- PC プリンター 278
- PFKEYS 列 (Q.PROFILES) 226
- PF_SETTING 列 (機能キー表) 315
- PRINT TABLE コマンド
 - 管理 401
 - 必要な SQL 特権 238
- PRINT コマンド
 - 最終使用日、設定の動作 259
 - 指名宛先への経路指定 273, 284
- PRINTER 列 (Q.PROFILES) 160, 226
- PROCOPT パラメーター、印刷 278
- PROFILE PREFIX ステートメント 408
- PROG 759 436
- PTF トレース・オプション (ストアード・プロシージャー・インターフェース) 160
- PUBLIC AT ALL LOCATIONS に対する付与 241
- PUBLIC に対する付与
 - 参照: PUBLIC AT ALL LOCATIONS に対する付与

Q

- Q 許可 ID 17
- QBE 照会
 - 印刷 287
 - 管理 401
 - SQL 特権 238
- QMF Analytics for TSO
 - サンプル表 132
- QMF Analytics for TSO の構成 129
 - 作業ファイル・ロケーションのセットアップ 130
 - HFS のカスタマイズ 130
 - ZFS のカスタマイズ 130
- QMF for TSO Analytics のカスタマイズ
 - インストール検査手順 (IVP) 132
- QMF VUE のご使用条件 23, 25
- QMF 管理者権限 18
 - 検査プロセスと特権 18
 - 無効にする 175
- QMF コマンドの同義語 291, 294, 301
 - アクティブ化の問題 294
 - 引用符 298
 - オブジェクト名 294
 - 構文 298
 - 初期化メッセージ 436, 441
 - 同義語定義 295
 - 同義語表の作成 299
 - 表の保守 302
 - 変数の使用 297
 - verb 294
- QMF タスクに必要な権限
 - インストール 17
 - インストール検査手順 (IVP) 93
 - QMF タスクに必要な権限 (続き)
 - 管理 18
 - サンプル表へのアクセス 135
 - QMF 管理者
 - 参照: QMF 管理者権限
 - QMF タスクの自動化
 - バッチ・モード QMF
 - プログラム・パラメーター 214
 - CICS 426
 - TSO、ISPF、ネイティブ z/OS 407
 - プロシージャー
 - 初期化後 215
 - ストアード・プロシージャー・インターフェースでの実行 143, 160
 - 大域変数を設定する初期化出口 175
- QMF 出口ルーチンの制御ブロック
 - 管理プログラム
 - DXEGOVA 380
 - DXEXCBA 385
- QMF で選択可能な翻訳
 - 参照: 各国語フィーチャー (NLF)
- QMF で必要な WLM アドレス・スペース
 - ストアード・プロシージャー・インターフェース 143
- QMF の DASD 要件 45
- QMF の PSP サブセット名 429
- QMF の PTF、適用 429
- QMF のアプリケーション・プラン
 - アクセス・タイプ 225
 - 必須ロケーション 8
- QMF のアンインストール
 - DB2 for iSeries および LUW データベース 124
 - DB2 for z/OS データベース 119
 - DB2 (VSE および VM 版) データベース 127
- QMF の移行
 - サポートされるデータベース・リリース 3
 - データベースのみのアップグレード・タスク 43
 - QMF の各種のリリースの共存 5
- QMF の開始
 - 一般エラー 433
 - ストアード・プロシージャー・インターフェース 143, 160
 - 参照: ストアード・プロシージャー
 - 大域変数の初期化 175
 - QMF 管理者権限のオーバーライド 175
 - QMF プロファイルの初期化 234
- QMF の旧リリース
 - 削除 119
 - バージョン 12.1 の互換性 113
- QMF の共通サービス、トレース 450
- QMF の言語機能
 - 参照: 各国語フィーチャー (NLF)
- QMF の構成
 - インストール検査手順 (IVP) 132
 - インストール・ロードマップ 15
 - オプション機能 129, 132
 - 各国語機能 (NLF) 135

QMF の構成 (続き)

- 計画
 - 参照: インストールの計画
- サーバー・データベース 81
- 分散作業単位 10
- 分散データ・アクセスの概要 5
- リモート作業単位 10
- リモート作業単位 (remote unit of work) 6
- ロード・ライブラリー 129
- HFS カスタマイズ 130
- QMF Analytics for TSO 129
- QMF Analytics for TSO サンプル表 132
- QMF Analytics for TSO ファイルの割り振り 129
- QMF Analytics for TSO 用のファイルの割り振り 129
- QMF for TSO Analytics のカスタマイズ 130, 132
- SAVE DATA コマンドのセットアップ 18
- zFS カスタマイズ 130
- z/OS 環境プリファレンスのカスタマイズ 64

QMF のデバイス要件 45

QMF のマシン要件 45

QMF プログラムのモジュール

- 参照: ロード・モジュール

QMF プロファイルの SYSTEM 行

- 使用する認証 143

QMF へのアクセスの制御 225

QMF へのログオン

- イネープリング 226
- 制限 233
- 参照: 認証

QMF ライブラリーを割り振るための LIBDEF ステートメント 168

QMF リリース間の後方互換性 113

QMF リリース間の互換性 4, 113

QMF リリース間の前方互換性 113

QMF をインストールするジョブ

- サーバー・データベース
 - 新規 バージョン 12.1 インストール 83
 - 3 部構成の名前によるアクセス 91
- リクエスト・データベース
 - 新規 V12.1 インストール 60

QMF をサポートするプロセッサ 45

QUEUENAME、QUEUETYPE キーワード 284

Q.COMMAND_SYNONYMS_n 表

- 作成するためのジョブ 135

Q.ERROR_LOG コントロール表 463

Q.GLOBAL_VARS 表

- 構造 180

Q.OBJECT_DIRECTORY

- LAST_USED 列 176, 259

Q.PROFILES コントロール表

- 管理者権限の確認 18
- 更新 236
- 表の構造 226
- ユーザー認証 234
- ユーザー変更 236
- ユーザー・プロファイルの削除 233

Q.PROFILES コントロール表 (続き)

- ユーザー・プロファイルの追加 234
- NLF に合わせた更新 135
- RESOURCE_GROUP フィールドの更新 363

Q.RESOURCE_VIEW、管理プログラム 361

R

RACF

- インストールの注意点 17
- グループ名
 - 参照: 2 次許可 ID
 - バッチのセキュリティー 407

RCT (リソース・コントロール表) 436

REMARKS 列 264

RESTRICTED 列

- 値を NO に変更 267
- 定義済み 259

REVOKE ステートメント

- 例 242

REXX サポート

- QMF for TSO ストアード・プロシージャー・インターフェース 143

RIGHT コマンド

- 管理 401

RLST 403

RUN QUERY コマンド

- 管理 401

RUN コマンド

- 記憶域
 - 参照: 記憶域の割り振り
 - コマンド同義語 295
 - 最終使用日、設定の動作 259
 - 必要な SQL 特権 238

S

SAVE DATA コマンド

- 管理 401
- 記憶域
 - 参照: 記憶域の割り振り
 - 表スペースを受け取る 18
 - LOB 記憶域 500

SAVE オプション

- EDIT TABLE コマンド
 - サポートされないシチュエーション 493

SAVE コマンド

- 拡張 254
- 最終使用日、設定の動作 259
- 必要な SQL 特権 238
- DATA キーワード 238
- QMF 管理者特権 18
- TABLE キーワード 238

SBCS 言語、CCSID 64, 75

SCOPE リソース・オプション 358

SECAUTH インストール・パラメーター 17
 SELECT ステートメント
 複数ステートメント照会に対する制約事項 225
 SEQ 列 263
 SET PROFILE コマンド 236
 SHARE パラメーター、SAVE コマンド 18
 SMALLINT データ・タイプ
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323
 SPACE 列 (Q.PROFILES) 226
 SQL
 ステートメント
 INSERT (新規のユーザー・プロファイル) 234
 UPDATE 237
 特権 234
 指示照会、QBE 照会に関する 238
 QMF コマンドのための 238
 戻りコード 451
 HEX 関数 445
 ID 234
 コマンド同義語表 304
 QMF の保管方法 263
 SQL コード
 正の処理の設定 451
 メッセージのサポート 451
 -551 175, 436
 -805 436
 SQL 照会
 32 KB よりも大きい
 管理 368
 SQL ステートメント
 DB2 リソース限定機能を使用する QMF の管理 401
 SUBTYPE 列、QMF コントロール表 259
 SUSPEND キーワード 284
 SYNONYMS 列 (Q.PROFILES) 226
 SYNONYM_DEFINITION 列 295
 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 226
 インストールの前提条件 17
 REVOKE ステートメント 242
 SYSIBM.SYSTEM_EBCDIC_CCSID セッション変数 64, 75
 SYSTEM プロファイル
 削除 233
 デフォルト値の変更 237
 SYSTSPRT データ・セット
 ストアード・プロシージャ・インターフェース、割り振り
 143

T

TIME データ・タイプ
 形式設定用の編集ルーチン 330
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323
 TIMESTAMP WITH TIME ZONE データ・タイプ
 形式設定用の編集ルーチン 330

TIMESTAMP WITH TIME ZONE データ・タイプ (続き)
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323
 TIMESTAMP データ・タイプ
 形式設定用の編集ルーチン 330
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323
 TOFAM キーワード、ADMMNICK 指定 278
 TOP コマンド
 管理 401
 TRACE 列 (Q.PROFILES) 226
 TRANSLATION 列 (Q.PROFILES) 226
 TSO (タイム・シェアリング・オプション)
 仮想記憶域 198
 管理プログラムへのインターフェース 369
 記憶域要件
 可変記憶域容量の設定 199
 記憶域の動的割り振りの設定 200
 固定記憶域の設定 199
 初期化 51
 報告書記憶域を比率として設定 199
 報告書操作 51
 割り込み機能 452
 QMF へのストアード・プロシージャ・インターフェース
 143, 160
 TSO ユーザー ID、QMF 認証で使用 197
 TYPE 列、QMF コントロール表 259
 TYPETERM 指定 465

U

U 編集コード、形式
 定義済み 326
 入力域 323
 UNIX
 参照： Linux、UNIX、および Windows プラットフォー
 ム

V

V 編集コード、書式
 定義済み 326
 入力域 323
 Value Unit Edition
 参照： VUE (Value Unit Edition)
 VARBINARY データ・タイプ
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323
 VARCHAR データ・タイプ
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323
 VARGRAPHIC データ・タイプ
 幅 202
 SQLTYPE、ユーザー編集ルーチン 323
 VERB 列 (同義語表) 294

- VM プラットフォーム
 - 管理で使用可能なパッケージ 401
 - 制約事項
 - 3 部構成の名前 10
 - データベースの前提条件 3
 - 必要な権限 17
 - リモート作業単位構成 6
 - QMF カタログの列の長さ 8
- VSAM データ・セット
 - 索引と表スペース用に使用される 270
- VSE プラットフォーム
 - 管理で使用可能なパッケージ 401
 - 制約事項
 - 3 部構成の名前 10
 - データベースの前提条件 3
 - 必要な権限 17
 - リモート作業単位構成 6
 - QMF カタログの列の長さ 8
- VUE (Value Unit Edition)
 - ご使用条件 23, 176
 - 前提条件となるデータベース・リリース 3, 45

W

- WIDTH 列 (Q.PROFILES) 226
- Windows
 - 参照： Linux、UNIX、および Windows プラットフォー
ム
- WLM 環境
 - ユーザー定義関数のための更新 513

X

- XML データ・タイプ
 - 制約事項
 - 複数行フェッチ 211
 - ユーザー編集コードの出口ルーチン 321
 - 予備データ用拡張記憶域 201, 202
 - XML 列のフェッチの管理 401

Z

- z/OS プラットフォーム
 - 管理で使用可能なパッケージ 401
 - データベースの前提条件 3
 - 必要な権限 17
 - 分散作業単位構成 10
 - リモート作業単位構成 6
 - DCB 異常終了のインターセプトと予備ファイル 447
 - QMF を除く DB2 のアップグレード 43



プログラム番号: 5697-QMF

Printed in Japan

GC43-3485-00



日本アイ・ビー・エム株式会社

〒103-8510 東京都中央区日本橋箱崎町19-21