Query Management Facility™ for Windows

Installing and Managing QMF for Windows

Version 7 Release 2

GC27-0722-01

Query Management Facility™ for Windows

Installing and Managing QMF for Windows

Version 7 Release 2

GC27-0722-01

Note!
Before using this information and the product it supports, be sure to read the general information under Appendix E, Notices.

Sixth Edition (March 2002)

This edition applies to Version 7, Release 2 of IBM QMF for Windows, a feature of the QMF Family with Version 7.1 of DB2 Server for OS/390, 5675-DB2, and for QMF Family with Version 7.2 of DB2 Server for VSE & VM, 5697-F42, and with Version 7.2 of DB2 UDB Universal Developer's Edition, 5648-D37, and with Version 7.2 of DB2 UDB Warehouse Manager, 5648-D35, and with Version 7.2 of DB2 Warehouse Manger iSeries, 5724-B08, and for Version 7.2 of QMF Windows for iSeries, 5697-G24, and for Version 7.2 of QMF for Windows for DB2 Workstation Databases, 5697-G22, and to all subsequent releases and modifications until otherwise indicated in new editions.

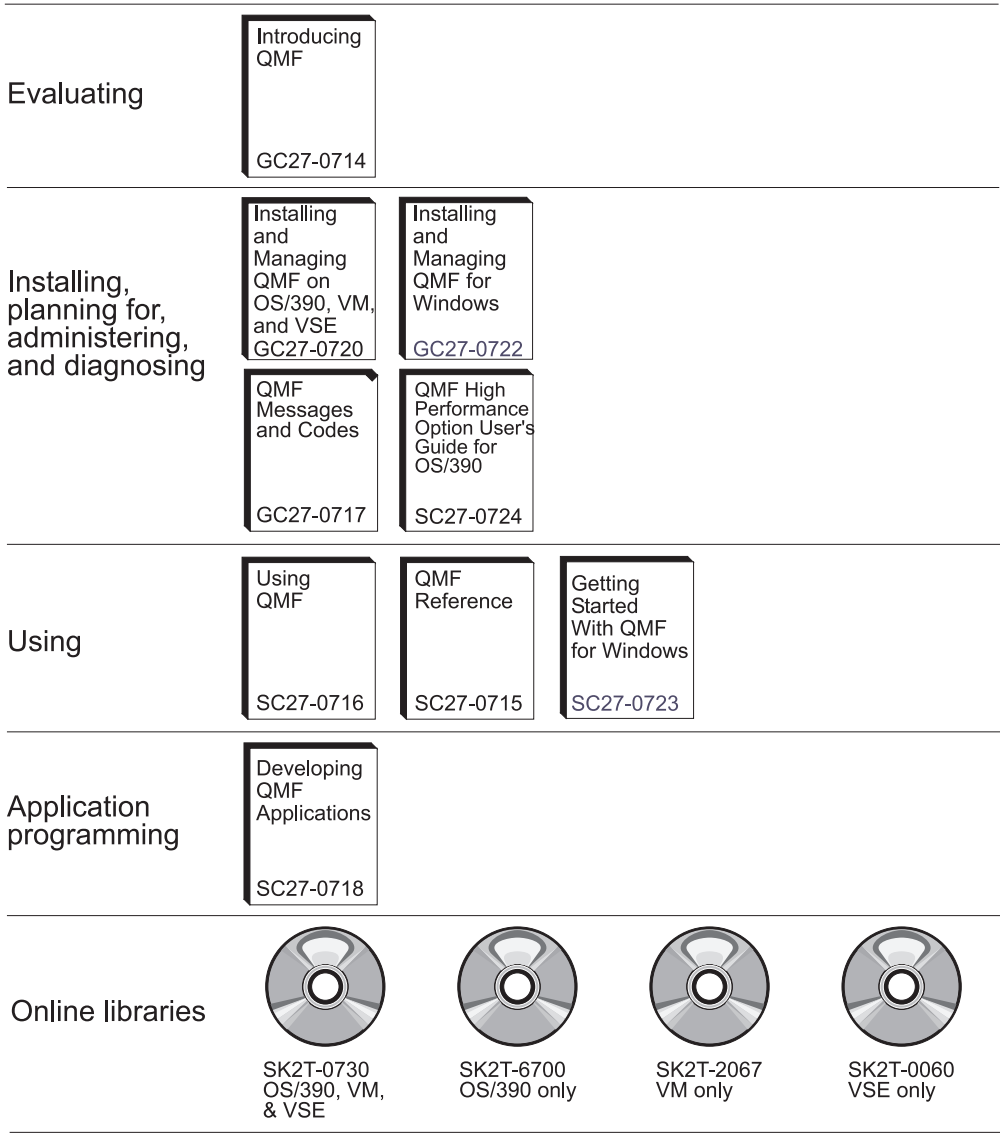
This edition replaces GC27-0722-00.

Contents

Tables

The QMF library

You can order manuals either through an IBM representative or by calling 1-800-879-2755 in the United States or any of its territories.



Chapter 1. Overview

This chapter provides general information on configuring and installing QMF for Windows Version 7.2 for z/OS™.

This book explains the basic concepts behind QMF for Windows Administrator. It describes the software required and its configuration. The intended audience is the person or persons responsible for planning for, installing, configuring, and maintaining QMF for Windows.

QMF for Windows Administrator is the administrative component of QMF for Windows. QMF Windows Administrator provides governing control over user actions and DB2® resource consumption.

See *Using QMF for Windows* for more information on how to use the installed product.

Prerequisites

The following operating system, applications, connectivity are required for using QMF for Windows.

Operating system

- Microsoft® Windows® 95
- Microsoft Windows 98
- Microsoft Windows ME
- Microsoft Windows NT 4.0
- Microsoft Windows 2000
- Microsoft Windows XP

Add-in applications

- Microsoft® Excel®: 7.0 or later
- Lotus® 1-2-3®: 97 or later

Connectivity

TCP/IP

The Microsoft TCP protocol installed with the Windows operating system should work with any product that provides a WinSock interface.

SNA
IBM Personal Communications 4.1 or later should work with any product that provides a WinCPIC interface.

ODBC
Most ODBC 2.x or 3.x drivers are acceptable.

CLI
IBM DB2 Universal Database™ runtime client, Version 6.1 or later.

QMF Query for Java™

IBM WebSphere® Version 3.02 or 3.5 should work with any application server product that implements JSDK 2.x or Java Servlet 2.2 or later. Applicable platforms and versions are listed below.

- UNIX and Linux
- Windows NT or Windows 2000
- iSeries™ Version 4.4 or later
- OS/390® Version2.5 or later, z/OS with UNIX System Services installed or MVS™

Note: IBM WebSphere Application Server is recommended (Versions 2.0, 3.02, and 3.5 are supported). However, QMF Query for Java is compatible with any application server that supports JDSK Version 2.0 and JDBC Version 1.0.

Database

IBM DB2 Version 5.x and above with a JDBC type 1.0 driver, with the applicable updates to enable JDBC.

Other functionality

ObjectREXX
IBM ObjectREXX for Windows Versions 1.0.2.3 or later.

Natural language query
- Linguistic Technology English Wizard Version 3.1
- EasyAsk Version 6.0

Audience

The person who is responsible for installing, configuring, and maintaining QMF for Windows should refer to this manual. This book is written for database administrators (DBAs), those who write SQL, and who are proficient using DB2. Additionally, the audience for this book is the person who is familiar with the infrastructure of the systems applications and communications.

Accessibility features

QMF for Windows allows for customization that will accommodate your vision or mobility needs. There are different categories of accessibility options available from within QMF for Windows and from the Windows operating system:

- **Keyboard accelerators:** these are shortcuts that enable you to perform activities directly through the keyboard rather than from a menu. These are also known as mnemonics.

  For example, pressing Ctl+S would be the same as selecting the Save option from a menu.

  Another example, files listed as 1, 2, 3 off a menu can be selected by simply typing the option number. For example, you would select Tables if you select option 2:

  ```
  1   Reports
  2   Tables
  3   Queries
  ```

- **Toolbar icons:** you can add icons to the toolbar to execute the most frequently-used options (such as save, run query, and others)
- **Tooltips:** tool tips (information about the option) display when you ″mouse-over″ (sweep the cursor) over an area of a tool tip-enabled window
- **Screen reader:** QMF objects can be read by screen readers

  Note: The query results (also known as the ″results grid″) cannot be read by a screen reader. Instead, export the file or save the file in HTML format.
- **High-contrast display:** you can change the display from the Control Panel so that icons and type are magnified to meet the needs of the visually impaired.
- **Microsoft Control Panel Accessibility Options:** These options are available from the control panel, and include options for:
  - Keyboard ″StickyKeys,″ ″FilterKeys,″ and ″ToggleKeys,″ plus the option to show extra keyboard help in programs
  - Display: contains an option for Windows to use colors and fonts designed for easy reading, plus an option for high-contrast (color scheme = white on black, black on white, or a custom scheme)
  - Mouse: the option to use MouseKeys, which controls the pointer with the numeric keypad on your PC (including the shortcut and pointer speed)
- **General:** turn on/off accessibility options; notifications; SerialKey devices, and administrative options

What is QMF for Windows?

IBM Query Management Facility for Windows (QMF for Windows) is a tightly integrated, powerful, and reliable query and reporting tool set for IBM's DB2 relational database management system. It provides an easy-to-use interface for enterprise reporting, including reports and procedures with IBM ObjectREXX calculations.

QMF for Windows accesses data that are stored in any database in the IBM DB2 family of databases. It also handles data from heterogeneous and non-heterogeneous data sources. QMF for Windows integrates with spreadsheet solutions such as Microsoft Excel and Lotus 1-2-3® for an alternate method of packaging and presenting report data. Java SQL from a browser and point-and-click, drag-and-drop functionality, including aggregations, summaries, and formatting from within the query results make QMF for Windows a robust query and reporting tool. The QMF for Windows Administrator component provides extensive, detailed management and control over your QMF environment to protect resources and monitor user actions.

There are two ways in which QMF for Windows can connect to DB2 to obtain this data:

- Via the Open Group's Distributed Relational Database Architecture™ (DRDA)
- Via the DB2 UDB for Windows Call Level Interface (CLI)

QMF for Windows is a query and reporting tool that executes queries against operational databases or data warehouses, resulting in reports or charts. QMF for Windows can:

- Access a company's DB2 database platforms
- Publish reports on the company's intranet
- Identify who is using what queries
- Identify which queries are not being used (and that can be removed from the reporting warehouse)
- Identify which queries are run most frequently
- Leverage stored procedures and static SQL to optimize performance
- Connect to multiple database servers and run multiple queries simultaneously

Benefits to the DBA

QMF for Windows Administrator offers the following benefits to the database administrator:

- Static SQL from Windows
- Centralized installation/elimination of database gateways and middleware
- Ability to protect DB2 from runaway queries and novice users
- Build governing into Windows applications
- Use existing DB2 security
- Centralize control over server resources
- Adjust governing limits by time of day, day of week, user groups, and server
- Set governing thresholds to warn users or to cancel queries and threads automatically
- Set limits by rows fetched, idle query time-outs, server response time-outs, and idle connection time-outs
- Allow or disallow 14 different SQL verbs
- Turn on or off the table editor and other features by group

QMF for Windows Administrator is installed when the system administrator installs QMF for Windows. QMF for Windows is a feature of DB2 for z/OS, DB2 VSE™ and VM®, or a stand-alone product to access DB2 workstation databases only.

Note: End users may install QMF for Windows Administrator. There is no security risk because database and file sharing security mechanisms restrict unauthorized users.

At installation, the QMF for Windows Administrator icon is added to the Windows 9x, NT 4.0, 2000 or ME or XP Start menu or to the QMF for Windows program group in the Windows NT 3.51 Program Manager. You can use this icon to start the Administrator, or you can run it directly by starting the program **rsadmin.exe**. QMF for Windows Administrator creates a flat file containing information that tells QMF for Windows how to connect to the database; this is the Server Definition File (SDF), **qmfsdf.ini**. QMF for Windows comes preconfigured with an SDF with one database server. You can use this database template with a simple TCP/IP connection to learn about QMF for Windows.

Once QMF for Windows Administrator connects to the database, it populates the tables and packages with the user groups and their corresponding authorities. When QMF for Windows, the end-user application, starts up, it reads the SDF and uses that information to find the database, packages, and tables.

Package

A package contains a set of related static SQL statements. Package names that contain ″RDBI″ are core packages for QMF for Windows, such as RDBI.PROFILE_VIEW. Some packages are predefined, and are included with the product.

CLI connections use different package names so that you can use a single collection name for DRDA and CLI connections when binding packages.

Collection

A collection contains a set of packages. Packages are bound together with a collection ID. If you use several versions of QMF for Windows, you can use different collection names for different versions to avoid naming conflicts.

System requirements

Minimum requirements

Workstation requirements

QMF for Windows requires the following software and hardware:

- QMF for Windows Version 7.2 must be installed on the client

  To verify the version, run QMF for Windows, then select Help/About QMF for Windows
- Microsoft Windows 9.x (R), NT (R), ME (R), XP (R) or 2000 (R)
- 386- or higher processor
- 10 MB hard disk space
- 4 MB of RAM
- DB2
- *For a CLI connection,* you must have Version 5.2 DB2 UDB, or higher
- *For DB2 UDB Personal Edition (C),* you must use a CLI connection
- *For TCP/IP,* you must have Version 5 DB2 or higher

iSeries™ requirements

- iSeries, Version 4.3 or higher
- Latest cumulative and DB2 group PTF applied
- iSeries TCP/IP services configured with valid host name and address
- DDM TCP/IP server started
- Local RDB name configured for the iSeries
- iSeries user profile CCSID must be set to a value other than 65525

iSeries™ terminology

QMF for Windows uses SQL terminology. The following table shows the relationship between iSeries system terms and SQL relational database terms. For more information, refer to the *DB2 UDB for iSeries Database Programming* book.

Table 1. iSeries and SQL terminology

iSeries Term	SQL Term
Library: Groups related objects and allows you to find objects by name.	**Collection:** Consists of a library, journal, journal receiver, SQL catalog, and, optionally, a data dictionary. A collection groups related objects and allows you to find the objects by name.
Physical file: A set of records.	**Table:** A set of columns and rows.
Record: A set of fields.	**Row:** The horizontal part of a table containing a serial set of columns.
Field: One or more characters of related information of one data type.	**Column:** The vertical part of a table of one data type.
Logical file: A subset of fields and records of one or more physical files.	**Package:** An object type that is used to run SQL statements.
User profile	Authorization name or authorization ID

Chapter 2. Introduction to QMF for Windows Administrator

The role of QMF for Windows administrator

The QMF for Windows administrator application performs the following roles:

- It creates a flat file containing information that tells QMF for Windows how to connect to the database. This is the server definition file, SDF (**qmfsdf.ini**).
- It connects to the specified database and creates a set of tables and packages for it and the end-user application to use
- It creates optional users groups and assigns governing limits to them. This information is stored in the tables above.

When the end-user application, QMF for Windows, starts up, it reads the SDF and uses that information to find the database, the packages, and the QMF for Windows tables.

QMF for Windows administrator concepts

This chapter explains the concepts for administering QMF for Windows and provides instructions for getting started. Use the online help for details about specific tasks.

There are four basic tasks that you perform with QMF for Windows Administrator:

- Defining and configuring the database servers that QMF for Windows will access
- Creating database objects, binding database packages, and granting permissions in each database server that QMF for Windows will access
- Creating sample database tables
- Governing and object tracking

Servers

You must define each database server that you access with QMF for Windows. When you do so, you give the database server a *server name*, which is a descriptive label for the server, used only in QMF for Windows. You need this name only to access the server. You do not need the actual server name or the technical details behind accessing the server.

Use the QMF for Windows Administrator application to:

- define each server
- name the servers
- specify technical information for QMF for Windows to access the server (this is similar to defining a data source in ODBC)

The parameters that you must specify include:

- The RDB name for the server, which is the *location name* for DB2 for z/OS or MVS; or the *database name* for DB2 Universal Database or DB2 Common Server
- Network connection information, which is the CPI-C symbolic destination name or the TCP/IP host name and port number
- For ODBC connections: the machine- or file data source name

  Machine data source names (DSN) lists available ODBC data sources, including the user's DSNs and machine DSNs from a pull-down window. User data source names are available only to the current user. Machine data source names (DSNs) are available to all logged-on users of that machine.

  File data source names contain the full path to the default file data source names (DSNs) directory. Under the directory, there are many files with the extension .dsn that store ODBC connectivity information to data sources. The file DSN directory can reside on a local machine or remotely on the network. File DSNs are available to all users logged on to the network.

Packages

To run distributed SQL at any DB2 database, you must bind a package at the database that contains the SQL that you want to run (excluding dynamic SQL). You use QMF for Windows Administrator to select the collection name and the options for the packages that it requires, and to bind the packages at a server automatically.

Note: For ODBC connections, you do not have to bind packages because dynamic SQL is used.

"Packages" refer to the set of DB2 tables that QMF for Windows uses, which might not exist at a server. Create these tables before binding the packages. QMF for Windows Administrator can determine which tables need to be created, and allows you to create them.

After you bind the QMF for Windows packages, grant authority to your end-users to execute the user packages. Again, QMF for Windows Administrator can grant or revoke this authority to the users that you specify.

User groups and governing

QMF for Windows incorporates a comprehensive resource governor (a ″predictive governor″), that restricts what a user can do in QMF for Windows and places limits on the resources a user can consume. With governing, you can provide distributed access to DB2 to your users without having an adverse impact on your database or network performance.

Using QMF for Windows Administrator, you can define sets of limits and restrictions, which are called *resource limits groups*. You can then assign users to specific resource limits groups, according to the governing that you want performed for those users.

The ″predictive″ governor feature enables you to set up thresholds by which the database can evaluate the cost of running a dynamic SQL statement. The governor will:

- return a warning if the cost exceeds the administrator's defined threshold, or
- cancel the statement automatically

Thresholds can vary by several different attributes, including the name of the package in which the SQL is being run.

Sometimes, QMF for Windows prepares dynamic SQL statements during internal processing instead of when running users' queries. Typically, predictive governing should not be applied to these queries. Predictive governing should be enabled only for the packages in which QMF for Windows runs users' queries.

If a DRDA connection (over TCP/IP or SNA) to the database is being used, predictive governing should be enabled for the following QMF for Windows packages, which are used only for users' queries (other QMF for Windows packages are used only for internal processing):

- RAARDBX1
- RAARDBX2
- RAARDBX3
- RAARDBX4
- RAARDBX5

The packages that are used when connecting to a database via CLI are determined by the DB2 client. Consult your DB2 client documentation for information about the packages used in this case.

Sample tables

QMF for Windows provides nine sample tables that you can use before you begin working with your own tables. The sample tables, listed below, are used throughout the *Getting Started with QMF for Windows* manual and online help as examples. They contain information about a fictitious electrical parts company.

Table 2. Sample tables

Sample Table Name	Contains information about
Q.APPLICANT	The prospective employees of the company.
Q.INTERVIEW	The interview schedule for prospective employees.
Q.ORG	Organization of the company by department (within division).
Q.PARTS	Materials supplied to the company.
Q.PRODUCTS	Products produced by the company.
Q.PROJECT	Company projects.
Q.SALES	Sales information for the company.
Q.STAFF	The employees of the company
Q.SUPPLIER	Other companies who supply materials to the company.

Chapter 3. Installation

QMF for Windows components

QMF for Windows is made up of three components:

- Client (PC code that users use to connect to the DB2 server)
- Administrator tool (PC code that administrators use to configure QMF for Windows)
- Host enabler (code that administrators use to upload to the database server via the administrator's tool, enabling the client to connect to DB2)

  This is more a post-installation task that needs to be performed by the database administrator or systems programmer. In this step, QMF objects (tables and views) are created on each server and the QMF packages are bound to the database.

The installation process

When you insert the product's CD-ROM you have the option of installing QMF for Windows only, or installing QMF for Windows and the QMF for Windows Administrator. The main installation process is straight-forward, as long as the database administrator installs the product using the distribution media. But this requires the installation CD-ROM to be passed among the potential users and each user has to perform the installation. In addition, this installs the *full version*, rather than the *thin client*.

To simplify the installation process, QMF for Windows allows the installation files to be placed on a central system and the users can start the installation process from there. This type of installation also allows the administrator to set certain variables for the users in advance, which simplifies the post-installation process for the end-users.

Role of the administrator

QMF for Windows must be installed with the Administrator component on the DBA's desktop, then migrated to the file server. QMF for Windows Administrator, the administrative component of QMF for Windows, is necessary for general administrative maintenance.

Only administrators should use QMF for Windows Administrator. There should not be any need for an end-user to run QMF for Windows Administrator, though end-users are restricted by existing database and file-sharing security mechanisms.

From QMF for Windows Administrator (Administrator), you set up the servers for QMF for Windows. The server definition file (SDF), set up through Administrator, is installed on the DBA's computer by default, not on the server.

Server definition file

The server definition file (SDF) is created by QMF for Windows Administrator. It is a flat file containing information that tells QMF for Windows how to connect to the database, including specifying the database servers that QMF for Windows uses. Enter the name and the path of the server definition file to be used by QMF for Windows. The server definition file specifies the database servers that QMF for Windows uses. Disk1 of the QMF Windows installation diskettes contains the **qmfsdf7.ini** file, which is stored on Disk1 of the QMF for Windows installation diskettes. By default, the **qmfsdf7.ini** file will be installed in `C:\WINNT` (for Windows NT and Windows 2000) or `C:WINDOWS` (for Windows 95, Windows 98, or Windows ME).

For example:

```
[Options]
ServerDefinitionsFile=f:\windows\sdf.ini
```

Note: Once the file server is configured, move the SDF from your computer to the server. Clients (QMF for Windows users) will download the SDF from the file server to install and run QMF for Windows.

This installation does not work with the trial version of QMF for Windows. This installation method enables you to apply an upgrade (a corrective service release) to the file server, and the users will be upgraded automatically.

Server install

Copy disk images

Before you can perform an installation, copy the disk images from the QMF for Windows CD. For information on locating the correct language and version of QMF for Windows, refer to the **readme.txt** file in the QMF for Windows CD root directory.

Then, using Windows Explorer, copy each disk image directory from the CD to a centrally accessible computer.

Edit qmfwin4.reg file

User options are stored in the registry which is updated during QMF for Windows installation based on the settings in the **qmfwin4.reg** file. This file is on Disk1 of the QMF for Windows installation diskettes. Edit the **qmfwin4.reg** file. This is a registry file that points to the server definition file (SDF). User

options in the **setup.ini** file are stored in the registry. The registry is updated during QMF for Windows installation based on the settings in the **qmfwin4.reg** file.

1. Right-click the **qmfwin4.reg** file on Disk1 and select Edit.
2. Specify the path to the SDF through your network. For example in the **qmfwin4.reg** file, enter the following information in the [HKEY_CURRENT_USER\Software\IBM\RDBI\Options] section:

   ```
   [HKEY_CURRENT_USER\Software\IBM\RDBI\Options]
   "ServerDefinitionsFile"="\\\\ServerName\\Root\\temp\\qmfsdf.ini"
   ```

   Note: Make sure all slashes are doubled (\\) and each string is surrounded by quotes (").

   To ensure you have the correct path to the SDF, go to Network Neighborhood or Explorer and confirm the path to the SDF.
3. If using an SNA connection to the server, edit the [HKEY_CURRENT_USER\Software\IBM\RDBI\Options] section of the **qmfwin4.reg** file. The CPI-C provider DLL defines which SNA product you are using. The CPI-C provider DLL defines which SNA product you are using.

   For example:

   ```
   [HKEY_CURRENT_USER\Software\IBM\RDBI\Options]
   "ServerDefinitionsFile"="\\\\ServerName\\Root\\temp\\qmfsdf.ini"
   "CPICDLL = "c:\\windows\\system\\wincpic.dll"
   ```

   Note: Make sure all slashes are doubled (\\) and each string is surrounded by quotes (").

If you are using a CLI or TCP/IP connection, you do not have to declare this setting.

Save predefinition file

After you edit and save the predefinition file, copy it to the Disk1 folder of the QMF for Windows installation diskettes.

Unattended installation

Edit setup.ini file

An unattended installation allows you to select the installation options for your QMF for Windows users before beginning the installation process. The advantage of this method is that you can designate the options of an installation rather than having to select the same options repeatedly for each installation. The options are defined when you edit **setup.ini**.

Using a simple text editor, edit the **setup.ini** file in Disk1. The **setup.ini** file controls the installation process and determines the settings used for installation.

See the ″Custom Installation Options″ appendix for more information about each parameter in the **setup.ini** file.

Sample setup.ini file

A sample **setup.ini** is presented below. It specifies the server name and root directory and the parameters for [Options] and [Components] located above the [Startup] line in the **setup.ini** file.

Note: The [Startup] parameters are the only entries in the **setup.ini** file provided at installation.

```
[Options]
AutoInstall=1
FileServerInstall=0
SetupType=2
InstallPath=\\ServerName\Root\Program Files\IBM\QMF for Windows
ProgramGroup=QMF Win 7.2

[Components]
Base=1
Admin=0
Excel=1

[Startup]
AppName=QMF for Windows 7.2
FreeDiskSpace=481
```

From the example above:

[Options]

- AutoInstall=1: specifies an unattended install
- FileServerInstall=0: installs all QMF for Windows files in the directory specified in the InstallPath. This indicates that QMF for Windows has not been installed previously.
- SetupType=2: specifies a custom install
- InstallPath=the directory where QMF for Windows will be installed. Use Network Neighborhood or the Explore function if you need to locate the directory.
- ProgramGroup=the name that will appear in the Start Menu on the user's PC
- AutoExit=this instruction will skip the final confirmation message at the end of the installation

[Components]

Specific components are declared in this section of the **setup.ini** file. Enter **1** to install a component; otherwise, enter **0**.

Note: The default parameter for each component is **1** (install). If you do not specify a **0** (do not install), the component will be installed automatically and it does not have to be declared.

From the example above:

- Base=1: specifies to install the QMF for Windows program files
- Admin=0: specifies not to install QMF for Windows Administrator
- Excel=1: specifies to install Microsoft Excel add-in files

Note: Other variations can be set. These include Report Center, Lotus 1-2-3, and IBM QMF Query for Java. See the ″Custom Installation Options″ appendix for more information.

Save the **setup.ini** file and copy it to Disk1 of the QMF for Windows installation diskettes.

Run QMF for Windows installation

After choosing your options, double-click on **setup.exe**. The installation proceeds automatically.

If the disks are on the Administrator's computer, the SDF is installed on the Administrator's computer, not on the server. Move the SDF from your computer to the server to allow end-user access.

Post installation

If you install QMF for Windows Administrator, an icon to start QMF for Windows Administrator is added to the Windows 9x, 2000, XP or NT 4.0 Start Menu or to the QMF for Windows program group in the Windows NT 3.51 Program Manager. You can use this icon to start QMF Administrator, or you can run it directly by starting the program **rsadmin.exe**. If this program file does not exist, you must reinstall QMF for Windows and copy this file from the installation disks. Specify the *Complete install* or *Custom install* option, then reinstall the QMF for Windows Administrator.

After installing QMF for Windows on the DBA's machine or the server, perform the client installation, as described below.

Client install

After you modify the **qmfwin4.reg** and the **setup.ini** files, as described in the ″Server Install″ section above and complete the initial installation of QMF for Windows and QMF for Windows Administrator on the server, you are ready to install QMF for Windows on other machines.

Configure for unattended installation

An unattended installation allows you to select the installation options for your QMF for Windows users prior to the installation process. This allows you to designate installation options rather than selecting the same options for each installation. The options are defined when you edit the **qmfwin4.reg** and **setup.ini** files, as described in the ″Server Install″ section above.

Sample setup.ini for client install

When performing a client install, the DBA edits the **setup.ini** file in Disk1 with options that control user actions. For example:

```
[Options]
AutoInstall=1
FileServerInstall=1
SetupType=2
InstallPath=\\ServerName\Root\Program Files\IBM\QMF for Windows
ProgramGroup=QMF Win 7.2
AutoExit=1

[Components]
Base=1
Admin=0
Excel=1
ReportCenter=1

[Startup]
AppName=QMF for Windows 7.2
FreeDiskSpace=481
```

These settings are specific to a client install:

- **FileServerInstall=1:** this indicates that QMF for Windows has been installed into the directory specified in the InstallPath. Therefore, the client computer will have a ″thin″ install. This allows users to maintain only minimal files on their machines.
- **InstallPath:** identifies the directory where QMF for Windows has been installed on the server. Go to Network Neighborhood or use the Explorer function to locate the path to the server, if needed.

Run QMF for Windows installation

Instruct the Client to point through the Network Neighborhood or Explorer to Disk1 on the server or remote computer. Open Disk1 and double-click **setup.exe**. QMF for Windows will be installed automatically with the specified choices.

Because QMF for Windows is installed on the server, the users maintain only minimal files on their client machines. The main QMF for Windows files reside on the server. The administrator performs general administrative maintenance and dispatches a new **setup.ini** file only when users need access to or should be excluded from a particular function.

A client installation must be performed in unattended mode.

Files installed on the client

The following files are installed on the client.

Table 3. Files installed on client

Platform	File name	Description	Version
	CTL3D32.DLL		2.3.1.000 (Windows NT 3.51 only)
	MFC42.DLL	MFC	4.21.7303
	MSVCRT.DLL	C runtime	5.00.7303

Chapter 4. Defining Servers

This chapter provides a step-by-step guide to configure and administer QMF for Windows, starting with the QMF for Windows Administrator application. It also provides detailed examples regarding the execution of DBA functions. The intended audience is the database administrator responsible for configuring and maintaining QMF for Windows.

Working with QMF for Windows Administrator

QMF for Windows Administrator is the administrative component of QMF for Windows. Using this application is strictly an administrative task. There are four basic tasks that you perform with QMF for Windows Administrator:

- Defining and configuring database servers that QMF for Windows will access
- Creating QMF for Windows database objects, binding packages, and granting permissions in each database server that QMF for Windows will access
- Creating QMF for Windows sample tables in each database server that QMF for Windows will access
- Administer QMF for Windows users, governing (resource limits), and object tracking (via Object Tracking Reports)

About the SDF file

When you use QMF for Windows Administrator, you edit a server definition file (SDF). The SDF contains the technical information needed by QMF for Windows to access database servers. There are two ways to use server definition files:

1. You can allow each user to have his- or her own SDF.
2. You can create a single SDF that is shared by users over a file-sharing network.

   The second approach has the advantage that it centralizes SDF administration. You only need to create and maintain a single file, and your users need only to point to that file when they run QMF for Windows.

You can create a new SDF for a user or group of users by selecting the File: New or File: Save As command. Select File: Open to open and work with a different SDF.

Note: The server definition file is created and edited using the QMF for Windows Administrator application. Editing this file using any other method is not recommended, because it can corrupt the file.

QMF for Windows comes with a server definition file (SDF) with a preconfigured database server, QMF Demo. The database can be used for trial purposes through a TCP/IP connection. QMF Demo can also be used as a template when you set up new servers.

Click **New** at the Servers dialog box to define a database server for QMF for Windows from the Administrator main window. The General parameters dialog box displays. Clicking **New** configures a new server by using the selected server as a template. A template server is included with QMF for Windows. You can customize the template server for your needs (such as connection type and name).

Information entered on this dialog and subsequent dialogs is written to the server definition file (SDF) to tell the QMF for Windows how to find the database. See the appendix, "The SDF File," for more information.

Important: The sections below describe how to **set up a new database server** and describe the purpose and content of every dialog box in the server setup routine.

Alternatively, a window with various tabs displays when you select the option to edit an existing server. The tabs on this window contain the same dialogs as you see when setting up the server.

Servers

This dialog box displays when you open QMF for Windows Administrator. Initially, only the sample server, *QMF Demo*, is listed. This server is delivered with QMF for Windows when you receive the product. Use QMF Demo as a template when you create a server. This dialog box enables you to add, edit, and delete servers from a server definition file (SDF).

A server definition file is an initialization file that contains the technical information that QMF for Windows needs to connect to database servers. The system administrator is responsible for creating and maintaining server definition files for QMF for Windows users. By default, the server definition file name is the same as the QMF for Windows .ini file.

Specify the name of a server definition file in the Options dialog box to use QMF for Windows.

Servers

Lists the database servers. A sample database, *QMF Demo*, is included when you install QMF for Windows. You can use this server as a template for the new servers you set up.

Enable dynamic ODBC server definitions

Check this box to enable dynamic ODBC server definitions. ODBC (Open Database Connectivity) is a programming interface that enables applications to access data in database management systems that use Structured Query Language (SQL) as the data access standard.

Because defining and configuring each user's ODBC data sources is not always an option for a central database administrator, QMF for Windows Administrator offers the option of dynamically defining ODBC data sources. These dynamically-defined servers are available for all locally-defined ODBC data sources, just as any other server. The ODBC data sources are defined within QMF for Windows, using the data source name as the server name, except for data source names that exceed 64 characters or those that are used by a server defined in the server definition file. Supply the catalog server and catalog name in the required fields to configure ODBC data sources as servers.

Procedure

Follow these steps to set up ODBC data sources as servers:

1. From the QMF for Windows Administrator startup window, check ″Enable dynamic ODBC server definitions.″
2. To the right of the check box, click **Properties**.

   The Server Parameters property sheet appears with nine tabbed windows.
3. Proceed with setup options.

   Note: The Connection, Packages, and Sample Tables tabs are disabled.

When you start up QMF for Windows, you can choose a server from the locally-defined ODBC data sources.

Table 4. Server parameters properties

Control	Description
Properties	Set properties if you are enabling dynamic ODBC server definitions. You advance to the Parameters for Dynamic ODBC Servers dialog box.
New	Create a database server. You advance to the General dialog box where you can name the server.

Table 4. Server parameters properties (continued)

Control	Description
Edit	Advance to the Parameters for [server] dialog box where you may update information on these tabs: • **General:** the server name • **Security:** set security parameters and handling instructions • **Connection:** the connection method between QMF for Windows and the database server • **Packages:** the collection ID, delimiters for SQL statements, and the Packages wizard • **Catalog:** the catalog to use at the database server, which identifies where to find and save content for procedures, forms, and queries • **Resource Limits:** governing and schedules for user access and system performance • **User Profiles:** creator and resource group • **Sample Tables:** sample tables for learning QMF for Windows • **Object Tracking Reports:** run specific performance reports
Delete	Deletes a server connection. 1. Select the database server you want to delete at the QMF for Windows Administrator main dialog box. 2. Click **Delete**. 3. Confirm your deletion. The server entry is deleted from the list of available servers. If you delete a server connection, the QMF for Windows installation objects created at that database server are not deleted. To clean up QMF for Windows objects, delete them explicitly at the server.

General

Name

At the General dialog box, the server. This name is used throughout QMF for Windows and QMF for Windows Administrator. Enter a name up to 64 characters.

Note: Because QMF for Windows users work with the server name, make the name descriptive, not technical. The server name is seen only by the end-users.

The server name is all that QMF for Windows needs to know to access the server. All technical details about how to access the server are hidden behind the server name in the SDF.

Table 5. General parameters: controls

Control	Description
Back	*This control is disabled.*
Next	Advance to the Connection dialog box after specifying the server name and setting up security parameters
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for the current dialog box

Security

User ID and password are required

The security option determines whether a user ID and password are required to connect to the server.

Encrypt passwords

Choose whether passwords are encrypted when they are sent over the network to a server

Allow users to change passwords

Choose whether to allow users to remotely change their passwords without logging on to the server.

Note: This applies only to DRDA TCP/IP, SNA connections. The server must support this capability.

Allow users to save passwords

Choose whether to allow users to save passwords, in encrypted format. After a user has logged on to Windows, the saved passwords are retrieved and utilized, without the user having to enter them again. By checking this box, users are allowed to save passwords.

Table 6. General parameters: security controls

Control	Description
Back	Return to the General dialog box.

Table 6. General parameters: security controls (continued)

Control	Description
Next	Advance to the Connection dialog box after specifying the server name and setting up security parameters
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for the current dialog box

Connection

Connection types

QMF for Windows accesses data stored in any database in the DB2 family of databases and a list of ODBC data sources. QMF for Windows can connect to servers using these connections:

- DRDA over TCP/IP
- DRDA over SNA
- DB2 for Windows CLI
- ODBC

Note: Prior to installing QMF for Windows, determine your connection type and obtain other important setting information.

The information entered here is written to the .ini file (Server Definition File).

Because QMF for Windows implements the DRDA requestor specification, it is capable of connecting to any database that adheres to and implements the DRDA server component. The IBM database products that contain a DRDA server component that can communicate directly with QMF for Windows are:

- DB2 UDB for z/OS, DB2 for z/OS, and DB2 for MVS
- DB2 Server for VSE&VM, and SQL/DS
- DB2 UDB for iSeries
- DB2 Universal Database and DB2 Common Server
- DB2 Parallel Edition
- DB2 DataJoiner

Use QMF for Windows Administrator to define each server by giving it a name and providing technical information for QMF for Windows to access it. This process is similar to defining a data source in ODBC.

Click **New** at the QMF for Windows Administrator main window for define a database server to QMF for Windows. Enter the server name (on the General

dialog box) and security values (on the Security dialog box), then specify the connection type on the Connection dialog box.

Once you click on a connection type, the lower half of the dialog box changes and presents parameters specific to the network connection.

Each connection type is described below.

Note: See the appendix, "Architecture, Communication, and Configuration," for a detailed description of each connection type.

Connect using DRDA over TCP/IP

This is the default connection method and the most highly used.

Note: Not all DB2 host servers (DB2 UDB for z/OS, DB2 for z/OS, and DB2 for MVS) support TCP/IP connections. If you do not know whether your DB2 z/OS is configured to use TCP/IP, see the Sync Port and host name in DB2xMSTR.

If DB2x is not configured to support TCP/IP, see the *DB2 Installation Guide* provided with your DB2 license for instructions on how to configure your system.

Host name

The IP address of the machine housing the database.

If you enter a TCP domain name for the host name, QMF for Windows Administrator resolves the name to an address using the `GetHostByName` socket call. Or, you can specify the host in dotted decimal notation (such as, 1.2.3.4).

Port number

The TCP/IP port for which the database server is configured with its listener port in the services file.

Note: Select one of the following options if you are unsure of the port number to use on DB2 MVS / z/OS:

- Run the Print Log Map Utility DSNJU004. Max to the bottom of the output to the second line up from the bottom of the output:

  ```
  LOCATION=DB2PDDF LUNAME=DB2PDDF PASSWORD=(NULL)
  GENERICLU=(NULL) PORT=NULL RPORT=NULL
  ```

  This gives you the RDB name listed as LOCATION=DB2PDDF
- Look at the DB2DMSTR job that is running on the mainframe as an address space. Find the following DSNL004I message:

```
,-------------------------------------------,SDSF OUTPUT DISPLAY
DB2DMSTR ,COMMAND INPUT ===>,
19.24.18 STC01839 DSNL004I - DDF START COMPLETE
LOCATION DB2DDDF
LU FFX1.DB2DDDF
GENERICLU -NONE
DOMAIN -NONE
TCPPORT 0
RESPORT 0
```

In the above examples the TCPPORT has been set to zero, which is a de-activated state. DB2 DRDA is not listening on any port.

You need to update your BSDS to set this port to 446 for DB2 to start listening. Run the Change Log Inventory utility.

1. Ensure that the DB2 is down.
2. Run the utility with the following SYSIN card:

   ```
   DDFPORT 446
   ```
3. Start up DB2 -start with DB2 on the console and then DDF facility -sta DDF on the DB2I panel.
4. Inform your network administrator to modify the TCP/IP profile to include the following:

   ```
   PORT 446 TCP OMVS               ; DRDA port for DB2D
   ```
5. Restart TCP/IP and DB2 should respond to your request.

The default port for DB2 to listen for DRDA TCP/IP conversations on the iSeries is 446. The default port (446) can already be assigned, but is not active. Or, the port may have been changed to another value. If so, you receive the WSAECONNREFUSED error message when QMF for Windows tries to connect.

Verify Listen State for iSeries

Follow these steps to verify whether the DRDA port is in a listen state:

1. Type **NETSTAT** on the iSeries command line.
2. Select option 3 (Work with TCP/IP connection status).
3. Look for DRDA under the Local Port column. It should be State = Listen.
   - If DRDA is under the Local Port column and the state is set to ″Listen,″ go to the next section, ″Modify User Profile CCSID.″
   - If DRDA is not there, follow these steps to start DDM services on the iSeries:
     a. Type **STRTCPSVR *DDM** at the iSeries command line.
     b. Verify that DRDA is in a Listen state (repeat step #3).

     Or,

a. Type **GO TCPADM** at the iSeries command line.
b. Select Option 3, "Select TCP/IP."
c. Verify that DRDA is in Listen state (restart Step #3).

Note: Follow these steps to ensure that the **DDM** server, which opens the DRDA port, will autostart when the **STRTCP** command is issued during system startup:

1. Type **GO TCPADM** at the iSeries command line.
2. Select Option 2, "Configure TCP/IP Applications."
3. Select Option 5, "Change DDM TCP/IP Attributes."
4. Change the Autostart Server prompt to ***yes**.

RDB name

The name of the relational database (RDB) server. For DB2 for z/OS or MVS, this is also known as the "location name." The RDB name is not visible to QMF for Windows users. In DB2 Universal Database (UDB) or DB2 Common Server technology, the RDB name is known as the "database name."

If you are unsure of the value to enter here, there is an easy way to determine the correct value. Using some tool other than QMF for Windows, run the following query at the server (you can use any table if SYSIBM.SYSTABLES does not exist):

```
SELECT DISTINCT CURRENT SERVER FROM SYSIBM.SYSTABLES
```

The resulting value is the RDB name for the database server.

The RDB name is the name of the database that is taken from the iSeries Relational Database Directory Entry. Use the following command from the iSeries command line to locate the RDB name:

```
DSPRDBDIRE
```

Enable load balancing

Load balancing is a way for QMF for Windows to maintain a balanced connection to the database server. If you enable load balancing and the DB2 supports this feature, QMF for Windows receives a list of IP addresses with rankings as to how often it wants to be connected to each address. QMF for Windows takes this list and goes to a specific IP address for its first connection. Next, it will go to the next address, then to the third address, and so on. If an IP address is busy or not working, load balancing will go to another address. This balances the load on each address and provides multiple addresses if the first address cannot be used.

Note: Disregard this setting if installing QMF for Windows on AIX, OS/2 or Windows UDB DB2. It applies only to z/OS and VM/VSE data sharing.

Modify user profile CCSID

The iSeries is shipped with the QCCSID system value set to 65535. In a DRDA environment, this CCSID does not allow for translations. Change the user profile for the users with QCCSID of 65535 to have a CCSID of your national language. For example, the CCSID for US English is 37.

Using English as an example, type this command from the iSeries command line to change the CCSID:

```
CHGUSRPRF USRPRF (USERID) CCSID(37)
```

For more information on iSeries CCSID considerations, please refer to the *OS/400 Distributed Database Programming Manual (SC41-5702)*.

Note: If you do not know your CCSID, refer to the i*Series National Language Support Manual (SC41-3101*) manual.

Connect using DRDA over SNA

DB2 UDB for z/OS, DB2 for z/OS, and DB2 for MVS are some of the IBM database products that contain a DRDA server component and are capable of communicating with QMF for Windows via SNA.

SNA Parameters

- **Symbolic destination name**

  Before configuring this connection, you must specify the SNA software that you are using to implement CPI-C in your Windows environment. This is a two-part process:

  1. Identify the **DLL** that your SNA software provides for CPI-C applications:
     a. From the QMF for Windows Administrator dialog box, select Edit: Options.
     b. Select the CPI-C tab on the Options dialog box.
     c. Specify the name of the DLL that your SNA software provides for CPI-C applications. The name of the provider DLL typically is **wcpic32.dll**.
  2. Define the CPI-C **symbolic destination name** for the server before you install QMF for Windows. The CPI-C symbolic destination name is defined in your SNA software.

- **RDB name**

  This is the name of the relational database (RDB) server. For DB2 for z/OS or MVS, this is also known as the "location name." The name is not visible

to QMF for Windows users. In DB2 UDB or DB2 Common Server technology, the RDB name is known simply as the ″database name.″

If you are unsure of the value to enter here, there is an easy way to determine the correct value. Using some tool other than QMF for Windows, run the following query at the server (you can use any table if SYSIBM.SYSTABLES does not exist):

```
SELECT DISTINCT CURRENT SERVER FROM SYSIBM.SYSTABLES
```

The resulting value is the RDB name for the database server.

- **Enable load balancing**

  When checked, this option enables load balancing for your database server.

Note: This configuration is outside the control of QMF for Windows. Work with your in-house networking staff to implement and support your network configuration.

Connect using DB2 for Windows CLI

Use the DB2 UDB facilities to define your database servers and how to connect to them in order for QMF for Windows to connect to DB2 via CLI.

Note: This configuration is outside the control of QMF for Windows. Work with your in-house networking staff to implement and support your network configuration.

QMF for Windows requires the DB2 UDB client, version 6.1 or later, to access the database. CLI connections are supported for these databases:

- DB2 Universal Database for z/OS, version 6 and later
- DB2 Common Server, Versions 1 and 2
- DB2 UDB for workstation databases, Version 5 and later
- DB2 for MVS, Versions 2.3, 3.1, and 4.1

  Note: DB2 for MVS requires that DB2 Connect is installed locally or as a gateway. Even though QMF for Windows supports CLI connectivity to DB2 for MVS, you should use a DRDA connection for your DB2 for MVS database.
- DB2 for z/OS, Version 5

  Note: DB2 for z/OS requires that DB2 Connect is installed locally or as a gateway. Even though QMF for Windows supports CLI connectivity to DB2 for MVS, you should use a DRDA connection for your DB2 for MVS database.
- DB2 DataJoiner, Version 2
- SQL/DS for VM/VSE, Version 3

- DB2 for VM/VSE, Version 5 and later
- DB2 for iSeries, Version 4.3 and later

DB2 Connect must be installed locally or as a gateway to connect to DB2 for MVS or z/OS.

Note: Although QMF for Windows supports CLI connectivity to DB2 for MVS and DB2 for z/OS, it is recommended that you use a DRDA connection for your DB2 for MVS or DB2 for z/OS databases.

Database alias

Once you configure the database server connection at the DB2 UDB client, specify the *database alias* to create the connection inside QMF for Windows Administrator.

A CLI connection uses a simplified configuration. If a database is defined in the DB2 UDB client, QMF for Windows only needs its alias. The network configuration is performed as part of the DB2 UDB client configuration rather than as a part of QMF for Windows configuration.

To access a DB2 UDB server using CLI, the 32-bit version of QMF for Windows must be able to establish a CLI connection from the local host (where QMF for Windows is running) to the remote host (where DB2 UDB is running) via the DB2 UDB client.

QMF for Windows requires the DB2 UDB client, version 6.1 or later, to access the database via CLI and supports CLI connections to DB2 UDB, DB2 Parallel Edition, and DB2 DataJoiner database servers.

CLI connections are supported for these databases:

- DB2 Universal Database for z/OS, Version 6 and later
- DB2 Common Server, Versions 1 and 2
- DB2 UDB for workstation databases, Version 5 and later
- DB2 for MVS, Versions 2.3, 3.1, and 4.1

  Note: DB2 for MVS requires that DB2 Connect is installed locally or as a gateway. Even though QMF for Windows supports CLI connectivity to DB2 for MVS, you should use a DRDA connection for your DB2 for MVS database.
- DB2 for z/OS, Version 5

Note: DB2 for z/OS requires that DB2 Connect is installed locally or as a gateway. Even though QMF for Windows supports CLI connectivity to DB2 for MVS, you should use a DRDA connection for your DB2 for MVS database.

- DB2 DataJoiner, Version 2
- SQL/DS for VM/VSE, Version 3
- DB2 for VM/VSE, Version 5 and later
- DB2 for iSeries, Version 4.3 and later
- DB2 Universal Database
- DB2 Common Server

You must have DB2 Connect installed locally or as a gateway to connect to DB2 for MVS, DB2 for z/OS, and DB2 UDB for z/OS using CLI. Although this connectivity is possible, it is recommended to connect to those platforms using a DRDA connection to avoid performance degradation at the DB2 Connect gateway. However, if DB2 Connect is installed and its performance is sufficient for the users accessing this gateway, this allows for an easy setup for QMF for Windows.

Connect using ODBC

ODBC, Open Database Connectivity, is an open standard method of accessing data. The data to be accessed needs:

- a driver (interpreter), which understands the format of the stored data
- a connection manager that determines how the connection needs to be made

This information is stored in a data source name (DSN). Use the 32-bit ODBC function in the Control Panel to set up an ODBC DSN.

Note: This is outside of QMF for Windows and must be performed by your DBA.

"Data source" refers to where data is kept. It can be a file, typically a database on a database management system, or a live feed. A data source gathers the technical information required to access the data (such as the driver name, network address, and network software) into a single place and make the data access transparent to the user.

Data sources are created by the DBA with a program called ODBC Administrator. When adding a data source, the ODBC Administrator presents a list of available database drivers. The administrator chooses a driver, then calls the selected driver. Use the driver's dialog box to enter information that the driver needs to connect to the data source. The driver stores this information on the system.

When an application needs to access the data from those databases, it calls the driver manager and passes the name of the machine data source or the path of the file data source.

Machine data source name

Machine data sources are stored on the system with a user-defined name. Associated with the data source name (DSN) is all the information the database driver needs to connect to the data source and that the driver manager needs to coordinate all data sources and drivers.

When an application needs to access the data from the database, it calls the driver manager and passes the name of the machine data source. When a machine data source name is passed, the driver manager searches the system to find a driver used by the data source. It loads the driver and passes the data source name to it. The driver uses the data source name to find the information it needs to connect to the data source. Finally, it connects to the data source, typically prompting the user for a user ID and password, which are generally not stored.

A system data source is typically used on one machine by all users of that machine. The information about the data is stored in the machine's registry under the HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBC.INI. Here, all information about the defined system data sources is stored. The information about the installed drivers is stored under the HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBCINST.INI registry key.

System DSNs are typically used when there is a need to access data from one machine. This usually is a server type process.

User data source name

User data source refers to data sources that are stored on your PC. The data sources available are user data sources (data sources available for a designated user) or a system data source (available for all users who log on to the PC).

File data source name

Refers to data sources that are stored on your PC. The data sources available are either user data sources (data sources available for a designated user) or a system data source (available for all users who log on to the PC).

When an application needs to access the data from the database, it calls the driver manager and passes the name of the file data source. When a file data source is passed, the driver manager opens the file and loads the specified driver. If the file contains a connection string, it is passed to the driver. Using

the information in the connection string, the driver connects to the data source. If no connection string is passed, the driver generally prompts the user for the necessary information.

A file data source stores the information about the data, and how to connect to it, in a disk file. File data sources can be shared among all users who have access to the file.

The control below applies only to ODBC connections.

Table 7. ODBC connections controls

Control	Description
Advanced	Enter driver-specific connection string keywords for this data source at the Advanced DSN Settings dialog box **Note:** You may be prompted to enter your iSeries log-on and password information.

The controls below apply to all Connection dialogs.

Table 8. Connection controls

Control	Description
Set User Info	Provide User ID, Password, and optional Account information. This information provides access to the server. The following message displays when you click **Set User Info**: `Users of the server definition file that you are currently editing must supply their own user information. The information you supply now applies only to you.` You are required to set user information and the Set User Info dialog box appears automatically before you establish a connection to the database server. The connection to the server is tested automatically after you supply a valid User ID and password.
Test	Test the connection to the server. You must provide a valid user ID and password on the Set User Info dialog box that displays automatically. If you do not click **Test**, the connection to the database server is tested automatically regardless of the connection type. You must establish a connection before you can continue setting up the server.
Back	Return to the Security dialog box

Table 8. Connection controls (continued)

Control	Description
Next	Advance to the Set User Information pop-up window where you specify your User ID, password, and optional account information, then test the connection to the server. **Note:** The Set User Information pop-up window displays only if user information has not been defined. Otherwise, the Packages dialog box displays.
Cancel	Undo your entries and return to the Servers dialog box
Help	View online help for the current dialog box

Set user information

Click **Set User Info** to enter user information to gain access to a database server. These settings affect the operations you perform at this database server from QMF for Windows Administrator. It does not affect or apply to any other users of the server definition file that you are editing.

The following message displays when you click this control:

```
Users of the server definition file that you are currently editing
must supply their own user information. The information that you supply
now applies only to you.
```

Click **OK** to return to the Connection dialog box.

Note: The Set User Information pop-up window appears automatically when you click **Test** to test the connection to the database. This is necessary because you must have permission to access the database server before you can connect to it.

Specify a valid user ID and password for the database server you are trying to access, not a local or LAN user ID and password. Case is preserved in the user ID and password you enter. For example, if your user ID or password are uppercase, make enter these values in uppercase here. Some types of database servers treat user IDs and passwords as case-sensitive, while others do not.

Note: The User ID and password combination are validated.

User ID

Specify a valid user ID to connect to the database server.

Note: The User ID and password combination are validated.

Password

Specify a valid password to connect to the database server.

Note: The User ID and password combination are validated.

Remember this password

Check **Remember this password** if you want QMF for Windows to remember your password across sessions.

The default is checked (″yes″).

Account

Specify an accounting string, which will send accounting information to the database server when connecting. The database server uses the account to track system usage by a department, for example. Accounting strings may be set up for a resource group at the Options tab on the Resource Limits Group Schedule dialog box. Also, options are available to override the account or to require the entry of an accounting string. See ″Resource limits″ below.

The controls below apply to the Set User Information dialog box.

Table 9. Set user information controls

Control	Description
OK	Accept and process your entries
Cancel	Undo your entries and return to the previous dialog box
Change	Display additional fields to enter user information

Change user information

Additional fields display when you click **Change** at the Set User Information dialog box. Make entries to these fields to change password or account information for the user. The current user ID displays along with the account number, if any.

You can only change the password and account number at this dialog box and specify whether to remember the password.

Note: You cannot change the user ID.

Procedure

1. Type the new password in the *New password* field.

   Note: This applies only if you are changing the user's password.
2. Press **Tab**.

3. Retype the new password in the *Confirm new password* field.
4. Check or uncheck the *Remember this password* checkbox, if applicable.
5. Enter the account number in the *Account* field, if applicable.
6. Click **OK** to change the user's information.

Table 10. Change user information controls

Control	Description
OK	Accept and process your entries
Cancel	Undo your entries and return to the previous dialog box
Hide	Hide the fields to change user information, including the *New password*, *Confirm new password*, *Remember this password*, and *Account* fields

Test connection

To ensure that QMF for Windows can establish a connection to the database server, select the server in the QMF for Windows Administrator main window and click **Edit**. Otherwise, for a new server, click **Test** to establish a network connection with the database server.

Note: The Set User Information pop-up window appears and must be completed before a connection to the database server can be established. You must have the authority to connect to the database server before you test the connection.

If installing on an iSeries, you must have SECADM authority.

There are only a few errors that can occur when establishing a connection to the server. A problem at this point usually indicates a problem with the network configuration rather than with QMF for Windows Administrator.

Use **Copy** to copy the error message to the clipboard to be saved, then used for further investigation.

Common errors

If there are problems with your network configuration, QMF for Windows Administrator displays an error message dialog box when trying to establish the connection.

Common errors include:

- Failure to activate the SNA software or start the SNA node
- Failure to activate the SNA link

- Failure to properly configure an LU 6.2 session between the QMF for Windows LU and the DB2 LU

  See ″LU6.2 connection errors,″ below.

LU 6.2 connection errors

If you are connecting to DB2 via an SNA network, the SNA link and the LU 6.2 session between the QMF for Windows LU and the DB2 LU must be active to establish the connection.

The LU 6.2 protocol provides a consistent method for programs to:

- Identify and negotiate the communications options to be used by each partner program
- Provide the name of the partner destination and program
- Supply end-user security parameters associated with the request on the remote platform
- Control message transmission
- Synchronize the processing between partner programs
- Perform coordinated commit processing

Tracing errors

If an error occurs, use the return code to diagnose the problem with your technical support services and your software vendor. Turn on tracing in QMF for Windows to investigate the problem:

1. Click **Copy** to copy the error message to the clipboard. You need this information to track down the specific error.
2. Select Edit: Options from the QMF for Windows Administrator main window.
3. Click **Traces** on the Options dialog box. This dynamically traces the types of errors you specify and write the results to a log file.
4. Review the listed trace file names; the defaults are probably acceptable.

   QMF for Windows writes data to the specified trace file, until the file size exceeds 512K bytes. Then QMF for Windows begins writing data to Trace file 2. When the size of this second file exceeds 512K bytes, QMF for Windows switches back to Trace file 1. This prevents QMF for Windows from consuming disk space with trace information while allowing for a reasonable running history of trace information.
5. Select the component to trace for errors. Choices are:
   - **DDM:** distributed data management architecture, which is used by the distributed file management and the distributed relational database protocol to define the protocol for communicating between two systems using the distributed file management and the distributed relational database.

Note: Select this option only if you use a DRDA connection.

- **TCP:** (transmission control protocol) TCP/IP is a host-to-host protocol that provides transmission in an internet environment. TCP assumes Internet Protocol (IP) is the underlying protocol.

  Note: Select this option only if you use a TCP/IP connection.
- **CPI-C:** CPI-C is a programming interface that implements the APPC verb set. Applications that require the APPC verb set can be written using CPI-C to achieve SNA vendor independence. QMF for Windows is written using CPI-C.

  Note: Select this option only if you use an SNA connection.
- **CLI:** CLI is an application programming interface for relational database access. Using CLI, client applications, such as QMF for Windows, can connect to DB2 UDB servers and execute SQL statements.

  Note: Select this option only if you use ODBC to access data in a database management system.
- **ODBC:** (Open Database Connectivity) ODBC is a standard that provides applications with a common interface to data held in relational databases. Any application that can act as an ODBC client can access data held in any relational database that has a suitable ODBC driver. ODBC accepts Structured Query Language (SQL) data requests from the client application and translates these into a form acceptable to the target database engine.

  Note: Select this option only if you use ODBC to access data in a database management system.
- **Embedded SQL:** Embedded SQL allows programmers to connect to a DB2 database using the DB2 native interface to manipulate and process data.

  Note: Select this option only if you use ODBC to access data in a database management system.
- **SQLAM:** (SQL Application Manager) SQLAM is a function that represents an application to the remote relational database manager. The SQL application manager can be present in an application requestor and an application server.

  Note: Select this option for any type of connection.
- **REXX:** IBM Object REXX is a programming language available for the mainframe and Windows environments. IBM Object REXX handles simple and complex calculations, logic, column definitions, detail variations, and conditions in QMF for Windows

Note: Select this option only if you are tracing a procedure that uses IBM Object REXX .

- **Internet Mail**
- **Messages**

6. Click **OK** on the Traces dialog box and then on the Options dialog box.
7. Repeat the action that caused the error.

A detailed trace of the calls that QMF for Windows Administrator made is written to the specified trace files. You can use this trace when seeking technical support.

Note: The user ID, password, and other sensitive data will be written in clear text in the trace files.

Important: Turn tracing on only when diagnosing a problem. Tracing can affect the performance of QMF for Windows.

Packages

Collection ID

The user-defined name that identifies the packages that are created for QMF for Windows and QMF for Windows Administrator. You should use a name that reflects the product and version used, such as *QMFW72*. The default name of the Collection ID is NULLID. Twelve (12) packages are bound in this collection ID. Six (6) packages are bound for a CLI connection:

Table 11. QMF packages

DRDA	CLI	DRDA	CLI
RAARDBII	N/A	RAARDBX1	N/A
RAARDBX2	N/A	RAARDBX3	N/A
RAARDBX4	N/A	RAARDBX5	N/A
RAARDBI2	RAARDBC2	RAARDBIA	RAARDBCA
RAARDBIL	RAARDBCL	RAACHUT1	RAASHTC1
RAARDBI1	RAARDBC1	RAARC1	RAARC1C

Note: RAARDBIL (RAARDBCL) is only available to servers that support large objects.

Decimal delimiter

Select the decimal delimiter to be used in SQL statements executed at the server. The default is a period [.].

String delimiter

Select the string delimiter to be used in SQL statements executed at the server. The default is an apostrophe [’].

Table 12. Packages controls

Control	Description
Wizard	Start the bind packages wizard
Back	Return to the previous dialog box
Next	Advance to the subsequent dialog box
Cancel	Undo your entries and return to the previous dialog box
Help	Obtain online help for this dialog box

Packages wizard

After you verify that QMF for Windows can establish a connection to the database server, you can bind the QMF for Windows packages using the bind packages wizard. Click **Wizard** at the Packages dialog box to start the bind packages wizard. The wizard creates the database objects and binds the packages during server setup.

Use the packages wizard for TCP/IP, SNA and CLI connections.

Catalog option

A QMF for Windows catalog is a set of database tables that contain resource limits, saved objects (queries, procedures, forms), users, reports, and other settings. A catalog is required for each server that hosts the QMF for Windows catalog. These tables can be created at the same server that is being accessed or at a different server. A catalog table is a ″table about tables,″ meaning it is not a data table.

Select a server on the server definition file that is capable of hosting the catalog, for example, the ODBC catalog tables. An ODBC catalog server can use the default catalog and it will share resource limits for the catalog server. Create these database tables if you want the current server to host a QMF for Windows catalog for its own use or for use from a different server.

Create the QMF for Windows catalog tables

Select this option to create the QMF for Windows catalog tables, which enable the server to host a QMF for Windows catalog. The default is ″yes.″

Do not create the QMF for Windows catalog tables

Select this option, which does not create the QMF for Windows catalog tables.

Table 13. Catalog controls

Control	Description
Back	*This control is disabled*
Next	Advance to the subsequent dialog window
Cancel	Undo your entries and return to the previous dialog box
Help	Obtain online help for this dialog box

Package properties

Specify properties for the QMF for Windows packages. Some of these properties may not be supported by the type and version of your database product. Consult the database documentation to see which properties it supports.

Note: When you define multiple server entries within the SDF, accessing the same database but using different network connection types, use different collection names for each network connection type. If not, only the users accessing the server using the last bound packages will be able to access the server.

Collection ID

The collection ID, also known as the collection name, is a user-defined name that groups the packages that are created for QMF for Windows and QMF for Windows Administrator. Use a name that identifies the product and its version, such as *QMFW72*. The collection name is limited to eight (8) characters. The collection name is limited to six (6) characters for a CLI connection.

In this collection, 12 packages will be bound and six (6) packages will be bound for a CLI connection.

Table 14. Collection IDs

DRDA	CLI	DRDA	CLI
RAARDBII	N/A	RAARDBIL	RAARDBCL
RAARDBI1	RAARDBC1	RAARDBX1	N/A
RAARDBI2	RAARDBC2	RAARDBX2	N/A
RAARDBIA	RAARDBCA	RAARDBX3	N/A
RAASHUT1	RAASHTC1	RAARDBX4	N/A
RAARC1	RAARC1C	RAARDBX5	N/A

Note: Packages containing the characters ″RC″ are packages for Report Center; those with an ″L″ are packages for large objects, and those with an ″X″ are packages for the extended catalog (which use a different isolation level).

Owner ID

To bind the packages you need authority at the server to execute the SQL that the packages contain. The configuration process requires a high level of authority to CREATE tables, bind packages, create views on system tables, and grant EXECUTE authority on these packages.

Leave this field blank if your primary authorization ID has these authorities. Enter your secondary authorization ID if you use this ID for administrative tasks.

Note: If you are connecting to an AIX DB2 and you receive an SQL Code =-4390 error message while binding the packages, clear the Owner ID field and bind again.

Use upper-case letters for the owner ID when you define a connection to DB2 UDB for z/OS, DB2 for z/OS or DB2 for MVS. QMF for Windows does not translate lower-case to upper-case.

Decimal delimiter

Select the decimal delimiter to be used in SQL statements executed at the server. The default is a period [.].

String delimiter

Select the string delimiter to be used in SQL statements executed at the server. The default is an apostrophe [’].

Advanced binding options

Click **Advanced** to specify advanced binding options. Five (5) tabs appear on the Bind options dialog box:

- Identification
- Formats
- Statement Performance
- Statements
- Miscellaneous

Note: Each tab is described below in a separate table.

Table 15. Package properties controls

IDENTIFICATION Tab	Field Description
Collection ID	A name that identifies the packages created for QMF for Windows and QMF for Windows Administrator. Use a descriptive name, such as *QMFW72*.
Owner ID	The ID that grants you authority at the server to execute the SQL contained in the package. The owner ID enables you to CREATE tables, bind packages, create views on system tables, and grant EXECUTE authority on these packages. Leave this field blank if your primary authorization ID has the required authorities. Use your secondary authorization ID if you use it for administrative tasks.
Qualifier	A qualifier for the Collection ID and the Owner ID.
Comment	[Optional]: A comment about the binding of these packages.
Replace existing package	Whether to replace existing packages. Check this box. All existing packages are replaced when you install a new version or service release of QMF for Windows. All clients with a full QMF for Windows installation will need to upgrade to the new version as well. All QMF for Windows installations with older versions will no longer be able to connect to the server.
Keep existing authorizations	Keep existing authorizations. The default is checked (″yes″).
OK	Accept your entries and return to the previous dialog box
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Table 16. Format parameters

FORMATS Tab	Field Description
Decimal delimiter	The decimal delimiter to be used in SQL statements executed at the server. Choices are: • Unspecified: no delimiter is specified in the package • Period [.]: a period delimiter is specified in the package. This is the default. • Comma [,]: a comma delimiter is specified in the package

Table 16. Format parameters (continued)

FORMATS Tab	Field Description
String delimiter	The string delimiter to be used in SQL statements executed at the server. Choices are: • Unspecified: no delimiter is specified in the package • Apostrophe ['']: an apostrophe delimiter is specified in the package. This is the default. • Quote ["]: a quote delimiter is specified in the package

Date format	The date format to use when binding packages. Choices are:
	• Unspecified: no date format is specified in the package
	• ISO: the International Organization for Standardization date format is specified in the package. The format is:
	`YYYY:MM:DD`
	• USA: the date format used in the United States is specified. The format is:
	`MM:DD:YYYY`
	• EUR: The European date format is specified. The format is:
	`DD:MM:YYYY`
	• JIS: The Japanese Industrial Standard date format is specified. The format is:
	`YYYY-MM-DD`

Time format	The time format to use when binding packages. Choices are:
	• Unspecified: no time format is specified in the package
	• ISO: the International Organization for Standardization time format is specified in the package. The format is:
	`HH.MM.SS`
	• USA: the time format used in the United States is specified. The format is:
	`HH:MM AM`
	• EUR: The European time format is specified. The format is:
	`HH:MM:SS`
	• JIS: The Japanese Industrial Standard time format is specified. The format is:
	`HH:MM:SS`

Table 16. Format parameters (continued)

FORMATS Tab	Field Description
OK	Accept your entries and you return to the previous dialog box
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Table 17. Statement Performance Parameters

STATEMENT PERFORMANCE Tab	Field Description
Isolation level	Six options are available: 1. Unspecified 2. Repeatable read (RR) 3. All (RS) 4. Cursor stability (CS) 5. Change (UR) 6. No commit (NC) When you choose Refresh List in QMF for Windows, this SQL is executed from the packages that were created from the Create Objects procedure in QMF for Windows Administrator. **Note:** See your database documentation for more information on these options.
Query blocking	Four options are available: 1. Unspecified 2. All 3. Unambiguous 4. None **Note:** See your database documentation for more information on these options.
Release	Three options are available: 1. Unspecified 2. Commit 3. Deallocate **Note:** See your database documentation for more information on these options.
Degree of parallelism	See your database documentation for more information on this option
Query optimization	See your database documentation for information on this option

Table 17. Statement Performance Parameters (continued)

STATEMENT PERFORMANCE Tab	Field Description
OK	Accept your entries and you return to the previous dialog box
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Table 18. Statements Parameters

STATEMENTS Tab	Field Description
Explain	Four options are available: 1. Unspecified 2. All 3. Yes 4. No **Note:** See your database documentation for your database for more information on these options.
Explain snapshot	Four options are available: 1. Unspecified 2. All 3. Yes 4. No **Note:** See your database documentation for your database for more information on these options.
Validate time	Four options are available: 1. Unspecified 2. Bind 3. Run **Note:** See your database documentation for your database for more information on these options.
Return SQL warnings	Specifies whether to return SQL warnings. The default is checked (″yes″). **Note:** See your database documentation for your database for more information on this option.
OK	Accept your entries and you return to the previous dialog box
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Table 19. Miscellaneous parameters

MISCELLANEOUS Tab	Field Description
Dynamic rules	Seven options are available: 1. Unspecified 2. Run 3. Bind 4. Invoker/Requester 5. Invoker/Owner 6. Definer/Requester 7. Definer/Owner **Note:** See your database documentation for your database for more information on this option.
Character sub-type	Five options are available: 1. Unspecified 2. FOR BIT DATA 3. FOR SBCS DATA 4. FOR MIXED DATA 5. Default **Note:** See your database documentation for your database for more information on this option.
Generic options	See your database documentation for more information on this option
OK	Accept your entries and you return to the previous dialog box
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Click **Next** to advance to the Object Listing dialog box to specify the types of objects to include in a list.

Check objects

After you set the package properties, QMF for Windows Administrator opens the Check Objects dialog box of the Packages Wizard if you selected the "Create the QMF for Windows catalog tables" option at the Catalog Option dialog box. At the Check Objects dialog box, specify whether you want to check which objects need to be created and determine which objects exist. Create QMF for Windows objects at each database server to which QMF for Windows will connect. Some objects might exist at the server from a prior QMF installation. QMF for Windows Administrator can determine which objects need to be created and it allows you to create them automatically. If you have QMF for Windows on an S/390 (z/OS, VM or VSE) or any QMF for

Windows version prior to 6.1 installed, QMF for Windows creates some of its own tables and views in the existing databases, typically using *RDBI* as the creator.

Version 6.1 and later of QMF for Windows stores user profile, resource limits, and authorization ID information in different tables, and uses different views than previous versions of host QMF.

The tables and views used to access these tables are created when you choose an option to check objects during the Packages wizard. If you look at the SQL that is generated to create these tables and views, you will notice that:

- If any previous host QMF tables are detected, INSERT statements will be created to copy all of the data stored in the old tables to the new tables
- In the CREATE VIEW statements for each of RDBI.PROFILE_VIEW and RDBI.RESOURCE_VIEW views, there are two versions of the FROM clause. One refers to a table owned by RDBI, and the other refers to a table owned by Q (which is commented out by default).

  To continue sharing information between host QMF and QMF for Windows, comment out the line referring to RDBI, and uncomment the line referring to Q. Otherwise, the information in the tables owned by Q will be unaffected, but will also not be used by QMF for Windows.

Note: Before proceeding, verify that you have the necessary privileges to create objects at the target server. See "Object privileges," for more information.

Assume that none of the objects already exist

Follow these steps if you are creating new objects on the server:

1. Select the option, "Assume none of the objects already exist."

   This is the default.
2. Click **Next**.

   The Object Listing Option dialog box displays.

Automatically check which objects already exist

This option checks for existing database objects, including databases, tablespaces, nodes, indexes, and views. Required objects will be created.

Note: This option is typically selected.

IMPORTANT

An error message is issued if a required object does not exist or you do not have the correct level of authority to that object.

Click **Next** to advance to the Check Objects dialog box to specify the types of objects to include in a list.

Assume that all of the objects already exist
This option does not create objects because they exist from a previous installation of QMF for Windows. You advance to the Bind Packages dialog box to specify whether to bind the database packages.

Migrating table information: Version 6.1 and later of QMF for Windows stores user profile, resource limits, and authorization ID information in different tables and views from previous versions and host QMF. If previous tables are detected while creating the new tables, all data from the old tables is migrated to the new tables. Customize the SQL in the Create Objects dialog box to continue sharing user profile and resource limits information with host QMF.

Sharing host information with QMF for Windows: In the CREATE VIEW statements for each of RDBI.PROFILE_VIEW, RDBI.RESOURCE_VIEW, and RDBI.AUTHID_VIEW views, there are two versions of the FROM clause referring to:

- a table owned by RDBI
- a table owned by Q (which is commented out by default)

To continue sharing information between host QMF z/OS and QMF for Windows, comment out the line referring to RDBI, and uncomment the line referring to Q. If you make no changes, the information in the tables owned by Q will be unaffected, but they will not be used by QMF for Windows.

Object privileges: The actual set of privileges required depends on the type and version of the database, the configuration of the database, and whether the package exists.

Note: Refer to your database documentation for more information about the privileges required to bind packages.

The table privileges also vary by whether a QMF for Windows catalog is created at the server. Therefore, privileges differ based on the privileges that are always required and privileges that are required if you create a QMF for Windows catalog. If one of your primary authorization IDs does not have these authorities but one of your secondary authorization IDs does, specify the secondary authorization ID as the Owner ID for the packages.

About secondary authorization IDs: Many companies using DB2 for z/OS use a security scheme featuring RACF and DB2 so that when a user signs on to DB2 with a user ID and password, the user is limited to setting the current

SQLID to a secondary authorization ID maintained by RACF. The secondary ID is the owner of all the tables and queries. Some users share the same secondary ID and can gain access to the same data and objects.

QMF for Windows is not granted access to this set of secondary authorization IDs when the Object List is created. Instead, it has its own table, RDBI.AUTHID_TABLE that must be populated by the DBA so users can see all familiar tables in the list. If the table is not populated, the users will not see these tables, but they will be able to write SQL queries against those tables.

The objects and their corresponding privileges include:

Table 20. Object privileges

Table Name	Authorities
Q.OBJECT_DATA	SELECT, INSERT, UPDATE (except with DB2 UDB for workstation platforms, version 5 or later), DELETE
Q.OBJECT_DIRECTORY	SELECT, INSERT, UPDATE, DELETE
Q.OBJECT_REMARKS	SELECT (DB2 UDB for workstation platforms, version 5 or later), INSERT, UPDATE, DELETE
Q.OBJ_ACTIVITY_DTL	SELECT, INSERT, UPDATE, DELETE
Q.OBJ_ACTIVITY_SUMM	SELECT, INSERT, UPDATE, DELETE
Q.RAA_OA_DTL_X	SELECT, INSERT, UPDATE, DELETE
Q.RAA_OA_SUMM_X	SELECT, INSERT, UPDATE, DELETE
Q.RAA_OBJECT_VIEW	SELECT
Q.RAA_OBJECT_VIEW_X	SELECT
Q.RAA_OBJ_DATA_X	SELECT, INSERT, UPDATE, DELETE
Q.RAA_OBJ_DIR_X	SELECT, INSERT, UPDATE, DELETE
Q.RAA_OBJ_REM_X	SELECT (DB2 UDB for workstation platforms, version 5 or later), INSERT, UPADTE, DELETE
Q.RAA_SUBTYPE	SELECT, INSERT, UPDATE, DELETE
Q.RC_NODE	SELECT, INSERT, UPDATE, DELETE
Q.RC_NODE_AUTH	SELECT, INSERT, UPDATE, DELETE
RDBI.AUTH_VIEW	SELECT
RDBI.CATALOG_DIR_X	SELECT, INSERT, DELETE
RDBI.PROFILE_VIEW	SELECT, INSERT, UPDATE, DELETE
RDBI.PROFILE_VIEW_X	SELECT, INSERT, UPDATE, DELETE
RDBI.RESERVED	SELECT

Table 20. Object privileges (continued)

Table Name	Authorities
RDBI.RESOURCE_VIEW	SELECT, INSERT, UPDATE, DELETE
RDBI.RESOURCE_VIEW_X	SELECT, INSERT, UPDATE, DELETE
RDBI.TABLE_VIEW2	SELECT
RDBI.TABLE_VIEW2_X	SELECT
RDBI.USER_ADMIN_VIEW	SELECT
RDBI.USER_ADMIN_VIEW_X	SELECT
RDBI.USER_AUTHID_VIEW	SELECT
RDBI.USER_AUTHID_VIEW_X	SELECT
SYSCAT.PROCPARMS	SELECT (DB2 for workstation platforms, version 5 or later)
SYSCAT.TABAUTH	SELECT (DB2 UDB for workstation platforms)
SYSCAT.TABLES	SELECT (DB2 UDB for workstation platforms)
SYSIBM.SYSPARMS	SELECT (DB2 UDB for z/OS version 6; DB2 UDB for z/OS; and z/OS version 7 and later)
SYSIBM.SYSPROCEDURES	SELECT (DB2 for MVS version 4; DB2 for z/OS version 5)
SYSIBM.SYSTABAUTH	SELECT (DB2 for MVS; DB2 for z/OS; DB2 UDB for z/OS; DB2 UDB for z/OS; and z/OS)
SYSIBM.SYSTABLES	SELECT (DB2 for MVS; DB2 for z/OS; DB2 UDB for z/OS; DB2 UDB for OS/390 and z/OS; DB2 UDB for workstation platforms)
SYSTEM.SYSCATALOG	SELECT (SQL/DS; DB2 Server for VSE & VM)
SYSTEM.SYSTABAUTH	SELECT(SQL/DS, DB2 Server for VSE & VM)
QSYS2.SYSPARMS	SELECT (DB2 for iSeries)
QSYS2.SYSTABLES	SELECT (DB2 for iSeries)

Note: If one of your primary authorization IDs does not have these authorities, but one of your secondary authorization IDs does, then specify the secondary authorization ID as the Owner ID for the packages.

Table 21. Check objects controls

Control	Description
Back	Return to the previous dialog window
Next	Advance to the subsequent dialog window
Cancel	Undo your entries and return to the previous dialog box

Table 21. Check objects controls (continued)

Control	Description
Help	Obtain online help for this dialog box

Object listing

The Object Listing dialog box controls whether an object is included in the list. QMF for Windows checks which required objects already exist. This dialog box displays when you select one of the following options at the Check Objects dialog box:

- Assume that none of the objects already exist
- Automatically check which objects already exist

QMF for Windows applies levels of filtering when producing lists of objects (queries, forms, procedures or tables). Filtering controls how authorizations on objects are considered and decides whether to include an object in the list. The actual access to an object is controlled by QMF for Windows or the database, depending on the type of object. Users cannot access objects to which they are not authorized. See ″Object privileges″ for more information.

Include all objects

This option includes all objects regardless of user access authority.

Note: This option requires the least database resources when producing the list, but may include objects the user cannot access.

Include objects that the user can access with the primary or current authorization ID

This filter results in a list of objects the user can access, based on the user's primary or current authorization.

After you select an object listing option, the status line at the bottom of the dialog box shows you the object checking operation that indicates which objects exist.

Include objects that the user can access with the primary or secondary authorization ID

This filter results in a list of objects the user can access, based on the user's primary or secondary authorization.

Note: This option requires the most database resources when producing the list.

Table 22. Object listing controls

Control	Description
Back	Return to the previous dialog window

Table 22. Object listing controls (continued)

Control	Description
Next	Advance to the subsequent dialog window
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Create objects

Create the QMF for Windows objects at each database server to which you want to connect to bind the installation packages. Some of these objects might exist at the server from a previous QMF installation. The QMF for Windows Administrator application can determine which objects need to be created and will create them automatically. Before proceeding, verify that you have the necessary DB2 privileges to create the objects at the target server.

SQL statements that create the QMF for Windows database objects currently allow object types of T (Tables) and V (Views) to be displayed in a list. Any iSeries file can be accessed using QMF for Windows. Modify the SQL script to allow all file types, including Physical and Logical files to be listed.

Procedure

Follow these steps to create objects from the Object Listing Option dialog box:

1. Select the option, ″Assume that none of the objects already exist.″
2. Click **Next**.

   The Object Listing Option dialog box displays.
3. Select the appropriate filtering option, which controls which objects are included in the list.

   The Create Objects dialog box displays and it contains the SQL generated for the objects. You can edit this information.

Listing physical and logical files for iSeries

Follow these steps to enable QMF for Windows to list iSeries physical and logical files:

1. On the Create Objects dialog, scroll down to the statement:

   ```
   CREATE VIEW RDBI.TABLE_VIEW2
   ```
2. In the ″CREATE″ statement, modify the ″WHERE″ clause to add ’P’ for Physical files and ’L’ for Logical files. For example:

   ```
   WHERE A.TABLE_TYPE IN ('T','V','P','L')
   ```
3. Click OK to finish the ″create.″

Procedure

Follow these steps to create the QMF installation objects for an existing server:

1. Select the server from the QMF for Windows Administrator main dialog box.
2. Click **Edit**.
3. Click the Packages tab at the Parameters for [server] dialog box.
4. Click **Wizard**.
5. Select the option to create the catalog tables at the Catalog Option dialog box, and click **Next**.
6. Set package properties at the Package Properties dialog box, and click **Next**.
7. Select one of the following options to create the required objects:
   - Assume that none of the objects already exist
   - Automatically check which objects already exist

   If any object needs to be created, the Create Objects dialog box opens, and displays a series of default SQL statements (separated by semicolons). QMF for Windows Administrator uses these SQL statements to create the table objects. Review these statements carefully and verify the syntax and naming conventions. Modify the statements, if necessary.

Two views are created when objects are created. These views determine how QMF for Windows lists are built. The views are:

- RDBI.TABLE_VIEW
- Q.RAA_OBJECT_VIEW

  These views may be altered during QMF for Windows installation to tailor the list building. These views differ based on DB2 platform.

Save the SQL statements by copying the text to notepad and save the file for future reference.

Note: Do not change the name of any object.

Use a semicolon (;) to separate multiple statements.

The packages now are bound and the Bind Packages dialog box opens.

After reviewing the statements, click **Next** to run the SQL at the server.

QMF for Windows lists for the extended catalog are built by the views created during the Create Objects phase of the post-installation process. For queries, forms, and procs, the view Q.RAA_OBJECT_VIEW_X is the key and it interacts with other views and tables.

The sample below is for DB2 z/OS catalog server:

Table 23. Object views: Q.RAA_OBJECT_VIEW_X

Q.RAA_OBJECT_VIEW_X
Looks at the catalog_ids and owner names in Q.RAA_OBJ_DIR_X and displays or fails to display each object based on the conditions below. ″Yes″ to any of the following questions causes the query, procedure or form to be displayed in the list: 1. Does the object owner match the current user ID? 2. Does the object owner match the current SQLID? 3. Does the object owner match the secondary authorization ID of the current user ID having the same catalog ID? 4. Is the object SHARED? (Restricted = N) 5. Is the user a SYSADM having the same catalog _ID?

Table 24. Object views: RDBI.USER_ADMIN_VIEW_X

RDBI.USER_ADMIN_VIEW_X
This view determines whether the user or a secondary authorization ID of the user is a SYSADM. It checks RDBI.ADMIN_VIEW and RDBI.CATALOG_DIR_X for: 1. The current user ID or SQLID. 2. Secondary authorization ID of the current user having the same catalog_ID in RDBI.CATALOG_DIR_X.

Table 25. Object views: RDBIUSER_AUTHID_VIEW_X

RDBI.USER_AUTHID_VIEW_X
This view gets all rows from the base table RDBI.AUTHID_VIEW_X, where the primary ID is that of the current user. It will list all secondary authorization IDs for the current user.

Table 26. Object views: RDBI.ADMIN_VIEW

RDBI.ADMIN_VIEW
This view gets a list of GRANTEEs from the base table SYSIBM.SYSUSERAUTH. It takes GRANTEEs who have been granted G or Y for SYSADMAUTH.

Table 27. Object views: RDBI.CATALOG_DIR_X

RDBI.CATALOG_DIR_X
This table contains all rows of internal identifier of the distinct catalog and name of the distinct catalog.

Table 28. Object views: RDBI_AUTHID_VIEW_X

RDBI_AUTHID_VIEW_X
This table copies all rows from the base table RDBI.AUTHID_TABLE_X.

Table 29. Object views: SYSIBM.SYSUSERAUTH

SYSIBM.SYSUSERAUTH

Table 30. Object views: RDBI_AUTHID_TABLE_X

RDBI_AUTHID_TABLE_X
A base table listing primary and secondary authorization IDs. This table must be populated by the system or database administrator. QMF for Windows does not have access to the preexisting RACF/DB2 relationships during or after installation, unless they are stored in a custom DB2 table of the administrator's design.

For queries, forms, and procedures, the view Q.RAA_OBJECT_VIEW_X is the key and it interacts with the other views and tables, as shown below.

Note: This view for the Object Listing Option, "Include only those objects that any of the user's primary or secondary authorization IDs are authorized to access. This option requires the most database resources when producing lists."

Table 31. Create View Q.RAA_OBJECT_VIEW_X

```
CREATE VIEW Q.RAA_OBJECT_VIEW_X (        CATALOG_ID,        OWNER, NAME,
TYPE, SUBTYPE, OBJECTLEVEL, RESTRICTED, MODEL, REMARKS
DELETED_BY_USER, DELETED_BY_SQLID, DELETED_TIMESTAMP ) AS SELECT
A.CATALOG_ID,        A.OWNER, A.NAME, A.TYPE, A.SUBTYPE,
A.OBJECTLEVEL, A.RESTRICTED, A.MODEL,        B.REMARKS,
A.DELETED_BY_USER, A.DELETED_BY_SQLID, A.DELETED_TIMESTAMP FROM
Q.RAA_OBJ_DIR_X A, Q.RAA_OBJ_REM_X B WHERE (A.CATALOG_ID = B.CATALOG_ID AND
       A.OWNER = B.OWNER AND        A.NAME = B.NAME AND
A.DELETED_TIMESTAMP IS NULL AND B.DELETED_TIMESTAMP IS NULL)   AND
(A.RESTRICTED = 'N'        OR A.OWNER IN (USER, CURRENT SQLID)        OR
A.OWNER IN (SELECT C.SECONDARY_ID                        FROM
RDBI_USER_AUTHID_VIEW_X C                       WHERE C.CATALOG_ID =
A.CATALOG_ID)        OR EXISTS (SELECT D.AUTHID                    FROM
RDBI.USER_ADMIN_VIEW_X D                   WHERE D.CATALOG_ID = A.CATALOG_ID)
)
```

Table 32. Create view RDBI.USER_ADMIN_VIEW_X

```
CREATE VIEW RDBI.USER_ADMIN_VIEW_X (     CATALOG_ID,     "AUTHID" ) AS SELECT
B.ID, A."AUTHID" FROM RDBI.ADMIN_VIEW A., RDBI.CATALOG_DIR_X B WHERE
A."AUTHID" IN (USER, CURRENT SQLID)    OR A."AUTHID" IN (SELECT
C.SECONDARY_ID                       FROM RDBI.USER_AUTHID_VIEW_X C
                     WHERE B.ID = C.CATALOG_ID)
```

Table 33. Create view RDBI.USER_AUTHID_VIEW_X

```
CREATE VIEW RDBI.USER_AUTHID_VIEW_X (     CATALOG_ID,     PRIMARY_ID,
SECONDARY_ID ) AS SELECT A.CATALOG_ID, A.PRIMARY_ID, A.SECONDARY_ID FROM
RDBI.AUTHID_VIEW_X A WHERE A.PRIMARY_ID = USER
```

Table 34. Create view RDBI.ADMIN_VIEW

```
CREATE VIEW RDBI.ADMIN_VIEW ( "AUTHID" ) AS SELECT A.GRANTEE FROM
SYSIBM.SYSUSERAUTH A WHERE A.SYSADMAUTH IN ('Y', 'G')
```

Table 35. Create table RDBI.CATALOG_DIR_X

```
CREATE TABLE RDBI.CATALOG_DIR_X (     ID SMALLINT NOT NULL     NAME VARCHAR
(128) NOT NULL )     IN RDBIDBX.RDBITSX1     CCSID EBCDIC
```

Table 36. Create view RDBI.AUTHID_VIEW_X

```
CREATE VIEW RDBI.AUTHID_VIEW_X (     CATALOG_ID,     PRIMARY_ID,
SECONDARY_ID ) AS SELECT A.CATALOG_ID, A.PRIMARY_ID, A.SECONDARY_ID FROM
RDBI.AUTHID_TABLE_X A
```

Table 37. Create SYSIBM.SYSUSERAUTH

SYSIBM.SYSUSERAUTH

Table 38. Create table RDBI.AUTHID_TABLE_X

```
CREATE TABLE RDBI.AUTHID_TABLE_X (    CATALOG_ID SMALL INIT NOT NULL,
PRIMARY_ID CHAR (8) NOT NULL,    SECONDARY_ID CHAR (8) NOT NULL )    IN
RDBIDBX.RDBITSX    CCSID EBCDIC
```

Note: This table must be populated by the system or database administrator. QMF for Windows does not have access to the pre-existing RACF/DB2 relationships during or after installation.

For other tables, the view RDBI.TABLE_VIEW2_X is the key and interacts with the other views and tables, as shown below.

Table 39. Object views for tables

RDBI.TABLE_VIEW2_X constructs a list of tables. First, it looks for tables listed in the base table SYSIBM.SYSTABAUTH. All tables in SYSIBM.SYSTABAUTH have, by definition, already had some type of authority granted to at least one user. SYSIBM.SYSTABAUTH must answer ″Yes″ to one of the following questions: 1. Is the GRANTEE = the current user? 2. Is the GRANTEE = the current SQLID? 3. Is the GRANTEE = PUBLIC? 4. Is the GRANTEE = PUBLIC*? 5. Is the GRANTEE = a secondary authorization ID of the current USER having the same catalog_ID? 6. Is the user a SYSADM having the same catalog_ID? 7. Is the user a SYSADM having the same catalog_ID? The table must have one of the following GRANTEE types: • ’ ’ • U • G The GRANTEE of the table must have Y or G authority on one of the following four categories: • DELETEAUTH • INSERTAUTH • SELECTAUTH • UPDATEAUTH

Table 40. Authority to RDBI.USER_ADMIN_VIEW_X

RDBI.USER_ADMIN_VIEW_X
This view determines whether the user or a secondary user ID of the user is a SYSADM. It checks the RDBI.ADMIN_VIEW and RDBI.CATALOG_DIR_X for: • The current user ID • Secondary user ID of the current user having the same catalog ID.

Table 41. Authority to RDBI.USER_AUTHID_VIEW_X

RDBI.USER_AUTHID_VIEW_X
This view gets all rows from the base table RDBI.AUTHID_VIEW_X where the primary ID is that of the current user. It will list all secondary authorization IDs for the current user.

Table 42. Authority to RDBI.ADMIN_VIEW

RDBI.ADMIN_VIEW
This view gets a list of GRANTEEs from the base table SYSIBM.SYSUSERAUTH. It takes GRANTEEs who have been granted G or Y for SYSADMAUTH.

Table 43. Authority to RDBI.AUTHID_VIEW_X

RDBI.AUTHID_VIEW_X
This view copies all rows from the base tableRDBI.AUTHID_TABLE_X.

Table 44. Tables and views: SYSIBM.SYSUSERAUTH

SYSIBM.SYSUSERAUTH

Table 45. Authority to RDBI.AUTHID_VIEW_X

RDBI.AUTHID_TABLE_X
A base table listing catalog_ID, and primary and secondary authorization IDs. **Note:** This table must be populated by the system or database administrator. QMF for Windows does not have access to the pre-existing RACF/DB2 relationships during or after installation, unless they are stored in a custom DB2 table of the administrator's design. Also, for tables, the view RDBI.TABLE_VIEW2_X is the key and interacts with the other views and tables, as shown below.

Table 46. Tables and views: RDBI.TABLE_VIEW2_X

RDBI.TABLE_VIEW2_X
(CREATE VIEW RDBI.TABLE_VIEW2_X) CATALOG_ID, OWNER, NAME, TYPE, SUBTYPE, OBJECTLEVEL, RESTRICTED, MODEL, REMARKS, DATABASE, TABLESPACE) AS SELECT DISTINCT C.ID, A.CREATOR, A.NAME, 'TABLE', A.TYPE, 0, 'Y', ' ', A.REMARKS , A.DBNAME, A.TSNAME FROM SYSIBM.SYSTABLES A, SYSIBM.SYSTABAUTH B, RDBI.CATALOG_DIR_X C WHERE (A.CREATOR = B.TCREATOR AND A.NAME = B.TTNAME) AND (B.GRANTEE IN (USER, CURRENT SQLID, 'PUBLIC', 'PUBLIC*') OR B.GRANTEE IN (SELECT D.SECONDARY_ID FROM RDBI.USER_AUTHID_VIEW_X D WHERE D.CATALOG_ID = C.ID) OR EXISTS (SELECT E.AUTHID FROM RDBI.USER_ADMIN_VIEW_X E WHERE E.CATALOG_ID = C.ID)) AND (B.GRANTEETYPE IN (' ', 'U', 'G')) AND (B.DELETEAUTH IN ('Y', 'G') OR B.INSTERTAUTH IN ('Y', 'G') OR B.SELECTAUTH IN ('Y', 'G') OR B.UPDATEAUTH IN ('Y', 'G'))

Table 47. Tables and views: SYSIBM.SYSTABAUTH

SYSIBM.SYSTABAUTH

Table 48. Tables and views: RDBI.USER_ADMIN_VIEW_X

RDBI.USER_ADMIN_VIEW_X
CREATE VIEW RDBI.USER_ADMIN_VIEW_X (CATALOG_ID, "AUTHID") AS SELECT B.ID, A."AUTHID" FROM RDBI.ADMIN_VIEW A, RDBI.CATALOG_DIR_X B WHERE A."AUTHID" IN (USER, CURRENT SQLID) OR A."AUTHID" IN (SELECT C.SECONDARY_ID FROM RDBI.USER_AUTHID_VIEW_X C WHERE B.ID = C.CATALOG_ID)

Table 49. Tables and views: RDBI.USER_AUTHID_VIEW_X

RDBI.USER_AUTHID_VIEW_X
CREATE VIEW RDBI.USER_AUTHID_VIEW_X (CATALOG_ID, PRIMARY_ID, SECONDARY_ID) AS SELECT A.CATALOG_ID, A.PRIMARY_ID, A.SECONDARY_ID FROM RDBI.AUTHID_VIEW_X A WHERE A.PRIMARY_ID = USER

Table 50. Tables and views: RDBI.ADMIN_VIEW

RDBI.ADMIN_VIEW
CREATE VIEW RDBI.ADMIN_VIEW ("AUTHID") AS SELECT A.GRANTEE FROM SYSIBM.SYSUSERAUTH A WHERE A.SYSADMAUTH IN('Y', 'G')

Table 51. Tables and views: RDBI.AUTHID_VIEW_X

RDBI.AUTHID_VIEW_X
CREATE VIEW RDBI.AUTHID_VIEW_X (CATALOG_ID, PRIMARY_ID, SECONDARY_ID) AS SELECT A.CATALOG_ID, A.PRIMARY_ID, A.SECONDARY_ID FROM RDBI.AUTHID_TABLE_X A

Table 52. Tables and views: SYSIBM.SYSUSERAUTH

SYSIBM.SYSUSERAUTH

Table 53. Tables and views: RDBI.AUTHID_TABLE_X

RDBI.AUTHID_TABLE_X
CREATE TABLE RDBI.AUTHID_TABLE_X (CATALOG_ID SMALLINT NOT NULL, PRIMARY_ID CHAR (8) NOT NULL, SECONDARY_ID CHAR (8) NOT NULL) IN RDBIDBX.RDBITSX2 CCSID EBCDIC **Note:** This table must be populated by the system or the database administrator. QMF for Windows does not have access to the pre-existing RACF/DB2 relationships during or after installation.

Table 54. Create objects controls

Control	Description
Back	Return to the previous dialog window
Next	Advance to the subsequent dialog window
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Bind packages

Bind the QMF for Windows packages at that database server to run distributed SQL at any database server. The QMF for Windows packages refer to the set of objects that the product uses. These packages may exist at the server, and can be replaced.

Use QMF for Windows Administrator to choose the collection name and bind options for the packages and to bind the packages at the server.

Twelve (12) packages will be bound; six (6) packages will be bound for a CLI connection:

Table 55. Binding packages

Package	CLI	Package	CLI
RAARDBI1: used only in the server configuration phase to create the database objects required by QMF for Windows in each connected database server	RAARDBII:used only in the server configuration phase to create the database objects required by QMF for Windows in each connected database server	RAARDBX2 (for a DRDA connection only); used for isolation levels RR, RS, CS, UR, and NC	N/A

Table 55. Binding packages (continued)

Package	CLI	Package	CLI
RAARDBIA: used by the QMF for Windows Administrator application, and contains the SQL required for administrative functions This package is used only in the server configuration phase to create the database objects required by QMF for Windows in each connected database server.		RAARDBX3 (for a DRDA connection only); used for isolation levels RR, RS, CS, UR, and NC	N/A
The remaining packages are used by the QMF for Windows application:			
RAASHUT1	RAASHTC	RAARDBX4 (for a DRDA connection only); used for isolation levels RR, RS, CS, UR, and NC	N/A
RAARDBI1	RAARDBC1	RAARDBX5 (for a DRDA connection only); used for isolation levels RR, RS, CS, UR, and NC	N/A
RAARDBI2	RAARDBC2	RAARDBIL: used for retrieving large objects	RAARDBCL: used for retrieving large objects
RAARDBX1: (for a DRDA connection only); used for isolation levels RR, RS, CS, UR, and NC	N/A	RAARC1: used by Report Center	RAARC1C: used by Report Center

Binding packages opens a connection to the database server. The status line indicate the progress of the bind operation as packages are being bound. Correct any errors, then repeat the bind. The packages will be bound and stored with the collection name.

Use this dialog box to specify whether to bind packages at the database server. QMF for Windows requires several packages to be bound, which can be performed automatically from this dialog box.

Note: If you have bound packages for this version of QMF for Windows, select default, "Do not bind the packages."

Do not bind the packages

Select this option to continue through the packages wizard without binding the packages at the database server. Use this option if you have bound your packages.

Bind the packages

Select this option to bind the packages at the database server. You need authority for the specified collection ID at the database server to bind these packages. Refer to your database documentation for information about what authority is needed to bind packages.

Note: If you are binding the packages, check the "Replace existing packages" and "Keep existing authorizations on packages" options.

Changing the default bind options can have an effect on the function of QMF for Windows. Only expert users should change bind options.

The following packages will be bound and used by QMF for Windows:

Table 56. Packages bound

DRDA Name	CLI Name	End-User?	Pre-requisites?	Notes	Cat-reqs
The name of the package when the connection to the server is via DRDA (over TCP/IP or SNA)	*The name of the package when the connection to the server is via the DB2 Call Level Interface (CLI)*	*Specifies whether the end-user requires EXECUTE authority on package (which is handled automatically by the GRANT process)*	*(Prerequisites): Conditions governing whether the package is actually bound and used. "None" indicates that the package is bound and used in all cases.*	*Additional information about the purpose and use of the package.*	*(Catalog required): Specifies the prerequisite conditions governing whether QMF for Windows catalog tables are required.* *"No" indicates that QMF for Windows catalog tables are not required.*

Table 56. Packages bound (continued)

DRDA Name	CLI Name	End-User?	Pre-requisites?	Notes	Cat-reqs
RAARDBII This package is used only in the server configuration phase to create the database objects required by QMF for Windows in each connected database server	n/a	No	None	Dynamic SQL is used during installation to check for the database object.	No
RAARDBX1	n/a	Yes	None	Dynamic user SQL runs with RR isolation level.	No
RAARDBX2	n/a	Yes	None	Dynamic user SQL runs with RS isolation level.	No
RAARDBX3	n/a	Yes	None	Dynamic user SQL runs with CS isolation level.	No
RAARDBX4	n/a	Yes	None	Dynamic user SQL run with UR isolation level.	No

Table 56. Packages bound (continued)

DRDA Name	CLI Name	End-User?	Pre-requisites?	Notes	Cat-reqs
RAARDBX5	n/a	Yes	None	Dynamic user SQL runs with NC isolation level.	No
RAARDBI1	RAARDBC1	Yes	None	Internal SQL.	No
RAARDBI2	RAARDBC2	Yes	QMF for Windows catalog tables	Internal SQL.	Yes
RAARDBIA	RAARDBCA	No	None	Internal SQL. for administrative functions only.	Yes
RAARDBIL	RAARDBCL	Yes	LOB support in database	Internal SQL for retrieving LOB values.	No
RAASHUT1	RAASHTC1	Yes	None	Internal SQL.	No
RAARC1	RAARC1C	Yes	None	Internal SQL for Report Center	No

Errors during bind

The status line indicates the progress of the bind operation if you select the option to bind the packages. If errors occur while binding packages, correct them, then repeat the bind. A few errors are described below.

SQL -204 error

An SQL code of -204 indicates that a required table does not exist. Click **Create Objects...** to create it.

SQL -551 or -552 error

An SQL code of -551 or -552 indicates that the authorization ID on the bind (your user ID or owner ID) is missing at least one authority required to bind the packages (create database objects at the selected database server). See ″Required table authorities″ for more information.

Table 57. Bind packages controls

Control	Description
Back	Return to the previous dialog window
Next	Advance to the subsequent dialog window
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Permissions

The Permissions dialog box displays after you bind the packages. After you bind the QMF for Windows packages, you must grant permission to users to execute the packages and work with the product.

The permissions procedure grants the authority required to execute the packages contained within a collection. QMF for Windows creates the collection containing these packages on the iSeries with authority PUBLIC * EXCLUDE. Grant the appropriate authority for users of that collection and library on the iSeries.

This grant is done by the QMF for Windows Administrator application. Specify the user IDs and the QMF for Windows Administrator can construct and execute the grant SQL at the server.

Note: Granting permissions is the final step in the Packages wizard.

Procedure

1. Enter the user IDs to which you want to grant authority to the packages.
2. Click **Grant** to give permission at the server.

   The following message displays after permission is granted:

   ```
   Permission to execute the user packages has been granted to the specified user IDs.
   ```

   Note: To revoke permissions, enter the user ID and click **Revoke**.
3. Click **Finish**.

   The Check Table Authorizations dialog box opens if you selected the option″Include only those objects that any of the user's primary or secondary authorization IDs are authorized to access. This option requires the most database resources when producing lists,″ at the Object Listing Option dialog box.

Table 58. Permissions controls

Control	Description
Back	Return to the previous dialog window
Next	Advance to the subsequent dialog window
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Check table authorizations

Check for tables with no permissions after you bind the packages and assign permissions. Tables that are missing permissions are not visible to QMF for Windows and do not appear in table lists. There must be at least one authorization on each table, recorded in the authorizations catalog table. Some tables can exist without authorizations and will not cause errors, but they not appear in table lists. QMF for Windows can detect and correct the status by granting privilege on the table to the specified user ID.

Procedure

1. Specify whether to check for tables with no authorizations and click **Next**.

   The Grant Table Authorizations dialog box opens if unassigned tables are found.
2. Grant table authorizations, if necessary. The user receives SELECT privilege by default.
3. Click **Finish**.

   The Table Authorizations are granted and the Packages Wizard closes.

Once tables with missing authorizations are located, you can assign permissions to these tables, making them visible in lists.

Table 59. Check table authorizations controls

Control	Description
Back	Return to the previous dialog window
Next	Advance to the subsequent dialog window
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Catalog server

The Catalog server is the final dialog box used when creating a server. Use this dialog box to select the server where the QMF for Windows catalog for the current server will be stored. The Catalog dialog box contains a drop-down list of the servers defined on the SDF.

Note: Only DB2 servers can host the QMF for Windows catalog.

Catalog server
The catalog server is the server where the QMF for Windows catalog for the current server will be stored. This must be a DB2 server.

Catalog name
Select which catalog to use at the catalog server selected above. The catalog named <<Default>> is the catalog shared with host QMF and previous versions of QMF for Windows.

You can add or delete a catalog name at this dialog box.

Table 60. Catalog server controls

Control	Description
Back	Return to the previous dialog window
Finish	Finish setting up the server and return to the Servers dialog box where the new server is listed
Cancel	Undo your entries and return to the previous dialog box
Help	View online help for this dialog box

Chapter 5. Resource Limits

What are resource limits?

QMF for Windows incorporates a resource governor that limits and controls database and communications resources. The resource governor restricts what actions a user can perform in QMF for Windows and places limits on the resources a user can consume. This governing feature allows you to provide the users with distributed access to DB2, while avoiding database or network degradation.

The governing function of QMF for Windows is always active. Governing based on default limits is performed even if you do not set up resource limits.

Note: It is recommended that you define your own sets of limits before your users access QMF for Windows.

Use the QMF for Windows Administrator module to define sets of limits and restrictions, which are called **resource limits groups**. Resource limits groups manage, control, and restrict resource usage. You can assign users to resource limits groups according to the governing level appropriate for these users. The QMF for Windows governor is optional and complementary to the rules for DB2. There are three basic steps to establishing explicit governing or resource limits:

1. Create a resource limits group.
2. Create schedules in the resource limits group.
3. Assign users to the resource limits group.

How resource limits work

The QMF for Windows resource limits are implemented before the query leaves the workstation. The query never reaches DB2 for evaluation if a resource limit is violated. Other QMF for Windows limits are invoked after DB2 has started to run the query. QMF for Windows does not replace any rules that are in place in DB2.

A resource limits group is a collection of limits and controls on the resources that are governed by QMF. You can control resource consumption by:

- user
- day of the week
- time of day

For example, a resource limits group can contain one set of limits that is in effect weekdays between 8:00 AM and 6:00 PM and another that is in effect on weekends and off-hours.

Resource limits groups are securely stored in a database table at the database server. This prevent users from bypassing the limits you establish. Resource limits groups are stored in the table RDBI.RESOURCE_TABLE. A view named RDBI.RESOURCE_VIEW is defined on this table to allow QMF for Windows users access the resource limits information.

Note: You must have the authorization to execute the QMF for Windows Administrator package to maintain resource limits groups. This prevents unauthorized users from changing the limits that you establish.

Users who are not assigned to a resource limits group are governed by the limits defined in the *default resource limits group*. You are responsible for creating and maintaining the default resource limits group, which is named **<Default>**.

Note: You can update the <Default> resource limits group to prevent access to users who are not registered in a resource limits group. Uncheck the following boxes in the Edit Resource Limits Group Schedule dialog box:

- SQL Verbs
- Options
- Save Data
- Binding
- Object Tracking

Creating a resource limits group

You must create a resource limits group and assign QMF for Windows users to that group to set up QMF for Windows governing. After a resource limits group is defined and a QMF for Windows user is assigned to the group, that user's resource usage is limited as defined by that group.

Note: A default resource limits group, which is named <Default>, is delivered with QMF for Windows Administrator.

Procedures

1. Select the server at the QMF for Windows Administrator main dialog box.
2. Click **Edit**.

   The Parameters dialog box opens.
3. Click the **Resource Limits** tab.

The Resource Limits dialog box opens and lists the resource groups defined at the server.

4. Select the resource limits group from which you want to model the new resource limits group, and click **New**.

   The New Resource Limits Group dialog box opens.

5. Type a name for the group in the Group name field. There are no restrictions on the name.
6. Click **OK**.
7. [Optional] Type up to 80 characters of comments that describe the resource limits group.
8. If the **Create this group using schedules from...** check box is checked, the group you selected as a model has schedules that you can copy into the new group.

   Check this box to create the new group with copies of the schedules in the model group. Otherwise, the new resource limits group contains no schedules.

   Note: After the resource limits group is determined, QMF for Windows determines which schedule group is in effect at the database server and checks for the server time zone.

   If the database server is accessed by users in different time zones than the server, their resource limit schedules determine their access rights in function of the server time.

9. Click **OK** to create the resource limits group.

   A confirmation pop-up window displays.

10. Click **OK**.

    The Edit Resource Limits Group dialog box displays.

Assigning users to resource limits groups

The relationship between a QMF for Windows user and a resource limits group is stored in a table (RDBI.PROFILE_TABLE) at that database server, accessed via RDBI.PROFILE_VIEW. QMF for Windows Administrator maintains user and resource limits group relationships in this table.

When QMF for Windows connects to a database server, the user must provide user information (user ID and password), which is validated by the database server. If the user information is valid, QMF for Windows locates the correct profile to determine which resource limits group to use. This is done by searching the CREATOR, ENVIRONMENT, and TRANSLATION columns in the RDBI.PROFILE_VIEW table.

Procedures

1. Click **Assign** on the Resource Limits dialog box (at the Parameters for [server] property sheet).

   The Assign User Profiles dialog box opens.

   Note: You cannot assign users to the default resource limits group.
2. Type the user ID you want to assign in the *Show user profiles with creator matching* field or a matching pattern if you want to work with multiple user IDs.
3. Click **Refresh List**.

   QMF for Windows Administrator retrieves all the user profiles stored in the RDBI.PROFILE_VIEW table that match the value you entered.

   The returned list of user profiles displays in the Not Assigned and Assigned columns.

   Note: If the user ID you want to assign to a resource limits group does not have an entry in the RDBI.PROFILE_VIEW table, click **Create New** to create the new user profile.
4. Select the user IDs to assign to a resource limits group and click **Assign** or **Unassign** to move the user profiles to either list.
5. Click **OK**.

Show profiles with ″creator″ matching

Specifies the creator, or owner, of the user profiles you want to list. You can enter matching patterns to list multiple user profiles.

When entering a query, form, and table names, you can use the percent (%) and underscore (_) characters to match patterns rather than entering a specific character.

- Use the percent character to match a string of any length, containing any characters. For example, to list all profiles with a creator beginning with an A, enter A%.
- Use the underscore to match a single character. For example, to list all profiles with a creator that has an A in the second position, enter _A%.
- If the pattern you enter contains special characters, enclose the entire pattern in quotation marks. For example, to include a space as part of a pattern, enter ″A B%″. A special character is any character not included in the list of regular characters: A-Z; 0-9; $, #, @ and _.

Refresh list

Click **Refresh List**, to build a list of user profiles that match the creator you specified in ″Show user profiles with 'creator' matching″ field. The user profiles are displayed in the User Profiles Not Assigned and User Profiles Assigned lists. QMF for Windows Administrator retrieves all user profiles

stored in RDBI.PROFILE_VIEW that match the value you entered. Make sure a % is in the ″Show user profiles with 'creator' matching″ field to list all user profiles. You can enter matching patterns to list multiple user profiles.

Note: You can access use profiles by choosing ″Manage user profiles.″ You can create a user profile, assign it to a resource group, and set the default tablespace where that user can save data. The tablespace is saved in RDBI.PROFILE_VIEW and will be the default if an * is entered for default tablespace in the Resource Group (under the Save Data tab).

You can also edit existing profiles.

User profiles not assigned

Lists the user profiles not assigned to the resource limits group that match the creator you entered in the ″Show user profiles with 'creator' matching″ field.

User profiles assigned

Lists the user profiles assigned to the resource limits group that match the creator you entered in the ″Show user profiles with 'creator' matching″ field. You cannot assign user profiles to the default resource limits group.

Create new

Displays the User Profile dialog box to create a new user profile. Once a profile is created, the new profile displays in the User Profiles Assigned list.

1. Click **Create New** to create a user profile. This is for a user who does not have an entry in the RDBI.PROFILE_VIEW table.
2. In the **Creator** field, specify the creator (or user) for the newly created profile. This is usually the same as the user ID.
3. **Resource group** displays the current resource limits group. QMF for Windows initially assigns new profiles to this group. You can open the Resource Limits Groups dialog box by clicking the box to the right of the input field (...).
4. **Space** displays a table space that holds tables created using SAVE DATA and IMPORT commands.

   Note: The value in this field is used when the user is assigned to a User Group with an asterisk (*) specified for the default table space. This value is set in the Resource Limits Group Schedule (Save Data tab) dialog box.

   If an asterisk (*) is displayed in this ″Default Table Space″ but there is no table space specified for this profile, the ″Default Table Space″ is blank.
5. Click **OK** to create the user profile.

6. Click **OK** at the confirmation pop-up window once the profile has been created.

   On the create Group and Resource limits editing, rows were inserted or updated in RDBI.RESOURCE_TABLE when you assigned a user to a resource limits group. Also, when you assigned users to the group, rows were inserted in RDBI.PROFILE_TABLE.

   You return to the Assign User Profiles dialog box.
7. Click **Assign...** to assign the user profile to the resource group. The Assign User Profiles dialog box opens.
8. Click **Refresh List**. QMF for Windows Administrator retrieves the user profiles stored in the Q.PROFILES table that match the value you entered and displays them in the Not Assigned or Assigned lists.

   Note: If the user ID you want to assign does not have an entry in the Q.PROFILES table, click **Create New...** to create the user profile.
9. Select a user profile from the Not Assigned list. Click Assign to assign in the "Show user profiles with 'creator' matching" field or a matching pattern to work with multiple user IDs.
10. Select the user IDs and use **Assign** and **Unassign** to move the user IDs to either list.
11. Click **OK**.

When the user signs on to QMF for Windows and provides a user ID, password, and (optional) account string to DB2, DB2 validates the information with the operating system then tells QMF for Windows to accept or reject the user. Then QMF for Windows looks up the User ID in RDBI.PROFILE_TABLE, finds the resource group, then goes to RDBI.RESOURCE_TABLE to retrieve the resource limits.

If the User ID is not found, QMF for Windows looks for the user, "SYSTEM." The user is granted the Default group limits if SYSTEM is there. The default group limits are not in RDBI.RESOURCE_TABLE. They are in the server definition file (SDF) for bootstrapping reasons.

Editing the resource limits group

Use the Edit Resource Limits Group dialog box to create a schedule for the resource limits group. This dialog box displays after the resource limits group is created.

Note: If you select a schedule in the Schedule List before you click this button, the selected schedule is used as a model for the new schedule.

Procedures

1. [Optional] Enter comments about the schedules for this resource group.

2. Specify whether the schedule is active or inactive. You can change this setting at any time.

   The default is ″Active.″
3. Click **New** to set up a schedule for this resource group or double-click (or click **Edit**) to update an existing schedule.

Resource limits group schedule options

After the resource limits group is determined, QMF for Windows determines which schedule is in effect for the group. The schedule is in effect at the designated database server and checks for the server time zone.

Note: If the database server is accessed by users located in different time zones than the server, the resource limit schedule to which they are assigned will determine their access rights in function of the server time.

Ten tabs are available at the Resource Limits Group Schedule window where you can enter resource limits information. The following schedules may be set up for a resource limits group:

- Main
- Timeouts
- Limits
- SQL verbs
- Options
- Save data
- Binding
- Object tracking
- LOB options
- Report Center

Once a schedule of limits is set, additional schedules can be created to vary the user's limits in time.

Main

The Main tab displays when you click **New** or **Edit** on the Edit Resource Limits Group dialog box. Use this page to set a schedule for the resource limits group. Use the Main dialog box to specify the times and days this schedule will be in effect.

A schedule is uniquely identified by schedule number. In addition to specifying a unique number, you must also specify an effective day of the week and time of day range. The From and To Time and the From and To Day values for the schedule define when the limits and controls are in effect.

All ranges are inclusive.

Table 61. Resource limits fields

Field	Description
Number	The priority assigned to this schedule. For example, if two schedules in the resource limits group span or overlap the same period, the schedule with the lower number is used. Specify a number greater than zero. No schedule can share the same schedule number; otherwise, the following message displays: `A schedule with this number is already defined.`
Day Range	The schedule is active between the From Day and the To Day, inclusively. You can specify a range that wraps around the end of the week. For example, if you select Friday as the ″From Day″ and Monday as the ″To Day,″ the schedule is active on Friday, Saturday, Sunday, and Monday, subject to time of day scheduling.
Time Range	The schedule is active between the From Time and the To Time, inclusively. You can specify a range that wraps around midnight. For example, if you select 20:00 as the ″From Time″ and 08:00 as the ″To Time,″ the schedule is active from 8:00 PM to 12:00 midnight and from 12:00 midnight to 8:00 AM, subject to day of week scheduling.
Status	Click **Active** to enable the schedule, subject to day of week and time of day scheduling. The status of the schedule is also subject to the *Active* or *Inactive* status of the resource group. Click **Inactive** to disable the schedule, regardless of day-of-week or time-of-day scheduling. **Note:** If more than one schedule is in effect at one time, the QMF for Windows governor will use the schedule with the lowest schedule number.

Procedures

1. Select the resource limits group for which you want to create schedules in the Resource Limits group list box.
2. Click **Edit**.

   The Edit Resource Limits Group dialog box opens.
3. Select a schedule in the Schedule List if you want to use it as a model for the new schedule.

4. Click **New**.

   The New Resource Limits Group Schedule dialog box [Main tab] opens. If you selected a schedule in the Schedule List, the selected schedule is used as a model for the new schedule.

5. Enter schedule information on each tab and click **OK**. Each field is described below.

Table 62. Resource Limits Controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. If there are no errors, the schedule is saved at the database server.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.
Help	Displays the QMF for Windows online help for this dialog box.

Timeouts

The Timeouts tab displays when you click **New** or **Edit** on the Edit Resource Limits Group dialog box. Use this page to set timeout thresholds for resource consumption for the resource limits group.

Note: Entries with a value of zero mean that no limit will be defined.

Table 63. Resource limits timeout fields

Field	Description
Idle Query Timeouts	This value limits the amount of time a query can remain idle. You can set two different timeouts: • *Warning Limit:* When this timeout expires, QMF for Windows reminds the user that the query is idle and prompts the user to cancel the query. • *Cancel Limit:* When this timeout expires, QMF for Windows cancels the query automatically. **Note:** A query might be idle when the first buffer of data has been returned to the user and QMF for Windows is waiting for that user to go to the bottom of the data before it fetches the next set of data.

Table 63. Resource limits timeout fields (continued)

Field	Description
Server Response Timeouts	This parameter limits the amount of time QMF for Wi ndows waits for a response from the database server before canceling a request. QMF for Windows waits asynchronously for a response time each time it sends a request to the database server. For example, when you run a query, QMF for Windows sends the request to the database server and waits asynchronously for the query results to return from the database server. **Note:** A lower timeout limit prevents long-running (runaway) queries. A higher timeout limit allows database requests to complete when the database server is slow due to resource contention or other reasons. • *Warning Limit:* When this timeout expires, QMF for Windows prompts the user to cancel the request. • *Cancel Limit:* When this timeout expires, QMF for Windows cancels the request automatically.
Idle Connection Timeout Cancel Limit	Limits the amount of time QMF for Windows retains an idle connection to the database server. This limit balances the tradeoff between the overhead of a connection and the idle connection resource consumption. **Note:** A lower timeout limit helps minimize the resources consumed at servers by idle connections. A higher timeout helps minimize the overhead of establishing connections. When this timeout expires, QMF for Windows closes the idle connection to the database server automatically.

Table 64. Resource limits timeout controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. If there are no errors, the schedule is saved at the database server.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.
Help	Displays the QMF for Windows online help for this dialog box.

Limits

The Limits tab displays when you click **New** or **Edit** on the Edit Resource Limits Group dialog box. Use this page to set various connection and retrieval limits for the resource limits group.

Table 65. Resource limits - limits fields

Field	Description
Maximum Rows to Fetch	Limits the number of rows of data the QMF for Windows retrieves from a database server when running a query. • *Warning Limit:* When this limit is reached, QMF for Windows prompts the user to fetch more data • *Cancel Limit:* When this limit is reached, QMF for Windows cancels the query automatically
Maximum Bytes to Fetch	Limits the number of bytes of data that QMF for Windows retrieves from a database server when running a query. See *Row Limits*, *Byte Limits*, and *Query Buffers* for more information. • *Warning Limit:* When this limit is reached, QMF for Windows prompts the user to continue fetching data • *Cancel Limit:* When this limit is reached, QMF for Windows cancels the query automatically.
Maximum Connections Cancel Limit	Limits the number of simultaneous connections that QMF for Windows establishes to the database server. Generally, connections are reused, so that if you run one query at a server and then run another query at the same server, only one connection is required. However, two connections are required if you run two queries simultaneously. If QMF for Windows requires another connection to a server and this limit is reached, an error is returned and the operation is not performed.

Row limits: When QMF for Windows retrieves data from a server, the data is not retrieved column by column or row by row. Instead, it is retrieved in blocks of data, where each block may contain multiple rows (or even portions of rows). This means that QMF for Windows typically fetches more than one row at a time. That is, buffers of data containing multiple rows are sent from the database server so that the number of network transmissions is minimized. Because of this buffering, you will typically see more rows of data than the row limit specifies. For example, if the row limit is 500 and the database server sends 1,000 rows in the first buffer, then all 1,000 rows are available for display. This "overshooting" of rows occurs because QMF for Windows never discards received data in order to meet the exact row limit. However, QMF for Windows will not request more data once the buffer that contains the 500th row is processed.

Byte limits: Consider the difference in network and database performance when you fetch 10,000 50-byte rows and when you fetch 10,000 5,000-byte rows. In the first case, the database server sends 500,000 bytes of data over the network. In the second case, the database server sends 50,000,000 bytes of data over the network. There is a significant performance difference between these two cases. In order to tightly govern database and network activity, QMF for Windows can limit the number of bytes that are retrieved from the database server. Because of the buffering discussed above, the same kind of ″overshooting″ that occurs with rows also occurs with bytes.

Retrieving large amounts of data: When retrieving large amounts of data, network communication typically limits performance. QMF for Windows tries to minimize the delay between requests to the database server to maximize performance. When QMF for Windows receives a buffer of data from the database server, it does not spend time processing that data before requesting more. Instead, it requests more data immediately, and then uses the time waiting for the new data to process the data received. When QMF for Windows requests more data, it checks how many rows have been retrieved, and compares this against the row limit in effect. If the row limit has been exceeded, it does not issue more requests. When it performs this check, it has not processed the data that has been received to see how many rows it contains. An accurate count of how many rows of data have been received is not available when determining whether to retrieve more data. As a result, QMF for Windows can fetch a large amount of data before determining that the row limit has been exceeded, sometimes it is much more data than specified for the row limit. This problem does not occur with byte limits. The number of bytes of data that have been received is available, even without processing the data. Therefore, in contrast to row limits, the evaluation of byte limits by QMF for Windows is always accurate (subject to the buffering). It is recommended that byte limits be used instead of row limits to effectively control the amount of data fetched.

Buffer size: You can control the size of the buffers that are retrieved. The QueryBlockSize parameter for a server in the SDF (server definition file) controls the maximum size of the buffers that are sent by the database server. Edit the server definition file manually to change this parameter.

Table 66. Resource limits - limits controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. If there are no errors, the schedule is saved at the database server.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.

Table 66. Resource limits - limits controls (continued)

Control	Description
Help	Displays the QMF for Windows online help for this dialog box.

SQL verbs

Select the SQL Verbs tab at the Edit Resource Limits Group Schedule dialog box to specify which SQL verbs the user is allowed to used when accessing a database server from QMF for Windows.

If a user tries to run a query that contains a *disallowed verb*, QMF for Windows cancels the query without sending the SQL to the database server. If a user attempts to run a query that contains an *allowed verb*, QMF for Windows sends the query to the database server and the database server's security authorization validation takes place.

Note: Turning off the permissions for UPDATE, DELETE and INSERT does not affect the ability to perform these actions using the table editor.

You can allow or disallow the SQL following SQL verbs from QMF for Windows:

Table 67. Allowing/dis-allowing SQL verbs

ACQUIRE	ALTER	CALL
COMMENT	CREATE	DELETE
DROP	EXPLAIN	GRANT
INSERT	LABEL	LOCK
REVOKE	SET	SELECT
SIGNAL	UPDATE	

Table 68. SQL verbs controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. If there are no errors, the schedule is saved at the database server.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.
Help	Displays the QMF for Windows online help for this dialog box.

Options

Select the Options tab at the Edit Resource Limits Group Schedule dialog box to specify levels of access to the database objects for the resource limits group.

Table 69. Resource limits options fields

Field	Description
Allow access to server from user interface	The user can access to the server from QMF for Windows.
Allow access to server from programming interface	The user can access the server from the QMF for Windows programming interface.
Fetch all rows automatically	Determines how the database server sends query results to QMF for Windows. QMF for Windows typically requests data from the database server only as required to display query results. For example, if 20 rows fill up the query window, QMF for Windows requests only 20 rows. When the user scrolls down to view the 21st row, QMF for Windows requests more data. If the user runs the query and then waits before scrolling down, the query remains active for that wait time. This consumes resources at the database server while the query is active. If you enable this parameter, QMF for Windows requests data repeatedly until it receives all data, independent of the user's scrolling requests.
Confirm database updates	Determines whether QMF for Windows prompts the user to confirm database changes resulting from the queries they run or the actions they perform when editing tables. Enable this option if you want QMF for Windows to prompt users to confirm database changes. Disable this option if you want database changes to be made without confirmation.
Allow exporting of data	Permits use of the Export Data command on the File menu or in procedures.
Allow table editing	Permits the use of the table editor for QMF for Windows.
Allow running of saved queries only	When checked, this parameter limits the user to running queries that have been saved at the database server. Additionally, the user cannot save new queries at the database server.

Table 69. Resource limits options fields (continued)

Field	Description
Isolation Level for Queries	Sets the isolation level for queries run by users. This option applies to the following types of servers: • DB2 for MVS, version 4 • DB2 for z/OS, version 5 • DB2 UDB for z/OS, version 6 • DB2 Server for VM & VSE, version 5 or higher
Account	Sets the default string for accounting information being sent to the database server when users in the resource limits connect to it.
Account can be overridden	Allows the user to override the default account by entering a new account string on the Set User Information dialog box. Uncheck this checkbox if this resource limits group should be prevented from accessing database objects that are not owned by the group.
Account must be specified	When selected, requires the user to specify a valid accounting string.

Table 70. Resource limits - options controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. If there are no errors, the schedule is saved at the database server.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.
Help	Displays the QMF for Windows online help for this dialog box.

Save data

Select the Save Data tab at the Edit Resource Limits Group Schedule dialog box to set parameters for a resource limits group to save data.

Table 71. Resource limits - save data fields

Field	Description
Allow Save Data command	Save data at the database server. **Note:** Saving data can be resource-intensive and have an impact on your database server and network performance.

Table 71. Resource limits - save data fields (continued)

Field	Description
Default table space	(Available only if you select *Allow Save Data* command.) The default target for tables created by the save data process. The syntax of the table space name you enter must conform to the database server's rules for table space names. This value is used as part of a CREATE TABLE SQL statement executed when the user saves data to a new table. **Note:** Entering an asterisk (*) will take the default table space from each user's profile in the Q.Profile table.
Default table space can be overridden	(Available only if you select the *Allow Save Data* command.) This parameter specifies whether the user is forced to use the tablespace specified in the *Default table space* field or can specify any table space, subject to database security authorizations. Select this check box to allow the user to specify any table space. **Note:** If you select this option but do not specify a default table space, the user cannot specify a table space and the database server uses a default.

Table 72. Resource limits - save data controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. If there are no errors, the schedule is saved at the database server.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.
Help	Displays the QMF for Windows online help for this dialog box.

Binding

Select the Binding tab at the Edit Resource Limits Group Schedule dialog box to set parameters for the bind process. The Binding tab controls static or bound queries. You control the user's ability to create and store these queries.

Table 73. Resource limits - binding fields

Field	Description
Allow binding of packages	Whether users can to bind static packages for their queries.
Allow dropping of packages	Whether users can drop static packages from the database server.
Default collection ID	The default collection ID for static packages bound by users.
Default collection ID can be overridden	Whether a user is forced to use the Default collection ID or can specify any collection ID, subject to database security authorizations.
Default isolation level for packages	The default isolation level for static packages bound by users: • *Repeatable read (RR):* The execution of SQL statements in the package is isolated (protected) from the actions of concurrent users for rows the requester reads and changes as well as phantom rows. • *All (RS):* The execution of SQL statements in the package is isolated (protected) from the actions of concurrent users for rows the requester reads and changes. • *Cursor stability (CS):* The execution of SQL statements in the package and the current row to which the database cursor is positioned are isolated (protected) from the actions of concurrent users for changes that the requester makes. • *Change:* The execution of SQL statements in the package is isolated (protected) from the actions of concurrent users for changes the requester makes. • *No Commit (UR):* The execution of SQL statements in the package is not isolated (protected) from the actions of concurrent users for changes the requester makes.
Default isolation level can be overridden	Whether a user is forced to use the Default isolation level or can specify any isolation level.

Table 74. Resource limits - binding controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. The schedule is saved at the database server if there are no errors.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.

Table 74. Resource limits - binding controls (continued)

Control	Description
Help	Displays the QMF for Windows online help for this dialog box.

Object tracking

Select the Object Tracking tab at the Edit Resource Limits Group Schedule dialog box to set parameters for different levels of object tracking.

QMF for Windows performs object tracking in the Q.OBJ_ACTIVITY_DTL and Q.OBJ_ACTIVITY_SUMM tables, which are created when you install QMF for Windows.

- Q.OBJ_ACTIVITY_DTL (the detail table) holds the detailed tracking options determined by the Object Tracking tab within the resource limits for the resource limits group.
- Q.OBJ_ACTIVITY_SUMM (the summary table) holds summary information for the objects.

Object Tracking enables you to

- run detailed history reports for QMF objects
- locate unused objects
- locate frequently accessed data sources (tables/columns)
- spot potential problem areas

Table 75. Resource limits - object tracking fields

Field	Description
Enable summary object tracking	Whether the object's use is tracked. Enable this option to track the number of times an object has been run, and the most recent times it was run and modified. This information is kept in Q.OBJ_ACTIVITY_SUMM.
Enable detailed object tracking	Whether to keep a detailed record of each action taken using the object. Enable this option to keep a record of each time an object has been run, who ran it, and the results. **Note:** This option must be enabled to use *ad hoc* object tracking or SQL text tracking. A large amount of data can be inserted into the Q.OBJ_ACTIVITY_DTL table if you enable this option.

Table 75. Resource limits - object tracking fields (continued)

Field	Description
Enable ad hoc object tracking	Whether to keep a record of each *ad hoc* query. Enable this option to keep a record of each *ad hoc* query, and the SQL text of that query. **Note:** Check ″Detailed object tracking″ to use *Ad hoc* object tracking.
Enable SQL text tracking	Whether to keep a record of the SQL text of each query. **Note:** Check ″Detailed object tracking″ to use SQL text tracking.

Object Tracking reports provide you with the results of object tracking.

Converting dynamic SQL to static SQL

Static queries are SQL queries that have been passed through the database server's preprocessor and the access plan to the data has been stored within the package. When the static query is executed, the database server no longer uses the query text and its preprocessor to determine the optimal access path, but it can use the access path stored within the package directly. This reduces resource consumption at the server and improves query execution.

All queries created through QMF for Windows use dynamic SQL. You can use object tracking to identify queries that are executed frequently. These queries could be converted to static SQL.

QMF for Windows allows this conversion to be performed for SQL queries only. Follow these steps to convert a prompted query to a static query:

1. Sign on as Admin, then open the prompted query in QMF for Windows.
2. Select Query: Convert to SQL.
3. Save the SQL query.

Then, follow these steps to select the query and convert it to a static query:

1. In QMF for Windows, open the SQL query you just created.
2. Select File: Bind Static Package.
3. Select the Package tab, enter the collection ID and a package name, and change any options.
4. [Optional] Click **Advanced** to create a more detailed bind option definition than using the main dialog box.

Use this dialog to set parameters relating to the Date and Time Format, Blocking and Degree of Parallelism, using Snapshot and Explain, and Dynamic Rules, and rules related to character subtypes such as FOR BIT DATA.

5. Select the Input Variables tab if the original query uses substitution variables. Here, all the substitution variables have to be translated to host variables for the static query. Not all of the substitution variables are easy to map to host variables, as they provide direct text substitution within the query text before being sent to the database server. Host variables, however, are sent to the database server as part of the query.

   Valid data types for host variables include:

   - CHAR(n)
   - VARCHAR(n)
   - INTEGER
   - SMALLINT
   - FLOAT
   - DECIMAL(p,s)
   - DATE
   - TIME
   - TIMESTAMP

6. Click **OK** to convert the query to a static query.

   The Bind Completed confirmation dialog displays.

   Note: The comment at the bottom of this dialog prompts you to save the query. If you do not save the query that you just translated into a static SQL query, it cannot be executed as static SQL later.

Table 76. Resource limits - object tracking controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. If there are no errors, the schedule is saved at the database server.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.
Help	Displays the QMF for Windows online help for this dialog box.

Object tracking maintenance

Two tables are created for object tracking during QMF for Windows installation:

- **Q.OBJ_ACTIVITY_DTL:** the detail table that holds the detailed tracking options determined by the Object Tracking tab within the resource limits for your resource group.
- **Q.OBJ_ACTIVITY_SUMM:** the summary table that holds the summary information for the objects.

Once you enable object tracking, you must perform maintenance on the Q.OBJ_ACTIVITY_DTL table that QMF for Windows creates during installation. You can create a scheduled task to run a DELETE statement that will do most of the cleanup for this table.

With object tracking features enabled, you can maintain a record of all queries and their SQL text. You can track how long queries take, how often they are run, and who runs them.

Object tracking data is added to the Q.OBJ_ACTIVITY_DTL table. Periodically, you must perform maintenance on the table and its associated table space (RAADB.RAATS2) and index (Q.RAAIX2).

The following are recommendations for optimum performance of the Q.OBJ_ACTIVITY_DTL table:

- Run RUNSTATS for the tablespace and index
- As the data volume increases, rebind the QMF for Windows packages to ensure the efficient use of the index
- Delete old data periodically

  For example, you can use this query to delete all rows older than 30 days:

```
DELETE FROM Q.OBJ_ACTIVITY_DTL WHERE "DATE" < (CURRENT DATE - 30 DAYS)
```

Note: The first-used, last-used, and last-modified summary statistics stored in the Q.OBJ_ACTIVITY_SUMM table are not affected when you delete detailed data.

LOB options

DB2 UDB has a data type called Large Object (LOB), able to store non-traditional data such as text files, multimedia files, images, video, photograph, video, sound, or virtually any type of object inside a database table. An LOB is a database entry containing a file that is stored within the database.

If the database that will be accessed with QMF for Windows contains tables that have LOB fields defined, you can prevent access to these tables or create a view of these tables that omits the LOB column.

You can define whether a user can retrieve LOBs, because LOBs are large and use a large amount of resources.

Table 77. LOB options

Field	Description
Disable LOB columns	(An LOB retrieval option): If selected, no data is returned to queries that query columns containing LOB data. The default is ″no.″
Disable LOB data retrieval	(An LOB retrieval option): If selected, users cannot retrieve LOB data. The default is ″no.″
Retrieve LOB data on demand	(An LOB retrieval option): Specific LOBs are returned when you request them individually. The default is ″yes.″
Retrieve LOB data automatically	(An LOB retrieval option): All LOBs in the query results are returned automatically. The default is ″no.″
Maximum LOB column size (KB)	The maximum size of a LOB column, in kilobytes. The default is 0, no maximum.
LOB options can be overridden	Provides users with the authority to override their default LOB options. The default is unchecked.

Table 78. LOB controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. If there are no errors, the schedule is saved at the database server.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.
Help	Displays the QMF for Windows online help for this dialog box.

Report Center

Use QMF Report Center to produce custom reports with shared QMF queries, forms, procedures, and tables. With quick access to these objects, you can specify data format preferences and produce custom reports that can be viewed and manipulated in a variety of applications.

Select this tab to set the resource group schedule options for the QMF for Windows Report Center.

Table 79. Report Center fields

Field	Description
Allow display of all object types	Display QMF object types (queries, forms, reports, and procedures) at the server in the QMF for Windows Report Center interface. Users will see the objects from which their reports are created.

Table 79. Report Center fields (continued)

Field	Description
Allow administration of root node	Administer the top-level folder (″root node″) in the Public Favorites folder, which allows users to add Public Favorites to the root node. Users can grant permission on these Public Favorites so that other users can read (run), write (amend or add to), or administer (grant permissions on) them.
Allow taking ownership of nodes	Take ownership of a Public Favorite. Allows the object to grant read, write, or administrative authority to another user to the Public Favorites. **Note:** By default, the owner of the Public Favorite (the person who created the Public Favorite) has read/write/administrative authority.

Note: See ″Report Center Security,″ for information on how the Report Center governing works. Also, see ″Using QMF Report Center″ in the Using QMF for Windows manual for more information.

Table 80. Report Center controls

Control	Description
OK	Saves the schedule. All values on the tabs of the Resource Limits Group Schedule dialog box are validated. If there are no errors, the schedule is saved at the database server.
Cancel	Discards any changes you have made to the schedule on any tab, then closes the dialog box.
Help	Displays the QMF for Windows online help for this dialog box.

Storing resource limits groups

Resource limits groups are stored in the Q.RESOURCE_TABLE table at the database server, which to prevents users from changing limits that you establish. Define the view Q.RESOURCE_VIEW on this table because QMF for Windows accesses that view, not the table.

QMF for Windows Administrator maintains resource limits groups. You must have the authorization to execute the QMF for Administrator package to maintain resource limits groups. This prevents unauthorized users from changing the limits you establish.

<Default> resource limits group

Users who are not assigned to a resource limits group are governed by the limits in the default resource limits group. The system administrator creates and maintains the default resource limits group, which is named **<Default>**.

Associating resource limits to users

The relationship between a QMF for Windows user and a resource limits group is stored in the Q.PROFILES table at the database server. QMF for Windows Administrator is used to maintain user and resource limits group relationships in this table.

Determining profile for resource limits group

When QMF for Windows connects to a database server, the user provides a user ID and password, which are validated by the database server. If the user information is valid, QMF for Windows determines which resource limits group to use by locating the user's profile. This is done by searching the CREATOR, ENVIRONMENT, and TRANSLATION columns in the Q.PROFILES table. QMF for Windows searches profile values in this order:

Table 81. Determining resource limits

CREATOR	ENVIRONMENT	TRANSLATION
userID	WINDOWS	from current national language of QMF for Windows
userID	NULL	from current national language of QMF for Windows
SYSTEM	WINDOWS	from current national language of QMF for Windows
SYSTEM	NULL	from current national language of QMF for Windows

Open and restricted enrollment

Every user must have a user profile.

- Under *restricted enrollment*, the user is denied access to the server if a user profile with a matching creator does not exist.
- With *open enrollment*, if a user profile with a matching creator does not exist, then QMF for Windows will look for a user profile with a creator equal to SYSTEM.

Access to the server will be granted only if a profile is found through one of these methods. With open enrollment, every user has access to the SYSTEM profile, giving every user potential access to the server.

You can create unique profiles for some users and allow other users to use the SYSTEM default profile. You can delete the SYSTEM profile, which prevents users without unique profiles from accessing the server.

Which resource limits group is in effect?

The RESOURCE_GROUP column in the Q.PROFILES table contains the name of the resource limits group that is in effect for users identified by the profile entry. If this field is NULL or blank, QMF for Windows will assume a default value of the user ID.

QMF for Windows then searches the RESOURCE_GROUP and RESOURCE_OPTION columns in the Q.RESOURCE_VIEW view, looking for these rows:

Table 82. Resource limits group in effect

RESOURCE_GROUP	RESOURCE_OPTION
RESOURCE_GROUP from Q.PROFILES	SHUTTLE_MAIN
RESOURCE_GROUP from Q.PROFILES	SHUTTLE_SCHED1
RESOURCE_GROUP from Q.PROFILES	SHUTTLE_SCHED2
RESOURCE_GROUP from Q.PROFILES	SHUTTLE_SCHED3
RESOURCE_GROUP from Q.PROFILES	SHUTTLE_SCHED4

If no matches are found, the user is governed by the **<Default>** resource limits group. Otherwise, the user is governed based on the limits and controls found in these rows.

DB2 predictive governor support

With V.6 of DB2 UDB for z/OS, the predictive governing function of the Resource Limit Facility (RLF) provides an estimate of the processing cost of SQL statements before they run. The cost estimate is expressed as a number of CPU milliseconds and service units (SUs).

Note: Statement cost estimation is supported for dynamic and static SELECT, INSERT, UPDATE or DELETE.

Predictive governing is only for dynamic statements.

The database server has to set up predictive governing so QMF will detect this layer of control. If predictive governing is set up for DB2, QMF will respond to its limits.

If an operation crosses a threshold set up in the predictive governor on DB2, DB2 will send an error message to QMF for Windows. The reactive governor set up in QMF for Windows, via the Resource Limits dialogs, works in conjunction with the predictive governor set up on DB2.

Cost estimation

To predict the cost of an SQL statement, you execute EXPLAIN to put information about the statement cost in DSN_STATEMNT_TABLE.

The governor controls only the dynamic SQL manipulative statements SELECT, UPDATE, DELETE, and INSERT. Each dynamic SQL statement used in a program is subject to the same limits. The limit can be a *reactive governing limit* or a *predictive governing limit*.

- If the statement exceeds a reactive governing limit, the statement receives an error SQL code.
- If the statement exceeds a predictive governing limit, it receives a warning or *error SQL code*.

You create, populate, and interpret the contents of DSN_STATEMNT_TABLE. You can establish the limits for individual plans or packages, for individual users, or for all users who do not have personal limits.

Each company has its own procedures defined by your installation for adding, dropping, or modifying entries in the Resource Limit Specification table.

How QMF for Windows handles predictive governing

If your installation uses predictive governing, QMF for Windows will check for the +495 and -495 SQLCODEs that predictive governing can generate after a PREPARE statement executes.

- Warning prompt (+495 SQL Error)
- Error message - Exceeded Limit (-495 SQL Warning)

Chapter 6. Report Center Security

QMF Report Center security is based on QMF for Windows governing limits. See the Resource Limits: Report Center dialog box for descriptions of the three resource limit settings. Security settings at the Public Favorites object level are modified in the Report Properties: Security dialog box in QMF Report Center.

Security concepts

The following provides an overview of QMF Report Center security concepts.

General

Users cannot see the Report Options: Security dialog box in QMF Report Center unless they have Administer rights or the ability to take ownership. See the "Allow taking ownership of nodes" option at the Report Center tab of the Resource Limits Group Schedule dialog box.

All users can see top-level Public Favorites folders, though their ability to access and view the underlying reports is controlled by the Read, Write, and Administer rights granted to them in QMF Report Center.

Groups

The PUBLIC group is set up in QMF Report Center automatically, and all users are assigned to PUBLIC. The PUBLIC group allows the administrator to grant permissions to Public Favorites for all users.

When a user belongs to several groups with varying permissions to each group, the group with the most restrictive access takes precedence, unless that group is the PUBLIC group (see below).

If a user is granted rights that provide higher or lower levels of access than the PUBLIC group, the higher or lower level rights take precedence.

Folders

When a user does not have permissions on a folder, the user can see the folder but cannot see the reports that are contained within the folder.

When a user has permissions on a folder, but does not have permissions on the reports contained within that folder, the user can see the reports but cannot access those reports.

Ownership

- Each report can only have one owner at a time.
- The default owner is the report creator.
- Allowing other users to take Ownership, eliminates the possibility of locking access to reports (if a user revokes his own Administrator rights without granting Administrator rights to another user).

  This access is determined by the QMF for Windows governing rights (in the "Allow taking ownership of nodes" field on the Report Center tab of the Resource Limits Group Schedule).

Chapter 7. Sample Tables

QMF for Windows provides nine sample tables that you can use while learning QMF for Windows and before you begin working with your own tables. The sample tables are used throughout the documentation as examples. They contain information about a fictitious electrical parts manufacturing company.

How to create sample tables

1. Select the server at which you want to create the sample tables.
2. Click **Edit**.

   The Parameters dialog box opens.
3. Click the **Sample Tables** tab.

   Existing sample tables will be overwritten if new tables are created.
4. Click **Create**.

   A confirmation message displays.
5. Click **Yes** to create or recreate the sample tables.

   The Set User Information dialog box displays.
6. Enter the password to connect to the database server where the sample tables will be created.
7. Click **OK** to create the tables.

   The status bar at the bottom of the dialog window lists the sample tables that are being created.
8. Click **OK** at the confirmation message. You return to the Parameters dialog box.

Table names and descriptions

These are the sample tables:

Table 83. Sample tables list and descriptions

Table name	Type of information
Q.APPLICANT	The company's prospective employees
Q.INTERVIEW	The interview schedule for the company's prospective employees
Q.ORG	Organization of the company by department, within division

Table 83. Sample tables list and descriptions (continued)

Table name	Type of information
Q.PARTS	Materials supplied by the company
Q.PRODUCTS	Products produced by the company
Q.PROJECT	The company's projects
Q.STAFF	The company's employees
Q.SALES	Sales information for the company
Q.SUPPLIER	Vendors that supply materials to the company

Chapter 8. Object Tracking Reports

The Object Tracking Reports provide information on the results of object tracking, which is a function of QMF for Windows Administrator's resource limits.Tool tips are available for each object tracking report. To view a tooltip, move the mouse over a report and wait approximately one second. Here is an example of a tooltip for the Object Tracking Reports:

```
This report shows which queries have been executed by a specified
user and the resources consumed by those queries. It is sorted in
descending order to highlight those queries with the largest resource
consumption.
```

Running object tracking reports

Follow these steps to run object tracking reports:

1. Open QMF for Windows Administrator.
   The Servers dialog box displays.
2. Select a server.
3. Click **Edit**.
   The Parameters for [server] dialog box displays.
4. Select the Object Tracking Reports tab.
   Nine default object tracking reports are available.

At the Object Tracking tab of the resource limits, check *Enable summary object tracking* to collect data for all summary tracking reports and *Enable detailed object tracking* to collect data for all detailed tracking reports. You can also query the tracking tables directly in QMF for Windows.

Summary and detailed tracking reports

The following reports are available:

Summary reports

- Cancellation summary by user
- Performance summary (average) by query
- Performance summary (cumulative) by user
- Performance summary (cumulative) for single query
- Performance summary (maximum) by query
- Time period performance summary (cumulative) by user

Detail reports

- Cancellation details for single user
- Performance details for single query
- Performance profile for single user

Cancellation details for single user

This report shows which users most frequently have their queries cancelled. High rates of cancellation may indicate that the user needs more training or greater resource privileges.

Cancellation summary by user

This report provides details about the queries that were cancelled for a specific user, including the query name, SQL text, and reason for cancellation.

Performance details for single query

This report shows performance data for queries, including their maximum elapsed time and number of times run. The report is sorted in descending order to highlight the queries with the most resource consumption.

Performance profile for single user

This report shows performance data for queries, including their average elapsed time and number of times run. It is sorted in descending order to highlight the queries with the most resource consumption.

Performance summary (average) by query

This report shows two things:

- which users frequently execute a specific query
- the elapsed time of the query

Use this report once a problem query is identified to determine the users who execute the query most frequently.

Performance summary (cumulative) by user

This report shows detail statistics for each execution of a query. It provides performance information, such as elapsed time and bytes fetched for each execution.

Performance summary (cumulative) for single query

This report shows which queries have been executed by a user and the resources consumed by the queries. The report is sorted in descending order to highlight the queries with the most resource consumption.

Performance summary (maximum) by query

This report shows which users are most actively using QMF and consuming the most resources.

Time period performance summary (cumulative) by user

This report shows, within a time period, which users most actively used QMF and consumed the most resources.

Chapter 9. Procedures

QMF for Windows allows procedures that automate administrative tasks. This type of procedure generally consists of a set of objects. Procedures can run in the background in three ways:

- update tables in a database
- run a query, then write results to a folder
- run a query, then display results

With QMF for Windows, you can use a single **linear procedure** to run QMF commands. A linear procedure, which runs a series of QMF commands with a single RUN command, has no logic built into it. Instead, it runs straight from top to bottom, without loops, jumps, or breaks. A QMF linear procedure is a QMF for Windows object that, instead of containing SQL commands, it contains procedure commands. Where SQL queries manipulate data, procedure commands manipulate QMF objects (tables, queries, forms, and even other procedures). Procedures enable the user to automate actions that would normally be done by clicking buttons and making menu selections.

Procedures are sets of commands that enable the DBA or other users to run queries, print reports, import and export data, as well as perform other functions. Like any other QMF object, procedures can be stored at the database server or saved in a file locally or on a file server. All commands issued through procedures are governed by the resource limits you have configured.

Note: In QMF for Windows, you cannot create a procedure with logic to run a series of QMF commands because QMF for Windows does not support IBM Object REXX procedures. It only supports form calculations using IBM Object REXX.

You can automate certain queries and procedures to be performed each night. To do so, you must create a procedure and schedule it to execute at a specific time.

A sample procedure is shown below:

Table 84. Sample procedure

<table>
<tr><td colspan="2">Main PROCEDURE

This procedure calls for all substitution vales to be used in the included procedures, queries, and forms. This typically includes the name of the current server and the name of the server on which the administrative tasks will be performed.</td></tr>
<tr><td colspan="2">Secondary PROCEDURE

This procedure runs the queries and forms and executes exports and imports. The typical form is:
<pre>RUN QUERY X
DISPLAY FORM X
EXPORT REPORT TO FILE.PRC (or FILE.QRY)
IMPORT PROC FROM FILE.PRC (or FILE.QRY)
RUN PROC</pre></td></tr>
<tr><td>Query X queries a system table or the object directory and produces a list of tables, queries, forms or procedures.</td><td>Form X contains SQL or QMF commands and formats the results as a PROCEDURE or a QUERY. This report is EXPORTED with an extension of .prc or .qry.</td></tr>
<tr><td colspan="2">This report is imported and run.</td></tr>
</table>

The procedure can:

- prompt for certain values (user = `PUBLIC` or `SMITH`)
- use wildcards (S% or %) and substitution variables
- specify an authority (`SELECT` or `ALL`) to an object

Note: You can turn off the *Confirm updates* parameter or run a procedure in batch to avoid database update prompts.

Chapter 10. Scheduling Procedures

You can automate procedures to be performed each night or at specific times during the month. First, create the procedure (via File: New Procedure) then schedule the procedure to execute at a certain time.

Three icons display when you create a procedure:

- Run job
- Schedule job
- View scheduled tasks

Procedure

Follow these steps to schedule procedures to be executed through QMF for Windows.

1. Create or select the procedure.

   Select or create a procedure containing the tasks that will be run automatically.
2. Schedule the procedure with the Schedule job command. The Schedule job icon displays when you create or select a procedure.

   The following warning message displays:

   ```
   WARNING: The encryption used when storing the password in the job file
   can easily be circumvented.
   ```

   This means that the password displays in ordinary text and not as a hidden, or encrypted, password.
3. Click **OK** at the warning message prompt.

   The Schedule Job dialog box displays.

Schedule Job

Task tab

Provide this information for the task (at the Task tab on the Schedule Job dialog box):

Table 85. Schedule job fields

Field	Description
Run	The path for the file to run as a scheduled task. If the task requires parameters, type them after the task's path. If the path to the task program includes spaces, type double quotation marks (") around the entire task path. [Optional]: Click **Browse** to search for the path where the file is located.
Start in	Specify the folder that contains the program file or related files if the program needs to use files from other locations.
Comments	Notes regarding the scheduled task.
Run as	[Optional]: type the user account that runs the scheduled task, such as domain name\username. **Note:** If the scheduled task requires administrator permissions to run, the account must be a member of the Administrators group.
Set password	Click **Set password** to enter the password for the user account you specified in the *Run as* field, if this applies.
Enabled	(Enabled, scheduled task runs at specified time): Check this check box to turn on scheduling for this task.

Schedule tab

Provide this information for the schedule (at the Schedule tab on the Schedule Job dialog box):

Table 86. Schedule information fields

Field	Description
[Unlabeled drop-down field]	This drop-down selection list lists the current schedules for the task.
Schedule task	Specifies how often the scheduled task runs. Choices are: • Daily • Weekly • Monthly • Once [a one-time run] • At system startup • At logon

Table 86. Schedule information fields (continued)

Field	Description
Start time	Specifies the time the scheduled task starts, if the task is scheduled daily, weekly, monthly, or only once.
Schedule task daily	**Schedule task daily [every ____ days]:** how often [expressed as the number of days] the scheduled task is run, starting with the specified date and time.
Show multiple schedules	Specifies whether to configure multiple schedules for this task.
Advanced	Click **Advanced** to set up start and end dates and a schedule for repeating the task, if applicable.

Advanced settings

Click **Advanced** at the Schedule tab on the Schedule Jobs dialog box to provide advanced information for the schedule:

Table 87. Schedule job advanced options

Field	Description
Start date	The date on which the scheduled task will begin.
End date	Specifies whether an end date has been defined for the scheduled task.
Repeat task	Specifies whether the scheduled task will run repeatedly, at the interval specified in Every. • **Every:** specifies how often, in minutes or hours, the scheduled task is repeated. • **Until:** the time until which the scheduled task will be repeated or the duration over which the scheduled task will be repeated. • **If the task is still running, stop it at this time:** specifies whether all instances of the scheduled task that are still running at the deadline in the Time or Duration fields are stopped. This option is useful if your tasks do not stop automatically. If this check box is not selected, the task continues running, even after the deadline. For example, if a task takes one hour to run but starts 15 minutes before the deadline and does not stop automatically when it has finished running, you might need to select this checkbox.

Settings tab

Provide additional settings for schedule (from the Settings tab on the Schedule Job dialog box):

Table 88. Schedule job settings fields

Field	Description
Scheduled task completed	• **Delete the task if it is not scheduled to run again:** specifies whether the task is deleted from your computer's hard disk after the task has finished running and it is not scheduled to run again. This option is useful for tasks you run only once. • **Stop the task if it runs for:** specifies whether the task is stopped after it has been running for the specified amount of time.
Idle time	• **Only start the task if the computer has been idle for at least:** specifies whether the scheduled task is started only after you have not used your keyboard or mouse for a specified period of time. If the task is scheduled to repeat, the first run of the task occurs only if the computer has been idle for the specified period of time. If the computer is not idle when the task first tries to start, Task Scheduler keeps checking to see if the computer is idle for the amount of time specified in the *Only start the task if the computer has been idle for at least* field. If the computer does not become idle during this time, no occurrences of the task run. If the computer has not been idle that long, *Retry for up to:* provides a space for you to type of the length of time (in minutes) that Task Scheduler keeps checking to see if the computer is idle. If you selected the *If the computer has not been idle for that long, retry for up to* field, and the computer is not idle at the scheduled time, you can also click on the scroll arrows to select a new setting. If the computer does not become idle during this time, no occurrences of the task run. • **Stop the task if the computer ceases to be idle:** specifies whether the scheduled task is stopped if you begin to use the computer while the task is running.

Table 88. Schedule job settings fields (continued)

Field	Description
Power management	• **Don't start the task if the computer is running on batteries:** specifies whether the scheduled task is prevented from starting while your computer is running on batteries. Some programs frequently access your hard disk, which drains the batters faster. You can extend the life of your batteries by selecting this checkbox. • **Stop the task if battery mode begins:** specifies whether the scheduled task stops running when your computer starts running on batteries. • **Wake the computer to run this task:** specifies whether the computer wakes to run the task at the scheduled time, even if the computer is in Sleep mode and uses OnNow power management.

Chapter 11. Defining Lists

QMF for Windows recognizes four types of objects:

- Queries
- Forms
- Procedures
- Tables

You can use lists to restrict the visibility of these objects to certain users, typically based on their business requirements. QMF allows you to create predefined lists that the users will see by default when working in QMF for Windows. Lists also simplify work by providing users with a tailored set of QMF for Windows objects by default. The user simply selects File: Open to see the list of tailored objects.

Procedures

Follow these steps to create and save a predefined list:

1. In QMF for Windows, select New: List from the File menu.
2. Select the server from which the list will be created.

   If the window does not list the server, go to List: Set Server to set the active server.
3. Specify the Object owner, the Object name, and the types of objects to be included in the list.
4. Click **Refresh List** to create the list.

   You can modify the list by removing or adding objects from the list.
5. Select File: Save As to save the list.

   This creates a file in the default installation directory that can be opened by using File: Open.

Note: This method does not prevent the user from seeing all objects by clicking **Refresh List** in the List window.

To further restrict users, you can edit the Create View permission for Object View or Table View when the collection is created. Different collections can then be used by different user groups using different server definition files (SDFs).

Appendix A. The SDF File

About the server definition file

When you use QMF for Windows Administrator, you edit a **server definition file** (SDF), which contains the technical information needed by QMF for Windows to access database servers.

Note: The SDF is an initialization file with technical information that QMF for Windows needs to connect to database servers. The system administrator creates and maintains SDFs for QMF for Windows users. By default, the SDF name is the same as the QMF for Windows .ini file.

There are two ways you can use server definition files:

- Each user can have an SDF, or
- You can create a single SDF that is shared by multiple users over a file-sharing network.

  The advantage of this method is that it centralizes SDF administration—you only need to create and maintain a single file. Your users point to that file when they run QMF for Windows. With this method, clients will have QMF for Windows, but no local SDF.

With either approach, you specify the SDF to use on the Options dialog box in QMF for Windows.

Procedure

Follow these steps to create an SDF for a user or group of users:

1. Open QMF for Windows Administrator.
2. Select File: New or File: Save As.

To open and work with a different SDF, select File: Open from the File menu.

Example

```
[Server Parameters]
Server1=QMF Demo
Server2=DB2NT
Server3=DB2AIX
[QMF Demo]
RDBName=SAMPLE
HostName=qmfdemo.rocketsoftware.com
Port=40000
```

```
SymDeskName=*TCP/IP*
DecimalDelimiter=Period
StringDelimiter=Apostrophe
RDBI-CollectionID-0000000000000012=QMFW72
QMFWin-CollectionID-0000000000000011=QMF72
DefaultSchedule1=Y00000078000003840009601710000012C 00000000000000000000A0000000007FDFFFE0
[DB2NT]
Timestamp=19990830205032
RDBName=SAMPLE
HostName=78-axfxb
Port=50000
SymDestName=*TCP/IP*
DriverName=
DefaultSchedule1=Y00000078000003840009601710000012C 00000000000000000000A0000000007FDFFFE0
DefaultSchedule2=@
0000000000000000
DefaultSchedule3=2
DefaultSchedule4=1
QueryBlockSize=32500
DecimalDelimiter=Period
StringDelimiter=Apostrophe
IsolationLevel=CursorStability
BindReplace=1
BindKeep=1
BindOwner=IMRES2
EnterpriseType=
DatabaseType=DB2
xSingleCCSID=1252
xMixedCCSID=0
xDoubleCCSID=0
xQMFCCSID=37
QMFWin-CollectionID-0000000000000011=NULLID
QMFWin-CollectionID-0000000000000014=NULLID
[DB2AIX]
Timestamp=19990831165455
RDBName=SAMPLE
HostName=AZOV
Port=60000
SymDestName=*TCP/IP*
DriverName=
DefaultSchedule1=Y00000078000003840009601710000012C 00000000000000000000A0000000007FDFFFE0
DefaultSchedule2=@
0000000000000000
DefaultSchedule3=2
DefaultSchedule4=1
QueryBlockSize=32500
DecimalDelimiter=Period
StringDelimiter=Apostrophe
IsolationLevel=CursorStability
BindReplace=1
BindKeep=1
BindOwner=db2inst1
EnterpriseType=
DatabaseType=DB2
xSingleCCSID=819
```

```
xMixedCCSID=0
xDoubleCCSID=0
xQMFCCSID=37
QMFWin-CollectionID-0000000000000011=NULLID
QMFWin-CollectionID-0000000000000014=NULLID
```

Appendix B. Architecture, Communication, and Configuration

Architecture

QMF for Windows accesses data stored in any database in IBM's DB2 family of databases.

QMF for Windows and DB2 are distributed relational database applications that operate together in a client/server relationship. Each component plays a separate and distinct role in this relationship:

- QMF for Windows, as the client or requester
- DB2, as the server

QMF for Windows and DB2 satisfy and adhere to a common architecture, IBM's Distributed Relational Database Architecture (DRDA). QMF for Windows uses DRDA to communicate to DB2. It does this by sending SQL requests (queries) to DB2. This means that QMF makes requests for the work then DB2 performs the requested work.

Note: QMF for Windows does not handle configuring DB2 to receive DRDA requests, nor does it configure the network connection from the client machine to your DB2. These tasks are performed at the DB2 level by your systems- or network administrator.

This architecture is a comprehensive and detailed blueprint that specifies the layers and functions required in a client/server distributed database application.

Because QMF for Windows implements the DRDA requester specification, it is capable of connecting to any database that adheres to and implements the DRDA server architecture. The following IBM database products contain a DRDA server component, and are capable of communicating with QMF for Windows:

- DB2 UDB for z/OS, DB2 for z/OS, and DB2 for MVS
- DB2 Server for VSE & VM and SQL/DS™
- DB2 Universal Database and DB2 Common Server
- DB2 Parallel Edition
- DataJoiner®

Your QMF for Windows license determines the DB2 product on which you can install QMF for Windows and use for connections.

Communication

DRDA describes the required communications protocol for the architecture. Specifically, requesters and servers must communicate via the SNA LU 6.2 architecture or TCP/IP protocols.

Important: Because TCP/IP connectivity is a more recent addition to the architecture, not all versions of all DB2 database products support TCP/IP connectivity.

The following servers support TCP/IP connections from DRDA clients:

- DB2 UDB Server for z/OS Version 6
- DB2 for z/OS Version 5
- DB2 Universal Database Version 5

Note: Check your DB2 documentation to see if its DRDA application server component supports TCP/IP.

LU 6.2 and CPI-C

LU 6.2 is an SNA communications architecture. APPC (Advanced Program-to-Program Communication) is a language based on the LU 6.2 architecture. A developer of SNA transaction programs may choose from many different implementations of APPC. Even though each implementation of APPC adheres to the LU 6.2 architecture, two implementations of APPC might not be the same. Therefore, programs that rely on one vendor's APPC implementation might not work with another vendor's implementation. CPI-C (Common Programming Interface-Communications) is a standard, common programming interface that solves this problem.

CPI-C is a programming interface that implements the APPC verb set. Therefore, applications that require the APPC verb set can be written using CPI-C to achieve SNA vendor independence. QMF for Windows is written using CPI-C.

TCP/IP and WinSock

TCP/IP is a collection of *protocols.* WinSock (Windows Sockets) is a standard, common programming interface that implements the TCP protocol. Applications that require TCP/IP protocols can be written using WinSock to achieve TCP/IP vendor independence. QMF for Windows is written using WinSock. See ″TCP/IP configuration″ below for more information.

ODBC

ODBC (Open Database Connectivity) is a programming interface that enables applications to access data in database management systems that use Structured Query Language (SQL) as a data access standard. See ″ODBC connectivity″ below for more information.

Configuration

Network

Work with your in-house networking team to install and configure the network infrastructure before you install, configure, or use QMF for Windows. QMF for Windows Technical Support cannot provide support for configuring your network infrastructure.

SNA

In an SNA network, QMF for Windows must establish an LU 6.2 session between itself and DB2, using the CPI-C interface.

Implementing LU 6.2 connectivity between Microsoft Windows and DB2 can be complex, depending on your SNA environment. This manual does not focus on each Windows-based SNA product or explain how it can be used.

Note: This connectivity is not provided with QMF for Windows. Instead, you must have a third-party product implement this connection. The product you use to provide connectivity must be installed, configured, and working before you install or use QMF for Windows.

QMF for Windows should work with other products that provide a Win CPIC-C interface.

Note: Make sure that you have the latest corrective service or maintenance for your SNA product. Contact your SNA software vendor's technical support for fixes, if any.

TCP/IP

To access a DB2 server using TCP/IP, QMF for Windows must be able to establish a TCP/IP connection from the local host (the system on which QMF for Windows is running) to the remote port (the port on which DB2 is listening). QMF for Windows requires a WinSock 1.1 interface or later to the installed TCP protocol stack.

Note: Work with your in-house TCP/IP networking staff and your TCP/IP software vendor's technical support services to implement and support your network configuration. QMF for Windows cannot provide support for these networking issues.

Because DB2 has been added to different platforms at different release points, check your DB2 database product documentation to see if its DRDA application server component supports TCP/IP.

CLI

A call level interface (CLI) component is included as part of the DB2 UDB for Windows 95 or Windows NT client. CLI is an application programming

interface (API) for relational database access. Using CLI, client applications, such as QMF for Windows, can connect to DB2 UDB servers and execute SQL statements.

For QMF for Windows to use CLI to connect to DB2, you first use the DB2 UDB facilities to define your database servers and how to connect to them.

Note: This configuration is outside the control of QMF for Windows.

> Connectivity information is defined in the DB2 UDB client. Work with your in-house networking staff to implement and support your network configuration. QMF for Windows Technical Support cannot provide support for these networking issues.

Once this configuration is complete, QMF for Windows needs the **database alias** in order to connect to it.

The primary advantage to using CLI is simplified configuration. If a database is defined in the DB2 UDB client, its alias is the only piece of configuration information needed in QMF for Windows.

The disadvantages to using CLI are:

- Generally, performance is substantially worse than when using a DRDA connection
- QMF for Windows only supports connecting to workstation and MVS or z/OS databases
- You must use DRDA connections to access VSE and VM or iSeries databases

To access a DB2 UDB server using CLI, the 32-bit version of QMF for Windows must be able to establish a CLI connection from the local host to the remote host via the DB2 UDB client.

Supported database connections

CLI connections require the runtime client from DB2 Universal Database (UDB), version 5.2 or later. CLI connections are supported to the following databases:

- DB2 for MVS - Version 2.3, Version 3.1, Version 4.1
- DB2 for z/OS - Version 5
- DB2 UDB Server for z/OS - Version 6 and later
- SQL/DS for VM/VSE - Version 3
- DB2 for iSeries - Version 4, release 3, and later
- DB2 Common Server - Version 1 and Version 2
- DB2 UDB (for workstation databases) - Version 5 and later

- DB2 DataJoiner - Version 2

DB2 Connect is required (either the Personal Edition installed locally or the Enterprise Edition installed on a gateway) for all databases except:

- DB2 Common Server - Version 1 and Version 2
- DB2 UDB (for workstation databases) - Version 5 and later
- DB2 DataJoiner - Version 2

Note: CLI is not supported for DB2 for VSE and VM.

CLI configuration considerations

To configure a CLI connection with QMF for Windows, you need to configure a connection to the DB2 Database using DB2 Connect or CAE or the Client Configuration Assistant.

This section describes how to make a connection via various protocols, including TCP/IP, to the database on a server.

Note: You will need the hostname, port number, and database name of the system to which you are connecting.

Add a database

The first phase in configuring a CLI connection with QMF for Windows is to add a database:

1. Start Client Configuration Assistant.
2. Add a new database.

   The Add Database SmartGuide dialog box displays.
3. Choose the option, *Manually configure a connection to a DB2 database*.
4. Press **Next**.
5. Specify the protocol you are using.
6. Select the target operating system.
7. Specify the hostname and port number of the system to which you are connecting.
8. Press **Next**.
9. Enter the database name, which is the location name or RDB name.

   Note: If you do not know the database name, run the following SQL statement from any querying utility:

   ```
   SELECT DISTINCT CURRENT SERVER FROM SYSIBM.SYSTABLES
   ```
10. Press **Next**.
11. Assign an alias to the database. The default is database name from the prior dialog box.

12. Press **Done**.

Set server parameters

The next phase is to set server parameters:

1. Go to QMF for Windows Administrator.
2. Click **New**.

   The General dialog box displays.
3. Enter a server name and set server security parameters.

   The server name can be any label you want to assign to the server definition on the main QMF for Windows Administrator screen.
4. Click **Next**.

   The Connection dialog box displays.
5. Select *Connect using DB2 UDB for Windows CLI*.
6. In the *Database alias* field at the bottom of the window, enter the name of the database from Step 3.

Bind packages directly on the server

Use a CLI connection to bypass QMF for Windows to bind packages on the server using DB2 Connect or CAE.

1. Identify which files to bind.
   a. Access your QMF for Windows directory and expand the bnd\folder.
   b. Select the folder that corresponds to your server's OS and version:
      - **DSN** (02, 03, 04) = MVS
      - **DSN** (05, 06) = z/OS
      - **SQLxx** = UDBs, such as Windows 95, Windows NT, Windows 2000, UNIX (AIX)
      - **QSQ** (03, 04) = iSeries (in QMF for Windows, version 7.0)
      - **ARI** (03, 05, 06, 07) = VM/VSE (in QMF for Windows, version 7.0)

      The binding files reside in the server folders.
2. Go to your DB2 Connect or CAE program and launch the Client Configuration Assistant.
   a. Select the server on which you want to bind the files.
   b. Click **Bind**.

      The Bind Database dialog box displays.
   c. Select *Bind user applications*.
   d. Click **Continue**.
   e. Enter your user name and password.
   f. Click **OK**.

      The Bind Applications dialog box displays.

g. Return to the files you accessed in Step 1.
h. Select these files to **Add**.
i. Click **OK**.

ODBC

ODBC enables QMF for Windows to access non-DB2 databases, including Microsoft® Access® , Oracle, Sybase and Informix. QMF for Windows can catalog the data source information from these non-DB2 servers and allows you to assign a catalog server. ODBC does not require you to bind packages because dynamic SQL is used.

To access an ODBC data source using QMF for Windows, you must define a QMF catalog on a DB2 database through QMF Administrator. To access ODBC data sources using ODBC, QMF for Windows must be able to establish an ODBC connection from the local host to the remote host via the ODBC driver manager.

QMF for Windows requires the Microsoft® ODBC Driver Manager® 3.0 (or later) to access ODBC data sources on various databases. The ODBC driver managers may be used however. Connections are supported to any data source that supports the ODBC protocols listed in your ODBC driver manager.

Note: QMF for Windows can connect using ODBC, as long as the ODBC protocol is supported by the data source, the ODBC driver manager, and the appropriate driver is installed on your computer. The connectivity information is defined in your ODBC driver manager.

Work with your in-house networking staff to implement and support your network configuration. QMF for Windows Technical Support cannot provide support for these issues.

ODBC driver manager files

The following files are included in the Microsoft ODBC Driver Manager 3.0 (or later):

- odbccp32.dll
- odbccp32.cpl
- odbccr32.dll
- odbc32.dll
- odbcint.dll
- odbccu32.dll
- odbcad32.exe

What ODBC supports

Within QMF for Windows, ODBC supports:

- all isolation levels, as long as they are supported by data sources
- binding methods by row and by column
- auto-commit-off, as long as they are supported by data sources
- user DSN, machine DSN, and file DSN
- the ODBC cursor library, if needed
- multi-row block fetch if the ODBC driver supports it
- stored procedures
- large objects (LOB), as long variable length objects

Table 89. What ODBC supports

ODBC Supports	ODBC Does Not Support	Data Sources
All isolation levels, as long as they are supported by data sources	Large objects (LOB)	Cross Access
Binding methods by row and column	Stored procedures	iSeries
Auto-commit-off, as long as they are supported by data sources	Bookmarks and any operations regarding bookmarks	MS Access
Long, varying binary and character values, up to 32,700 bytes	Any interval data types	MS Excel
User DSN, machine DSN, and file data sources		SQL Server
The use of the ODBC cursor library, if needed		Oracle
The use of multi-row block fetch, if the ODBC driver supports it		dBASE
		Paradox
		Delimited Text Files (*.CSV, *.TXT)

Data sources

QMF for Windows allows an ODBC connection to several data sources through the catalog server. Data sources are referenced in the QMF for Windows program by their DSN names. A user ID and password are required to make the connection if it is required by the ODBC data sources.

Most data sources can be used as long as the particular data source drivers are available. Here is a sample list of data sources that can be accessed by QMF for Windows via ODBC:

- CrossAccess
- IBM iSeries
- Microsoft Access
- Microsoft Excel
- Microsoft SQL Server
- Oracle
- dBase
- Paradox
- Sybase
- Informix
- Delimited Text Files (*.CSV, *.TXT)

ODBC drivers

The following ODBC drivers have been tested for use with QMF for Windows.

Note: While this manual discusses different ways in which you can connect and use QMF for Windows, the connectivity products and their uses are too numerous and complex for this document to describe in detail.

Table 90. Tested ODBC drivers

Model	Version	Manufacturer
CrossAccess 32	2.21.3777.00	CrossAccess
Client Access ODBC Driver (32-bit)	7.00.00.00	IBM
Client Access ODBC Driver (32-bit)	5.00.08.00	IBM
IBM DB2 ODBC Driver	7.01.00.00	IBM
IBM DB2 ODBC Driver	6.01.00.00	IBM
IBM DB2 ODBC Driver	5.02.00.00	IBM
SQL Server	3.70.06.23	Microsoft
Microsoft dBase Driver (*.dbf)	4.00.3711.08	Microsoft
Microsoft dBase VFP Driver (*.dbf)	6.00.8428.00	Microsoft
Microsoft Excel Driver (*.xls)	4.00.3711.08	Microsoft
Microsoft Access Driver (*.mdb)	4.00.3711.08	Microsoft
Microsoft Paradox Driver (*.db)	4.00.3711.08	Microsoft
Microsoft Text Driver (*.txt, *.csv)	4.00.3711.08	Microsoft

Table 90. Tested ODBC drivers (continued)

Model	Version	Manufacturer
Oracle ODBC Driver	8.01.06.00	Oracle

Establish ODBC connectivity

You can use an ODBC connection, through QMF Administrator, to define a server to host the QMF catalog. With an ODBC connection, the QMF catalog can contain catalog information from several data sources, including non-DB2 data sources. Also, you can click *Enable Dynamic ODBC Server Definitions* on the QMF for Windows Administrator Window.

Benefits of using ODBC

- access to read-only data sources on DB2 and non-DB2
- centralized data storage

ODBC architecture

QMF Administrator follows these steps to communicate with an ODBC data source:

1. Connects with the data source, which can be DB2 or non-DB2.
2. Defines a DB2 server to host the QMF catalog.

   Selects a catalog name for the ODBC data source.

What is a catalog

Before ODBC, one QMF catalog resided on one DB2 data source. The catalog, named <Default>, is the catalog shared with host QMF and previous versions of QMF for Windows. Now, with an ODBC connection, you can:

- access other non-DB2 data sources
- have QMF read the specific data stored on that particular server
- catalog the data on a DB2 host server

  The catalog server hosts the catalogs that contain QMF objects.

The catalog server is the database on which the QMF for Windows catalog resides. When defining a server through QMF Administrator, a catalog is created on the server that contains database objects such as tables, views, and tablespaces. These tables contain information that QMF for Windows uses to track QMF objects.

Only servers defined using a DRDA or CLI connection can hold a QMF catalog. Catalogs, taken from an ODBC server, are read-only.

Create server with ODBC connection

Configure a database server in QMF Administrator before establishing an ODBC connection. A QMF server, using either a DRDA or CLI connection, is

required in your server definition file (SDF) before establishing an ODBC connection, which can be the catalog server for the ODBC server.

All DB2 and non-DB2 data sources are accessed via ODBC in the same manner, as long as the data sources provide ODBC support. You need to set up each data source individually to set up multiple data sources.

Follow these steps to establish ODBC connectivity on a new server. You need to define a server before you can establish the ODBC connection.

1. Open QMF for Windows Administrator.
2. Click **New**.

   The General window displays.
3. Enter the name of the server and set security information.
4. Click **Next**.

   The Connections profile window displays.
5. Choose *Connect using ODBC*.

   The ODBC parameters group box displays at the bottom of the window.
6. Enter one of the following parameters to establish the connections:
   - Machine data source name

     Machine data sources are stored on the system with a user-defined name. Associated with the data source name (DSN) is all the information the database driver needs to connect to the data source and that the driver manager needs to coordinate all data sources and drivers.

     When an application needs to access the data from the database, it calls the driver manager and passes the name of the machine data source. When a machine data source name is passed, the driver manager searches the system to find a driver used by the data source. It then loads the driver and passes the data source name to it. The driver uses the data source name to find the information it needs to connect to the data source. Finally, it connects to the data source, typically prompting the user for a user ID and password, which are generally not stored.
   - File data source name

     Data sources stored on your PC. The data sources available are user data sources (data sources available for a designated user) or a system data source (available for all users who log onto the PC).

     File data source refers to data sources that are stored on your PC. The data sources available are either user data sources (data sources available for a designated user) or a system data source (available for all users who log on to the PC).

File data sources are stored in a file and allow connection information to be used repeatedly by a single user or shared among several users. When a data file source is used, the driver manager makes the connection to the data source using the information from a.dsn file. This file can be manipulated like any other text file. A file data source does not have a data source name, as does a machine data source, and is not registered to any user or system.

When an application needs to access the data from the database, it calls the driver manager and passes the name of the file data source. When a file data source is passed, the driver manager opens the file and loads the specified driver. If the file also contains a connection string, it is passed to the driver. Using the information in the connection string, the driver connects to the data source. If no connection string is passed, the driver generally prompts the user for the necessary information.

A file data source stores the information about the data, and how to connect to it, in a disk file. File data sources can be shared among all users who have access to the file.

7. [Optional]

   The control below applies only to ODBC connections.

   Click **Advanced** to add data source-specific keyword/values pairs on the Advanced DSN Settings window. Enter driver-specific keywords to connect to a server not listed under Machine data source. Click OK after completing this dialog to return to the Connection profile dialog.

8. Click **Set User Info**. You may receive the following message:

   ```
   Users of the server definition file that you are currently editing must
   supply their own user information. The information that you supply now
   applies only to you.
   ```

   Click **OK**.

   The Set User Information dialog box appears. This setting affects only the operations you perform at this database server from QMF for Windows Administrator. It does not affect or apply to any other users of the server definition file that you are editing.

   Note: Click **Test** at any time to test the connection.

9. Enter your user ID and password.
10. Click **OK**.
11. Click **Next**.

    The Catalog profile window appears.

12. Select a DB2 catalog server and identify which catalog will be used at the server.

- Add catalog names to the list by clicking **Add**.
- Delete catalog names by clicking **Delete**.

13. Click **Finish** to complete the server and connection set-up.

Note: You do not need to bind packages with an ODBC connection.

Sample tables are not provided as they are with other types of connections because an ODBC connection accesses data sources other than QMF data sources.

Appendix C. Custom Installation Options

Use an unattended installation to select the installation options for your QMF for Windows users before beginning the installation process. The advantage of using this method is that you can designate all the installation options rather than selecting the options repeatedly for each installation. These options are defined when you edit the **setup.ini** file.

To perform an unattended installation, use a simple text editor to edit the **setup.ini** file. This file, on Disk1 of the installation diskettes, controls the installation process and determines the settings used for the installation. For each component listed below, **1** means ″install″ and **0** means ″do not install.″

The .ini file contains the following parameters at installation:

```
[Server Parameters]
Server1=QMF Demo

[QMF Demo]
RDBName=SAMPLE
HostName=qmfdemo.rocketsoftware.com
Port=40000
SymDestName=*TCP/IP*
DecimalDelimiter=Period
StringDelimiter=Apostrophe
RDBI-CollectionID-0000000000000049-QMFWin=QFW72
QMFWin-CollectionID-0000000000000067-QMFWin=QFW72
```

See Chapter 3, ″Installation″ for more information about installing and configuring QMF for Windows.

Setup.ini variables

This section presents the variables available for each option in the **setup.ini** file when performing a custom installation (SetupType=2).

[Options]

AutoInstall = 0 | 1
Specifies whether to perform an unattended/automatic installation.

Note: If this value is not specified, the FileServerInstall, SetupType, InstallPath, and ProgramGroup settings are ignored.

FileServerInstall = 0 | 1
Specifies whether a file server installation is performed. In a file server installation, the QMF for Windows files are assumed to be installed in the directory specified by the InstallPath setting. Only the Windows system files are installed to the local machine.

SetupType 0 | 1 | 2
Specifies the type of installation to perform.

- 0 = typical
- 1 = compact
- 2 = custom

  If 2 (the recommended setting) is specified, settings in the [Components] section determine which optional components will not be installed. If a component is flagged to be skipped (0 = ″do not install″), all components will be installed (1 = ″install″). The default is to install all components.

  Note: Option 2, the custom installation, is strongly recommended because most users do not require the Administrator component, or the add-ins. Custom installations enable the system administrator provide the user with the most efficient installation.

  You may, however, list all optional components and specify whether to install each.

  The base program files are always installed.

InstallPath = <path>
Specifies the directory in which QMF for Windows will be installed (if FileServerInstall = 0) or is already installed (FileServerInstall = 1).

ProgramGroup = <group>
Specifies the folder on the Start Menu where shortcuts for QMF for Windows will be created.

AutoExit = 0 | 1
Specifies whether the ″completed″ message will be displayed when installation ends (displayed=0; skipped=1). This should be set to 1 to perform a completely unattended installation.

OverwriteINI = 0 | 1
Specifies whether to overwrite existing **rdbi.ini** and **qmfsdf.ini** files in the Windows directory with the corresponding files on Disk1 of the installation diskettes.

[Components]

Components are specified only if SetupType in the [Options] section is set to 2, for a custom installation.

Base = 0 | 1
Specifies whether to install the main QMF for Windows program files.

Note: If FileServerInstall=1, files are not copied to the local machine; instead, program group icons are created so users can access QMF for Windows from the server.

Admin = 0 | 1
Specifies whether to install the QMF for Windows Administrator application.

Note: Set this to 0 for clients.

Excel = 0 | 1
Specifies whether to install the QMF for Windows add-in for Microsoft Excel.

ReportCenter = 0 | 1
Specifies whether to install the QMF for Windows Report Center.

QMFJAVA = 0 | 1
Specifies whether to install QMF for Windows SQL Query for the Web. A 1 specifies to install QMF Query for Java; whereas, 0 specifies to install this application (that is a companion product to QMF for Windows). See Appendix D, ″QMF Query for Java.″

[Shortcuts]

Common = 0 | 1
Specifies whether to create shortcuts on the user-specific Start Menu (0) or the common Start Menu (1).

Note: This is used only on Windows NT/2000.

Main = 0 | 1
Specifies whether to create a shortcut for the QMF for Windows application.

Admin = 0 | 1
Specifies whether to create a shortcut for the QMF Windows Administrator application.

Note: Always set this option to 0 for user installations; this value is ignored if the Administrator is not installed.

Help = 0 | 1
Specifies whether to create a shortcut for the QMF for Windows online help.

APIHelp = 0 | 1
Specifies whether to create a shortcut for the QMF for Windows API online help.

ReadMe = 0 | 1
Specifies whether to create a shortcut for the QMF for Windows readme.txt file.

WhatsNew = 0 | 1
Specifies whether to create a shortcut for the QMF for Windows ″What's New″ online help.

Uninstall = 0 | 1
Specifies whether to create a shortcut for the QMF for Windows uninstall program.

Note: Even if the shortcut is not created, the unistall is still available from the Add/Remove Programs option on the Control Panel.

Appendix D. Installing and Configuring QMF Query for Java

Overview

QMF Query for Java opens data stored in any member of the DB2 family of databases to any web browser client. It uses 100% pure Java technology to provide server platform independence. It runs in any Java servlet application server environment, including IBM WebSphere Application Server.

This chapter describes the setup and configuration of an HTTP server, a Web application server, and the JDBC drivers on a Windows NT/2000 platform.

Features

QMF Query for Java enables you to create and run SQL queries against DB2 databases and format the results for viewing or use in other applications.

After logging into the database (by specifying the database name, user ID, and password), you can select QMF for Windows SQL queries, prompted queries, and database tables. You can also create a new SQL query manually or by using a prompted query tool. Queries can be parameterized, and you can specify different input values each time you run an SQL query.

Once a query is run and a result set is obtained, there are several ways to view or export the results data. You can:

- apply a QMF form from the server or from your computer to create a report
- generate the default QMF form and apply it to the query results
- export the data in text format for import into other applications
- export the data in comma-separated value (*.CSV) format for import into spreadsheet applications
- save the data to a new or existing database table

The resulting output report or export file (and the database connection) is maintained for the lifetime of the web server session.

Object tracking

Object tracking, similar to that in QMF for Windows, is performed automatically. The tracking data for executed queries is stored in the Q.OBJ_ACTIVITY_SUMM and Q.OBJ_ACTIVITY_DTL tables.

Requirements

JDK

QMF Query for Java requires JDK version 1.1.6, or higher.

JDBC

The version of JDBC is implied by the version of the JDK. QMF Query for Java uses a basic set of JDBC, so the version of JDBC provided by JDK 1.1.6 or later is sufficient.

JDBC driver

A suitable JDBC driver must also be provided. Your DB2 client version should be 5.2 or higher.

The JDBC driver support needed for DB2 servers is included with DB2 UDB database servers, DB2 Connect, and DB2 DataJoiner.

Web application server

Use a Web application server product that supports the Java servlet standard. Supported Web application servers include:

- WebSphere Application Server for AIX, version 3.0 or later
- WebSphere Application Server for zSeries, version 3.0 or later
- WebSphere Application Server for Windows NT/2000, version 3.0 or later

Browser

It is recommended that you use a browser that supports HTML version 4.0 or higher.

You can view error messages in a separate window if your browser is enabled to run Java applets via an "applet" tag. Otherwise, error messages display in the top right of the page.

Installation and configuration

This section describes how to deploy QMF Query for Java to the Web application server.

Servlet properties should be set to enable access to IBM QMF for Java via URLs. The application server must also have its JVM's classpath set to include the IBM JDBC driver classes. Set the `TempFileRealDir` and `TempFileURLDir` parameters as servlet configuration parameters. See "Sample Installation" below for an example installation using IBM WebSphere Application Server and IBM HTTP server.

Configuration parameters

You can specify additional configuration parameters for QMF Query for Java in the QMFWebDir/Config/servlet.properties file:

Table 91. Configuration parameters

Parameter	Description
TempFilePrefix	Sets the name prefix for temporary files generated by QMF Query for Java. The default value is QMF.
BackgroundColor	Sets the background color of QMF Query for Java HTML pages. The default value is **#ffffcc**, which is light yellow.
DriverName	Sets the database JDBC driver. The default value is **COM.ibm.db2.jdbc.app.DB2Driver**.
DatabaseURLPrefix	Sets the database URL prefix for database connections. The default value is **'jdbc:db2:'**.
DatabaseName	The default database name. The user can override this at login unless the `AllowUserLogin` parameter is set to false.
UserID	The default user login name. The user can override this at login unless the `AllowUserLogin` parameter is set to false.
Password	The default user password. The user can override this at login unless the `AllowUserLogin` parameter is set to false.
AllowUserLogin	If this parameter is set to false, the user cannot override pre-set database, login, and password parameters. This may be used to provide access to the product for demonstration purposes only. If this parameter is set to true (by default), the user can override the default parameters for database and login information.
SessionAPIEnabled	By default, QMF Query for Java provides its own implementation of session tracking API (to distinguish between concurrent users and prevent them from mixing data). However, if the servlet engine or application server used to run the product supports user sessions, this parameter can be set to false. In this case, the servlet engine or application server session tracking implementation is used.
MaxPostSize	Sets the maximum size (in bytes) of a QMF form file that can be uploaded from the user's computer. The default is 100,000 bytes.

Table 91. Configuration parameters (continued)

Parameter	Description
AppUser AppPassword	QMF Query for Java will use these parameters to establish a special application connection for operations on QMF tables. AppUser should have all authorities on the QMF tables listed below. In this case, the user may not have any privileges to QMF tables, except for some QMF views. See below for more information.
If you set the AppUser and AppPassword parameters, QMF for Java will create two database connections, such as AppConnect and UserConnect to DB2. The administrator will give authorities to the application user. The user may not have any privileges to QMF tables. **Note:** If the AppUser and AppPassword parameters are not set up, you must provide all the users with authorities to QMF tables (instead of AppUser), as specified below.	

The IBM DB2 JDBC driver must be available in the servlet environment, and must be configured properly to access the databases. The default driver used is `COM.ibm.db2.jdbc.app.DB2Driver`, but you can override this with a configuration parameter, described above.

Database objects

To access a database, QMF for Windows must be installed at the database to create the required database objects. The **AppUser** must have SELECT authority on the following objects:

- Q.OBJECT_DIRECTORY
- Q.OBJECT_REMARKS
- Q.OBJECT_DATA
- Q.OBJ_ACTIVITY_SUMM
- Q.OBJ_ACTIVITY_DTL
- Q.RAA_SUBTYPE
- Q.RAA_OBJECT_VIEW
- Q.RAA_SERVER_INFO
- RDBI.USER_AUTHID.VIEW
- RDBI.USER_ADMIN_VIEW
- RDBI.TABLE_VIEW2

Object authority

In order to save queries with QMF Query for Java, the **AppUser** must have INSERT, UPDATE, SELECT, and DELETE authorities on the objects listed below.

- Q.OBJECT_DIRECTORY*

- Q.OBJECT_REMARKS*
- Q.OBJECT_DATA*
- Q.RAA_SUBTYPE*

The AppUser must have SELECT authority on the following object:

- Q.RAA_OBJECT_VIEW

View authority

The user has to have SELECT authority on the following QMF views:

- Q.RAA_OBJECT_VIEW
- RDBI.TABLE_VIEW2
- RDBI.USER_ADMIN_VIEW
- RDBI.USER_AUTHID_VIEW

Note: AppUser has to have all other authorities to the views listed above.

Tracking authority

To allow a user to track SQL queries, the **AppUser** must have INSERT and UPDATE authorities to the objects Q.OBJ_ACTIVITY_SUMM and Q.OBJ_ACTIVITY_DTL.

Sample installation with WebSphere and HTTP Server

The following is a sample installation using IBM WebSphere Application Server 3.0 (or 3.5) and IBM HTTP Server.

Installation Summary

To install QMF Query for Java, copy a file from the distribution CD-ROM, configure WebSphere's properties using WebSphere's Administration Console, create a directory, and add a directory from the distribution CD-ROM to the new directory.

Procedure

1. Open the ″Topology″ page in WebSphere Administration Console.
2. Choose a WebApplication Server and Servlet Engine for QMF Query for Java.
3. Choose a Virtual Host for QMF Query for Java.
   For example, the name selected for this example is ″default_host″.
4. Stop the Web Application Server (WAS).
5. Create a Web Application for this WAS (under the Servlet Engine).
6. Use the default_host as a host for this application.
7. Set the ″Share context″ option to ″False″ during the creation process.
   For this example, assume that the name of the application is `QMFApp`.

8. Add the following files to the ″Class path″ property of the web application:

   ```
   c:\sqllib\java\db2java.zip
   ```

   This assumes that the SQLLIB directory has been installed on drive C:\.
9. Copy the QMFJavaDir from the QMF for Windows installation CD-ROM to:

   ```
   C:\WebSphere\AppServer\hosts\default_host
   ```
10. For IBM WebSphere Application Server 3.0
    a. Return to the WebSphere Administration Console.
    b. Go to the ″Task″ page.
    c. Create a servlet for QMFJavaApp.
    d. Continue with step #12.
11. For IBM WebSphere Application Server 3.5
    a. Return to the WebSphere Administration Console.
    b. Select:

       ```
       Console  ----> Tasks ----> Add Servlet
       ```
    c. Create a servlet for QMFJavaApp.
    d. Continue with step #12.
12. Set the following values for the servlet to the appropriate input fields:

    Table 92. Sample servlet input fields

Servlet input fields	Value
jar-file	`C:\WebSphere\AppServer\hosts\default_host\QMFApp \servlets\QMFJava.jar`
class name for the servlet	`com.ibm.qmf.QMFJava`
Web path for the servlet	`QMFJava`

    Additionally, add two parameters to the servlet:

    ```
    TempFileURLDir=\QMFJavaDir
    TempFileRealDir=C:\WebSphere\AppServer\hosts\
                              default_host\QMFJavaApp\web\QMFJavaDir
    ```

    Note: The real path you use depends on the location where your WebSphere Application Server has been installed.
13. Restart the WebApplication Server.

Reference

For more information, refer to the appendix, ″Using QMF Query for Java″ in the *Getting Started with QMF for Windows* manual.

Appendix E. Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10594-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will

be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this publication to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is as your own risk.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J74/G4
555 Bailey Avenue
San Jose, CA 95161-9023
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM
Advanced Peer-to-Peer Networking
AIX
AIX/6000
C/370
CICS
CICS/ESA
CICS/MVS
CICS/VSE
COBOL/370
DATABASE 2
DataJoiner
DB2
DB2 Universal Database
Distributed Relational Database Architecture
DRDA
DXT
GDDM
IBM
IBMLink
iSeries
IMS
Language Environment
MVS
MVS/ESA
MVS/XA
OfficeVision/VM
OS/2
OS/390
PL/I
PROFS
QMF
RACF
S/390
SQL/DS
Virtual Machine/Enterprise Systems Architecture
Visual Basic
VM/XA
VM/ESA
VSE/ESA
VTAM
z/OS

Java or all Java-based trademarks and logos, and Solaris are trademarks of Sun Microsystems, Inc. in the United States, other counrtries, or both.

Lotus and 1-2-3 are trademarks of Lotus Development Corporation in the Unites States, other counrties, or both.

Microsoft, Windows, and Windows NT are trademarks or registered trademarks of Microsoft Corporation.

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

Index

Special Characters

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

Q

R

S

T

Readers' Comments — We'd Like to Hear from You

Query Management Facility™ for Windows
Installing and Managing QMF for Windows
Version 7 Release 2

Publication No. GC27-0722-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
GC27-0722-01

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Department HHX/H3
555 Bailey Ave.
SAN JOSE, CA
USA
95161-9023

Fold and Tape **Please do not staple** Fold and Tape

GC27-0722-01

Cut or Fold
Along Line

Program Number: 5697-F42, 5675-DB2, 5697-G24, 5697-G22, 5724-B08,
5648-D35, 5648-D37

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GC27-0722-01