

IBM<sup>®</sup> DB2<sup>®</sup> Query Patroller



# Administration Guide

*Version 7*



IBM<sup>®</sup> DB2<sup>®</sup> Query Patroller



# Administration Guide

*Version 7*

Before using this information and the product it supports, be sure to read the general information under "Appendix D. Notices" on page 117.

This document contains proprietary information of IBM. It is provided under a license agreement and is protected by copyright law. The information contained in this publication does not include any product warranties, and any statements provided in this manual should not be interpreted as such.

Order publications through your IBM representative or the IBM branch office serving your locality or by calling 1-800-879-2755 in the United States or 1-800-IBM-4YOU in Canada.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1999, 2000. All rights reserved.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>Welcome to DB2 Universal Database!</b> . . . . .	<b>v</b>	Result Destinations. . . . .	26
Conventions . . . . .	v	Listing Result Destinations . . . . .	28
<b>Chapter 1. System Overview</b> . . . . .	<b>1</b>	Creating a Result Destination . . . . .	28
DB2 Query Patroller Components . . . . .	1	Editing a Result Destination. . . . .	30
DB2 Query Patroller Server . . . . .	2	Removing a Result Destination. . . . .	30
DB2 Query Patroller Agent . . . . .	3	Job Queue Administration . . . . .	30
Command Line Interface . . . . .	3	Viewing Job Queues . . . . .	30
QueryAdministrator. . . . .	4	Creating a Job Queue . . . . .	31
Tracker . . . . .	4	Editing a Job Queue . . . . .	31
QueryEnabler . . . . .	4	Removing a Job Queue . . . . .	31
QueryMonitor. . . . .	4	System Administration . . . . .	32
DB2 Query Patroller Control Tables . . . . .	4	Listing System Parameters . . . . .	32
How Components Communicate . . . . .	7	Editing System Parameters . . . . .	32
How Jobs are Processed . . . . .	7	<b>Chapter 4. DB2 Query Patroller E-Mail</b>	
<b>Chapter 2. Getting Started.</b> . . . . .	<b>11</b>	<b>Notification . . . . .</b>	<b>35</b>
The iwmm Administrative User Account . . . . .	11	E-mail Notification on UNIX . . . . .	35
Updating the dqpnodes.cfg File . . . . .	11	E-Mail Message Formatting . . . . .	35
Enabling Query Management . . . . .	12	E-mail Notification on Windows . . . . .	35
Starting and Stopping DB2 Query Patroller . . . . .	12	Configuring E-mail Notification on	
Starting and Stopping DB2 Query Patroller		Windows . . . . .	36
as a Service on Windows. . . . .	14	<b>Chapter 5. DB2 Query Patroller Profile</b>	
<b>Chapter 3. Using QueryAdministrator to</b>		<b>Variables. . . . .</b>	<b>39</b>
<b>Administer DB2 Query Patroller.</b> . . . . .	<b>15</b>	DB2 Profile Variables . . . . .	39
Using QueryAdministrator . . . . .	15	Server Profile Variables . . . . .	39
Starting QueryAdministrator . . . . .	15	Client Profile Variables . . . . .	42
Stopping QueryAdministrator . . . . .	16	Windows Specific DB2 Profile Variables. . . . .	44
Displaying Columns . . . . .	16	<b>Chapter 6. Performance Tuning . . . . .</b>	<b>45</b>
Resizing Columns . . . . .	17	Profiling . . . . .	45
User Administration . . . . .	17	Estimating and Monitoring . . . . .	45
Adding Users . . . . .	17	Modifications . . . . .	45
How User Profiles are Resolved . . . . .	18	Configuring the Data Warehouse . . . . .	46
Creating a User Profile . . . . .	18	<b>Chapter 7. Configuring a DB2 Query</b>	
Editing a User or Group Profile . . . . .	21	<b>Patroller Client. . . . .</b>	<b>47</b>
Removing a User or Group Profile . . . . .	21	<b>Chapter 8. Managing Jobs . . . . .</b>	<b>51</b>
Querying for the Effective User Profile . . . . .	21	Releasing and Canceling Jobs . . . . .	51
Data Source Administration. . . . .	22	Controlling Job Flow and Execution . . . . .	51
Listing Data Sources . . . . .	22	Exit Analysis. . . . .	52
Editing a Data Source Definition . . . . .	22		
Node Administration . . . . .	25		
Listing All Nodes . . . . .	25		
Changing the Node Status . . . . .	25		
Result Destination Administration. . . . .	26		

<b>Chapter 9. Replacing the DB2 Query Patroller Database</b> . . . . .	<b>53</b>	syserr.log . . . . .	91
<b>Chapter 10. Using the Command Line Interface to Administer DB2 Query Patroller</b> . . . . .	<b>55</b>	DB2 Diagnostic Log . . . . .	91
Command Line Interface Environment . . . . .	55	Processes . . . . .	91
Submit Command (iwm_submit) . . . . .	55	DB2 Query Patroller Server . . . . .	91
Control Command (iwm_cmd) . . . . .	59	DB2 Query Patroller Agent . . . . .	92
Common iwm_cmd Parameters . . . . .	59	Process Failures . . . . .	92
Monitoring Jobs . . . . .	59	Generating a Trace File . . . . .	92
Controlling Jobs . . . . .	61	Common Server Problems . . . . .	92
Monitoring and Controlling Nodes . . . . .	62	<b>Appendix B. Troubleshooting DB2 Query Patroller Clients</b> . . . . .	<b>95</b>
Listing and Controlling Data Sources . . . . .	63	Troubleshooting QueryEnabler . . . . .	95
Listing and Controlling System Parameters . . . . .	65	Common QueryEnabler Problems . . . . .	95
<b>Chapter 11. Monitoring the DB2 Query Patroller System</b> . . . . .	<b>67</b>	Troubleshooting QueryMonitor and QueryAdministrator . . . . .	96
The Tracker Tool . . . . .	67	Common QueryMonitor and QueryAdministrator Problems . . . . .	97
Creating Tracker Data on the Database Server (iwm_tracker) . . . . .	67	Troubleshooting Tracker . . . . .	98
Tracker Reports . . . . .	68	Common Tracker Problems . . . . .	98
Using Tracker . . . . .	71	<b>Appendix C. Using the DB2 Library</b> . . . . .	<b>99</b>
Job Accounting . . . . .	82	DB2 PDF Files and Printed Books . . . . .	99
Job Accounting Table Columns . . . . .	82	DB2 Information . . . . .	99
<b>Chapter 12. Log Monitoring</b> . . . . .	<b>85</b>	Printing the PDF Books . . . . .	108
Running the Log Monitor . . . . .	85	Ordering the Printed Books . . . . .	109
Log Monitor Configuration File . . . . .	86	DB2 Online Documentation . . . . .	110
Example Configuration File . . . . .	86	Accessing Online Help . . . . .	110
Log File . . . . .	86	Viewing Information Online . . . . .	112
INTERVAL Command . . . . .	87	Using DB2 Wizards . . . . .	114
LOGFILE Command . . . . .	87	Setting Up a Document Server . . . . .	115
MAILTO Command . . . . .	88	Searching Information Online . . . . .	116
INCLUDE and EXCLUDE Commands . . . . .	88	<b>Appendix D. Notices</b> . . . . .	<b>117</b>
<b>Appendix A. Troubleshooting the DB2 Query Patroller Server</b> . . . . .	<b>91</b>	Trademarks . . . . .	120
Distinguishing Between DB2 and DB2 Query Patroller Failures . . . . .	91	<b>Index</b> . . . . .	<b>123</b>
		<b>Contacting IBM</b> . . . . .	<b>127</b>
		Product Information . . . . .	127

---

# Welcome to DB2 Universal Database!

The DB2 Query Patroller Administration Guide provides information about administering and operating the DB2 Query Patroller system.

The audience generally consists of any person involved in day-to-day operations of DB2 Query Patroller. In many organizations, the system administrator or the database administrator is responsible for these tasks.



---

## Conventions

This book uses the following highlighting conventions:

- **Boldface** indicates commands or graphical user interface (GUI) controls such as names of fields, folders, icons, or menu choices.
- *Italics* indicates variables that you should replace with your own value. It is also used to indicate book titles and to emphasize words.
- Monospace indicates file names, directory paths, and examples of text you enter exactly as shown.



This icon marks a fast path. A fast path guides you to information specific to your configuration where multiple options are available.



This icon marks a tip. It provides additional information that can help you complete a task.

For related documentation and a complete description of the DB2 library, see “Appendix C. Using the DB2 Library” on page 99.



- If you do not follow the documented installation method with the recommended defaults, it may be necessary to refer to the *Administration Guide* and the *Command Reference* to complete the installation and configuration.
- The term *Windows 32-bit operating systems* refers to Windows 95, Windows 98, Windows NT, or Windows 2000.
- The term *Windows 9x* refers to Windows 95 or Windows 98.
- The term *DB2 client* refers to a DB2 Run-Time Client, a DB2 Administration Client, or a DB2 Application Development Client.
- In this book, the term *DB2 Universal Database* refers to DB2 Universal Database on OS/2, UNIX, and Windows 32-bit operating systems, unless otherwise stated.



---

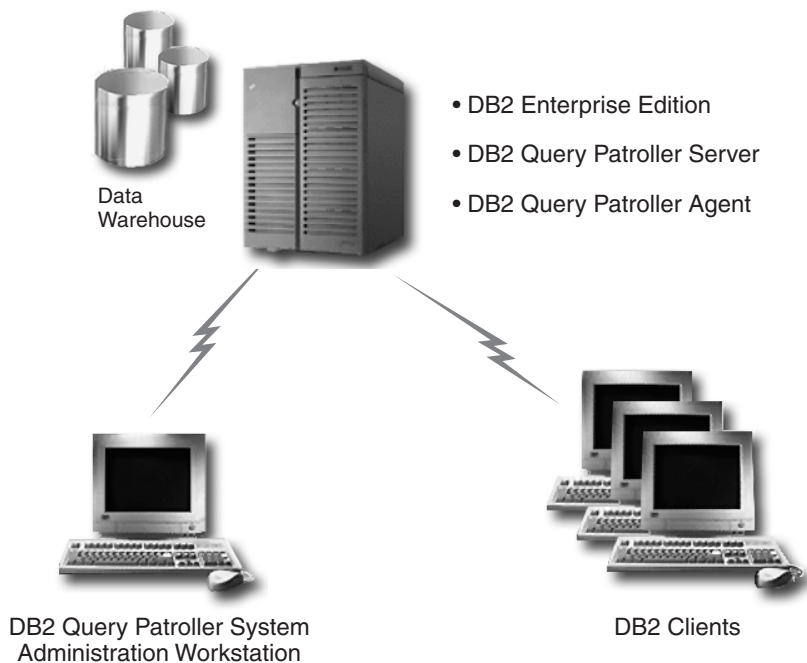
## Chapter 1. System Overview

This chapter provides an overview of the DB2 Query Patroller system, its components, and the tables used by DB2 Query Patroller. Also provided is information on how DB2 Query Patroller components communicate, and an explanation of how queries are handled and processed.

---

### DB2 Query Patroller Components

The DB2 Query Patroller system consists of the DB2 Query Patroller server and agents, the DB2 Query Patroller system administration workstation, and client workstations. DB2 Query Patroller can be deployed on a system running DB2 Enterprise Edition or DB2 Enterprise-Extended Edition. The diagram below illustrates the architecture of DB2 Query Patroller deployed on a system running DB2 Enterprise Edition. Refer to the *DB2 Query Patroller Installation Guide* for a diagram of DB2 Query Patroller deployed on a system running DB2 Enterprise-Extended Edition.



DB2 Query Patroller consists of the following components.

## DB2 Query Patroller Server

The DB2 Query Patroller server executes on a single database node. The server accepts, analyzes, prioritizes, and schedules database requests and optionally notifies users when their requests have been processed.

The server consists of the following software components:

### request server

The request server provides management and coordination for requests received from the various DB2 Query Patroller components.

### cost analyzer

The cost analyzer component determines the relative cost to execute a query using the cost estimate provided by DB2. For more information about DB2 cost estimates, refer to the *DB2 Administration Guide*. The cost analyzer executes once per query. The job scheduler uses the calculated cost to determine when the query should be run. Because the cost analyzer uses the database catalog to determine the relative cost of a full table scan, **RUNSTATS** should periodically be run to keep the catalog entries current. The catalog reflects the status of a table as of the most recent **RUNSTATS**. If it has never been run, the catalog will be empty.

The processing described in the previous paragraph determines if DB2 Query Patroller should proceed with query execution or put the query on hold for later manual intervention by the system administrator. DB2 Query Patroller assigns a hold status to a query if the computed cost exceeds a user's cost threshold as defined in the user's profile. The notifier component of DB2 Query Patroller notifies the user in the event that a query has been placed on hold. For information about the User Profile table, see "DB2 Query Patroller Control Tables" on page 4. See "User Administration" on page 17 for information on setting the cost threshold for a user.

### job scheduler

The job scheduler schedules the query to a DB2 Query Patroller agent for execution. For the scheduling of queries, DB2 Query Patroller uses a scheduling technique that takes into account:

- Current number of queries executing on the system
- Cost of all queries currently executing
- Number of nodes participating in the DB2 Query Patroller system
- Individual user priorities
- Number of queries executing for each user

### notifier

The notifier scans the DB2 Query Patroller tables for newly completed queries at a user-specified time interval. This time interval is specified

by the DQP\_INTERVAL profile variable. For more information, see “DB2 Profile Variables” on page 39. The notifier component notifies the user through system mail when the query completes. If job accounting status is active, the notifier adds a row to the Job Accounting table for each query completed. Information in the Job Accounting table is used by the Tracker tool to provide reports and display database usage history.

The following list outlines the query completion status returned to the user through system mail:

- **Done:** The query completed successfully.
- **Aborted:** An abnormal termination has occurred.
- **Hold:** The query has been placed on hold.
- **Cancelled:** The query was canceled.

### log monitor

The DB2 Query Patroller server and agent components write error information to a log file. The log monitor program periodically activates and examines the log file for new entries. You may want to consider using a product that consolidates system status information, including the DB2 Query Patroller log file. If you do not have such a product, use the DB2 Query Patroller log monitor to send new log entries to users through system mail. See “Chapter 12. Log Monitoring” on page 85 for more information.

## DB2 Query Patroller Agent

The DB2 Query Patroller agent executes on one or more database nodes and processes database requests. On a single-processor or non-clustered SMP machine, the agent and server components run on the same machine. On an MPP machine, or clustered SMP machines, the server component runs on one node and the agents may run on several nodes, including the same node as the server component.

The DB2 Query Patroller agent consists of the following software components:

### node manager

The node manager executes any work scheduled to its node by the job scheduler.

### executor

The executor component runs SQL statements and system commands.

## Command Line Interface

The command line interface enables the system administrator to monitor and control DB2 Query Patroller from a command line prompt. The system administrator can also submit and monitor jobs using the command line interface. When combined with shell scripts or languages such as Perl, awk,

and REXX, you can use the command line interface as an application programming interface (API) to DB2 Query Patroller. See “Chapter 10. Using the Command Line Interface to Administer DB2 Query Patroller” on page 55 for more information.

### **QueryAdministrator**

QueryAdministrator tool is a Java application that provides a GUI interface for administering the DB2 Query Patroller system.

QueryAdministrator enables the system administrator to manage the DB2 Query Patroller system parameters, create or delete profiles for DB2 Query Patroller users, and manage nodes, result destinations, data sources and job queues. See “Using QueryAdministrator” on page 15 for more information.

### **Tracker**

The Tracker is a system administration GUI tool that provides reports displaying database usage history for queries that have been managed by the DB2 Query Patroller system. For example, the Tracker tool allows you to determine which tables and columns have been accessed most frequently, which tables have returned the most result rows, and which jobs have been completed within a specific time period. See “The Tracker Tool” on page 67 for more information. The Tracker tool is a Java application.

### **QueryEnabler**

QueryEnabler places submitted queries, including dynamic queries submitted through any DB2 interface, under the management of the DB2 Query Patroller system. With QueryEnabler, you can wait for the query results to return, or you can have the query results returned at a later time to free your workstation to submit other queries or perform other work. Refer to the *DB2 Query Patroller User's Guide* for more information. The QueryEnabler tool is a Java application.

### **QueryMonitor**

QueryMonitor is a GUI tool used for monitoring and managing queries that have been submitted through the DB2 Query Patroller system. With QueryMonitor, you can monitor query status, view job details, cancel jobs, submit new jobs, drop result tables, and resubmit completed jobs. Refer to the *DB2 Query Patroller User's Guide* for more information about QueryMonitor. The QueryMonitor tool is a Java application.

---

## **DB2 Query Patroller Control Tables**

DB2 Query Patroller control tables belong to the IWM schema. The IWM schema and the control tables are created in the target database during the DB2 Query Patroller installation. The control tables contain all of the information DB2 Query Patroller requires to process queries. For example, the User Profile table contains information such as the user's ID, authority level,

and the maximum number of jobs a user can have running simultaneously. When a user submits a query, DB2 Query Patroller will reference the User Profile table for these parameters.

The following list describes the control tables used by DB2 Query Patroller:

### **Job Table**

The Job table stores information about queries under the control of DB2 Query Patroller. This information includes the ID and user name of the user that submitted the query, the priority of the query, when the query was submitted, and the status of the query. Various components of DB2 Query Patroller access and manipulate the data stored in the Job table. For example, when a user views a job's status through the QueryMonitor tool, it is the information in the Job table that is being displayed, or when a system administrator changes a job's status using the QueryMonitor tool, information in the Job table is updated.

### **User Profile Table**

The User Profile table stores user profile information for DB2 Query Patroller users and groups. The User Profile table contains information such as the user or group ID, authority level, and the maximum number of jobs a user or group member can have running simultaneously. A user or group profile record must exist in the User Profile table before the corresponding user or group member can access the DB2 Query Patroller system. The system administrator uses the QueryAdministrator tool to add or manage user and group profiles for the DB2 Query Patroller system. See "User Administration" on page 17 for more information.

### **Node Information Table**

The Node Information table contains information concerning the status of each node in the system. The information contained in the Node Information table includes the Node ID, node status, the number of scheduled jobs for the node, and node CPU utilization. The DB2 Query Patroller server and agent components maintain the information in this table. The system administrator can alter the status of any node using the QueryAdministrator tool or the command line interface. For more information on changing node status using the QueryAdministrator tool, see "Node Administration" on page 25. For more information about using the command line to administer DB2 Query Patroller, see "Chapter 10. Using the Command Line Interface to Administer DB2 Query Patroller" on page 55.

### **System Parameters Table**

The System Parameters table contains system-wide information that DB2 Query Patroller uses to control processing. Information contained in the System Parameters table includes the maximum cost allowed

for a query, and the maximum number of jobs allowed to be running. The system administrator maintains this table using the QueryAdministrator tool or the command line interface. For more information about the QueryAdministrator tool, see “System Administration” on page 32. For more information about using the command line to administer DB2 Query Patroller, see “Chapter 10. Using the Command Line Interface to Administer DB2 Query Patroller” on page 55.

### **Result Destinations Table**

Normally, the database stores query results as tables. The Result Destinations table contains the descriptions of alternate destinations for query results. Information stored in the result destinations table includes the destination name, destination type, and description. The system administrator maintains this table using the QueryAdministrator tool. For more information, see “Result Destination Administration” on page 26.

### **Data Source Table**

The Data Source table contains information that DB2 Query Patroller uses to control processing for the database instance. The Data Source table stores information such as data source status and the maximum number of jobs that can be running simultaneously against the data source. Presently, a single installation of DB2 Query Patroller controls only one data source. The system administrator maintains the Data Source table using the QueryAdministrator or command line interface. For more information about maintaining the Data Source table using the QueryAdministrator tool, refer to “Data Source Administration” on page 22. For more information about using the command line to administer DB2 Query Patroller, see “Chapter 10. Using the Command Line Interface to Administer DB2 Query Patroller” on page 55.

### **Job Queues Table**

The Job Queues table defines the job queues for each data source. The Job Queues table contains information such as the maximum number of jobs for a queue and the maximum cost for a job on a particular queue. Queues receive job assignments based on the job’s cost. Each queue can have different limits on the number of jobs run. The system administrator maintains this table using the QueryAdministrator tool. For more information about maintaining the Job Queue table using the QueryAdministrator tool, see “Job Queue Administration” on page 30.

### **Job Accounting Table**

The Job Accounting table contains information on completed jobs. This information is used by the Tracker tool to provide reports that display database usage history for queries managed by DB2 Query Patroller. For this table to be populated with data, accounting status

must be activated. This can be done using the System Administration page of the QueryAdministrator tool. For more information about using the QueryAdministrator tool to set the job accounting status, see “System Administration” on page 32.

When job accounting status is set to active, the notifier component of DB2 Query Patroller adds one row to the Job Accounting table each time a job completes. Job accounting status is set through the System Administration page of the QueryAdministrator tool. Only successful jobs are parsed by the Tracker tool. For more information about using the Tracker tool, see “The Tracker Tool” on page 67. For more information about job accounting, see “Job Accounting” on page 82.

### **Result Tables**

Each time DB2 Query Patroller executes a job containing an SQL statement, it normally creates a temporary table containing the results of the query. QueryEnabler retrieves results from these tables and returns the results to the originating query application.

QueryEnabler and QueryMonitor can be used to drop result tables. If result tables are not dropped by the user after retrieval, the notifier component of DB2 Query Patroller can automatically purge result tables after a period of time specified by the system administrator. For information about QueryMonitor or QueryEnabler, refer to the *DB2 Query Patroller User's Guide*.

---

## **How Components Communicate**

The QueryEnabler component communicates with DB2 Query Patroller through the DB2 Query Patroller control tables. The command line interface, agent component, and some parts of the server component communicate with the DB2 Query Patroller request server through sockets. To reduce the number of TCP/IP socket connections required, DB2 Query Patroller has a process called `iwm_net` that runs on the server and agent nodes. The `iwm_net` process allows communication between components on a single node to communicate with each other using TCP/IP sockets and concentrates all communication between nodes to a single outgoing socket connection reducing the number of TCP/IP socket connections required.

---

## **How Jobs are Processed**

This section describes the processing of a job from the time a query is submitted to DB2 Query Patroller until the time that notification of job completion is received.

### **1. Job Submission**

Queries are typically submitted through a query application or dynamically through a DB2 interface. Jobs can also be submitted through

QueryMonitor and the command line interface. When a job is submitted, DB2 Query Patroller saves the SQL statement, the user ID, the date and time the job is scheduled to be run, the maximum cost permitted, and the cost management threshold. The Cost Management Threshold parameter determines whether or not the query will be intercepted by the QueryEnabler tool for user intervention. The Maximum Cost and the Cost Management Threshold parameters are defined for each user or group profile using the User Administration page of the QueryAdministrator tool.

## 2. Cost Analysis

After a user submits a job to DB2 Query Patroller, the server component estimates the cost to execute the query using the cost estimate from DB2. For more information about DB2 cost estimates, refer to the *DB2 Administration Guide*.



DB2 Query Patroller can invoke a customer-written exit program just prior to performing cost analysis for a query. The exit program can examine the SQL statement and prevent DB2 Query Patroller from executing the statement. For more information, see “Exit Analysis” on page 52.

The cost of the job is measured against the Maximum Cost and Cost Management Threshold parameters obtained from the profile of the submitting user. If the computed cost is greater than the maximum cost, the job is placed on hold. The cost management threshold parameter determines whether or not the query will be intercepted for user intervention by the QueryEnabler tool.

If the job does not exceed the cost management threshold, the query will not be intercepted and will execute immediately. If the job exceeds the cost management threshold, and does not exceed the maximum cost, the query will be intercepted by the QueryEnabler tool for user intervention, and the job will run as soon as the scheduled date and time pass and sufficient system resources become available.

If a job is placed on hold, the DB2 Query Patroller system administrator or operator must release or cancel the job. For jobs submitted through the command line interface, cost analysis is performed as part of job submission, not as a separate step.

## 3. Job Scheduling

The server component’s job scheduler activates periodically to scan for new jobs to be run. It processes jobs that are ready to run in order of priority. Jobs of equal priority are processed in the order that they were submitted.



To determine if a job can run, the scheduler looks to see if the number of queries already running has reached the limit set in the User Profile. The scheduler also looks to see if the total number of queries running or the cost of the new job plus the total cost of any running jobs has reached the limits set for the data source and the system. Next, based on the job's cost, the scheduler checks for an associated job queue and determines the limit on the number of jobs for the queue.

If a job cannot be scheduled due to system limits, the scheduler bypasses the job and processes the next ready job. The scheduler processes any jobs skipped during a scheduling cycle the next time it activates.

If a job can be scheduled, it must be assigned to a DB2 Query Patroller node to be executed. The job scheduler selects a node based on the limits set for the data source and on the CPU and disk-utilization statistics being gathered by the node managers. The node selected is the one with the fewest number of assigned jobs. If two or more nodes are running the same number of jobs, the job scheduler selects the node with the lowest CPU utilization.

#### **4. Job Execution**

The node manager of each agent activates periodically to scan for new jobs scheduled to be run on that node. The node manager creates a new process to run the executor for each new job. Normally, for a job containing an SQL statement, the executor component opens a cursor to fetch the result rows. When each job completes, the executor component sends the completion status and execution statistics back to the server component to be recorded in the job table.

In addition to running jobs newly scheduled for the node, each time the node manager activates it also collects CPU and disk utilization statistics and checks for jobs to be canceled or that have aborted. Jobs may be canceled any time from submission until they have been scheduled. Database connections for running jobs are removed when a job is canceled.

#### **5. Notification**

The server component's notifier activates periodically to scan for jobs that complete successfully or unsuccessfully, or for any jobs with a hold status. This component notifies users of these events through e-mail and adds a row to the Job Accounting table for each completed job when the job accounting status is active. The notifier component also purges expired job table entries and result tables.



---

## Chapter 2. Getting Started

This chapter introduces you to the `iwm` administrative user account for DB2 Query Patroller, provides information for updating the `dqpnodes.cfg` file on systems running DB2 Enterprise-Extended Edition, and provides procedures for starting and stopping the DB2 Query Patroller system.

---

### The `iwm` Administrative User Account

During the installation of DB2 Query Patroller, the `iwm` administrative user account was created. To perform DB2 Query Patroller administrative tasks you must be logged on as this user. The `iwm` account is the user account under which the DB2 Query Patroller server is run, and the user account that owns the IWM schema, which contains the DB2 Query Patroller control tables.

---

### Updating the `dqpnodes.cfg` File

If DB2 Query Patroller is installed on a system that is running DB2 Enterprise-Extended Edition, you must update the `dqpnodes.cfg` file for each agent node before starting DB2 Query Patroller. The `dqpnodes.cfg` file is created in the following location during installation:

**UNIX:**

`instance_path/dqpnodes.cfg`

where `instance_path` is the path for the DB2 instance containing the target data source.

**Windows:**

`instance_directory\dqpnodes.cfg`

where `instance_directory` is the directory for the DB2 instance containing the target data source.

The `dqpnodes.cfg` file must contain entries for each node that has a DB2 Query Patroller server or agent installed. Entries in the `dqpnodes.cfg` file are in the following format:

```
node_number host_name {server | agent | none}
```

where `node_number` is the number given to the DB2 Query Patroller node, and `host_name` is hostname or IP address on UNIX, or the computer-name on Windows.

The node number you provide must be a value from 0 to 999. The number you choose for a DB2 Query Patroller node can be the same number used for the DB2 node on which the DB2 Query Patroller agent or server is installed.

Each entry in the `dqpnodes.cfg` file must be defined as `server`, `agent` or `none`. There must be only one entry per line, and only the node where the DB2 Query Patroller server is installed should specify the `server` option. Entries for nodes where the DB2 Query Patroller agent is installed must specify the `agent` option. To suspend a DB2 Query Patroller node entry for whatever reason, you can use the `none` option.

A `dqpnodes.cfg` file on Windows would contain entries similar to the following:

```
0 MY1STNODE server
1 MY2NDNODE agent
3 MY3RDNODE agent
```

To avoid confusion, it is recommended that you specify the same node number for a DB2 Query Patroller node and the DB2 node on which the DB2 Query Patroller server or agent is installed. For example, if you are installing a DB2 Query Patroller server on a DB2 node that has 0 as its designated node number in the `db2nodes.cfg` file, you would specify a node number of 0 in the `dqpnodes.cfg` file for the DB2 Query Patroller server. In this case, you would use the following commands to start both the DB2 node and DB2 Query Patroller server node:

```
db2start nodenum 0
dqpstart nodenum 0
```

---

## Enabling Query Management

For queries to be managed by DB2 Query Patroller, the `DYN_QUERY_MGMT` database configuration parameter must be set to `ENABLE` on the DB2 Query Patroller server. Setting this parameter will allow the QueryEnabler component to intercept submitted queries, and place them under the management of the DB2 Query Patroller system. You must have `SYSADMIN`, `SYSCTRL`, or `SYSMAINT` authority for the database to set database configuration parameters. For information on setting database configuration parameters, refer to the *DB2 Administration Guide*.

---

## Starting and Stopping DB2 Query Patroller

To start DB2 Query Patroller across all nodes, log on as the administrative user `iwm` and enter the following command:

```
dqpstart
```

To stop DB2 Query Patroller across all nodes, log on as the administrative user `iwm` and enter the following command:

```
dqpstop
```

If you have DB2 Query Patroller installed on a system that is running DB2 Universal Database Enterprise-Extended Edition, you likely have the DB2 Query Patroller server installed on one node, and the DB2 Query Patroller agent distributed across other nodes that are participating in the database instance. In this case, you have the option of starting or stopping a DB2 Query Patroller server or agent on a specific node.

To start a DB2 Query Patroller server or agent on a specific node, log on as the administrative user `iwm` and enter the following command:

```
dqpstart nodenum node_number
```

where *node\_number* is the DB2 Query Patroller server or agent node number as defined in the `dqpnodes.cfg` file.

To stop a server or agent on a specific node, log on as the administrative user `iwm` and enter the following command:

```
dqpstop nodenum node_number
```

If you specify the *node\_number* option, you must be aware of the order in which to stop and start the DB2 Query Patroller server and agents.

- Remember to always start the DB2 Query Patroller server first, followed by the agents.
- When stopping DB2 Query Patroller, you should stop all DB2 Query Patroller agents first, followed by the server.

On a system running DB2 Universal Database Enterprise-Extended Edition, the `dqpstart` and `dqpstop` commands can be issued from any node participating in the instance, even nodes that do not have a DB2 Query Patroller server or agent installed.



If you experience an error attempting to start or stop DB2 Query Patroller, check the log files for the `db2start` and `db2stop` commands. These files are located in the following path on UNIX operating systems:

```
db2_instance/log/dqpstart.timestamp.errlog  
db2_instance/log/dqpstop.timestamp.errlog
```

Windows operating systems:

```
db2_instance\log\dqpstart.timestamp.errlog  
db2_instance\log\dqpstop.timestamp.errlog
```

where *db2\_instance* is the path or directory of the DB2 instance, and *timestamp* is the time the log file was created.

## Starting and Stopping DB2 Query Patroller as a Service on Windows

Starting and stopping the DB2 Query Patroller server and agents on the Windows operating system can be done using the preceding instructions or by starting and stopping the DB2 Query Patroller service in the Windows Services window. The service name for DB2 Query Patroller is DB2QP. Before starting DB2 Query Patroller on Windows, ensure that you have set the `DQP_STARTUP` profile variable for each node. For more information refer to “DB2 Profile Variables” on page 39.

To start the DB2 Query Patroller service on Windows you must open the Windows Services window on the node where the DB2 Query Patroller server or agent is installed, select the DB2 Query Patroller service (DB2QP) to highlight it, and click on the **Start** push button. To stop the DB2QP service, select the started DB2QP service to highlight it and click on the **Stop** push button.

If you start DB2 Query Patroller as a service on Windows, and you have DB2 Query Patroller installed on a DB2 Enterprise-Extended Edition system, you must start the DB2 Query Patroller service on the server node first, and then start the service on the agent nodes. When stopping DB2 Query Patroller, you must stop the service on the agent nodes first, then stop the service on the server node.

The `dqpstart` and `dqpstop` commands use the `dqpnodes.cfg` file to ensure the proper start and stop sequence and allow you to start or stop the entire DB2 Query Patroller system from a single node. This is the recommended method for starting and stopping DB2 Query Patroller.

---

## Chapter 3. Using QueryAdministrator to Administer DB2 Query Patroller

The QueryAdministrator tool is a Java application that provides an easy to use interface for setting DB2 Query Patroller parameters and administering the DB2 Query Patroller system on a day-to-day basis. The QueryAdministrator can be used for the following administrative tasks:

- “User Administration” on page 17
- “Data Source Administration” on page 22
- “Node Administration” on page 25
- “Result Destination Administration” on page 26
- “Job Queue Administration” on page 30
- “System Administration” on page 32

If you are setting up your system for the first time you should use the QueryAdministrator tool to set the DB2 Query Patroller parameters according to your system requirements. You can also set most DB2 Query Patroller parameters and administer DB2 Query Patroller using the command line interface. For information about using the command line interface, refer to “Chapter 10. Using the Command Line Interface to Administer DB2 Query Patroller” on page 55.

---

### Using QueryAdministrator

#### Starting QueryAdministrator

A user must have administrator authority on the DB2 Query Patroller system to start the QueryAdministrator tool. Use the following procedure to start QueryAdministrator:

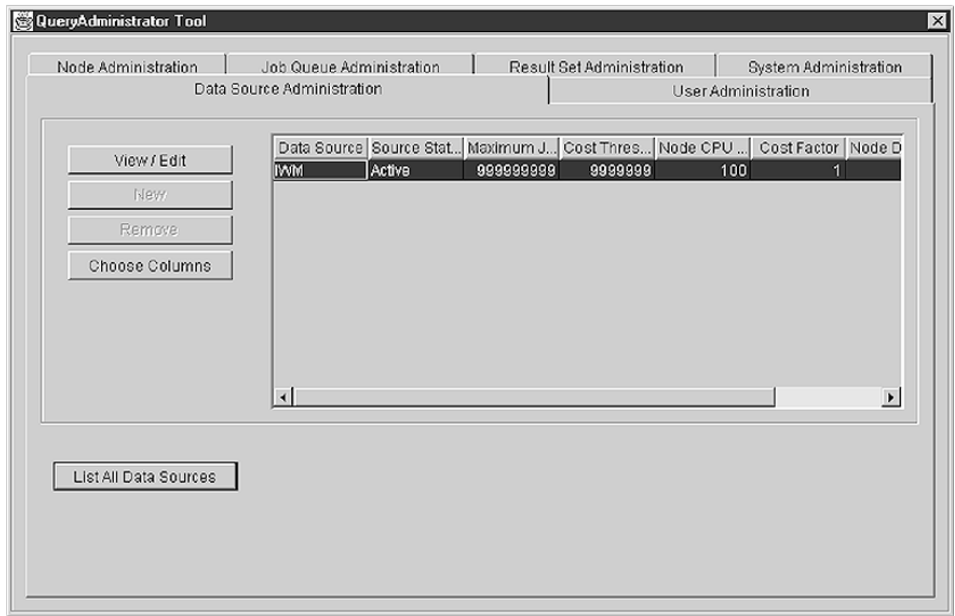
1. If you are using Windows, you can select **DB2 Query Patroller** —> **QueryAdministrator** from the **IBM DB2** program group. On UNIX or Windows you can enter the following command in a system or DB2 command window:

```
QueryAdmin
```

The QueryAdministrator Login window opens.

2. Enter the user name, password, and the database alias, and click **OK**.

The QueryAdministrator main window opens:



## Stopping QueryAdministrator

To stop QueryAdministrator, close the window.

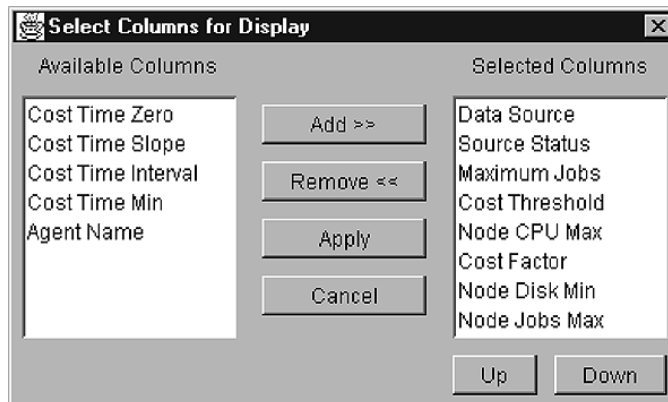
## Displaying Columns

When viewing information on any of the pages of the QueryAdministrator main window, you can select the columns you want to display.

To select the columns you want to display, perform the following steps:

1. Click on **Choose Columns**.

The Select Columns for Display window opens:





2. To add a column to the display, do the following:
  - a. Select the column to be added from the **Available Columns** list.
  - b. Click on **Add>>** to add the column to the **Selected Columns** list.
  - c. Select the newly added column in the **Selected Columns** list.
  - d. Click on **Up** or **Down** to move the newly added column to the position where you want it to display.
3. To remove a column from the display select the column to be removed from the Selected Columns list and click on **Remove<<**.
4. Click on **Apply** when you have finished adding and removing columns.

Any changes you make to the columns displayed are retained for the current session only.

## Resizing Columns

Whenever you are viewing information on a page in the QueryAdministrator main window you can resize any column to a preferred width.

To resize a column, complete the following steps:

1. Move the mouse pointer to the vertical line forming the right edge of the column heading. The mouse pointer changes to a double arrow.
2. Left-click and hold the mouse button on the vertical line; move the line to resize the column.
3. Release the mouse button.

Any changes you make to the columns displayed are retained for the current session only.

---

## User Administration

This section provides information on creating user or group profiles (adding users) for the DB2 Query Patroller system, editing user profiles, and steps for removing user profiles. In order for a user's queries to be managed by DB2 Query Patroller, the user must be defined in the DB2 User Profile table, or belong to a group that has a profile in the User Profile table. The task of adding user or group profiles is performed using the QueryAdministrator tool.

### Adding Users

In order to be added to the DB2 Query Patroller system, a user must have a system account on the target database server system with the CONNECT database privilege for DB2. Because DB2 Query Patroller for DB2 UDB Version 7 offers group support, a user may also belong to a group that has

been granted the CONNECT privilege. For information on creating DB2 users and groups and granting the CONNECT privilege, refer to the *DB2 Administration Guide*.

Once the user has a system account on the target database server with the CONNECT privilege, or belongs to a group with the CONNECT privilege, you can add the user or group to the DB2 Query Patroller system using the User Administration page of the QueryAdministrator tool. When a user or group is added using the DB2 QueryAdministrator tool, the information provided for the user or group is added to the User Profile table.

## How User Profiles are Resolved

If a user belongs to more than one group defined in the DB2 Query Patroller User Profile table, the authorities for the two profiles will be merged, and the highest authorities will be granted to the user. For example, user jsmith belongs to Group A and Group B. If the Group A profile specifies that jsmith can run a maximum of 20 queries simultaneously, and the Group B profile specifies that the maximum is 30, jsmith will have the authority to run 30 queries simultaneously. The same logic applies for a user who is defined as both an individual user and as part of a group.

For the Management Threshold parameter, the highest value is considered to have the greatest authority. The Management Threshold parameter value determines whether or not a query will be intercepted by the QueryEnabler tool for user intervention. If a query does not meet this threshold value, the query will not be intercepted by the QueryEnabler tool, and will execute against the data source immediately, without user intervention.

If a user that belongs to more than one group, and a value for either the e-mail address parameter or User ID parameter is not provided in one group, the user's groups will be searched in alphabetical order by group name for the first non-null parameter value. See "Creating a User Profile" for more information about user profile parameters.

To query the effective user profile for a user that belongs to more than one group, or for a user that is defined individually, and as part of a group, you can use the **Query** button on the User Administration page of the QueryAdministrator tool. For more information, see "Querying for the Effective User Profile" on page 21.

## Creating a User Profile

Use the following procedure to create a user profile:

1. In the QueryAdministrator main window, select the **User Administration** tab.
2. Click on **Create User**.

A user profile window opens:

The screenshot shows a dialog box titled "QueryAdministrator - Information about new user". It contains the following fields and controls:

- User ID: Text input field.
- Type: Dropdown menu with "User" selected.
- Cost Analysis: Dropdown menu with "Do Cost Analysis" selected.
- Authority Level: Dropdown menu with "User" selected.
- Maximum Queries: Text input field.
- Normal Priority: Text input field.
- Low Priority: Text input field.
- High Priority: Text input field.
- User Threshold: Text input field.
- Management Threshold: Text input field with "0" entered.
- Maximum Elapsed Time: Text input field with "0" entered.
- Maximum Result Rows: Text input field with "0" entered.
- Account ID: Text input field.
- Email Addr: Text input field.
- OK and Cancel buttons at the bottom right.

3. Enter a value in each of the fields and click on **OK** when finished.

The following list provides information for each user profile parameter:

#### **User ID**

Provides the ID for the user or group.

**Type** Indicates whether the profile is for a user or a group.

#### **Cost Analysis**

Provides a setting for cost analysis:

- **Do Cost Analysis** indicates that cost analysis is performed on all jobs for this user or group.
- **Don't Do Cost Analysis** indicates that the user or group can determine whether cost analysis is to be performed on specific jobs. If this option is selected, the user can use the **Cost analyze query** check box in the QueryEnabler window to enable or disable cost analysis for a specific job.

#### **Authority Level**

Provides the authority level for the user or group:

- **Not Allowed** indicates that this user or group is not allowed to use DB2 Query Patroller.
- **User** indicates that this user or group has normal authority to submit queries and manage the status of those jobs in the DB2 Query Patroller system.
- **Operator** indicates that the user or group has operator authority. With operator authority a user can use the QueryMonitor tool to manage jobs for all users and groups.

- **Administrator** indicates that this user or group has administrative authority to alter the DB2 Query Patroller configuration and manage jobs for all users and groups.

### **Maximum Queries**

Indicates the maximum number of jobs that a user is able to run simultaneously. Maximum Queries must be an integer from 0 to 99. When creating a group profile, note that the value set for this parameter applies to each user. If this value were set to 10 for Group A, each user belonging to Group A has the authority to run 10 jobs simultaneously.

### **Low Priority**

Provides the priority assigned to a job when the job is submitted on low priority. Low Priority must be an integer from 0 to 999.

### **Normal Priority**

Provides the priority assigned to a job when the job is submitted on normal priority. Normal Priority must be an integer from 0 to 999.

### **High Priority**

Provides the priority assigned to a job when the job is submitted on high priority. High Priority must be an integer from 0 to 999.

### **User Threshold**

Provides the threshold, which if exceeded by a job, causes the job to be placed on hold. User threshold is measured in timeron units. A timeron is a unit of measurement used to give a rough relative estimate of the resources, or cost, required by the database server to execute two plans for the same query. The resources calculated in the estimate include weighted CPU and I/O costs. Refer to the *DB2 Administration Guide* for more information.

### **Management Threshold**

This value determines whether or not a dynamic SQL query will be trapped by the QueryEnabler component. If the cost of the query does not exceed the set value, the query will not be trapped by the QueryEnabler and will execute without user intervention. Query cost is measured in timeron units. A timeron is a unit of measurement used to give a rough relative estimate of the resources, or cost, required by the database server to execute two plans for the same query. The resources calculated in the estimate include weighted CPU and I/O costs. Refer to the *DB2 Administration Guide* for more information.

### **Maximum Elapsed Time**

This option specifies the maximum number of seconds the query will be permitted to run against the DB2 Query Patroller server. If this value is set to 0 or -1 the query will always run to completion.

### **Maximum Result Rows**

This option specifies the maximum number of rows that will be returned in the answer set. If this value is set to 0, the complete answer set will be returned.

### **Account ID**

Provides a user-defined accounting ID that is written in the Job Accounting table.

### **Email Address**

Provides an address for notification when a job completes. If Email Address is null, the user ID is used as the e-mail address. For more information on e-mail notification, refer to “Chapter 4. DB2 Query Patroller E-Mail Notification” on page 35.

## **Editing a User or Group Profile**

Use the following procedure to edit a user profile:

1. In the QueryAdministrator main window, select the **User Administration** tab.
2. Click on **List All Users** to list all user profiles.
3. Select a profile.
4. Click on **View / Edit**. The Detailed User Profile window opens.
5. Edit each field that you want to change.

**Note:** Once a profile has been created, you cannot change the **User ID** or **Type**.

6. Click on **OK** after all new values have been entered.

## **Removing a User or Group Profile**

Use the following procedure to remove a user profile:

1. On the **User Administration** page, select a profile.
2. Click on **Remove**.
3. Click on **Yes** to verify that you want to remove the selected user profile.

**Note:** User iwm cannot be deleted.

## **Querying for the Effective User Profile**

To query the effective user profile for a user, you can use the **Query** button on the User Administration page of the QueryAdministrator tool. Perform the following steps:

1. Click the **Query** button on the User Administration page.
2. Enter a User ID and click **Apply**. The different profiles for the user will be merged providing an effective user profile, which combines the highest authorities from each of the user’s profiles. For information on how an

effective user profile is resolved when more than one profile exists for a user, see “How User Profiles are Resolved” on page 18.

## Data Source Administration

This section describes the procedures for listing and editing data sources.

### Listing Data Sources

Use the following procedure to list data sources:

1. In the QueryAdministrator main window, select the **Data Source Administration** tab.
2. Click on **List All Data Sources** to list all data sources.

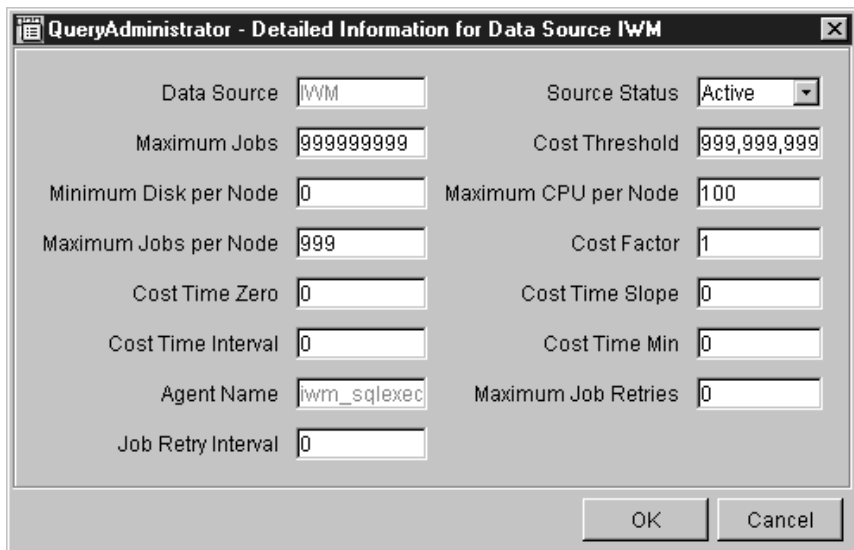
Each data source is listed in tabular format on the Data Source Administration page.

**Note:** DB2 Query Patroller only supports a single DB2 data source.

### Editing a Data Source Definition

Use the following procedure to edit a data source definition:

1. On the Data Source Administration page, select a data source.
2. Click on **View / Edit**. The Detailed Information for Data Source window opens:



Data Source	IWM	Source Status	Active
Maximum Jobs	999999999	Cost Threshold	999,999,999
Minimum Disk per Node	0	Maximum CPU per Node	100
Maximum Jobs per Node	999	Cost Factor	1
Cost Time Zero	0	Cost Time Slope	0
Cost Time Interval	0	Cost Time Min	0
Agent Name	iwm_sqlexec	Maximum Job Retries	0
Job Retry Interval	0		

3. Edit each field that you want to change. See the list below for a description of each field. For example, if you want to put the data source on hold or make the data source inactive, change the **Status** field to the appropriate value. Click **OK** after all new values have been entered.

**Note:** Once a data source has been created, you cannot change its name.

The following list provides information for each data source parameter:

#### **Data Source**

Provides the name of the data source. Only one data source, IWM, is recognized by DB2 Query Patroller. This field cannot be edited.

#### **Source Status**

Provides the status of the data source:

- **Active** indicates that the data source is accepting and scheduling new jobs. Active is the default value for this parameter.
- **Held** indicates that the data source is accepting new jobs and placing them on hold. Queued, scheduled, and running jobs will be allowed to finish.
- **Inactive** indicates that the data source is not accepting new jobs, no additional jobs will be scheduled, but running jobs will be allowed to finish.

#### **Maximum Jobs**

Indicates the maximum number of jobs that can be running simultaneously for this data source. Once this threshold of running jobs is reached, no more jobs will be scheduled for this data source. Maximum Jobs must be greater than or equal to 0. Estimate the number of jobs your data source can handle simultaneously, and modify this value accordingly. This value will vary depending on the performance of your database, the number of users submitting queries, and the average cost of the queries being submitted at any given time.

#### **Cost Threshold**

Indicates the total cost threshold for jobs running simultaneously for this data source. Once this cost threshold for running jobs is reached, no more jobs will be scheduled for this data source. Cost Threshold must be greater than 0. Cost Threshold is measured in timeron units. A timeron is a unit of measurement used to give a rough relative estimate of the resources, or cost, required by the database server to execute two plans for the same query. The resources calculated in the estimate include weighted CPU and I/O costs. Refer to the *DB2 Administration Guide* for more information.

#### **Minimum Disk per Node**

If disk utilization is being monitored for a node, jobs will not be scheduled to that node if the number of bytes available is less than this value. Minimum Disk per Node must be greater than or equal to 0.

#### **Maximum CPU per Node**

If CPU utilization is being monitored for a node, jobs will not be

scheduled to that node if the CPU utilization is greater than this value. Maximum CPU per Node is measured in percentage and must be an integer from 0 to 100.

### **Maximum Jobs per Node**

Indicates the maximum number of jobs that can be simultaneously scheduled on each node. Jobs will not be scheduled to a node if the total number of jobs already scheduled on that node is equal to or greater than maximum. Maximum Jobs per Node must be an integer from 1 to 9999.

### **Cost Factor**

Provides the accounting multiplier for the estimated cost of a job, which is used when writing a job's estimated cost in the accounting table. Since the units for estimated cost are megabytes scanned, the units for the cost factor are dollars per megabyte. Cost Factor must be greater than or equal to 0.

### **Cost Time Zero**

Provides the estimated time (in seconds) for a zero-cost query. If zero is entered, no time estimate is provided for a zero-cost query. Cost Time Zero must be greater than or equal to 0.

### **Cost Time Slope**

If Cost Time Slope is greater than zero, the time estimate in seconds is computed as:  $\text{static cost} * \text{cost time slope} + \text{cost time interval}$ . If Cost Time Slope is zero, no time estimate is provided. Cost Time Slope must be greater than or equal to 0.

### **Cost Time Interval**

See the description for Cost Time Slope.

### **Cost Time Min**

Specifies a minimum estimated time. If Cost Time Min is greater than zero and greater than the estimated time, then this minimum is used instead of the estimated time. If Cost Time Min is zero, then the estimated time is set to Cost Time Zero as if the static cost were zero. Cost Time Min must be greater than or equal to zero.

### **Agent Name**

This field is set by default to `iwm_sql` exec and cannot be modified. This program is used to execute SQL statements.

### **Maximum Job Retries**

Specifies the maximum number of attempts that will be made to run a query through to completion before the query is cancelled.

### **Job Retry Interval**

Specifies the time interval in seconds between attempts to execute a query against the data source.



---

## Node Administration

This section describes the procedures for listing nodes and changing node status.

### Listing All Nodes

Use the following procedure to list all nodes:

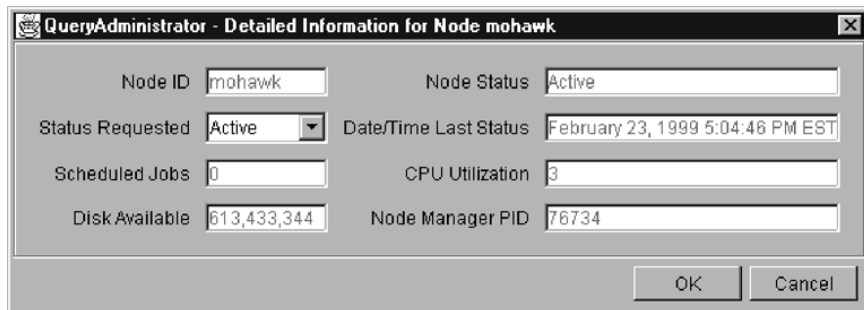
1. In the QueryAdministrator main window, select the **Node Administration** tab.
2. Click on **List All Nodes**. Each node is listed in tabular format on the Node Administration page.

### Changing the Node Status

Use the following procedure to change the node status:

1. On the Node Administration page, select a node.
2. Click on **View / Edit**.

The Detailed Information for Node window opens.



The screenshot shows a dialog box titled "QueryAdministrator - Detailed Information for Node mohawk". It contains several fields for node information:

Node ID	mohawk	Node Status	Active
Status Requested	Active	Date/Time Last Status	February 23, 1999 5:04:46 PM EST
Scheduled Jobs	0	CPU Utilization	3
Disk Available	613,433,344	Node Manager PID	76734

At the bottom right of the dialog box are "OK" and "Cancel" buttons.

3. Select the new status in the **Status Requested** field.

**Note:** **Status Requested** is the only field in the Detailed Information for Node window that can be changed; all other fields display values that have been supplied by DB2 Query Patroller.

4. Click on **OK**.

The following list provides information for each node parameter:

#### Node ID

Provides the ID for the node.

#### Node Status

Contains the current node status

- **Active** indicates that the node is able to run jobs.
- **Inactive** indicates that the node is not available to DB2 Query Patroller.

- **Quiescing** indicates that the node is in transition to the quiescent state. Running jobs will complete but no new jobs will be scheduled.
- **Quiesced** indicates that the node is quiescent. The node is available to DB2 Query Patroller but no new jobs are being scheduled.

### **Status Requested**

Indicates what the node status will be changed to:

- **Active** indicates that the node will be made active.
- **Inactive** indicates that the node will be made inactive. Running jobs will complete and no new jobs will be scheduled.
- **Force** indicates that the node will be made inactive immediately. Running jobs are terminated immediately and no new jobs will be scheduled.
- **Quiesced** indicates that the node will be made quiescent. Running jobs will complete.

### **Date/Time Last Status**

Indicates the date and time node status was last changed.

### **Scheduled Jobs**

Provides the number of jobs scheduled to run plus the number of jobs running on this node.

### **CPU Utilization**

Provides the CPU utilization of the node as a percentage (0 - 100). If CPU utilization information is not being collected, the value is -1.

### **Disk Available**

Indicates the bytes available in the file system where results are created. If disk utilization is not being monitored, the value is -1.

### **Node Manager PID**

Indicates the process ID of the node manager process.

## **Result Destination Administration**

This section provides information and procedures for administering result destinations.

### **Result Destinations**

Normally, DB2 Query Patroller stores query results as database tables. These tables reside in the tablespace defined by the DQP\_RES\_TBLSPC profile variable. If this profile variable does not designate a tablespace, result tables are placed into tablespaces determined by DB2. For more information on setting DB2 Query Patroller profile variables, see “DB2 Profile Variables” on page 39.

In the DB2 Query Patroller Result Destinations table, you can specify descriptions for alternative query result destinations. An alternative destination can be a file or a named pipe. If results are written to a file or pipe, the results are output in comma-separated variable format.

When the results are written to a file, the destination description designates the file name. When the results are written to a pipe, the description designates the name of the executable to which the results will be piped, along with any command line parameters.

### **Result Destination Variables**

For either files or piped destinations, the description can contain the following variables. The run-time values replace these variables before the query executes:

#### **\$(JOB)**

A seven digit number replaces this variable.

#### **\$(USER)**

The user ID of the job's owner replaces this variable.

#### **\$(HOME)**

The path name of the home directory of the job's owner replaces this variable. On Windows, this will be C:\.

If all three variables were used in a result destination description, the description might look similar to the following:

```
$(HOME)\$(USER)\job$(JOB)_results\result.txt
```

After the result destination variables are resolved, the result destination will look similar to the following:

```
C:\qpuser\job0000002_results\result.txt
```

Another variable that can be specified is the \$(PIPE) variable, which is used to represent a pipe that is named in the pipe program. Instead of writing the data to the command's STDIN, DB2 Query Patroller can write data to a named pipe.

If the \$(PIPE) variable is used in the result destination description, the name of the pipe used by the DB2 Query Patroller server will be passed in as a program argument. For example, if the result destination description is:

```
C:\pipes\testprogram.exe $(PIPE)
```

the \$(PIPE) variable will be resolved to the name of the pipe and passed into the program. The program must use this argument to open a connection to the named pipe in order to receive the result set. Sample pipe programs are provided on the DB2 Query Patroller CD-ROM in the db2/samples/c directory.

## Piped Result Destinations

The pipe process can be a shell script or any other executable. DB2 Query Patroller invokes this process under the job owner's user ID, but with the environment variables set as they were when the agent started. The process can read the results from STDIN unless the system administrator specifies the \$(PIPE) variable symbol in the result destination description, in which case it can be read from a named pipe created by the DB2 Query Patroller SQL executor. If the process terminates before reading all of the results, the job aborts. On UNIX, to create a result file on the machine named "remote", you can use the following destination description:

```
rsh remote "cat > job$(JOB)_results"
```

## Pipes to STDIN on Windows

In a Windows environment, pipes to STDIN are handled differently than on other operating systems. Because of the way Windows passes handles between applications, an additional argument is needed. The additional argument, *-9 handleNumber*, is appended to the call to the pipe program. For example, if the piped result destination is C:\pipes\testprogram.exe \$(HOME)\ \$(USER) \\$(JOB), the call to the pipe program, after the result destination variables have been resolved, will be similar to the following:

```
C:\pipes\testprogram.exe C:\qpuser\0000001 -9 handleNumber
```

The additional argument, *-9 handleNumber*, is the parent process's pipe write handle. The pipe program must close this passed handle to allow the server process to break the pipe when it is finished writing. If the handle is not closed, the pipe process will hang waiting for data, and the server process will hang waiting for the pipe process to terminate.



Pipe program examples are included on the DB2 Query Patroller CD-ROM in the db2/samples/c directory. Samples include:

- pipe\_stdin\_NT.c
- pipe\_named\_NT.c
- pipes.c

## Listing Result Destinations

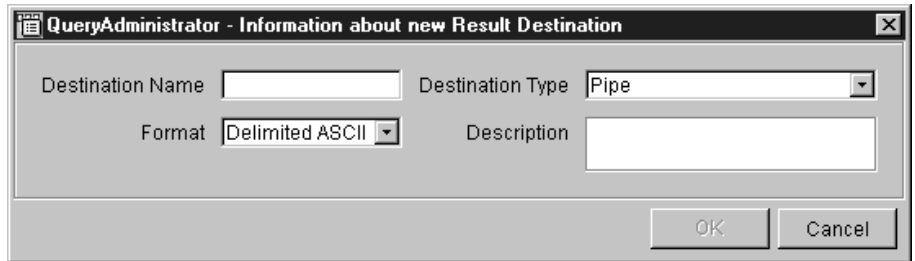
Use the following procedure to list result destinations:

1. In the QueryAdministrator main window, select the **Result Set Administration** tab.
2. Click on **List All Result Sets** to list all result destinations. Each result destination is listed in tabular format on the Result Set Administration page.

## Creating a Result Destination

Use the following procedure to create a result destination:

1. In the QueryAdministrator main window, select the **Result Set Administration** tab.
2. Click on **New**. An empty result destination window opens.



3. Enter a value in each of the fields.
4. Click on **OK** after all values have been entered.

The following list provides information for each node parameter.

#### **Destination Name**

Provides the name for the result destination. QueryEnabler displays these names in the Result Destinations field when a user schedules a new query.

#### **Destination Type**

Indicates the type of destination:

- File indicates that the destination is a file.
- Pipe indicates that the destination is an executable program.

#### **Format**

The only format supported is delimited ASCII.

#### **Description**

For a file, the description provides the name of the file. For a pipe, the description provides the command line for the executable program. Descriptions for both types may contain any of the following substitution variables:

- \$(USER) is the user ID.
- \$(HOME) is the user's home directory.
- \$(JOB) is the DB2 Query Patroller job number.

Another variable that can be specified is the \$(PIPE) variable, which is used to represent a pipe that is named in the pipe program code. Instead of writing the data to the command's STDIN, DB2 Query Patroller can write data to a named pipe.

If the \$(PIPE) is used in the result destination description, the name of the pipe used by the DB2 Query Patroller server will be passed in as a program argument. For example, if the result destination description is:

```
C:\pipes\testprogram.exe $(PIPE)
```

the \$(PIPE) variable will be resolved to the name of the pipe and passed into the program. The program must use this argument to open a connection to the named pipe in order to receive the result set. Sample pipe programs are provided on the DB2 Query Patroller CD-ROM in the db2/samples/c directory.

## Editing a Result Destination

Use the following procedure to edit a result destination:

1. On the Result Set Administration page, select a result destination.
2. Click on **View / Edit**. The Detailed Information for Result Destination window opens.
3. Edit the destination type, the description, or both.
4. Click on **OK**.

**Note:** The Destination Name cannot be changed. Only the type and description of the destination can be changed.

## Removing a Result Destination

Use the following procedure to remove a result destination:

1. On the Result Set Administration page, select the result destination you want to remove.
2. Click on **Remove**.
3. Click on **Yes** to verify that you want to remove the selected result destination.

---

## Job Queue Administration

This section describes the procedures for creating, editing, and removing job queues.

### Viewing Job Queues

Use the following procedure to view the job queues:

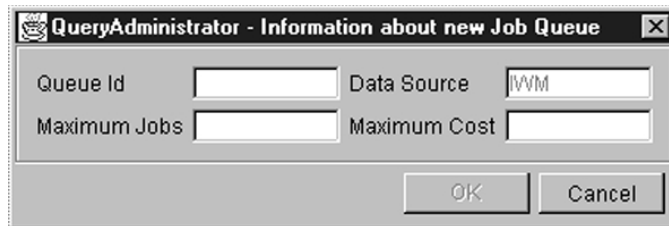
1. In the QueryAdministrator main window, select the **Job Queues** tab.
2. Click on **View / Edit** to list the system job queues. Each job queue is listed in tabular format on the Job Queues page.

## Creating a Job Queue

Use the following procedure to create a job queue:

1. In the QueryAdministrator main window, select the **Job Queues** tab.
2. Click on **New**.
3. Enter a value in each of the fields listed in the table below.
4. Click on **OK** after the values have been entered.

An empty job queue information window opens.



The screenshot shows a dialog box titled "QueryAdministrator - Information about new Job Queue". It contains four input fields arranged in a 2x2 grid. The top-left field is labeled "Queue Id" and is empty. The top-right field is labeled "Data Source" and contains the text "IWM". The bottom-left field is labeled "Maximum Jobs" and is empty. The bottom-right field is labeled "Maximum Cost" and is empty. At the bottom of the dialog, there are two buttons: "OK" and "Cancel".

The following list provides information for each job queue parameter:

### Queue Id

Provides an ID for the job queue. This value must be unique.

### Data Source

The only data source supported is IWM. This field cannot be altered.

### Maximum Jobs

Maximum number of jobs allowed to run in the queue.

### Maximum Cost

Indicates the maximum query cost that will be accepted by the job queue. DB2 Query Patroller assigns jobs to the queue that has the smallest cost limit that is not less than the cost of the job.

## Editing a Job Queue

Use the following procedure to edit a job queue:

1. On the Job Queues page, select a job queue.
2. Click on **View / Edit**. A detailed job queue window opens.
3. Edit the queue ID, the maximum jobs value, or the maximum cost value.
4. Click on **OK**.

## Removing a Job Queue

Use the following procedure to remove a job queue:

1. On the Job Queues page, select the job queue you want to remove.
2. Click on **Remove**.
3. Click on **Yes** to verify that you want to remove the selected job queue.

---

## System Administration

This section describes the procedures for listing and editing system parameters.

### Listing System Parameters

Use the following procedure to list the system parameters:

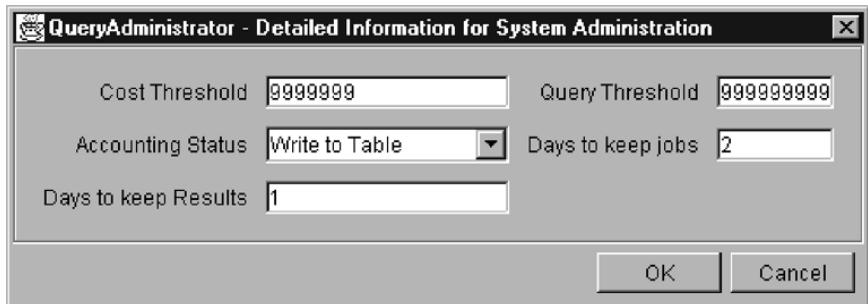
1. In the QueryAdministrator main window, select the **System Administration** tab.
2. Click on **List System Administration** to list the system parameters.

### Editing System Parameters

Use the following procedure to edit system parameters:

1. In the QueryAdministrator main window, select the **System Administration** tab.
2. Click on **List System Administration** to list the system parameters.
3. Click on **View / Edit**.

The Detailed Information for System Administration window opens:



The screenshot shows a dialog box titled "QueryAdministrator - Detailed Information for System Administration". It contains the following fields and values:

Cost Threshold	9999999	Query Threshold	99999999
Accounting Status	Write to Table	Days to keep jobs	2
Days to keep Results	1		

Buttons: OK, Cancel

4. Enter a value in each of the fields.
5. Click on **OK** after all values have been entered.

The following list provides information for each system administration parameter.

#### Cost Threshold

If the total estimated cost of all running queries is greater than the cost threshold, no more queries will be scheduled to run on any node.

#### Query Threshold

If the total count of running queries is greater than the query threshold, no more queries will be scheduled to run on any node.

#### Accounting Status

Indicates whether information is written to the Job Accounting table for use by Tracker tool. Write to Table indicates that accounting is on; Do Not Write to Table indicates that accounting is off.



**Days to Keep Jobs**

Indicates the number of days to retain an entry in the Job table after the job completes. Use 0 to retain job entries indefinitely.

**Days to Keep Results**

Indicates the number of days to retain result sets after the job completes. Use 0 to retain result sets indefinitely.



---

## Chapter 4. DB2 Query Patroller E-Mail Notification

Use the following information to enable e-mail notification for your DB2 Query Patroller System.

---

### E-mail Notification on UNIX

The task of adding users includes defining an e-mail address to which DB2 Query Patroller can send job completion notifications. This can be done using the User Administration Page of the QueryAdministrator tool. Mail can be sent to other mail packages given the proper configuration of the mail system. If not set up properly, the notification message bounces back to the `iwm` account. The mail for the `iwm` account should be monitored, either directly or by forwarding it to another user.

#### E-Mail Message Formatting

On UNIX, if you do not set the `DQP_MAIL` profile variable, the DB2 Query Patroller notifier component uses the shell script `DQP_RUNTIME/bin/iwm_mail.sh` to format e-mail messages, where `DQP_RUNTIME` is the installation path for DB2 Query Patroller. You can also create your own script to format the messages and then set the `DQP_MAIL` profile variable to the name of your script. If you want to create your own formatting script, use the `iwm_mail.sh` script supplied by IBM as a guide.

The formatting script executes each time the notifier needs to send e-mail to a user about a job that has either completed normally, aborted, been canceled, or been put on hold. The script receives the job details using `STDIN` and must format and write the message body to `STDOUT`. The `iwm_mail.sh` script supplied by IBM is written as a KornShell script. You may write a customized formatting script in any language.

**Note:** E-mail message formatting is not required on the Windows platform.

---

### E-mail Notification on Windows

DB2 Query Patroller will send e-mail notification on Windows using the Windows Message Application Programming Interface (MAPI). When DB2 Query Patroller sends e-mail notification, it will look for the `DQP_MAPI_PROFILE` variable in the DB2 Profile Registry. The `DQP_MAPI_PROFILE` should be set to the name of the MAPI profile for your e-mail application. Refer to “DB2 Profile Variables” on page 39 for information on setting profile variables. E-mail applications such as Microsoft Internet Mail and Microsoft Exchange Server are MAPI-compliant service providers.

Mail notification is only sent from the server component of DB2 Query Patroller. DB2 Query Patroller agent components do not require configuration for e-mail notification.

DB2 Query Patroller runs as a service on Windows. There are specific restrictions associated with the Windows Message API and Windows services which may prevent some mail services from being used with DB2 Query Patroller. Specifically, DB2 Query Patroller will only be able to send mail notification if the message store and transport provider is able to directly route mail without assistance from the MAPI spooler. This is referred to as a tightly-coupled mail service. For additional information, refer to Microsoft documentation on MAPI and your mail service provider documentation.

**Note:** Lotus Notes is not supported as a MAPI-compliant mail service provider.

## Configuring E-mail Notification on Windows

To set up mail notification on Windows, perform the following steps:

1. Ensure that MAPI has been installed on the server where DB2 Query Patroller resides. This should be done prior to installing a mail service client to ensure that the mail service will be registered with the Windows messaging subsystem.
2. Create a MAPI mail profile. To create this profile, you must be logged on as user `iwm` on the server where the DB2 Query Patroller server component resides. The `iwm` user account must be associated with the DB2 Query Patroller service, and if using a mail program such as Microsoft Exchange, the `iwm` user account must also have access to the domain where the Microsoft Exchange server resides. Only specify one mail service provider in the mail profile. For more information on how to define a MAPI profile, refer to your mail service provider documentation.
3. Configure DB2 Query Patroller to use a MAPI mail profile when sending e-mail notification. This requires setting one or more of the following profile variables. Refer to “DB2 Profile Variables” on page 39 for information on setting profile variables.

- **DQP\_MAPI\_PROFILE**

The `DQP_MAPI_PROFILE` variable is set to the name of the mail profile that will be used when logging on to MAPI. The mail profile specifies the mail service that will be used to send e-mail notification. In the address book for the mail profile, list the addresses that mail should be sent to. The default address for mail notification must be set to `iwm`. Setting the `DQP_MAPI_PROFILE` variable is required for notification to work. If `DQP_MAPI_PROFILE` is not set, notification will be written to the DB2 Query Patroller log file.

- **DQP\_MAPI\_PASSWORD**

If the DQP\_MAPI\_PASSWORD profile variable is set, the password provided will be used to logon to the MAPI subsystem. It should not be necessary to set this variable when using Microsoft Exchange or Microsoft Internet Mail.

- **DQP\_MAIL\_ACCOUNT**

If the DQP\_MAIL\_ACCOUNT profile variable is set, DB2 Query Patroller will attempt to send mail under the account you specify. Use the following format to specify account information: *username, password*. Setting the DQP\_MAIL\_ACCOUNT profile variable is not required for Microsoft Exchange or Microsoft Internet Mail.

4. Specify the target e-mail address in the DB2 Query Patroller User Profile. Specifying an e-mail address can be done on the User Administration page of the QueryAdministrator tool. The e-mail address must be in a form recognizable by your mail application.



---

## Chapter 5. DB2 Query Patroller Profile Variables

This chapter provides information for each DB2 Query Patroller profile variable.

---

### DB2 Profile Variables

DB2 Query Patroller profile variables are stored in the DB2 Profile Registry. The system administrator can set profile variables to alter the behavior of DB2 Query Patroller. Because DB2 Query Patroller profile variables are located in the DB2 Profile Registry, the system does not require rebooting after a change is made. Many of the following profile variables are set during installation.

Registry information is stored in files containing variable names and values. However these files should not be edited directly. To update registry values, use the `db2set` command. For example, to set the default database value for your database instance, you would enter the following command in a DB2 command window:

```
db2set DB2DBDFT=default_database -i instance
```

Where *default\_database* is the name of the default database, and *instance* is the name of the DB2 instance containing the target database. For additional information and the complete syntax of the `db2set` command, refer to the *Command Reference*.

### Server Profile Variables

#### DB2DBDFT

Set the DB2DBDFT profile variable to the target database name. Queries submitted through DB2 Query Patroller that do not specify a database name will run against this default database.

#### DQP\_DISKMON

Set the DQP\_DISKMON profile variable to a valid file system path where the result sets are created. Each agent node can monitor the space available in a different file system. When available disk space goes below a system threshold, no more jobs will be submitted to that node. If DQP\_DISKMON is not set, disk space is not monitored and creation of result tables will fail when the file system becomes full.

#### DQP\_EXIT\_AN

DB2 Query Patroller invokes the exit analysis program if you set DQP\_EXIT\_AN to the path name of the executable program before

starting the server processes. The exit can be a shell script or any other executable. DB2 Query Patroller invokes the exit under the iwm user ID.

On UNIX, DB2 Query Patroller passes the job owner's user ID as the only argument to the exit. On Windows, additional arguments, *-9 handleNumber*, are passed and the exit program must close the passed handle before the SQL statement can be read from STDIN. If the exit routine terminates with an exit code of zero, and without having written anything to STDOUT, DB2 Query Patroller continues to process the job normally.

If the exit routine terminates with a non-zero exit code or writes anything to STDOUT, the job ends abnormally and sends a message to STDOUT. For more information about exit analysis, see "Exit Analysis" on page 52.

### **DQP\_INTERVAL**

Set the DQP\_INTERVAL profile variable to the number of seconds that each server and agent process sleeps between searches for work. The default value is 20, which can be appropriate on a system with a moderate load. On a system with a light load, use a smaller value, such as 5, so DB2 Query Patroller will be more responsive to new requests.

### **DQP\_LOCAL\_SERVANTS**

By default, when you start the DB2 Query Patroller system using *dqpstart*, or start the DB2 Query Patroller service on Windows, five *iwm\_local* processes are started. If a different number of *iwm\_local* processes is required, set the DQP\_LOCAL\_SERVANTS profile variable to that number. The new number of processes will be realized when the DB2 Query Patroller system is restarted. To increase system performance, you may want to increase the number of processes.

### **DQP\_LOG**

Set the DQP\_LOG profile variable to the directory into which DB2 Query Patroller writes its log files. The default value on UNIX is *DQP\_RUNTIME/log*, where *DQP\_RUNTIME* is the installation path for DB2 Query Patroller. On Windows, the default is *instance\_directory\log*. For more information about log monitoring, see "Chapter 12. Log Monitoring" on page 85.

### **DQP\_LOGMON**

If you set DQP\_LOGMON, the log monitor will start when the DB2 Query Patroller server processes are started, and will stop when the DB2 Query Patroller server processes are stopped. The DQP\_LOGMON profile variable should be set to the path name of the log monitor configuration file. For more information about log monitoring, see "Chapter 12. Log Monitoring" on page 85.



## DQP\_MAIL

You can set DQP\_MAIL to cause the notifier component to use the program named to format e-mail messages. If you do not set this variable DB2 Query Patroller uses *DQP\_RUNTIME/bin/iwm\_mail.sh*. This option is not available on Windows.

## DQP\_NET

DQP\_NET must be set to the local host TCP/IP address and port. Use the format *address:port*, where *address* represents either the dotted notation for the address or a name resolvable in the hosts file or through DNS, and *port* represents either the port number or a name resolvable in the services file. DQP\_NET must be defined on each node and cannot be set to the same value as DQP\_SERVER. Using the same port number across all nodes is recommended. The port defined will be used for communication between processes residing on a single node. On the server node, this port will be used to listen for connections from processes running on other nodes in the system.

## DQP\_NO\_CPU

If this profile variable is set, DB2 Query Patroller will not gather CPU utilization statistics. You may want to set this parameter to reduce the system load.

## DQP\_NOEXPLAIN

If you want to ensure that all jobs run and that none are disqualified due to their estimated cost, disable cost analysis for all DB2 Query Patroller jobs by setting the DQP\_NOEXPLAIN profile variable to any non-null value. Setting this variable prevents DB2 Query Patroller from performing a cost analysis on queries.

## DQP\_PURGEHOURS

If DQP\_PURGEHOURS is set to Y, this will cause the value of the Job Purge Days system parameter to be read as hours causing jobs to be purged from the Job table in hours rather than days. For more information on setting the Job Purge Days system parameter, see "System Administration" on page 32.

## DQP\_RES\_TBLSPC

Set the DQP\_RES\_TBLSPC profile variable to the name of the tablespace that will hold result tables. Each node may specify a different name, or multiple nodes may share a name. If this variable is not set, the result tables will be placed into tablespaces determined by DB2.

## DQP\_RUNTIME

DQP\_RUNTIME profile variable is set to the path where the DB2 Query Patroller software is installed. This parameter is set during installation.

## DQP\_SERVER

Set the DQP\_SERVER profile variable to the TCP port and IP address of the DB2 Query Patroller server component. DQP\_SERVER must be set to the same value across all DB2 Query Patroller nodes. Use the format *address:port*. The address represents either the dotted notation for the address or a name resolvable in the *hosts* file or through the Domain Name Server (DNS). The port represents either the port number or a name resolvable in the *services* file. DQP\_SERVER must be defined on each DB2 Query Patroller node and cannot be set to the same value as DQP\_NET. The port defined will be used for communication across DB2 Query Patroller nodes.

## Client Profile Variables

### DQP\_ABORTRESULT

If set to Y, result sets that exceed the value provided for the Maximum Result Rows parameter as defined in the User profile table will be aborted. No result set will be returned. To define the Maximum Result Rows parameter for a user or group, see “User Administration” on page 17. If DQP\_ABORTRESULT is not set, the result set will be truncated once the row count reaches the defined number of rows specified by the Maximum Result Rows parameter.

### DB2DBDFT

Set the DB2DBDFT profile variable to the target database name. Queries submitted through DB2 Query Patroller that do not specify a database name will run against this default database.

### DQP\_LAST\_RESULT\_DEST

If DQP\_NTIER variable is set to RUN or CHECK, DQP\_LAST\_RESULT\_DEST may be set to the name of a valid result destination. Result destinations are defined using the Result Set Administration page of the QueryAdministrator tool. If the DQP\_NTIER variable is set, and DQP\_LAST\_RESULT\_DEST is not set, the result destination will be a table determined by DB2.

### DQP\_NET

DQP\_NET must be set to the local host TCP/IP address and port. Use the format *address:port*, where *address* represents either the dotted notation for the address or a name resolvable in the *hosts* file or through DNS, and *port* represents either the port number or a name resolvable in the *services* file. DQP\_NET must be defined on each node and cannot be set to the same value as DQP\_SERVER. Using the same port number across all nodes is recommended. The port defined will be used for communication between processes residing on a single node. On the server node, this port will be used to listen for connections from processes running on other nodes in the system.

## DQP\_NTIER

This option is provided to allow queries to be posted to the DB2 Query Patroller server without requiring user intervention. This profile variable can be set to one of the following values:

- RUN

When the n-tier option is set to RUN, queries will be submitted to the DB2 Query Patroller server in a Submit and Wait mode. User intervention through the QueryEnabler tool is not required. The RUN value has a time out option: RUN: *timeout*. This option is in units of seconds. If this option is not set, QueryEnabler will wait until the query has been processed before returning control to the query application.

- CHECK

When the n-tier option is set to CHECK, it will behave similarly to RUN, but in this case, DB2 Query Patroller will check to determine if an answer set already exists for the query being submitted. If an answer set does exist, that result set will be returned, as opposed to the same query being executed again. The CHECK value has a time out option: CHECK: *timeout*. This option is in units of seconds. If this option is not set, QueryEnabler will wait until the query has been processed before returning control to the query application.

**Note:** The default value for the DQP\_NTIER variable is OFF.

## DQP\_RUNTIME

DQP\_RUNTIME profile variable is set to the path where the DB2 Query Patroller software is installed. This parameter is set during installation.

## DQP\_SHARE

The DQP\_SHARE profile variable only requires setting if the DQP\_NTIER profile variable is set. Otherwise, the option to share results sets is specified using the QueryEnabler interface. Set the value of this profile variable to Y to have the user's result tables granted public access. If the DQP\_NTIER profile variable is set, and DQP\_SHARE is not, only the submitting user and system administrator will have access to the user's result sets.

## DQP\_TRACEFILE

This profile variable specifies a file name for the QueryEnabler Java Trace file. Set this profile variable on the DB2 Query Patroller client.

---

## Windows Specific DB2 Profile Variables

The DB2 Query Patroller system is integrated with the operating system features of Windows and runs as a service. For this reason, the DB2 Query Patroller system on Windows has a slightly different implementation. The following profile variables are specific to DB2 Query Patroller running on a Windows operating system, and are set on the server.

### DQP\_MAPI\_PROFILE

The DQP\_MAPI\_PROFILE environment variable is set to the name of the mail profile that will be used when logging on to MAPI. The mail profile specifies the mail service that will be used to send e-mail notification. In the address book for the mail profile, list the addresses that mail should be sent to. The default address for mail notification must be set to `iwm`. Setting the DQP\_MAPI\_PROFILE variable is required for notification to work. If DQP\_MAPI\_PROFILE is not set, notification will be written to the DB2 Query Patroller log file. For more information about DB2 Query Patroller e-mail notification, see “Chapter 4. DB2 Query Patroller E-Mail Notification” on page 35.

### DQP\_MAPI\_PASSWORD

If the DQP\_MAPI\_PASSWORD profile variable is set, the password provided will be used to logon to the MAPI subsystem. It should not be necessary to set this variable when using Microsoft Exchange or Microsoft Internet Mail. For more information about DB2 Query Patroller e-mail notification, see “Chapter 4. DB2 Query Patroller E-Mail Notification” on page 35.

### DQP\_MAIL\_ACCOUNT

If the DQP\_MAIL\_ACCOUNT profile variable is set, DB2 Query Patroller will attempt to send mail under the account you specify. Use the following format to specify account information: *username, password*. Setting the DQP\_MAIL\_ACCOUNT environment variable is not required for Microsoft Exchange or Microsoft Internet Mail. For more information about DB2 Query Patroller e-mail notification, see “Chapter 4. DB2 Query Patroller E-Mail Notification” on page 35.

### DQP\_STARTUP

The DQP\_STARTUP profile variable must be set on each node. It can be set to `all`, `server`, or `agent`. The default setting, `all`, specifies that both the server and agent processes are to be started. The `server` option starts only the server processes. The `agent` option starts only the agent processes. Only one node per system should be started with the `all` or `server` option.

---

## Chapter 6. Performance Tuning

The following section provides information and outlines approaches for configuring the DB2 Query Patroller server for best performance.

There are no specific guidelines for setting the DB2 Query Patroller parameters because each data warehouse has its own set of unique attributes, such as size of the database, the complexity of the queries, the number of users supported, and the availability of hardware resources. This section describes an approach that can be taken to configure the DB2 Query Patroller server.

---

### Profiling

If feasible, profile the set of queries that will access the data warehouse through DB2 Query Patroller by submitting each query in isolation. This method provides valuable statistics, such as the cost estimate of the query, the time for the query to be executed, and the size of the answer set. Additional information, such as the behavior patterns of the end users and hardware resources, helps you determine the appropriate setting for DB2 Query Patroller configuration parameters.

---

### Estimating and Monitoring

Once you have profiled the set of queries that will access the database, the behavior patterns of end users, and the hardware resources, apply what you estimate to be the proper parameter configuration. In this initial phase, monitor the data warehouse during peak hours to determine if the data warehouse is being under utilized or is performing poorly in an attempt to process more queries than it can manage. If DB2 Query Patroller is configured too conservatively, the queries may not be submitted to the data warehouse even though resources are available. On the other hand, if DB2 Query Patroller configuration is too liberal, too many queries may be running against the data warehouse, causing the system to waste valuable resources paging or thrashing.

---

### Modifications

It is important to gather information from DB2 Query Patroller in order to determine the most effective configuration. The process of gathering this information must be performed systematically. Typically, you should modify only one parameter at a time. After each modification, observe the system to determine the specific impact of the modification.

---

## Configuring the Data Warehouse

DB2 Query Patroller stores valuable information in DB2 tables. This information can be accessed to help configure the data warehouse. Since the DB2 Query Patroller schema resides in the same DB2 database as the data warehouse, the data warehouse should be configured to account for the requirements of DB2 Query Patroller.

Run the following queries to measure the DB2 Query Patroller requirements. These queries should be executed against the DB2 Query Patroller schema directly through DB2 and not through the DB2 Query Patroller client interface:



To avoid having these queries intercepted by the QueryEnabler tool, configure your user account to have a high management threshold. This can be done using the User Administration page of the QueryAdministrator tool.

1. To determine the maximum number of queries currently under the management of DB2 Query Patroller and the cost of the queries:  

```
SELECT COUNT(*), SUM(STATIC_COST) FROM IWM003_JOB_TABLE WHERE JOB_STATUS='R'
```
2. To determine how many jobs are in the queue state waiting to run, but are held up due to a resource limitation:  

```
SELECT COUNT(*), SUM(STATIC_COST) FROM IWM003_JOB_TABLE WHERE JOB_STATUS='R' AND TIMESTAMP(DTIME_START_AFTER) < CURRENT_TIMESTAMP
```
3. To check the load on the DB2 Query Patroller nodes to determine their individual workloads:  

```
SELECT NODE_ID, SCHEDULED_JOBS, CPU_UTILIZATION FROM IWM003_NODE_TABLE WHERE NODE_STATUS='ACTIVE'
```

**Note:** The number of scheduled jobs includes both active jobs and jobs waiting to run.

You can run other queries against the DB2 Query Patroller schema to derive additional information. When accessing the DB2 Query Patroller schema, keep in mind that when DB2 Query Patroller is running, it is constantly accessing these tables to update status and retrieve information. Access to these tables must be shared with DB2 Query Patroller. If any of these tables are locked indefinitely, DB2 Query Patroller will appear to be hung as it waits for the database lock to be freed.

---

## Chapter 7. Configuring a DB2 Query Patroller Client

A typical DB2 Query Patroller client consists of two utilities. The QueryEnabler and the QueryMonitor. The QueryMonitor tool allows the user to manage queries, and the QueryEnabler tool is the component that intercepts submitted queries and places them under the control of the DB2 Query Patroller system. This chapter provides steps to help configure a DB2 Query Patroller client.

1. Before you can submit a query from a DB2 Query Patroller client workstation, you must have a user profile on the DB2 Query Patroller system.
2. The DYN\_QUERY\_MGMT database configuration parameter must be set to ENABLE on the DB2 Query Patroller server. Setting this parameter will allow the QueryEnabler component to intercept dynamically submitted queries.
3. There are several important parameters you must set when adding a user or group profile to the DB2 Query Patroller system. All of the parameters listed below will affect performance of the DB2 Query Patroller client:

### User Threshold

Provides the threshold, which if exceeded by a job, causes the job to be placed on hold. If a job is placed on hold, only a user with administrator or operator authority can release it. Be sure to set this parameter at a value high enough to accommodate the queries that a user will be submitting. User threshold is measured in timeron units. A timeron is a unit of measurement used to give a rough relative estimate of the resources, or cost, required by the database server to execute two plans for the same query. The resources calculated in the estimate include weighted CPU and I/O costs.

### Management Threshold

This value determines whether or not a dynamic SQL query will be intercepted by the QueryEnabler component. If the cost of the query does not exceed the set value, the query will not be intercepted by the QueryEnabler and will execute without user intervention. For example, if you want all queries with a cost greater than 10 to be intercepted by the QueryEnabler component for user intervention, set this parameter to a value of 10.

Query cost is measured in timeron units. A timeron is a unit of measurement used to give a rough relative estimate of the resources, or cost, required by the database server to execute two

plans for the same query. The resources calculated in the estimate include weighted CPU and I/O costs.

#### **Maximum Elapsed Time**

This option specifies the maximum number of seconds the query will be permitted to run against the DB2 Query Patroller server. If this value is set to 0 or -1 the query will always run to completion. If the value for this parameter is too small, queries will not be able to complete executing.

#### **Maximum Result Rows**

This option specifies the maximum number of rows that will be returned in the answer set. If this value is set to 0, the complete answer set will be returned. Ensure that this parameter is set to an appropriate value.

Other parameters you can configure for the DB2 Query Patroller client include Low, Normal, and High Priority, and a users e-mail address for notification. For a complete listing of User Profile Parameters, and for instructions on how to add a user to the DB2 Query Patroller system, see “User Administration” on page 17.

4. Update the Database manager configuration file for the client workstation. You must set values for the following parameters:
  - The JAVA\_HEAP\_SZ specifies the maximum size of the heap that is used by the Java interpreter. For non-partitioned database systems, one heap is allocated for the instance; for partitioned database systems, one heap is allocated for each database partition server. The DB2 Query Patroller GUI tools are Java applications.
  - The JDK11\_PATH parameter specifies the directory under which the Java Development Kit 1.1 is installed. The CLASSPATH and other environment variables used by the Java interpreter are computed from the value of this parameter. The DB2 Query Patroller GUI tools are Java applications.
5. DB2 Query Patroller profile variables provide a means to further configure a DB2 Query Patroller client. For instructions on how to set the following profile variables, see “DB2 Profile Variables” on page 39.

#### **DQP\_ABORTRESULT**

If set to Y, result sets that exceed the value provided for the Maximum Result Rows parameter as defined in the User profile table will be aborted. No result set will be returned. To define the Maximum Result Rows parameter for a user or group, see “User Administration” on page 17. If DQP\_ABORTRESULT is not set, the result set will be truncated once the row count reaches the defined number of rows specified by the Maximum Result Rows parameter.



## DQP\_LAST\_RESULT\_DEST

If DQP\_NTIER variable is set to RUN or CHECK, set DQP\_LAST\_RESULT\_DEST to the name of a valid result destination. Result destinations are defined using the Result Set Administration page of the QueryAdministrator tool. If the DQP\_NTIER variable is set, and DQP\_LAST\_RESULT\_DEST is not set, the result destination will be a table determined by DB2.

## DQP\_NTIER

This option is provided to allow queries to be posted to the DB2 Query Patroller server without requiring user intervention. It can be set to one of the following values:

- RUN

When the n-tier option is set to RUN, queries will be submitted to the DB2 Query Patroller server in a Submit and Wait mode. User intervention through the QueryEnabler tool is not required. The RUN value has a time out option: RUN: *timeout*. This option is in units of seconds. If this option is not set, QueryEnabler will wait until the query has been processed before returning control to the query application.

- CHECK

When the DQP\_NTIER profile variable is set to CHECK, it will behave similarly to RUN, but in this case, DB2 Query Patroller will check to determine if an answer set already exists for the query being submitted. If an answer set does exist, that result set will be returned, as opposed to the same query being executed again. The CHECK value has a time out option: CHECK: *timeout*. This option is in units of seconds. If this option is not set, QueryEnabler will wait until the query has been processed before returning control to the query application.

**Note:** The default value for the DQP\_NTIER variable is OFF.

## DQP\_SHARE

The DQP\_SHARE profile variable only requires setting if the DQP\_NTIER profile variable is set. Otherwise, the option to share results sets is specified using the QueryEnabler interface. Set the value of this profile variable to Y to have the user's result tables granted public access. If the DQP\_NTIER profile variable is set, and DQP\_SHARE is not, only the submitting user and system administrator will have access to the user's result sets.

## DQP\_TRACEFILE

This profile variable specifies a file name for the QueryEnabler Java Trace file. Set this profile variable on the DB2 Query Patroller client.

6. A server side DB2 Query Patroller profile variable that will have a direct affect on the performance of a DB2 Query Patroller client is the DQP\_INTERVAL profile variable. This profile variable specifies the number of seconds that each server and agent process sleeps between searches for work. For example, if the value for this variable is set to 20, a user may have to wait up to twenty 20 seconds plus the time it takes for a query to execute and return the result. Adjust this value according to your system and user requirements.

---

## Chapter 8. Managing Jobs

This chapter provides information to help you manage jobs on the DB2 Query Patroller system, and information about setting up a customer-written exit analysis program to execute prior to cost analysis being performed on a query.

---

### Releasing and Canceling Jobs

If a user submits a query to DB2 Query Patroller and DB2 Query Patroller estimates the job's cost to be greater than the user's limit as defined in the user's profile, then the job receives a hold status. In order for a job with a hold status to run, the system administrator or operator must release the job using QueryMonitor or the command line interface. While a job is in the DB2 Query Patroller system, the system administrator, operator, or the job's owner may cancel the job. A list of the jobs with a hold status or those jobs with some other status can be obtained through the QueryMonitor or command line interface. See the *DB2 Query Patroller User's Guide* for more information about viewing a job's status using the QueryMonitor tool. For more information about using the command line interface, see "Chapter 10. Using the Command Line Interface to Administer DB2 Query Patroller" on page 55.

If a user's jobs are regularly put on hold you may want to grant authority to submit more expensive jobs. You can grant a user or group this authority by adjusting the User Threshold parameter in the user or group profile. This task can be performed using the QueryAdministrator tool. For more information, see "User Administration" on page 17.

---

### Controlling Job Flow and Execution

You can control the number of jobs, where they run, and the load put on the database by altering the data source, job queue, and system parameters. See the following sections:

- "Data Source Administration" on page 22
- "Job Queue Administration" on page 30
- "System Administration" on page 32

If your system is not processing jobs fast enough, you may want to adjust the value of the DQP\_INTERVAL profile variable. Decreasing this value will decrease the amount of time between searches for work by the DB2 Query Patroller system. For more information, see "DB2 Profile Variables" on page 39.

To view statistics and reports for jobs that have run against your data source, use the Tracker tool. For more information, see “The Tracker Tool” on page 67. For information on performance tuning, see “Chapter 6. Performance Tuning” on page 45.

---

## Exit Analysis

DB2 Query Patroller can invoke a customer-written exit program just prior to performing cost analysis for a query. The exit program can examine the SQL statement and prevent DB2 Query Patroller from executing the statement. DB2 Query Patroller invokes the exit program for all jobs containing a SELECT, INSERT, UPDATE, or DELETE statement unless the user bypassed cost analysis.



The user can only bypass cost analysis if the system administrator sets the Cost Analysis parameter to Don't Do Cost Analysis in the user's profile. For more information, see “Creating a User Profile” on page 18.

DB2 Query Patroller invokes the exit analysis program if you set DQP\_EXIT\_AN to the path name of the executable program before starting the server processes. The exit program can be a shell script or any other executable. DB2 Query Patroller invokes the exit program under the iwm user ID. An SQL statement is passed in through STDIN. On UNIX, DB2 Query Patroller passes the job owner's user ID as the only argument to the exit. On Windows, additional arguments, *-9 handleNumber*, are passed and the exit program must close the passed handle before the SQL statement can be read from STDIN. If the exit routine terminates with an exit code of zero and without having written anything to STDOUT, DB2 Query Patroller continues to process the job normally. If the exit routine terminates with a non-zero exit code or writes anything to STDOUT, the job aborts. The job abort message includes the message written to STDOUT.



An example exit analysis program, *userexit\_NT.c*, is included on the DB2 Query Patroller CD-ROM in the *db2/samples/c* directory.

---

## Chapter 9. Replacing the DB2 Query Patroller Database

This chapter provides steps for replacing the DB2 Query Patroller Database. To replace the DB2 Query Patroller database, perform the following steps:

1. Log on to the system using as the administrative user `iwm`.
2. Set the default database for the instance to the name of the replacement database. This may be a new database you have created or an already existing database. You can set the default database using the following command:

```
db2set DB2DBDFT=default_database -i instance
```

Where *default\_database* is the name of the database, and *instance* is the name of the DB2 instance where the target database is defined. For additional information and the complete syntax of the `db2set` command, refer to the *Command Reference*.

3. You require a tablespace for the IWM schema. You can use an existing tablespace in the replacement database, or you can create a separate tablespace to hold the IWM schema. Refer to the *DB2 Administration Guide* for information on creating tablespaces.
4. You require a tablespace for the DB2 Query Patroller result tables. You can use an existing tablespace in the replacement database, or you can create a separate tablespace to hold the DB2 Query Patroller result tables. Refer to the *DB2 Administration Guide* for information on creating tablespaces.
5. Set the profile variable `DQP_RES_TBLSPC` to the result tablespace name.

```
db2set DQP_RES_TBLSPC=result_tablespace
```

Where *result\_tablespace* is the name of the result tablespace that will hold the DB2 Query Patroller result tables. For additional information and the complete syntax of the `db2set` command, refer to the *Command Reference*

6. Use the tool `db2_qp_schema.exe` to create the IWM schema and IWM control tables. `db2_qp_schema.exe` is located in the `DB2_RUNTIME\bin` directory, where `DB2_RUNTIME` is the DB2 installation path. From the `DB2_RUNTIME\bin` directory on Windows, or the `DB2_RUNTIME/bin` path on UNIX, enter the following command:

```
db2_qp_schema.exe iwm_schema.sql dsn iwm password tablespace
```

where *password* is the password for user `iwm`, *tablespace* is the name of the tablespace where the IWM schema will exist, and *dsn* is the name of the new database to be managed by DB2 query Patroller.

7. Create the IWM schema explain tables. In a DB2 command window, execute the following commands from the *DB2\_RUNTIME*\misc directory:

```
db2 connect to database user iwm using password
db2 -tvf EXPLAIN.DDL
db2 commit
```

where *database* is the name of the database you have added, and *password* is the password for the administrative user account, iwm.

8. Bind the DB2 Query Patroller bind files to the database. From the *DB2\_RUNTIME*\bnd directory on Windows, or the *DB2\_RUNTIME*/bnd path on UNIX, enter the following commands:

```
db2 connect to database user iwm using password
db2 bind @ db2qp.lst blocking all grant public
db2 bind @ db2qp_sp.lst
db2 commit
```

where *database* is the database the replacement database that will be managed by DB2 Query Patroller, and *password* is the password for the administrative user account, iwm.

---

## Chapter 10. Using the Command Line Interface to Administer DB2 Query Patroller

The command line interface for DB2 Query Patroller enables the system administrator to monitor and control DB2 Query Patroller from an operating system shell. The command line interface also enables end users to submit and monitor jobs from the system prompt. When combined with shell scripts or languages such as Perl, awk, and REXX, you can use the command line interface as an application programming interface (API) to DB2 Query Patroller.

The command line interface consists of the following two commands:

- Submit command (**iwm\_submit**)
- Control command (**iwm\_cmd**)

You can execute these commands on any machine with a TCP/IP connection to the server node.

---

### Command Line Interface Environment

To use the DB2 Query Patroller command line interface a user must have an entry in the User Profile table and the following profile variables must be defined. See “DB2 Profile Variables” on page 39 for more information.

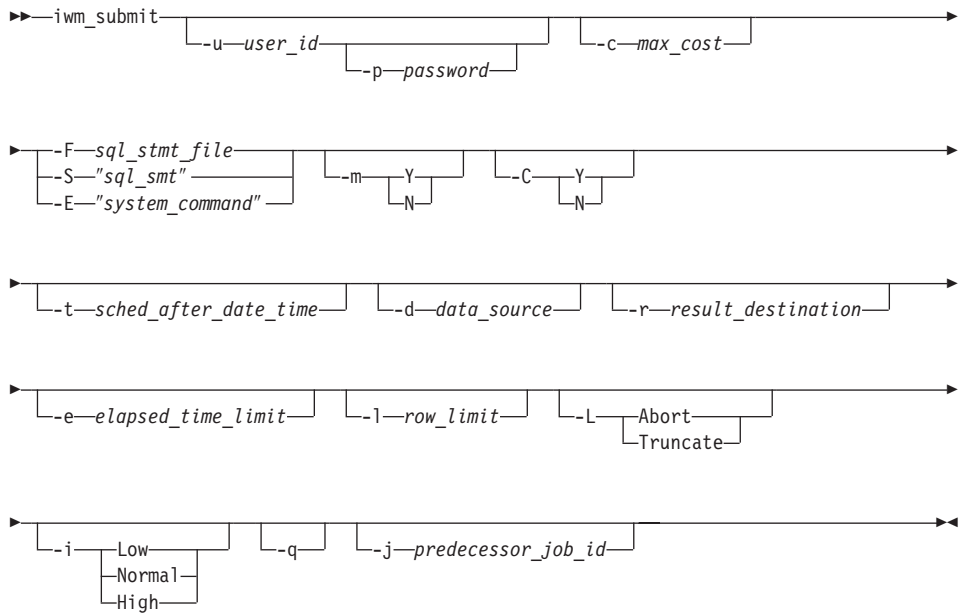
- **DQP\_RUNTIME** must be set to the path name into which DB2 Query Patroller was installed.
- **DQP\_NET** must be set to the TCP/IP address and port of the server node. Use the format *address:port*, where *address* represents either the dotted notation for the address or a name resolvable in the hosts file or through DNS, and *port* represents either the port number or a name resolvable in the services file.
- The default executable path on UNIX systems must contain *DQP\_RUNTIME/bin*. On Windows systems, the default executable path must contain *DQP\_RUNTIME\bin*.

---

### Submit Command (**iwm\_submit**)

The submit command (**iwm\_submit**) enables users to submit jobs to DB2 Query Patroller. You can use **iwm\_submit** to submit SQL commands and system commands.

The following diagram provides the syntax for the **iwm\_submit** command:



**Note:** When executing a SELECT statement against a lowercase table or column name using **iwmm\_submit**, the SELECT statement must be defined in an SQL statement file (*sql\_stmt\_file*) and submitted using the **-F** parameter. See the following table for more information on the **-F** parameter and SQL statement files.

Table 1. *iwmm\_submit* Parameter Descriptions

Parameter	Description
-u <i>user_id</i>	User identification. This parameter enables the submitting user to specify a different owner for the job. This gives the DB2 Query Patroller system administrator the ability to submit queries or jobs on behalf of end users. The user ID must be a valid system and database ID, and a registered DB2 Query Patroller user. The command <b>iwmm_submit</b> executes under the submitting user's environment. The user ID must conform to the system user identification syntax.  The job owner defaults to the submitting user.
-p <i>password</i>	User password. If you provide the -u parameter and the submitting user is neither root nor the DB2 Query Patroller system administrator, this password parameter must also be specified. The password must be the user's database password.



Table 1. *iwm\_submit* Parameter Descriptions (continued)

Parameter	Description
-c <i>max_cost</i>	<p>Cost threshold. This option enables the user to lower the cost threshold from what is defined in the DB2 Query Patroller user profile. Only root and the DB2 Query Patroller system administrator can increase the cost threshold with this option. The cost threshold must be a numeric value.</p> <p>By default, DB2 Query Patroller uses the cost threshold defined in the DB2 Query Patroller database.</p>
-F <i>sql_stmt_file</i>	<p>Specifies the name and path of a file that contains a single SQL statement. The <i>sql_stmt_file</i> must be a valid file path name. The file contents are immediately placed in the DB2 Query Patroller database.</p>
-S " <i>sql_stmt</i> "	<p>Specifies a single SQL command enclosed in quotes. The SQL statement is immediately placed in the DB2 Query Patroller database.</p>
-E " <i>system_command</i> "	<p>Specifies an entire system command including any optional arguments. The command is executed from the user's home directory, and its source directory must be available in the search path.</p>
-m {Y   N}	<p>Mail Notification. This parameter specifies whether or not the user should be notified through e-mail when the job completes or receives a hold status.</p> <p>By default, the user receives notification upon completion of the submitted job.</p>
-C {Y   N}	<p>Cost analysis override. If the user's DB2 Query Patroller profile allows queries to be submitted without cost analysis, this parameter determines whether the cost analysis should be performed. The option must be either Y or N. Y directs <b>iwm_submit</b> to perform cost analysis; N indicates no cost analysis.</p> <p>By default, DB2 Query Patroller performs cost analysis on SQL queries. DB2 Query Patroller ignores this option for system commands.</p>
-t <i>sched_after_date_time</i>	<p>Schedule after date/time. This parameter specifies the date and time after which the job will run. The parameter default is the present date and time.</p>
-d <i>data_source</i>	<p>Data source. Specifies the data source to run the job on. The only supported data source is IWM. This is the default value.</p>

Table 1. *iwm\_submit* Parameter Descriptions (continued)

Parameter	Description
-r <i>result_destination</i>	Result Destination. This parameter specifies the name of an alternate destination where results are sent instead of being written to a table in the database.
-e <i>elapsed_time_limit</i>	Elapsed time limit. This option enables the user to set a limit for the elapsed time the job will be permitted to run. If the time limit is exceeded, DB2 Query Patroller aborts the job. The time limit value is in seconds. By default, DB2 Query Patroller will not impose an elapsed time limit on the job.
-l <i>row_limit</i>	Result row limit. This option enables the user to set a limit for the number of rows returned by an SQL SELECT statement. If the row limit is exceeded, the action taken by DB2 Query Patroller depends on the setting of the -L parameter. By default, DB2 Query Patroller will not impose a result row limit.
-L {Abort   Truncate}	Result row limit action. This option enables the user to specify the action taken by DB2 Query Patroller should the job return more rows than the result row limit. If it is not specified or set to Abort, DB2 Query Patroller aborts the job. Otherwise, if it is set to Truncate, DB2 Query Patroller truncates the result table at the row limit.
-i {Low   Normal   High}	User Priority. This parameter enables the user to select the priority of the job. The priority value can be Low, Normal, or High. Only the first letter is significant. These values may also be specified as 0, 1, or 2. The user's DB2 Query Patroller profile defines the actual priority levels that correspond to these values.  By default, DB2 Query Patroller assigns the user's normal priority to the job.
-q	Quiet mode. This parameter directs DB2 Query Patroller to print only the job number upon successful job submission. This may be useful if the <b>iwm_submit</b> command is used from within a shell script.
-j <i>predecessor_job_id</i>	Predecessor job number. This parameter specifies a job number of a previously submitted DB2 Query Patroller job. The new job cannot be scheduled until the predecessor job completes.

---

## Control Command (**iwm\_cmd**)

The control command (**iwm\_cmd**) enables the system administrator to monitor and control DB2 Query Patroller from the operating system shell. It also enables end users to monitor their jobs.

Specifically, the system administrator and end user can use **iwm\_cmd** when:

- Monitoring jobs
- Controlling jobs
- Monitoring and controlling nodes
- Listing and controlling data sources
- Listing and controlling system parameters

### Common **iwm\_cmd** Parameters

Each function of **iwm\_cmd** has a different syntax. Parameters that are common to all functions are described in the following table. Following this are syntax diagrams and parameter descriptions for each **iwm\_cmd** function.

*Table 2. iwm\_cmd Common Parameter Descriptions*

Parameter	Description
<code>-u user_id</code>	User identification. This parameter provides either the DB2 Query Patroller user ID for the user whose jobs will be monitored or controlled, or the DB2 Query Patroller system administrator's ID ( <b>iwm</b> ).  By default, DB2 Query Patroller uses the current system user ID.
<code>-p password</code>	User password. If the user provides the <code>-u</code> parameter and the current user is not the DB2 Query Patroller system administrator, the password parameter must also be specified. The password must be the user's database password.

### Monitoring Jobs

**iwm\_cmd** allows users to monitor their jobs and the system administrator to monitor the status of all incomplete jobs. DB2 Query Patroller provides an interface to make it easy for programs invoking **iwm\_cmd** to process information about each job. If a user invokes **iwm\_cmd** without any parameters, all jobs for that user display.

Syntax:

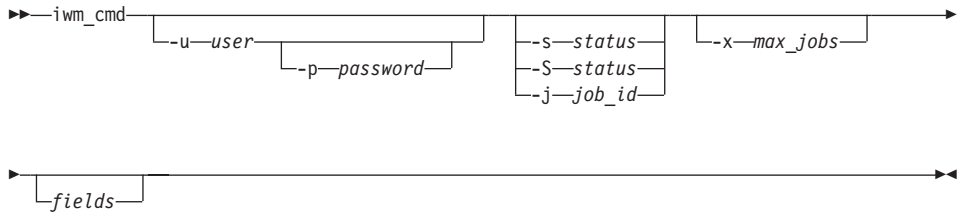


Table 3. iwmm\_cmd Parameters for Monitoring Jobs

Parameter	Description
-s status	<p>User job status. This parameter specifies that jobs belonging to the user with the given status should display. The status must be one of the following:</p> <ul style="list-style-type: none"> <li>• Q - Queued</li> <li>• S - Scheduled</li> <li>• H - Held</li> <li>• R - Running</li> <li>• U - Unfinished</li> <li>• D - Done</li> <li>• A - Aborted</li> <li>• C - Canceled</li> <li>• F - Finished</li> </ul> <p>Only the first letter is significant. It can be in either uppercase or lowercase. The unfinished status includes any queued, scheduled, held, or running jobs. The finished status includes any done, aborted, or canceled jobs.</p>
-S status	<p>Unfinished job status. This parameter specifies that all DB2 Query Patroller jobs with the given status should display. The status must be one of the following:</p> <ul style="list-style-type: none"> <li>• Q - Queued</li> <li>• S - Scheduled</li> <li>• H - Held</li> <li>• R - Running</li> <li>• U - Unfinished</li> </ul> <p>Only the first letter is significant. It can be in either uppercase or lowercase. The Unfinished status includes queued, scheduled, held, or running jobs. This parameter may only be used by the DB2 Query Patroller system administrator.</p>

Table 3. *iwm\_cmd* Parameters for Monitoring Jobs (continued)

Parameter	Description
<code>-j job_id</code>	Job identifier. This parameter specifies that only the requested job should display. If someone other than the DB2 Query Patroller system administrator uses this parameter, the user must be the owner of the job.
<code>-x max_jobs</code>	Maximum job count. This parameter specifies the maximum number of jobs to be listed. If this parameter is not specified, a maximum of 100 jobs display. If <i>max_jobs</i> contains the value 0, no limit is applied.
<code>fields</code>	Job fields. There are two job fields: Job and Status. You can enter these fields together or separately. This parameter is case sensitive. If you provide a field parameter, the jobs do not display in an easy-to-read format. Instead, the values of the fields are written to STDOUT in a format that other programs can easily process. A horizontal tab character separates each field from the next. Newline characters separate each job's fields from the next.

## Controlling Jobs

`iwm_cmd` allows users to cancel their jobs. It also allows the system administrator to cancel or release a job.

Syntax:

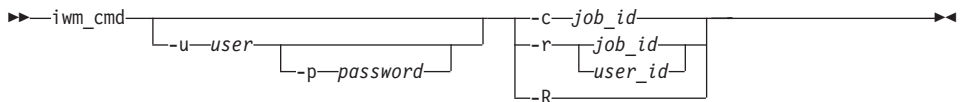


Table 4. *iwm\_cmd* Parameters for Controlling Jobs

Parameter	Description
<code>-c job_id</code>	Cancel a job. This parameter specifies that the given job should be canceled. A job need not have finished in order for it to be canceled. If the user is not the DB2 Query Patroller system administrator, the job must be owned by the user.
<code>-r job_id</code>	Release a job. This parameter specifies that the given job should be released. The job must currently have a status of held. This parameter may only be used by the DB2 Query Patroller system administrator.

Table 4. *iwm\_cmd* Parameters for Controlling Jobs (continued)

Parameter	Description
-r <i>user_id</i>	Release all jobs for a user. This parameter specifies that all jobs owned by a user with a current status of held should be released. This parameter may only be used by the DB2 Query Patroller system administrator.
-R	Release all jobs. This parameter specifies that all held jobs should be released. This parameter may only be used by the DB2 Query Patroller system administrator.

## Monitoring and Controlling Nodes

*iwm\_cmd* enables the DB2 Query Patroller system administrator to monitor and control the status and activity of agent nodes.

Syntax:

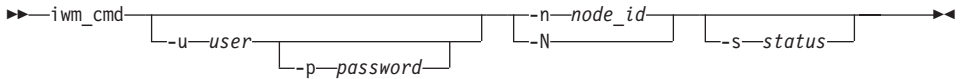


Table 5. *iwm\_cmd* Parameters for Monitoring and Controlling Nodes

Parameter	Description
-n <i>node_id</i>	Node identifier. This parameter specifies that the status of the given node should be displayed or be altered.
-N	All nodes. This parameter specifies that the status of all agent nodes should be displayed or be altered.

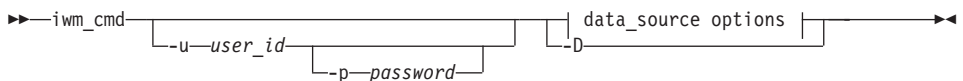
Table 5. *iwm\_cmd* Parameters for Monitoring and Controlling Nodes (continued)

Parameter	Description
-s <i>status</i>	<p>Node status. This parameter specifies that the status of the node(s) should be altered and provides the new requested status. The status must be from the following list. Only the first letter is significant. It can be either uppercase or lowercase.</p> <ul style="list-style-type: none"> <li>• Active. The server component can schedule new jobs to run on the node.</li> <li>• Quiesced. Jobs already scheduled to run on the node continue to run, but no more will be scheduled.</li> <li>• Inactive. Jobs already scheduled to run on the node continue to run, but no more will be scheduled to run on the node. After all previously scheduled jobs complete, the node manager process running on the node will terminate.</li> <li>• Force. The node manager process running on the node will terminate immediately without waiting for jobs already running on the node to complete.</li> </ul> <p><b>Note:</b> If <i>iwm_cmd</i> is used to terminate node manager processes, the server or agent component must be stopped and restarted in order to turn the node manager processes on again.</p>

## Listing and Controlling Data Sources

*iwm\_cmd* enables the DB2 Query Patroller system administrator to list and change the data source parameters for one or all data sources.

Syntax:



**data\_source options:**

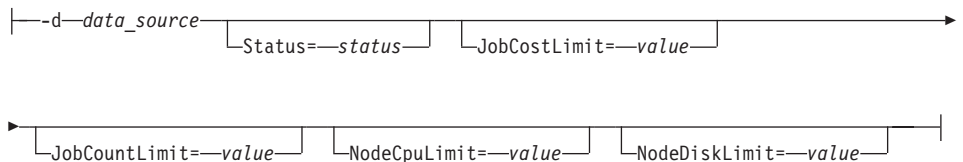


Table 6. *iwm\_cmd* Parameters for Listing and Controlling Data Sources

Parameter	Description
<code>-d data_source</code>	Data source name. This parameter specifies the name of the data source to be listed or altered. In the current release, the only valid value is IWM.
<code>Status=status</code>	Data source status. This parameter specifies that the status of the data source should be altered and provides the new status. The status must be one of the following: <ul style="list-style-type: none"> <li>• <b>Active.</b> Jobs will be scheduled normally.</li> <li>• <b>Hold.</b> All incoming jobs will be held, but any jobs already in the queue will be allowed to proceed.</li> <li>• <b>Inactive.</b> All requests to submit a new job will be rejected.</li> <li>• <b>ReleaseAll.</b> All jobs in the data source with a status of held will be released and the data source status will be set to active.</li> <li>• <b>HoldAll.</b> All jobs in the data source that are not yet running will be held. The status of the data source will be set to hold.</li> </ul>
<code>JobCostLimt=value</code>	Alter job cost limit. This parameter specifies that the limit of the total estimated cost of running jobs against this data source should be altered. The new value must be at least zero.
<code>JobCountLimt=value</code>	Alter job count limit. This parameter specifies that the limit of the total number of jobs running against this data source should be altered. The new value must be at least zero. If zero, scheduling of new jobs is disabled.
<code>NodeCpuLimt=value</code>	Alter node CPU limit. This parameter specifies that the per-node CPU utilization maximum for scheduling new jobs against this data source should be altered. The new value must be between 0 and 100. If CPU utilization is not being collected, the value of this parameter is ignored.
<code>NodeDiskLimit=value</code>	Alter node disk space limit. This parameter specifies that the per-node disk space minimum for scheduling new jobs against this data source should be altered. The value is in bytes and must be at least zero. If disk space utilization is not being collected, the value of this parameter is ignored.
<code>-D</code>	List data sources. This parameter specifies that summary information on all data sources should display.



## Listing and Controlling System Parameters

`iwm_cmd` enables the DB2 Query Patroller system administrator to list and change system parameters.

Syntax:

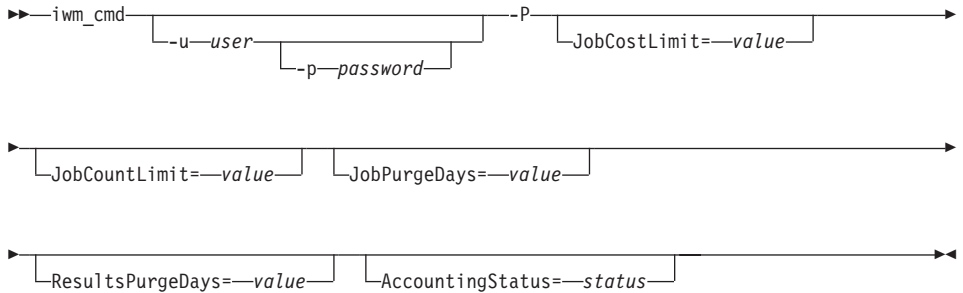


Table 7. `iwm_cmd` Parameters for Listing and Controlling System Parameters

Parameter	Description
-P	System parameters. This parameter specifies that the values of the DB2 Query Patroller system parameters should be displayed or altered.
JobCostLimit= <i>value</i>	Alter job cost limit. This parameter specifies that the limit of the total estimated cost of running jobs should be altered. The new value must be at least zero.
JobCountLimit= <i>value</i>	Alter job count limit. This parameter specifies that the limit of the total number of running jobs should be altered. The new value must be at least zero. If zero, scheduling of new jobs is disabled.
JobPurgeDays= <i>value</i>	Alter job information purging. This parameter specifies that the purge age for DB2 Query Patroller jobs should be altered. The new value must be between 0 and 999. If zero, automatic purging of job information is disabled. If not zero, the value of JobPurgeDays must be equal to or greater than the value of ResultsPurgeDays.
ResultsPurgeDays= <i>value</i>	Alter result table purging. This parameter specifies that the purge age for result tables should be altered. The new value must be between 0 and 999. If zero, automatic purging of result tables information is disabled. If the value of JobPurgeDays does not equal zero, it must be equal to or greater than the value of ResultsPurgeDays.

Table 7. *iwm\_cmd* Parameters for Listing and Controlling System Parameters (continued)

Parameter	Description
AccountingStatus= <i>status</i>	Alter job accounting status. This parameter specifies that the status of the DB2 Query Patroller job accounting feature should be altered. The status must be either active or inactive. If Active, DB2 Query Patroller inserts a row into the job accounting table as each job completes, aborts, or is canceled.

---

## Chapter 11. Monitoring the DB2 Query Patroller System

Monitoring the performance of DB2 Query Patroller is done using the Job Accounting table and the Tracker tool. Statistics are gathered and stored in the Job accounting table, and those statistics are used by the Tracker tool to produce reports. This chapter provides instructions for using the Tracker tool and detailed information for the Job Accounting table.

---

### The Tracker Tool

The Tracker tool provides reports that display a database usage history for queries that have been managed by the DB2 Query Patroller system. For example, Tracker allows you to determine which tables and columns have been accessed most frequently, which tables have returned the most result rows, and which jobs have been completed within a specific time period.

#### Creating Tracker Data on the Database Server (iwm\_tracker)

The information displayed by Tracker is gathered by DB2 Query Patroller running on your database server. DB2 Query Patroller collects job accounting data, which is then analyzed by the `iwm_tracker` program. The `iwm_tracker` program is the back-end parser for DB2 Query Patroller, which analyzes data for successfully completed job runs. Each time the `iwm_tracker` program runs, it examines any job accounting entries added since its last run.

**Note:** Because the user `iwm` owns the DB2 explain tables, you must be logged on as the user `iwm` to run `iwm_tracker` successfully. The Tracker tool uses data from DB2 explain tables to create reports.

Before running Tracker for the first time, you must perform the following steps:

1. In DB2 Query Patroller, set the Accounting Status system parameter to Write to Table. This task can be performed using the System Administration page of the QueryAdministrator tool. This must be completed before jobs are submitted. Information will be gathered for the Tracker tool only after the Accounting Status system parameter is set.
2. Determine when to run the `iwm_tracker` program. Choose a time period that is appropriate for the query volume and usage pattern of your database. You might want to run `iwm_tracker` at a time when the database load is light. Run the `iwm_tracker` program at least once to ensure that data is available the first time you run the Tracker tool.

## Tracker Reports

This section describes the different reports that can be viewed using the Tracker tool.

### Understanding the Reports

There are four basic types of predefined reports that are generated when the Tracker tool is run:

- *Total Table Hits*. This report tells you which tables were hit and how often, and how many rows were returned from each table during a particular time period.
- *Columns Hit in Table X*. This report tells you how often the different columns in a table were hit. You can also view this data by particular users, telling you how often a particular column in a table was accessed by a particular user.
- *Users that Hit Table X*. This report tells you how many times a particular user hits a particular table. You can also view a report detailing the number of times a particular user hits a particular column in a table.
- *Job Activity over Time*. This report gives you information about the levels of job activity over a defined period of time.

### Total Table Hits Report

The Total Table Hits report identifies the tables that have been accessed by queries submitted against the database within the defined date range. It tells you the number of times each table has been accessed, and the number of rows that have been returned from each table. This type of report may reveal certain tables to be heavily accessed, making them good candidates for being located on rapid-access devices.

For each table hit, the Total Table Hits report gives you the following information:

- The table name.
- The number of hits on the table during the specified time period.
- The percentage of hits. This represents the proportion of hits on a table relative to the number of hits on all the tables.
- The number of rows returned. This represents the total number of rows returned from the table for all queries within the defined date range.
- The percentage of rows returned. This represents the proportion of rows returned from a particular table relative to the number of rows returned from all tables during the defined time period.

An example of a Total Table Hits report is given below. This report shows the number of hits on the table IWM003\_JOB\_ACCT and the number of rows

returned from that table.

Table	Number of hits	Percentage of hits	Number of rows	Percentage of rows
IWM003_JOB...	62	63.27	52	10.66
IWM003_TUN...	13	13.27	195	39.96
SYSCOLUMNS	5	5.1	0	0
SYSTABLES	18	18.37	241	49.39

### Columns Hit in Table X Report

The Columns Hit in Table X report lists each column in a specific table and the number of times the column has been accessed by queries within the defined date range. The report also shows the percentage of hits per column relative to the hits on all columns in the table. This type of report displays the columns in a table that are queried most frequently. To optimize access, you may want to create indexes for the most frequently accessed columns.

An example of a Columns Hit in Table X report is given below. This report displays the number of hits on each of the columns in the table IWM003\_JOB\_ACCT.

Column	Number of hits	Percentage of hits
ACCOUNT_ID	4	6.45
DATA_SOURCE	3	4.84
ELAPSED_TIME	3	4.84
COMPLETION_STA...	2	3.23
DTIME_COMPLET...	2	3.23
DTIME_CREATED	2	3.23

### Users That Hit Table X Report

The Users That Hit Table X report lists the ID of each user accessing a specific table and the number of times the table is accessed by that user within the defined date range. The report also shows the percentage of hits on the table by a user relative to hits by all users. You can examine the details of a particular user's access to a table by viewing the Columns Hit by User Y Hitting Table X report.

This type of report might reveal trends of heavy resource use by certain individuals, perhaps indicating a need for stricter user constraints.

**Columns Hit by User Y Hitting Table X Report:** The Columns Hit by User Y Hitting Table X report lists the columns accessed by a specific user for a specific table and the number of times those columns were accessed within the defined date range. The report also shows the percentage of times each column was accessed by a specific user relative to the total hits on all columns in that table by that user.

An example of a Columns Hit by User Y in Table X Report is given below. This report displays a list of columns that were hit by the user IWM in the table IWM003\_JOB\_ACCT. For each column, the report displays the number of times it was accessed by the user IWM.

Column	Number of hits	Percentage of hits
ACCOUNT_ID	4	6.45
DATA_SOURCE	3	4.84
ELAPSED_TIME	3	4.84
COMPLETION_STA...	2	3.23
DTIME_COMPLET...	2	3.23
DTIME_CREATED	2	3.23

## Job Activity over Time Report

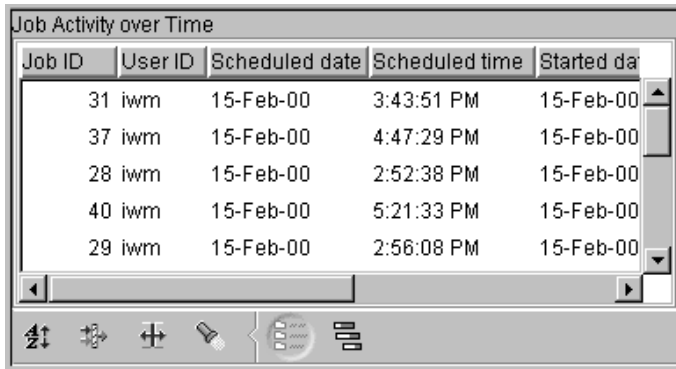
The Job Activity over Time report displays all the jobs that were completed between the defined start and end dates. This type of report might reveal job trends, such as periods of heavy usage or slow response times. This type of information might point to a need to reallocate resources during high-demand periods, or to reschedule non-critical jobs to alleviate some of the database workload.

For each job, the report provides the following information:

- The job ID.
- The user ID.
- The date and time the job was scheduled, started, and completed.

Additional job detail information, such as execution cost, the number of result rows, result set destination, and the SQL statement, can be viewed by double-clicking on a specific job in the list to open the Job Detail Information window.

An example of a Job Activity over Time Report is given below. This report displays the details of jobs executed from 1:00 pm to 6:00 pm on February 15, 2000.



Job ID	User ID	Scheduled date	Scheduled time	Started da
31	iwm	15-Feb-00	3:43:51 PM	15-Feb-00
37	iwm	15-Feb-00	4:47:29 PM	15-Feb-00
28	iwm	15-Feb-00	2:52:38 PM	15-Feb-00
40	iwm	15-Feb-00	5:21:33 PM	15-Feb-00
29	iwm	15-Feb-00	2:56:08 PM	15-Feb-00

## Using Tracker

This section describes how to start and stop Tracker and how to use Tracker to display job history data.

### Starting Tracker

Complete the following steps to start the Tracker tool:

1. On Windows, you can select **DB2 Query Patroller** → **Tracker** from the **IBM DB2** program group. On Windows or UNIX you can, enter the following command in a DB2 or system command window:

```
db2track
```

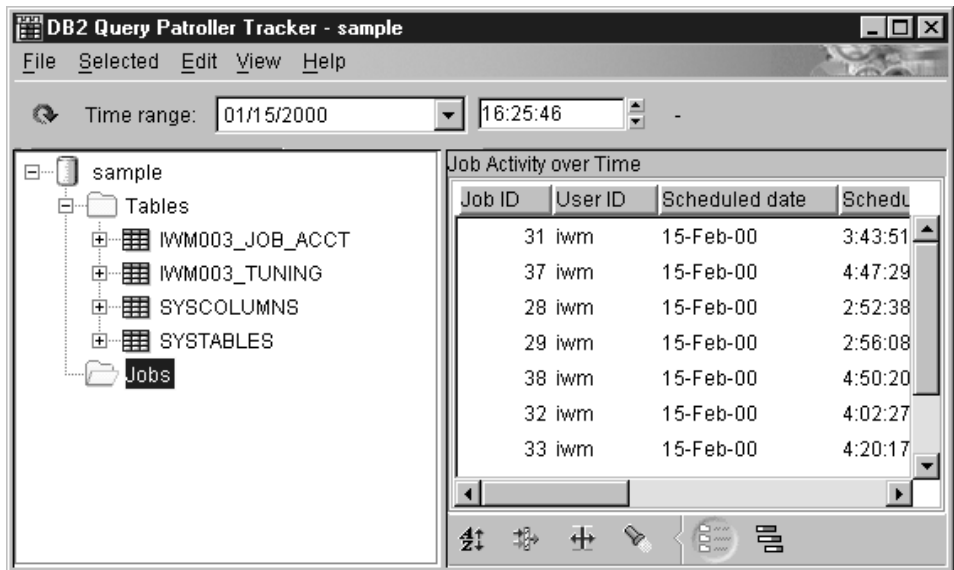
The Tracker Login window opens.

2. Type your user ID, password, and database alias, and click on **OK**. The Tracker report window opens.

**Note:** The user ID and password are case sensitive.

### The Tracker Window

The Tracker Window is illustrated below:



There are four major areas in the Tracker window. At the top of the window is the menu bar, consisting of drop-down menus of actions to perform within the Tracker tool. Below the menu bar is the Time Range settings area. In this area, you can set the time range for Tracker reports. For information on changing the date and time range, see “Changing the Date and Time Range” on page 80.

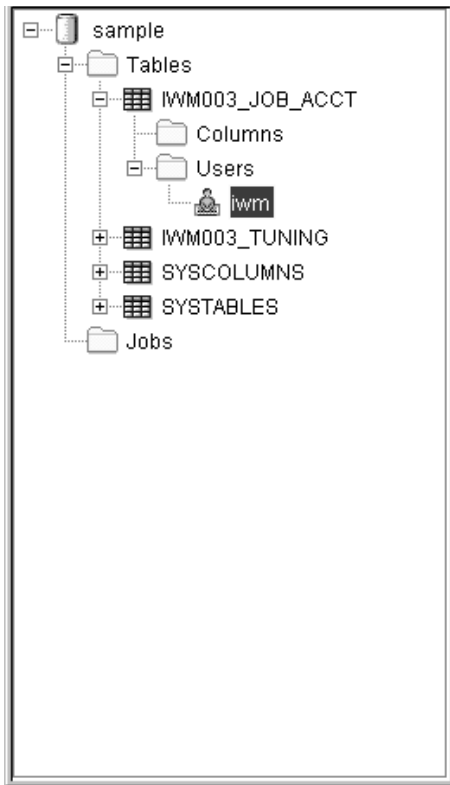
**Note:** If you reduce the size of the Tracker window, the settings for the end of the time range disappear, as shown in the example above. To view these settings, simply increase the size of the Tracker window.

The remainder of the Tracker window is split into two areas. On the left side is a navigation tree for selecting reports to view. The area on the right side of the screen displays the selected report. The report that is displayed when you open the Tracker tool is the Total Table Hits report. For information on viewing other reports, see “Using the Tracker Report Navigator”.

**Using the Tracker Report Navigator:** On the left side of the Tracker window is an area containing a directory tree of folders and reports. An example



directory is illustrated below:



At the top of the directory tree is a database icon representing the database that was used in the tracked queries. Below this icon is a folder labeled Tables and a folder labeled Jobs. When each of these folders is highlighted, the folder icon is open, and the corresponding report is displayed in the report area on the right of the Tracker window. A + sign next to an icon indicates that the directory may be expanded to reveal dependents. To expand the directory, click on the +.

When you click on the + next to the Tables folder icon, you see a list of all the tables that have been accessed during the specified time range. Clicking on the icon for each table reveals a short report in the report area of the Tracker window. This report tells you how many times that table was hit during the time range, and what percentage those hits represent relative to the number of hits on all the tables.

If you click on the + next to a particular table icon, two folders appear below that icon: a Columns folder and a Users folder.

If you click on the Columns folder under a table icon, the Columns Hit in Table X report appears in the report area of the Tracker window. For more information on this type of report, see “Columns Hit in Table X Report” on page 69.





If you click on the Users folder under a table icon, the Users that Hit Table X appears in the report area of the Tracker window. For more information on this type of report, see “Users That Hit Table X Report” on page 69. To expand this directory, click on the + next to the Users folder. Below the Users folder for a particular table is an icon for every user that has accessed that table. Clicking on the icon for User Y reveals a Columns Hit by User Y in Table X report in the report area of the Tracker window.



The Jobs folder has no dependent items so there is no + next to it. If you click on the Jobs folder, the Job Activity over Time report appears in the report area of the Tracker window. For more information on this type of report, see “Job Activity over Time Report” on page 70.

### Customizing the Report Display

The Tracker application allows you to customize the display of reports in several different ways. You can sort the items in the report by any column, apply filters to reduce the scope of the report, and select a subset of columns to display. You can also choose to view the report as a chart or as a graph.

At the bottom of the report area in the Tracker window is a menu of icons that allow you to customize the view of the report. The icons are described in the table below:

	Sort. This icon allows you to sort by different columns in ascending or descending order.
	Filter. This icon allows you to apply a filter to the report and display only the data meeting specified criteria.
	Customize Columns. This icon allows you to display only selected columns of the report.
	Find. This icon allows you to find a specified string in the report.

	View Details. This icon allows you to view the report in a detailed chart.
	View Graph. This icon allows you to view the report as a graph.

**Sorting Report Data:** You can sort a Tracker report when it is displayed in chart view. To do a customized sort on a report, perform the following steps:

1. Click on the **Sort** icon at the bottom of the report area in the Tracker window. The Sort dialog window appears.
2. From the Available columns area of the window, select the column or columns you wish to sort by. The selected columns are highlighted.
3. Click the > button to move the selected column to the Sort columns area of the Sort dialog window. To move the entire list of columns to the Sort columns area, click the >> button.
4. By default, the sort order is ascending. To sort by a column in descending order, select the desired column in the Sort columns area and click the **Descending** radio button. The arrow to the left of the columns in the Sort columns area indicates whether the sort order for that column is ascending or descending. Ascending order is indicated by an arrow pointing upward, and descending is indicated by an arrow pointing downward.
5. The report is sorted first according to the first item in the Sort columns list, second according to the second item, and so on. To change the order of the columns in this list, use the **Move Up** and **Move Down** buttons. To move an item up in the list, select that item and click the **Move Up** button. The item will move up one place in the list.
6. When you have finished selecting the settings for the sort, click **OK**. The Sort dialog window closes and the report in the Tracker window appears in the specified order.

**Applying Filters to Reports:** You can customize reports produced by the Tracker tool by applying filters to the report contents. To apply a filter to a report, perform the following steps:

1. Click on the **Filter** icon at the bottom of the report area of the Tracker window. The Filter window opens, displaying a list of all the columns.
2. For the columns you want to apply a filter to, specify a value to use for the filter and the operator that you want applied to the value. The possible operators in the Comparison column are: In, Not in, Equal to, and Not equal. When specifying more than one value, separate the different values with a space.

3. When you are finished setting all of the filters to all of the targeted columns, click **OK**.

**Customizing Columns in Reports:** You can customize the reports generated in the Tracker tool by selecting the columns that appear in the report, and specifying the order they appear in. To customize the columns, perform the following steps:

1. Click the **Customize Columns** icon. The Customize Columns window opens.
2. Select columns that you want displayed from the Available columns area. Click the > button to move the selected column to the Displayed columns area. To move the entire list of columns to the Displayed columns area, click the >> button.
3. The columns are listed in the Displayed columns area in the order that they appear in the report. To change the order of a column, select that column in the Displayed columns area. Click the **Move Up** or **Move Down** buttons to position the column correctly.
4. When you have finished selecting the columns and setting the order that you want them in, click **OK**. The report appears with the selected columns and ordering.

**Finding Strings in Reports:** To find a string in a report, perform the following steps:

1. Click on the **Find** icon. The Find window appears.
2. Type the text that you are searching for in the Find string field.
3. If you want your search to be case-sensitive, click the Case sensitive check box.
4. Click **OK**. If the specified string is found in the report, the row containing the first occurrence of that string is highlighted.

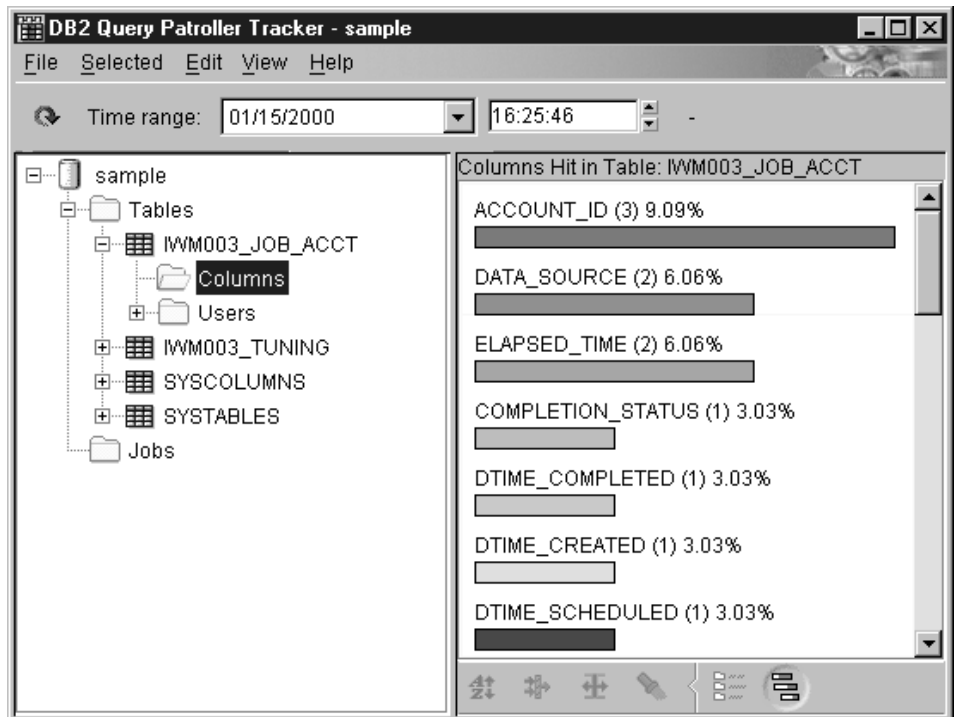
**Viewing Reports in Chart Format:** To view a report in chart format, click the **View Details** icon at the bottom of the report area in the Tracker window. You can also use the menu bar at the top of the window, selecting **View** → **By Details**. The report appears as a set of row items and columns.

In this format, you can change the sort order for a column from ascending to descending by clicking on the title area of the column. Clicking once on the column title sorts the data by that column in ascending order. An arrow pointing upward appears next to the column title.

Clicking a second time sorts the data by that column in descending order. An arrow pointing downward appears next to the column title. A third click returns the order to the default order.

**Viewing Reports in Graph Format:** To view a report in graph format, click the **View Graph** icon at the bottom of the report area in the Tracker window. You can also use the menu bar at the top of the window by selecting **View** → **By Graph**. All the reports appear as bar graphs, except the Job Activity over Time report. The graph format of this type of report is discussed in “Displaying Job History Data” on page 78.

An example of a report in graph format is illustrated below:



The graph reports represent the proportion of hits recorded for a particular table, column, or user relative to other hits in the same category. For example, the graph report for Columns Hit in Table X shows the relative proportion of hits received by each column in Table X. For each column, the name of the column, the number of hits, and the percentage of hits overall is displayed. The length of each bar represents the proportion of hits on the column relative to the hits on all the columns in the table.

Where there are two types of data represented in the table, you can choose which figures to represent in the graph format report. For example, the Total Table Hits report contains both the number and percentage of hits on each table, and the number and percentage of rows returned from each table. When you select the graph format for this report, a drop-down menu appears next

to the **View Graph** icon at the bottom of the report area in the Tracker window. The menu gives you the option of viewing the Number of hits data or the Number of rows returned in graph format.

When viewing reports in graph format, you cannot perform the same customizations that you can perform when viewing reports in chart format. You cannot sort or apply filters to the data or customize the columns. If you perform any of these customizations in chart format and then switch to viewing the report in graph format, these customizations do not appear. If you return to viewing the report in chart format, the customizations reappear.

The find function is also only available when viewing a report in chart view.

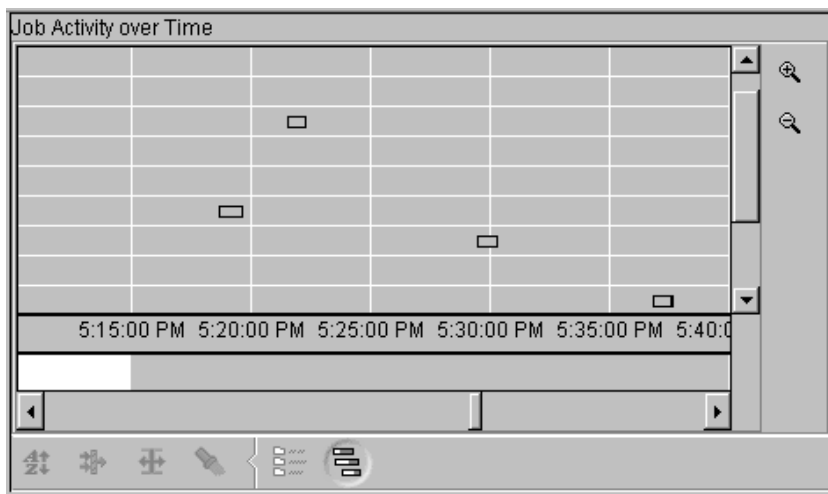
The **Sort, Filter, Customize Columns, Find** icons are inactive when viewing reports in graph format.

### Displaying Job History Data

Tracker allows you to view job history data so that you can tune the database for heavily used tables and columns. For a description of each report available in Tracker see “Tracker Reports” on page 68.

The Job Activity over Time report allows you to view job activity data in a detailed chart format or in a easy-to-read graph. When viewing this type of report in the detailed chart format, you may customize the report presentation as described in “Customizing the Report Display” on page 74.

An example of a graph view of a Job Activity over Time report is illustrated below:



Individual jobs scheduled and completed during the specified time range are plotted in the grid area that makes up the greater part of the graph. Each row in the grid represents a job. To see all of the jobs scheduled during the time range, you may need to scroll down using the scroll bar control to the right of the grid area. Hover help provides you with the job ID associated with the row.

Additional job detail information, such as execution cost, the number of result rows, result set destination, and the SQL statement, can be viewed by double-clicking on a specific job row to open the Job Detail Information window.

The grid represents time from left to right. The time intervals are displayed along the bottom of the grid. The timing of each scheduled job is represented by a bar in the row associated with that job. The position and size of the bar tells you when the job was scheduled and how long it took to complete. For example, a job that began at 3:00 pm on February 15th and completed at 4:00 pm the same day is represented with a bar that spans the space between 3:00 and 4:00 on the time line.

The bar that represents a job may have a transparent portion and a solid portion. The transparent portion represents the time from when the job is scheduled to the time it actually runs. The solid portion represents the time that the query is actually executing.

Below the grid is a strip with shaded segments corresponding to the time intervals on the grid. The relative intensity of the shading in a segment represents the relative level of activity during that time interval: the darker the shade, the more jobs that were scheduled during that interval. The shading represents a relative measure: the interval associated with the greatest proportion of jobs will have the darkest shading. You can move your cursor over a particular segment of the strip to display the number of jobs scheduled in that interval.

**Zooming In and Out on a Job Activity Graph:** When you first open a Job Activity over Time report to view in graph format, the graph displays the entire date range specified at the top of the Tracker window. This determines the size of the time intervals displayed on the graph. For example, if the specified date range spans 10 days, the time intervals shown at the bottom of the graph are one day. You can change the scale of the graph, increasing or decreasing the size of the time intervals by using the magnifying glass icons to the right of the graph. You can also use the **Zoom In** and **Zoom Out** items under **View** on the menu at the top of the application window.

To view a more detailed graph of the job activity for a smaller period of time, you can zoom in by clicking on the magnifying glass icon with the + sign in it

or by double-clicking on the time interval and job activity level strips at the bottom of the graph. This will decrease the size of the time intervals on the graph, and provide a more detailed representation of job activity distribution. You can continue to decrease the size of the time intervals by clicking repeatedly on the icon. You can also use the menu bar at the top of the window to zoom in on a report by selecting **View** —> **Zoom In**. Double clicking on the time interval strip or the shaded job activity concentration strip will also allow you to zoom in.

Zooming in decreases the number of days displayed in the report. You may find that, as you zoom in closer, the job distribution spreads out and you need to use the horizontal scroll bar at the bottom of the graph to view all of the job activity for that time period.

When the graph is displaying hour time intervals, the dates are no longer visible at the bottom of the graph. To see the date represented in a particular view of the graph, move your cursor over the time interval in question. The date that is represented appears.

As you zoom in on a particular report, the time period represented in the graph decreases in size. However, the date range of the entire report remains the same, as shown at the top of the Tracker window.

If you wish to view a time period that is not shown when you zoom in closely on a report, you can adjust the date range for the report to reduce the scope of the report. For example, if you want to view the job activity during a particular day, you can change the date range (see “Changing the Date and Time Range”) to focus on that day.

To Zoom Out, click on the magnifying glass icon with the - sign in it. You can also use the menu bar at the top of the window to zoom out on a report by selecting **View** —> **Zoom Out**.

### **Changing the Date and Time Range**

A date range exists for an entire Tracker session, which means that each new report opened during that session uses the same specified date range. Therefore, if you change the date range for a report, it resets the date range for the next report you open within the session.

When you open the Tracker tool, the default date range is set to one month, beginning one month before the session started and ending at the date and time the Tracker session started. The report that displays when you open the application reflects this default time range. To generate a set of reports covering a different time range, change the time range as described below.



If you change the time range in a Job Activity over Time report, the scale of the displayed data may also change. For example, if the date range is less than 30 days, the increments displayed in the Job Activity over Time report are days. If the date range is greater than a year, the increments displayed in the Job Activity over Time report are years. If the date range is 24 hours or less, the increments displayed in the Job Activity over Time report are hours. See “Displaying Job History Data” on page 78 for information on the Job Activity over Time report display features.

**Setting the Date Range:** There are two ways to change the date range for a session. You can either type over the existing date in the date field at the top of the Tracker window using the format MM/DD/YYYY, or you can use the drop-down calendar tool. To use this tool, perform the following steps:

1. Click on the drop-down arrow next to the date field. The calendar appears. The month that appears is the month of the current setting. The currently specified date is highlighted in the month.
2. To select a date within the month displayed, click on the date. The calendar disappears.
3. To select a date in a later month, click on the >> button to advance to another month. To select a date in a previous month, click on the << button to display earlier months.
4. Select a date within the month by clicking on the desired day.
5. After you have finished setting the date and time range for the Tracker session, refresh the screen by clicking on the refresh icon to the left of the Time Range display.

**Note:** For new data to be available to display, `iwm_tracker` (the back-end process for Tracker) must have been run to analyze any job accounting data that accrued since its last run.

**Setting the Time Range:** To set the time boundaries for a Tracker session, you can either type over the existing time in the time field at the top of the Tracker window or you can use the up and down arrows next to the field as follows:

1. Highlight the hour in the time field at the top of the Tracker window. Use the up and down arrows at the right of the time field to set the desired hour.

**Note:** The time is set according to a 24 hour clock. This means that 1:00 pm is represented as 13:00.

2. Click on the minute portion of the time display field. Use the up and down arrows at the right of the time field to set the desired minute.
3. Click on the second portion of the time display field. Use the up and down arrows at the right of the field to set the desired second.

4. After you have finished setting the date and time range for the Tracker session, refresh the screen by clicking on the refresh icon to the left of the Time Range display.

**Note:** For new data to be available to display, `iwm_tracker` (the back-end process for Tracker) must have been run to analyze any job accounting data that accrued since its last run.

### **Exiting Tracker**

To exit Tracker, select **File** → **Exit** or click on the **Close** button in the upper right hand corner of the application window.

---

## **Job Accounting**

This section provides information about the job accounting process and the columns in the Job Accounting table.

If you set the Accounting Status system parameter to `Write to Table`, the notifier component adds a row to the Job Accounting table each time a job completes.

This information is used by the Tracker tool to provide reports that display a database usage history for queries managed by the DB2 Query Patroller system. For example, this information will be used by Tracker to create reports showing which tables and columns have been accessed most frequently, which tables have returned the most result rows, and which jobs have been completed within a specific time frame.

### **Job Accounting Table Columns**

The following list explains each of the columns in the Job Accounting table.

#### **ACCOUNT\_ID**

The `ACCOUNT_ID` column represents the accounting identifier associated with the user from the User Profile table.

#### **COMPLETION\_STATUS**

The `COMPLETION_STATUS` column contains a single letter code indicating how the job completed. Values are `D` for done (normal completion), `C` for canceled, or `A` for aborted.

#### **DATA\_SOURCE**

The `DATA_SOURCE` column identifies the database instance against which the query was run. Currently, the only supported data source identifier is `IWM`.

#### **DTIME\_COMPLETED**

The `DTIME_COMPLETED` column identifies the date and time that the job was completed.

**DTIME\_CREATED**

The DTIME\_CREATED column identifies the date and time that the user submitted the job to DB2 Query Patroller.

**DTIME\_SCHEDULED**

The DTIME\_SCHEDULED column identifies the date and time after which the user scheduled the job to be run.

**DTIME\_STARTED**

The DTIME\_STARTED column identifies the date and time that the job started running. If the job was canceled or aborted before the job started running, this column contains a NULL value.

**ELAPSED\_TIME**

The ELAPSED\_TIME column identifies the elapsed time in seconds from when the job started running until it completed.

**ELAPSED\_TIME\_LIMIT**

The ELAPSED\_TIME\_LIMIT column identifies the maximum number of elapsed seconds a job will be permitted to run.

**ERROR\_REASON**

The ERROR\_REASON column identifies the reason why the job did not complete successfully.

**ESTIMATED\_COST**

The ESTIMATED\_COST column identifies the estimated cost for the job in accounting units, if one was derived. This column contains the value of the estimated cost for the job multiplied by the cost factor from the Data Source table.

**JOB\_ID**

The JOB\_ID column contains a number that identifies the job.

**JOB\_PRIORITY**

The JOB\_PRIORITY column contains the priority level number assigned to the job.

**JOB\_TEXT**

The JOB\_TEXT column contains either the job's SQL statement or its system command.

**JOB\_TYPE**

The JOB\_TYPE column contains either the letter S for SQL jobs or the letter C for jobs containing system commands.

**NODE\_ID\_EXECUTED**

The NODE\_ID\_EXECUTED column identifies the host name of the node where the job was run.

**NODE\_ID\_SUBMITTED**

The `NODE_ID_SUBMITTED` column identifies the node from which the job was submitted.

**PREDECESSOR\_JOB**

If the job was assigned a predecessor job (a job which must have completed before this job could run), the `PREDECESSOR_JOB` column contains the predecessor's job identifier. If no predecessor was assigned, this column contains a NULL value.

**RESULT\_DESTINATION**

If the user selected an alternative result destination, the `RESULT_DESTINATION` column contains the identifier of the destination. If the results were to be written to a result table created in the database, this column contains a NULL value.

**RESULT\_ROWS**

The `RESULT_ROWS` column contains the number of database rows that a job selected, updated, deleted, or inserted.

**RESULT\_ROWS\_LIMIT**

The `RESULT_ROWS_LIMIT` column identifies the maximum number of rows a SQL `SELECT` statement is allowed to return.

**RESULT\_TABLE\_OWNER**

The `RESULT_TABLE_OWNER` column contains the database identifier of the owner of the result table, if it was created.

**RESULT\_TABLE\_NAME**

The `RESULT_TABLE_NAME` column contains the table name of the result table, if it was created.

**SUBMIT\_SOURCE**

The `SUBMIT_SOURCE` column contains the name of the program that submitted the job.

**SUBMITTER\_USER\_ID**

The `SUBMITTER_USER_ID` column contains the system identifier of the submitter of the job. This identifier could be different from the job's owner if one user submitted the job on behalf of another user.

**SYSTEM\_TIME**

The `SYSTEM_TIME` column contains the number of CPU seconds of system time consumed by the job.

**USER\_ID**

The `USER_ID` column contains the system identifier of the job owner.

**USER\_TIME**

The `USER_TIME` column contains the number of CPU seconds of user time consumed by the job.

---

## Chapter 12. Log Monitoring

This chapter provides information about running the log monitor and using the log monitor configuration file.

To be alerted to a severe problem, you should monitor the system error log file produced by the server and agent components.

The log monitor program periodically activates and examines the log file for new entries. New messages found in the log file are tested against filter commands specified in the configuration file. Messages that are not filtered out during this process are sent through system e-mail to a list of recipients defined in the configuration file.

---

### Running the Log Monitor

**Note:** If you set the DQP\_LOGMON profile variable to the path name of the configuration file, the log monitor will be started when the server processes are started. See “Starting and Stopping DB2 Query Patroller” on page 12 for information on starting and stopping DB2 Query Patroller.

Once the DQP\_LOGMON profile variable has been set, the log monitor can only be started when the server processes are started. If the DQP\_LOGMON variable has *not* been set, the log monitor can be started by executing the following command:

```
iwm_logmon -c configuration_file
```

If the DQP\_LOGMON profile variable *is* set, any invocation of the **iwm\_logmon -c *configuration\_file*** command will cause an error to occur and the log monitor will fail to run.

**Note:** When using the **iwm\_logmon** command on Windows, ensure that the necessary DQP\_MAPI profile variables have been defined and ensure that a MAPI mail profile for the mail service provider has been created for the currently logged on user. The DQP\_MAPI\_PROFILE variable must be set to the name of the MAPI profile for the mail service provider. For more information on setting up e-mail notification on Windows, see “E-mail Notification on Windows” on page 35.

---

## Log Monitor Configuration File

The log monitor configuration file must contain one or more configuration command lines. These commands define the log file to be monitored, the wake-up interval, the list of message recipients, and the filter criteria.

Each line in the configuration file may be empty or may contain a comment or configuration command. Comment lines start with a pound sign (#). One or more white space characters can precede the comment indicator or the first character in the configuration command. You can also enter comments at the end of each configuration command line when preceded by a pound sign.

### Example Configuration File

The following represents a sample configuration file:

```
# DB2 Query Patroller sample log monitor configuration file
INTERVAL 15 # wake up every 15 seconds
LOGFILE * TRUNCATE # truncate the log at start-up
MAILTO iwm root # send e-mail to these users
MAILTO admin@customer.com # add this one too
INCLUDE SEV=E # include error messages
EXCLUDE ALL # exclude all others
```

### Log File

Messages written to the log file display in the following format:

```
<date-time> <node> <program> <pid> <message-id> <source-line> <text>
```

where:

- **<date-time>** represents the date and time that the message was recorded.
- **<node>** represents the host name of the node that recorded the message.
- **<program>** is the name of the program that recorded the message.
- **<pid>** represents the process ID that recorded the message.
- **<message-id>** represents the message identifier in the format `iwm-<number>-<severity>`. The severity code is **V** (verbose - messages displayed for information only), **I** (informational messages), **W** (warning messages), or **E** (error messages).
- **<source-line>** is the line of code where the error originated.
- **<text>** represents the text of the message. It may include a message from the database software, identified in message descriptions below as `<dbms-error>`. Refer to your database documentation for descriptions of these messages. The text may also include a message from the operating system.

The default log file name is `syserr.log`. DB2 Query Patroller creates this in the `$DQP_RUNTIME/log` directory on UNIX, or in the `sql1lib\db2\log` directory

on Windows. However, if the user sets the DQP\_LOG profile variable, DB2 Query Patroller creates the log file in the directory specified.

## INTERVAL Command

Syntax:

►►—INTERVAL—*seconds*—►►

The INTERVAL command accepts one parameter: the number of seconds that the log monitor should wait between checking the log file for new messages. If you do not specify an INTERVAL command, the parameter defaults to 30 seconds. If you specify multiple INTERVAL commands, log monitor uses the last specified command.

## LOGFILE Command

Syntax:

►►—LOGFILE—*file\_name*—

TRUNCATE
CONTINUE

—►►

The LOGFILE command defines the log file name and the action that should be taken when the log monitor starts. You must first specify the log file name parameter, which may be specified as an asterisk (\*) if you want to use the default log file name. The default log file name is syserr.log. DB2 Query Patroller creates this in the DQP\_RUNTIME/log directory on UNIX, or in the *instance\_directory*\log directory on Windows. However, if the user sets the DQP\_LOG profile variable, DB2 Query Patroller creates the log file in the directory specified.

The action to take is specified by TRUNCATE, CONTINUE, or by omitting these keywords altogether. If you specify TRUNCATE, DB2 Query Patroller removes all messages from the log file when the log monitor initializes. If you specify CONTINUE, DB2 Query Patroller sends only messages added to the log file after the log monitor starts. If you omit these keywords, DB2 Query Patroller sends all messages already in the log file, as well as any new messages.

If you specify multiple LOGFILE commands, DB2 Query Patroller uses the last command. If you do not specify a LOGFILE command, the log monitor monitors the default log file and any messages already in the log file at startup are sent.

## MAILTO Command

Syntax:

►MAILTO—address—►

The MAILTO command defines the recipients of messages. Each MAILTO command must include one or more e-mail addresses. You must specify at least one MAILTO command. If you are running DB2 Query Patroller on Windows, e-mail notification is sent using the Windows Message Application Programming Interface (MAPI). The e-mail addresses specified with the MAILTO command must be in a format recognizable by the MAPI service provider. For information on setting up e-mail notification on Windows, see “E-mail Notification on Windows” on page 35.

## INCLUDE and EXCLUDE Commands

Syntax:

► [INCLUDE  
EXCLUDE] test1 test2 test3 ►

Each *test* is one of: **SEV** (*message severity*), **NUM** (*message number*), **PGM** (*program name*), or **ALL**.

The INCLUDE and EXCLUDE commands define the message filtering criteria. The log monitor tests each new message against the filter commands. Recipients receive those messages that match an INCLUDE command or do not match any filter commands. The log monitor tests each message against the filter commands in the order specified in the configuration file. The first command that matches the message on all tests specified causes the message to be included or excluded.

You cannot specify the ALL test with any other test. It should appear only on the last INCLUDE or EXCLUDE command in the configuration file because it matches all messages, causing all subsequent filter commands to be ignored.

Of the other tests (SEV, NUM, and PGM), you can specify up to three on each filter command, but you can only specify one of each type on each command. If you specify more than one test on a command, all tests must match the message for the command to apply. A comparison operator and comparison value must follow each test type name. The operator must either be the equal sign (=) or an exclamation mark followed by an equal sign (!=). The equal sign means that in order to match, the test must be equal to the value. The



exclamation mark and equal sign specifies that the test must not be equal to the value. The following list explains the filter test types.

- **SEV** (*message severity*)

The severity comparison value must be one of the following letters: V (Verbose), W (Warning), I (Informational), or E (Error). For example, to test for error messages, use SEV=E; to test for non-verbose messages use SEV!=V.

- **NUM** (*message number*)

The message number comparison value must either be a single one to five digit number or a range of message numbers specified by two numbers separated by a colon (:). For example, to test for message 100, use NUM=100; to test for all messages greater than 100, specify NUM!=0:100.

- **PGM** (*program name*)

The program name comparison must be the name of the program that generated the message. For example, to test for messages generated by the iwm\_analyze program, use PGM=iwm\_analyze; to test for messages not generated by the iwm\_sqlexec program, use PGM!=iwm\_sqlexec.



---

## Appendix A. Troubleshooting the DB2 Query Patroller Server

This appendix provides information to assist in troubleshooting the DB2 Query Patroller server and describes common problems that may be encountered.

---

### Distinguishing Between DB2 and DB2 Query Patroller Failures

DB2 Query Patroller is composed of components that run on the DB2 Query Patroller server, and components that run on the DB2 Query Patroller client. The client does not communicate directly with the server. DB2 Query Patroller uses DB2 tables to pass information between the client and the server. For this reason an administrator or operator needs to be able to distinguish between DB2 failures and DB2 Query Patroller failures.

Refer to the *Troubleshooting Guide* if a problem or system failure is suspected to have originated from DB2.

---

### syserr.log

The DB2 Query Patroller server maintains a system error log named `syserr.log` in the path referenced by the profile variable `DQP_LOG`. This file contains diagnostic information for the DB2 Query Patroller server and may explain the cause of the system failure.

**Note:** DB2 Query Patroller agents should be configured to log their diagnostic information to a unique location. This will assist in determining where the error originated.

---

### DB2 Diagnostic Log

If the DB2 database manager configuration parameter `DIAGLEVEL` is set to 4, all SQLCAs with a negative `SQLCODE` will be dumped to the diagnostic log. This information is useful when trying to determine why the DB2 Query Patroller server failed when attempting to access DB2.

---

### Processes

#### DB2 Query Patroller Server

On a properly functioning DB2 Query Patroller server, the following processes will be active:

- iwm\_server
- iwm\_nodemgr
- iwm\_local (one or more)
- iwm\_sched
- iwm\_remote
- iwm\_net

## DB2 Query Patroller Agent

On a properly functioning DB2 Query Patroller agent, the following processes will be active:

- iwm\_net
- iwm\_nodemgr

**Note:** Other DB2 Query Patroller processes are transitory and will be started and stopped as required.

## Process Failures

If any of these processes have quit before completion, the DB2 Query Patroller system should be restarted. See “Starting and Stopping DB2 Query Patroller” on page 12 for instructions.

---

## Generating a Trace File

DB2 Query Patroller is also integrated with DB2 trace. The IBM service organization may request a DB2 trace in order to help identify and localize a problem. Refer to the *Troubleshooting Guide* for details on how to perform a DB2 trace.

---

## Common Server Problems

1. The DB2 Query Patroller server fails to start:
  - Make sure DB2 is started.
  - Make sure that the DB2 Query Patroller TCP/IP port is not being used by another process.
  - Check DB2 Query Patroller profile variables to ensure that they are set correctly.
  - Check the `syserr.log` file for specific process errors.
2. The DB2 Query Patroller agent fails to start:
  - Ensure that the DB2 Query Patroller agent and DB2 Query Patroller server communications configuration specify the same TCP/IP port.
  - Check the setting of the DB2 profile variable `DQP_SERVER` to ensure that the correct TCP host name for the DB2 Query Patroller server is specified.

- Ensure that this TCP/IP port is not in use on the DB2 Query Patroller agent system.
3. A job is in the queue but will not run:
    - Check the DB2 Query Patroller error log to make sure there have been no DB2 Query Patroller process failures.
    - Check the DB2 Query Patroller system configurations to ensure that they are set correctly.
    - Restart the DB2 Query Patroller server.
  4. DB2 Query Patroller Service on Windows fails to start:
    - Check to make sure DB2 is started.
    - Check that the user account associated with the service has the correct system access rights and a valid password.
    - Check all DB2 Query Patroller profile variables to see if they are set correctly.
    - Make sure that there are no orphan DB2 Query Patroller processes prior to starting the system.
    - Make sure a valid license key has been registered.
  5. Sending of e-mail notification on Windows fails:
    - Make sure that a MAPI profile for the user account associated with the DB2 Query Patroller service has been created.
    - Ensure that all DQP\_MAPI environment variables have been properly set.
    - Make sure the account associated with the DB2 Query Patroller service has the necessary access to the mail provider.
    - If the mail provider requires a password to logon to the mail service, set the DQP\_MAPI\_PASSWORD to this value and re-start the service.
    - If the MAPI service provider requires the sender of the mail to be a user account different from the user account associated with the DB2 Query Patroller service, try to use the DQP\_MAIL\_ACCOUNT environment variable.
    - Check the syserr.log for DQP\_MAIL.EXE specific errors.



---

## Appendix B. Troubleshooting DB2 Query Patroller Clients

This appendix provides information to assist in troubleshooting the DB2 Query Patroller client and describes common problems that may be encountered.

---

### Troubleshooting QueryEnabler

Whenever a problem occurs, you should check the following:

- The DB2 server is started.
- The application can connect to the DB2 server directly using the DB2 cataloged names.

**Note:** You may also want to check the DB2 diagnostic log for DB2-related problems.

In order to help identify problems, IBM support representatives may employ the following tracing mechanisms collectively or separately:

- JDBC trace
- DB2 CLI trace
- DB2 trace
- ODBC trace (if you are using ODBC)

DB2 trace will activate the Java Trace tool for QueryEnabler. The Java trace file will be created in the `DQP_RUNTIME/log/` directory on UNIX, or in the `DQP_RUNTIME\log\` directory on Windows. Once a log file has been created, trace information created subsequently will be appended to this log file. You can name your own Java log file by setting the DB2 Query Patroller profile variable `DQP_TRACEFILE`. For more information on setting DB2 Query Patroller profile variables, see “DB2 Profile Variables” on page 39.

Trace facilities are described in the *Troubleshooting Guide*. Refer to this document for detailed instructions.

### Common QueryEnabler Problems

1. The application is unable to connect to DB2:
  - Verify that the application is able to connect to DB2. If this fails, refer to the *Troubleshooting Guide* for information on how to resolve a connectivity problem.
  - Ensure that the user account and user password are correct.

2. The application is unable to access the QueryEnabler query submission dialogs:
  - Check the path for the database configuration parameter, JDK11\_PATH and check the setting for JAVA\_HEAP\_SZ. JDK11\_PATH and JAVA\_HEAP\_SZ are both used to start the Java Virtual Machine (JVM).
  - Ensure that QueryEnabler has been installed correctly.
  - Ensure that the DYN\_QUERY\_MGMT database configuration parameter has been set to ENABLE in the database configuration file on the server.
3. The QueryEnabler dialogs appear unexpectedly:
  - Ensure the application is using the correct database alias name.
  - Increase the Cost Management Threshold parameter in the profile of the submitting user. This will allow queries that do not exceed the threshold to run against the data source without user intervention via the QueryEnabler tool.
4. The query application is unable to submit a query to the DB2 Query Patroller server using QueryEnabler:
  - Ensure the DB2 Query Patroller server stored procedures have been installed on the DB2 server.
  - Ensure the user has been granted the proper DB2 access to the stored procedures.
  - Ensure that the User Management Threshold parameter for the user is set to an appropriate value. If the cost of the query does not exceed the management threshold, the QueryEnabler interface will not appear.
5. The application has submitted a query to the DB2 Query Patroller server but the query is not being executed:
  - Ensure that the DB2 Query Patroller server is running correctly. It may be necessary to restart the DB2 Query Patroller server.
  - Use QueryMonitor to inspect the details of the Job. The job may be held because the query exceeded user thresholds. A DB2 Query Patroller administrator or operator must release this job.
  - The job may be in the queued state because system resources have exceeded configured limitations. Under these conditions, the query will execute once the system resources fall within acceptable limits as configured by the DB2 Query Patroller administrator.

---

## Troubleshooting QueryMonitor and QueryAdministrator

QueryMonitor and QueryAdministrator are Java applications that access DB2 using JDBC. Diagnostic information can be obtained from four sources:

- DB2 diagnostic log
- JDBC trace
- DB2 trace



In order to help identify problems, IBM support representatives may employ the following tracing mechanisms collectively or separately:

- JDBC trace
- DB2 CLI trace
- DB2 trace

Typically those trace tools are launched using a Java Virtual Machine (JVM) that does not have a console (JREW.EXE). These trace tools may also be run in a JVM that has a console (JRE.EXE), in which case DB2 Query Patroller will print information to the console. You can use this information to help identify the cause of a problem.

QueryAdministrator and QueryMonitor may be launched from the command line with a `-t` switch, which will cause a console to appear for a Java trace, or a `-tf filename` switch causing the trace information to be output to a file. Note that the current directory must be set to `DQP_RUNTIME\qp` directory on Windows, or the `DQP_RUNTIME/qp` path on UNIX. To initiate a Java trace for the QueryAdministrator and have the trace information output to a file, execute the following command from the `DQP_RUNTIME\qp` directory on Windows, or the `DQP_RUNTIME/qp` path on UNIX:

```
QueryAdmin -tf filename
```

where *filename* is a fully qualified path and file name for the Java trace to place diagnostic information.

To initiate a Java trace for the QueryMonitor and have the trace information output to a file, execute the following command from the `DQP_RUNTIME\qp` directory:

```
QueryMonitor -tf filename
```

where *filename* is a fully qualified path and filename for the Java trace to place diagnostic information.

## Common QueryMonitor and QueryAdministrator Problems

1. The tool is unable to connect to DB2:
  - Ensure that the DB2 data source name is used in the connect dialog.
  - Ensure that you have supplied the correct user name and password.
2. The application stops responding:
  - Close and restart the application. If the problem persists, attempt to run the application from a JVM with a console to obtain additional diagnostic information.

---

## Troubleshooting Tracker

Tracker is a JAVA application that uses JDBC to access the DB2 Query Patroller schema in DB2 in order to provide reports that display database usage history for queries that have been managed by DB2 Query Patroller.

Diagnostic information for Tracker can be obtained from four sources:

- the DB2 diagnostic log
- ODBC traces
- CLI traces
- DB2 traces

The Tracker tool may be launched from the command line with a `-t` switch, which will cause a console to appear for a Java trace, or a `-tf filename` switch causing the trace information to be output to a file. Note that the current directory must be set to `DQP_RUNTIME\qp` directory on Windows, or the `DQP_RUNTIME/qp` path on UNIX, where `DQP_RUNTIME` is the DB2 Query Patroller installation path. To initiate a Java trace for the Tracker tool and have the trace information output to a file, execute the following command from the `DQP_RUNTIME\qp` directory on Windows, or the `DQP_RUNTIME/qp` path on UNIX:

```
db2track -tf filename
```

where *filename* is a fully qualified path and file name for the Java trace to place diagnostic information.

### Common Tracker Problems

1. Tracker is unable to access DB2:
  - Ensure that the correct DB2 data source name, user name, and password have been specified.
2. Tracker does not display any data:
  - Ensure that the DB2 Query Patroller server has been configured to capture this information.
  - From QueryAdministrator, check that the Accounting option has been enabled.
  - Ensure that the date range specified in the Tracker application is set correctly to a time period when queries were in progress.
3. The data displayed by Tracker is not up-to-date:
  - Ensure the Tracker server on the DB2 Query Patroller server is scheduled to execute periodically. The Tracker server component may be executed interactively to generate up-to-date statistics.

---

## Appendix C. Using the DB2 Library

The DB2 Universal Database library consists of online help, books (PDF and HTML), and sample programs in HTML format. This section describes the information that is provided, and how you can access it.

To access product information online, you can use the Information Center. For more information, see “Accessing Information with the Information Center” on page 113. You can view task information, DB2 books, troubleshooting information, sample programs, and DB2 information on the Web.

---

### DB2 PDF Files and Printed Books

#### DB2 Information

The following table divides the DB2 books into four categories:

##### **DB2 Guide and Reference Information**

These books contain the common DB2 information for all platforms.

##### **DB2 Installation and Configuration Information**

These books are for DB2 on a specific platform. For example, there are separate *Quick Beginnings* books for DB2 on OS/2, Windows, and UNIX-based platforms.

##### **Cross-platform sample programs in HTML**

These samples are the HTML version of the sample programs that are installed with the Application Development Client. The samples are for informational purposes and do not replace the actual programs.

##### **Release notes**

These files contain late-breaking information that could not be included in the DB2 books.

The installation manuals, release notes, and tutorials are viewable in HTML directly from the product CD-ROM. Most books are available in HTML on the product CD-ROM for viewing and in Adobe Acrobat (PDF) format on the DB2 publications CD-ROM for viewing and printing. You can also order a printed copy from IBM; see “Ordering the Printed Books” on page 109. The following table lists books that can be ordered.

On OS/2 and Windows platforms, you can install the HTML files under the `sql1lib\doc\html` directory. DB2 information is translated into different

languages; however, all the information is not translated into every language. Whenever information is not available in a specific language, the English information is provided

On UNIX platforms, you can install multiple language versions of the HTML files under the `doc/%L/html` directories, where `%L` represents the locale. For more information, refer to the appropriate *Quick Beginnings* book.

You can obtain DB2 books and access information in a variety of ways:

- “Viewing Information Online” on page 112
- “Searching Information Online” on page 116
- “Ordering the Printed Books” on page 109
- “Printing the PDF Books” on page 108

Table 8. DB2 Information

Name	Description	Form Number PDF File Name	HTML Directory
<b>DB2 Guide and Reference Information</b>			
<i>Administration Guide</i>	<i>Administration Guide: Planning</i> provides an overview of database concepts, information about design issues (such as logical and physical database design), and a discussion of high availability.	SC09-2946 db2d1x70	db2d0
	<i>Administration Guide: Implementation</i> provides information on implementation issues such as implementing your design, accessing databases, auditing, backup and recovery.	SC09-2944 db2d2x70	
	<i>Administration Guide: Performance</i> provides information on database environment and application performance evaluation and tuning.	SC09-2945 db2d3x70	
	You can order the three volumes of the <i>Administration Guide</i> in the English language in North America using the form number SBOF-8934.		
<i>Administrative API Reference</i>	Describes the DB2 application programming interfaces (APIs) and data structures that you can use to manage your databases. This book also explains how to call APIs from your applications.	SC09-2947 db2b0x70	db2b0

Table 8. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>Application Building Guide</i>	Provides environment setup information and step-by-step instructions about how to compile, link, and run DB2 applications on Windows, OS/2, and UNIX-based platforms.	SC09-2948 db2axx70	db2ax
<i>APPC, CPI-C, and SNA Sense Codes</i>	Provides general information about APPC, CPI-C, and SNA sense codes that you may encounter when using DB2 Universal Database products.	No form number db2apx70	db2ap
	Available in HTML format only.		
<i>Application Development Guide</i>	Explains how to develop applications that access DB2 databases using embedded SQL or Java (JDBC and SQLJ). Discussion topics include writing stored procedures, writing user-defined functions, creating user-defined types, using triggers, and developing applications in partitioned environments or with federated systems.	SC09-2949 db2a0x70	db2a0
<i>CLI Guide and Reference</i>	Explains how to develop applications that access DB2 databases using the DB2 Call Level Interface, a callable SQL interface that is compatible with the Microsoft ODBC specification.	SC09-2950 db2l0x70	db2l0
<i>Command Reference</i>	Explains how to use the Command Line Processor and describes the DB2 commands that you can use to manage your database.	SC09-2951 db2n0x70	db2n0
<i>Connectivity Supplement</i>	Provides setup and reference information on how to use DB2 for AS/400, DB2 for OS/390, DB2 for MVS, or DB2 for VM as DRDA application requesters with DB2 Universal Database servers. This book also details how to use DRDA application servers with DB2 Connect application requesters.	No form number db2h1x70	db2h1
	Available in HTML and PDF only.		

Table 8. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>Data Movement Utilities Guide and Reference</i>	Explains how to use DB2 utilities, such as import, export, load, AutoLoader, and DPROF, that facilitate the movement of data.	SC09-2955 db2dmx70	db2dm
<i>Data Warehouse Center Administration Guide</i>	Provides information on how to build and maintain a data warehouse using the Data Warehouse Center.	SC26-9993 db2ddx70	db2dd
<i>Data Warehouse Center Application Integration Guide</i>	Provides information to help programmers integrate applications with the Data Warehouse Center and with the Information Catalog Manager.	SC26-9994 db2adx70	db2ad
<i>DB2 Connect User's Guide</i>	Provides concepts, programming, and general usage information for the DB2 Connect products.	SC09-2954 db2c0x70	db2c0
<i>DB2 Query Patroller Administration Guide</i>	Provides an operational overview of the DB2 Query Patroller system, specific operational and administrative information, and task information for the administrative graphical user interface utilities.	SC09-2958 db2dwx70	db2dw
<i>DB2 Query Patroller User's Guide</i>	Describes how to use the tools and functions of the DB2 Query Patroller.	SC09-2960 db2wwx70	db2ww
<i>Glossary</i>	Provides definitions for terms used in DB2 and its components.  Available in HTML format and in the <i>SQL Reference</i> .	No form number db2t0x70	db2t0
<i>Image, Audio, and Video Extenders Administration and Programming</i>	Provides general information about DB2 extenders, and information on the administration and configuration of the image, audio, and video (IAV) extenders and on programming using the IAV extenders. It includes reference information, diagnostic information (with messages), and samples.	SC26-9929 dmbu7x70	dmbu7
<i>Information Catalog Manager Administration Guide</i>	Provides guidance on managing information catalogs.	SC26-9995 db2dix70	db2di

Table 8. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>Information Catalog Manager Programming Guide and Reference</i>	Provides definitions for the architected interfaces for the Information Catalog Manager.	SC26-9997 db2bix70	db2bi
<i>Information Catalog Manager User's Guide</i>	Provides information on using the Information Catalog Manager user interface.	SC26-9996 db2aix70	db2ai
<i>Installation and Configuration Supplement</i>	Guides you through the planning, installation, and setup of platform-specific DB2 clients. This supplement also contains information on binding, setting up client and server communications, DB2 GUI tools, DRDA AS, distributed installation, the configuration of distributed requests, and accessing heterogeneous data sources.	GC09-2957 db2iyx70	db2iy
<i>Message Reference</i>	Lists messages and codes issued by DB2, the Information Catalog Manager, and the Data Warehouse Center, and describes the actions you should take.  You can order both volumes of the Message Reference in the English language in North America with the form number SBOF-8932.	Volume 1 GC09-2978  db2m1x70 Volume 2 GC09-2979  db2m2x70	db2m0
<i>OLAP Integration Server Administration Guide</i>	Explains how to use the Administration Manager component of the OLAP Integration Server.	SC27-0787 db2dpx70	n/a
<i>OLAP Integration Server Metaoutline User's Guide</i>	Explains how to create and populate OLAP metaoutlines using the standard OLAP Metaoutline interface (not by using the Metaoutline Assistant).	SC27-0784 db2upx70	n/a
<i>OLAP Integration Server Model User's Guide</i>	Explains how to create OLAP models using the standard OLAP Model Interface (not by using the Model Assistant).	SC27-0783 db2lpx70	n/a
<i>OLAP Setup and User's Guide</i>	Provides configuration and setup information for the OLAP Starter Kit.	SC27-0702 db2ipx70	db2ip
<i>OLAP Spreadsheet Add-in User's Guide for Excel</i>	Describes how to use the Excel spreadsheet program to analyze OLAP data.	SC27-0786 db2epx70	db2ep

Table 8. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>OLAP Spreadsheet Add-in User's Guide for Lotus 1-2-3</i>	Describes how to use the Lotus 1-2-3 spreadsheet program to analyze OLAP data.	SC27-0785 db2tpx70	db2tp
<i>Replication Guide and Reference</i>	Provides planning, configuration, administration, and usage information for the IBM Replication tools supplied with DB2.	SC26-9920 db2e0x70	db2e0
<i>Spatial Extender User's Guide and Reference</i>	Provides information about installing, configuring, administering, programming, and troubleshooting the Spatial Extender. Also provides significant descriptions of spatial data concepts and provides reference information (messages and SQL) specific to the Spatial Extender.	SC27-0701 db2sbx70	db2sb
<i>SQL Getting Started</i>	Introduces SQL concepts and provides examples for many constructs and tasks.	SC09-2973 db2y0x70	db2y0
<i>SQL Reference, Volume 1 and Volume 2</i>	Describes SQL syntax, semantics, and the rules of the language. This book also includes information about release-to-release incompatibilities, product limits, and catalog views.  You can order both volumes of the <i>SQL Reference</i> in the English language in North America with the form number SBOF-8933.	Volume 1 SC09-2974 db2s1x70  Volume 2 SC09-2975 db2s2x70	db2s0
<i>System Monitor Guide and Reference</i>	Describes how to collect different kinds of information about databases and the database manager. This book explains how to use the information to understand database activity, improve performance, and determine the cause of problems.	SC09-2956 db2f0x70	db2f0
<i>Text Extender Administration and Programming</i>	Provides general information about DB2 extenders and information on the administration and configuring of the text extender and on programming using the text extenders. It includes reference information, diagnostic information (with messages) and samples.	SC26-9930 desu9x70	desu9



Table 8. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>Troubleshooting Guide</i>	Helps you determine the source of errors, recover from problems, and use diagnostic tools in consultation with DB2 Customer Service.	GC09-2850 db2p0x70	db2p0
<i>What's New</i>	Describes the new features, functions, and enhancements in DB2 Universal Database, Version 7.	SC09-2976 db2q0x70	db2q0
<b>DB2 Installation and Configuration Information</b>			
<i>DB2 Connect Enterprise Edition for OS/2 and Windows Quick Beginnings</i>	Provides planning, migration, installation, and configuration information for DB2 Connect Enterprise Edition on the OS/2 and Windows 32-bit operating systems. This book also contains installation and setup information for many supported clients.	GC09-2953 db2c6x70	db2c6
<i>DB2 Connect Enterprise Edition for UNIX Quick Beginnings</i>	Provides planning, migration, installation, configuration, and task information for DB2 Connect Enterprise Edition on UNIX-based platforms. This book also contains installation and setup information for many supported clients.	GC09-2952 db2cyx70	db2cy
<i>DB2 Connect Personal Edition Quick Beginnings</i>	Provides planning, migration, installation, configuration, and task information for DB2 Connect Personal Edition on the OS/2 and Windows 32-bit operating systems. This book also contains installation and setup information for all supported clients.	GC09-2967 db2c1x70	db2c1
<i>DB2 Connect Personal Edition Quick Beginnings for Linux</i>	Provides planning, installation, migration, and configuration information for DB2 Connect Personal Edition on all supported Linux distributions.	GC09-2962 db2c4x70	db2c4
<i>DB2 Data Links Manager Quick Beginnings</i>	Provides planning, installation, configuration, and task information for DB2 Data Links Manager for AIX and Windows 32-bit operating systems.	GC09-2966 db2z6x70	db2z6

Table 8. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>DB2 Enterprise - Extended Edition for UNIX Quick Beginnings</i>	Provides planning, installation, and configuration information for DB2 Enterprise - Extended Edition on UNIX-based platforms. This book also contains installation and setup information for many supported clients.	GC09-2964 db2v3x70	db2v3
<i>DB2 Enterprise - Extended Edition for Windows Quick Beginnings</i>	Provides planning, installation, and configuration information for DB2 Enterprise - Extended Edition for Windows 32-bit operating systems. This book also contains installation and setup information for many supported clients.	GC09-2963 db2v6x70	db2v6
<i>DB2 for OS/2 Quick Beginnings</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database on the OS/2 operating system. This book also contains installation and setup information for many supported clients.	GC09-2968 db2i2x70	db2i2
<i>DB2 for UNIX Quick Beginnings</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database on UNIX-based platforms. This book also contains installation and setup information for many supported clients.	GC09-2970 db2ixx70	db2ix
<i>DB2 for Windows Quick Beginnings</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database on Windows 32-bit operating systems. This book also contains installation and setup information for many supported clients.	GC09-2971 db2i6x70	db2i6
<i>DB2 Personal Edition Quick Beginnings</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database Personal Edition on the OS/2 and Windows 32-bit operating systems.	GC09-2969 db2i1x70	db2i1
<i>DB2 Personal Edition Quick Beginnings for Linux</i>	Provides planning, installation, migration, and configuration information for DB2 Universal Database Personal Edition on all supported Linux distributions.	GC09-2972 db2i4x70	db2i4

Table 8. DB2 Information (continued)

Name	Description	Form Number PDF File Name	HTML Directory
<i>DB2 Query Patroller Installation Guide</i>	Provides installation information about DB2 Query Patroller.	GC09-2959 db2iwx70	db2iw
<i>DB2 Warehouse Manager Installation Guide</i>	Provides installation information for warehouse agents, warehouse transformers, and the Information Catalog Manager.	GC26-9998 db2idx70	db2id
<b>Cross-Platform Sample Programs in HTML</b>			
Sample programs in HTML	Provides the sample programs in HTML format for the programming languages on all platforms supported by DB2. The sample programs are provided for informational purposes only. Not all samples are available in all programming languages. The HTML samples are only available when the DB2 Application Development Client is installed.  For more information on the programs, refer to the <i>Application Building Guide</i> .	No form number	db2hs
<b>Release Notes</b>			
<i>DB2 Connect Release Notes</i>	Provides late-breaking information that could not be included in the DB2 Connect books.	See note #2.	db2cr
<i>DB2 Installation Notes</i>	Provides late-breaking installation-specific information that could not be included in the DB2 books.	Available on product CD-ROM only.	
<i>DB2 Release Notes</i>	Provides late-breaking information about all DB2 products and features that could not be included in the DB2 books.	See note #2.	db2ir

**Notes:**

1. The character *x* in the sixth position of the file name indicates the language version of a book. For example, the file name *db2d0e70* identifies the English version of the *Administration Guide* and the file name *db2d0f70* identifies the French version of the same book. The following letters are used in the sixth position of the file name to indicate the language version:

Language	Identifier
Brazilian Portuguese	b

Bulgarian	u
Czech	x
Danish	d
Dutch	q
English	e
Finnish	y
French	f
German	g
Greek	a
Hungarian	h
Italian	i
Japanese	j
Korean	k
Norwegian	n
Polish	p
Portuguese	v
Russian	r
Simp. Chinese	c
Slovenian	l
Spanish	z
Swedish	s
Trad. Chinese	t
Turkish	m

2. Late breaking information that could not be included in the DB2 books is available in the Release Notes in HTML format and as an ASCII file. The HTML version is available from the Information Center and on the product CD-ROMs. To view the ASCII file:
  - On UNIX-based platforms, see the `Release.Notes` file. This file is located in the `DB2DIR/Readme/%L` directory, where `%L` represents the locale name and `DB2DIR` represents:
    - `/usr/lpp/db2_07_01` on AIX
    - `/opt/IBMDB2/V7.1` on HP-UX, PTX, Solaris, and Silicon Graphics IRIX
    - `/usr/IBMDB2/V7.1` on Linux.
  - On other platforms, see the `RELEASE.TXT` file. This file is located in the directory where the product is installed. On OS/2 platforms, you can also double-click the **IBM DB2** folder and then double-click the **Release Notes** icon.

## Printing the PDF Books

If you prefer to have printed copies of the books, you can print the PDF files found on the DB2 publications CD-ROM. Using the Adobe Acrobat Reader, you can print either the entire book or a specific range of pages. For the file name of each book in the library, see Table 8 on page 100.

You can obtain the latest version of the Adobe Acrobat Reader from the Adobe Web site at <http://www.adobe.com>.

The PDF files are included on the DB2 publications CD-ROM with a file extension of PDF. To access the PDF files:

1. Insert the DB2 publications CD-ROM. On UNIX-based platforms, mount the DB2 publications CD-ROM. Refer to your *Quick Beginnings* book for the mounting procedures.
2. Start the Acrobat Reader.
3. Open the desired PDF file from one of the following locations:
  - On OS/2 and Windows platforms:  
*x:\doc\language* directory, where *x* represents the CD-ROM drive and *language* represent the two-character country code that represents your language (for example, EN for English).
  - On UNIX-based platforms:  
*/cdrom/doc/%L* directory on the CD-ROM, where */cdrom* represents the mount point of the CD-ROM and *%L* represents the name of the desired locale.

You can also copy the PDF files from the CD-ROM to a local or network drive and read them from there.

## Ordering the Printed Books

You can order the printed DB2 books either individually or as a set (in North America only) by using a sold bill of forms (SBOF) number. To order books, contact your IBM authorized dealer or marketing representative, or phone 1-800-879-2755 in the United States or 1-800-IBM-4YOU in Canada. You can also order the books from the Publications Web page at <http://www.elink.ibm.com/pbl/pbl>.

Two sets of books are available. SBOF-8935 provides reference and usage information for the DB2 Warehouse Manager. SBOF-8931 provides reference and usage information for all other DB2 Universal Database products and features. The contents of each SBOF are listed in the following table:

Table 9. Ordering the printed books

SBOF Number	Books Included
SBOF-8931	<ul style="list-style-type: none"> <li>• Administration Guide: Planning</li> <li>• Administration Guide: Implementation</li> <li>• Administration Guide: Performance</li> <li>• Administrative API Reference</li> <li>• Application Building Guide</li> <li>• Application Development Guide</li> <li>• CLI Guide and Reference</li> <li>• Command Reference</li> <li>• Data Movement Utilities Guide and Reference</li> <li>• Data Warehouse Center Administration Guide</li> <li>• Data Warehouse Center Application Integration Guide</li> <li>• DB2 Connect User's Guide</li> <li>• Installation and Configuration Supplement</li> <li>• Image, Audio, and Video Extenders Administration and Programming</li> <li>• Message Reference, Volumes 1 and 2</li> <li>• OLAP Integration Server Administration Guide</li> <li>• OLAP Integration Server Metaoutline User's Guide</li> <li>• OLAP Integration Server Model User's Guide</li> <li>• OLAP Integration Server User's Guide</li> <li>• OLAP Setup and User's Guide</li> <li>• OLAP Spreadsheet Add-in User's Guide for Excel</li> <li>• OLAP Spreadsheet Add-in User's Guide for Lotus 1-2-3</li> <li>• Replication Guide and Reference</li> <li>• Spatial Extender Administration and Programming Guide</li> <li>• SQL Getting Started</li> <li>• SQL Reference, Volumes 1 and 2</li> <li>• System Monitor Guide and Reference</li> <li>• Text Extender Administration and Programming</li> <li>• Troubleshooting Guide</li> <li>• What's New</li> </ul>
SBOF-8935	<ul style="list-style-type: none"> <li>• Information Catalog Manager Administration Guide</li> <li>• Information Catalog Manager User's Guide</li> <li>• Information Catalog Manager Programming Guide and Reference</li> <li>• Query Patroller Administration Guide</li> <li>• Query Patroller User's Guide</li> </ul>

## DB2 Online Documentation

### Accessing Online Help

Online help is available with all DB2 components. The following table describes the various types of help.

Type of Help	Contents	How to Access...
<i>Command Help</i>	Explains the syntax of commands in the command line processor.	<p>From the command line processor in interactive mode, enter:</p> <pre>? <i>command</i></pre> <p>where <i>command</i> represents a keyword or the entire command.</p> <p>For example, ? catalog displays help for all the CATALOG commands, while ? catalog database displays help for the CATALOG DATABASE command.</p>
<i>Client Configuration Assistant Help</i>	Explains the tasks you can perform in a window or notebook. The help includes overview and prerequisite information you need to know, and it describes how to use the window or notebook controls.	From a window or notebook, click the <b>Help</b> push button or press the <b>F1</b> key.
<i>Command Center Help</i>		
<i>Control Center Help</i>		
<i>Data Warehouse Center Help</i>		
<i>Event Analyzer Help</i>		
<i>Information Catalog Manager Help</i>		
<i>Satellite Administration Center Help</i>		
<i>Script Center Help</i>		
<i>Message Help</i>	Describes the cause of a message and any action you should take.	<p>From the command line processor in interactive mode, enter:</p> <pre>? XXXnnnnn</pre> <p>where XXXnnnnn represents a valid message identifier.</p> <p>For example, ? SQL30081 displays help about the SQL30081 message.</p> <p>To view message help one screen at a time, enter:</p> <pre>? XXXnnnnn   more</pre> <p>To save message help in a file, enter:</p> <pre>? XXXnnnnn &gt; filename.ext</pre> <p>where <i>filename.ext</i> represents the file where you want to save the message help.</p>

Type of Help	Contents	How to Access...
<i>SQL Help</i>	Explains the syntax of SQL statements.	<p>From the command line processor in interactive mode, enter:</p> <pre>help <i>statement</i></pre> <p>where <i>statement</i> represents an SQL statement.</p> <p>For example, help SELECT displays help about the SELECT statement.</p> <p><b>Note:</b> SQL help is not available on UNIX-based platforms.</p>
<i>SQLSTATE Help</i>	Explains SQL states and class codes.	<p>From the command line processor in interactive mode, enter:</p> <pre>? <i>sqlstate</i> or ? <i>class code</i></pre> <p>where <i>sqlstate</i> represents a valid five-digit SQL state and <i>class code</i> represents the first two digits of the SQL state.</p> <p>For example, ? 08003 displays help for the 08003 SQL state, while ? 08 displays help for the 08 class code.</p>

## Viewing Information Online

The books included with this product are in Hypertext Markup Language (HTML) softcopy format. Softcopy format enables you to search or browse the information and provides hypertext links to related information. It also makes it easier to share the library across your site.

You can view the online books or sample programs with any browser that conforms to HTML Version 3.2 specifications.

To view online books or sample programs:

- If you are running DB2 administration tools, use the Information Center.
- From a browser, click **File** —> **Open Page**. The page you open contains descriptions of and links to DB2 information:
  - On UNIX-based platforms, open the following page:

```
INSTHOME/sql1lib/doc/%L/html/index.htm
```

where %L represents the locale name.

- On other platforms, open the following page:

```
sql1lib\doc\html\index.htm
```

The path is located on the drive where DB2 is installed.



If you have not installed the Information Center, you can open the page by double-clicking the **DB2 Information** icon. Depending on the system you are using, the icon is in the main product folder or the Windows Start menu.

### **Installing the Netscape Browser**

If you do not already have a Web browser installed, you can install Netscape from the Netscape CD-ROM found in the product boxes. For detailed instructions on how to install it, perform the following:

1. Insert the Netscape CD-ROM.
2. On UNIX-based platforms only, mount the CD-ROM. Refer to your *Quick Beginnings* book for the mounting procedures.
3. For installation instructions, refer to the `CDNAVnn.txt` file, where *nn* represents your two character language identifier. The file is located at the root directory of the CD-ROM.

### **Accessing Information with the Information Center**

The Information Center provides quick access to DB2 product information. The Information Center is available on all platforms on which the DB2 administration tools are available.

You can open the Information Center by double-clicking the Information Center icon. Depending on the system you are using, the icon is in the Information folder in the main product folder or the Windows **Start** menu.

You can also access the Information Center by using the toolbar and the **Help** menu on the DB2 Windows platform.

The Information Center provides six types of information. Click the appropriate tab to look at the topics provided for that type.

<b>Tasks</b>	Key tasks you can perform using DB2.
<b>Reference</b>	DB2 reference information, such as keywords, commands, and APIs.
<b>Books</b>	DB2 books.
<b>Troubleshooting</b>	Categories of error messages and their recovery actions.
<b>Sample Programs</b>	Sample programs that come with the DB2 Application Development Client. If you did not install the DB2 Application Development Client, this tab is not displayed.
<b>Web</b>	DB2 information on the World Wide Web. To access this information, you must have a connection to the Web from your system.

When you select an item in one of the lists, the Information Center launches a viewer to display the information. The viewer might be the system help viewer, an editor, or a Web browser, depending on the kind of information you select.

The Information Center provides a find feature, so you can look for a specific topic without browsing the lists.

For a full text search, follow the hypertext link in the Information Center to the **Search DB2 Online Information** search form.

The HTML search server is usually started automatically. If a search in the HTML information does not work, you may have to start the search server using one of the following methods:

#### On Windows

Click **Start** and select **Programs** → **IBM DB2** → **Information** → **Start HTML Search Server**.

#### On OS/2

Double-click the **DB2 for OS/2** folder, and then double-click the **Start HTML Search Server** icon.

Refer to the release notes if you experience any other problems when searching the HTML information.

**Note:** The Search function is not available in the Linux, PTX, and Silicon Graphics IRIX environments.

## Using DB2 Wizards

Wizards help you complete specific administration tasks by taking you through each task one step at a time. Wizards are available through the Control Center and the Client Configuration Assistant. The following table lists the wizards and describes their purpose.

**Note:** The Create Database, Create Index, Configure Multisite Update, and Performance Configuration wizards are available for the partitioned database environment.

Wizard	Helps You to...	How to Access...
<i>Add Database</i>	Catalog a database on a client workstation.	From the Client Configuration Assistant, click <b>Add</b> .
<i>Backup Database</i>	Determine, create, and schedule a backup plan.	From the Control Center, right-click the database you want to back up and select <b>Backup</b> → <b>Database Using Wizard</b> .

<b>Wizard</b>	<b>Helps You to...</b>	<b>How to Access...</b>
<i>Configure Multisite Update</i>	Configure a multisite update, a distributed transaction, or a two-phase commit.	From the Control Center, right-click the <b>Databases</b> folder and select <b>Multisite Update</b> .
<i>Create Database</i>	Create a database, and perform some basic configuration tasks.	From the Control Center, right-click the <b>Databases</b> folder and select <b>Create</b> → <b>Database Using Wizard</b> .
<i>Create Table</i>	Select basic data types, and create a primary key for the table.	From the Control Center, right-click the <b>Tables</b> icon and select <b>Create</b> → <b>Table Using Wizard</b> .
<i>Create Table Space</i>	Create a new table space.	From the Control Center, right-click the <b>Table Spaces</b> icon and select <b>Create</b> → <b>Table Space Using Wizard</b> .
<i>Create Index</i>	Advise which indexes to create and drop for all your queries.	From the Control Center, right-click the <b>Index</b> icon and select <b>Create</b> → <b>Index Using Wizard</b> .
<i>Performance Configuration</i>	Tune the performance of a database by updating configuration parameters to match your business requirements.	From the Control Center, right-click the database you want to tune and select <b>Configure Performance Using Wizard</b> .  For the partitioned database environment, from the Database Partitions view, right-click the first database partition you want to tune and select <b>Configure Performance Using Wizard</b> .
<i>Restore Database</i>	Recover a database after a failure. It helps you understand which backup to use, and which logs to replay.	From the Control Center, right-click the database you want to restore and select <b>Restore</b> → <b>Database Using Wizard</b> .

## Setting Up a Document Server

By default, the DB2 information is installed on your local system. This means that each person who needs access to the DB2 information must install the same files. To have the DB2 information stored in a single location, perform the following steps:

1. Copy all files and subdirectories from `\sql11ib\doc\html` on your local system to a Web server. Each book has its own subdirectory that contains all the necessary HTML and GIF files that make up the book. Ensure that the directory structure remains the same.

2. Configure the Web server to look for the files in the new location. For information, refer to the NetQuestion Appendix in the *Installation and Configuration Supplement*.
3. If you are using the Java version of the Information Center, you can specify a base URL for all HTML files. You should use the URL for the list of books.
4. When you are able to view the book files, you can bookmark commonly viewed topics. You will probably want to bookmark the following pages:
  - List of books
  - Tables of contents of frequently used books
  - Frequently referenced articles, such as the ALTER TABLE topic
  - The Search form

For information about how you can serve the DB2 Universal Database online documentation files from a central machine, refer to the NetQuestion Appendix in the *Installation and Configuration Supplement*.

### **Searching Information Online**

To find information in the HTML files, use one of the following methods:

- Click **Search** in the top frame. Use the search form to find a specific topic. This function is not available in the Linux, PTX, or Silicon Graphics IRIX environments.
- Click **Index** in the top frame. Use the index to find a specific topic in the book.
- Display the table of contents or index of the help or the HTML book, and then use the find function of the Web browser to find a specific topic in the book.
- Use the bookmark function of the Web browser to quickly return to a specific topic.
- Use the search function of the Information Center to find specific topics. See “Accessing Information with the Information Center” on page 113 for details.

---

## Appendix D. Notices

IBM may not offer the products, services, or features discussed in this document in all 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 10504-1785  
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 information 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 at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licenses 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 Canada Limited  
Office of the Lab Director  
1150 Eglinton Ave. East  
North York, Ontario  
M3C 1H7  
CANADA

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 measurements 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.

This information may contain 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 may contain 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.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. \_enter the year or years\_. All rights reserved.

---

## Trademarks

The following terms, which may be denoted by an asterisk(\*), are trademarks of International Business Machines Corporation in the United States, other countries, or both.

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2
BookManager	OS/390
CICS	OS/400
C Set++	PowerPC
C/370	QBIC
DATABASE 2	QMF
DataHub	RACF
DataJoiner	RISC System/6000
DataPropagator	RS/6000
DataRefresher	S/370
DB2	SP
DB2 Connect	SQL/DS
DB2 Extenders	SQL/400
DB2 OLAP Server	System/370
DB2 Universal Database	System/390
Distributed Relational Database Architecture	SystemView
DRDA	VisualAge
eNetwork	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WebExplorer
	WIN-OS/2

The following terms are trademarks or registered trademarks of other companies:

Microsoft, Windows, and Windows NT are trademarks or registered trademarks of Microsoft Corporation.

Java or all Java-based trademarks and logos, and Solaris are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Tivoli and NetView are trademarks of Tivoli Systems Inc. in the United States, other countries, or both.



UNIX is a registered trademark in the United States, other countries or both and is licensed exclusively through X/Open Company Limited.

Other company, product, or service names, which may be denoted by a double asterisk(\*\*) may be trademarks or service marks of others.



---

# Index

## A

- add database wizard 114, 115
- adding users 17
- administration
  - data sources 22
  - DB2 Query Patroller 14
  - nodes 25
  - result destination 26
  - system parameters 32
  - user profiles 18
  - using the
    - QueryAdministrator 14
- administrative user iwmm 11
- agent processes 92
- agents 3, 9
- applying filters
  - reports 75

## B

- backup database wizard 114
- books 99, 109

## C

- canceling jobs 51
- changing the date and time range 80
- clients
  - configuring 47
  - troubleshooting 95
- columns
  - displaying 16
  - Job Accounting table 82
  - resizing 17
- Columns Hit by User Y Hitting Table X report 70
- Columns Hit in Table X report 69
- command line interface 3, 55
- commands
  - dqptest 12
  - dqptestop 12
  - EXCLUDE 88
  - INCLUDE 88
  - INTERVAL 87
  - iwmm\_cmd 55, 59
  - iwmm\_submit 55
  - LOGFILE 87
  - MAILTO 88
- common parameters 59
- communication
  - component 7

- components 1
  - agent 3, 9
  - command line interface 3
  - communications 7
  - cost analyzer 2
  - executor 3, 9, 28
  - job scheduler 2, 8
  - log monitor 3, 85
  - node manager 3, 9
  - notifier 2, 9, 35, 82
  - QueryAdministrator 4
  - QueryEnabler 4
  - QueryMonitor 4
  - request server 2
  - server 2
  - Tracker 4
- configuration
  - client 47
  - data warehouse 46
  - estimating and monitoring 45
  - monitoring 45
- configuration file
  - example 86
  - log monitor 86
- configuration parameters
  - DYN\_QUERY\_MGMT 48
  - JAVA\_HEAP\_SZ 48
  - JDK11\_PATH 48
- configure multisite update wizard 114
- configuring DB2 Query Patroller server 45
- connections
  - socket 7
- control command 59
- controlling
  - data sources 63
  - job execution 51
  - job flow 51
  - jobs, using iwmm\_cmd 61
  - nodes, using iwmm\_cmd 62
  - system parameters, using iwmm\_cmd 65
- conventions v
- cost analysis 2, 8
  - exit 52
- cost analyzer 2
- CPU utilization 9
- create database wizard 115

- create table space wizard 115
- create table wizard 115
- creating
  - Tracker data 67
  - user profiles 18
- customizing columns
  - reports 76
- customizing display
  - reports 74

## D

- data source administration 22
- data source parameters
  - data source status 64
  - job cost limit 64
  - job count limit 64
  - node CPU limit 64
  - node disk limit 64
- data source status 64
- data sources
  - controlling using iwmm\_command 63
  - editing 22
  - listing 22
  - listing using iwmm\_command 63
- Data Sources table 6
- data warehouse configuration 46
- database replacement 53
- date range 80
- DB2\_ABORTRESULT 42
- DB2 diagnostic log 91
- DB2 failures 91
- DB2 library
  - books 99
  - Information Center 113
  - language identifier for books 107
  - late-breaking information 108
  - online help 110
  - ordering printed books 109
  - printing PDF books 108
  - searching online
    - information 116
  - setting up document server 115
  - structure of 99
  - viewing online information 112
- wizards 114
- DB2 Query Patroller
  - components 2
  - failures 91

DB2 Query Patroller (*continued*)

- starting 12, 14
- stopping 12, 14
- system architecture 1
- tables 4

DB2DBDFT 39, 42

diagnostic log 91

disk space monitoring 39, 64

displaying columns 16

displaying job history data 71, 78

DQP\_ABORTRESULT 48

DQP\_DISKMON 39

DQP\_EXIT\_AN 39, 52

DQP\_INTERVAL 40, 50

DQP\_LAST\_RESULT\_DEST 42, 49

DQP\_LOCAL\_SERVANTS 40

DQP\_LOG 40

DQP\_LOGMON 85

DQP\_MAIL 41

DQP\_MAPI\_ACCOUNT 44

DQP\_MAPI\_PASSWORD 44

DQP\_MAPI\_PROFILE 44

DQP\_NET 41, 42, 55

DQP\_NO\_CPU 41

DQP\_NOEXPLAIN 41

DQP\_NTIER 43, 49

DQP\_PURGEHOURS 41

DQP\_RES\_TBLSPC 41

DQP\_RUNTIME 41, 43, 55, 86, 87

DQP\_SERVER 42

DQP\_SHARE 43, 49

DQP\_STARTUP 44

DQP\_TRACEFILE 43, 49

DQPLOG 86, 87

dqpnodes.cfg 11

DYN\_QUERY\_MGMT 48

**E**

- e-mail 2, 9, 35, 41, 85, 88
- e-mail message format 35
- e-mail notification on UNIX 35
- e-mail notification on Windows 35
- editing profiles 21
- effective user profile 21
- environment variables
  - DQP\_EXIT\_AN 52
  - DQP\_NET 55
  - DQP\_RUNTIME 55
  - PATH 55
- estimating system configuration 45
- example configuration file 86
- EXCLUDE command 88
- executor 3, 9, 28
- exit analysis 52
- exiting Tracker 82

**F**

- failures 91
  - DB2 91
  - DB2 Query Patroller 91
  - process 92
- filters
  - NUM 89
  - PGM 89
  - SEV 89
- formatting e-mail 35

**G**

- generating a trace file 92

**H**

- highlighting conventions v
- HTML
  - sample programs 107

**I**

- INCLUDE command 88
- index wizard 115
- Information Center 113
- installing
  - Netscape browser 113
- INTERVAL command 87
- iwm account 11, 39
- iwm\_cmd 55, 59
  - controlling data sources 63
  - controlling jobs 61
  - controlling nodes 62
  - controlling system
    - parameters 65
  - listing data sources 63
  - listing system parameters 65
  - monitoring jobs 59
  - monitoring nodes 62
- iwm\_net 92
- iwm\_nodemgr 92
- iwm\_submit 55

**J**

- JAVA\_HEAP\_SZ 48
- JDK11\_PATH 48
- job accounting
  - status 65, 82
  - table 2, 6, 9, 65, 82
  - table columns 82
- Job Activity over Time report 70
- job cost limit 64, 65
- job count limit 64, 65
- job execution
  - controlling 51
- job flow 7
  - controlling 51
  - cost analysis 8

- job flow 7 (*continued*)
  - job execution 9
  - job notification 9
  - job scheduling 8
  - job submission 7
- job history data 71, 78
- job purge days 65
- Job Queues table 6
- job scheduler 2, 8
- Job table 5
- jobs
  - accounting 82
  - canceling 51
  - controlling 59, 61
  - controlling using iwm\_cmd 61
  - execution 9
  - flow 7
  - managing 51
  - monitoring 59
  - monitoring using iwm\_cmd 59
  - processing 7
  - releasing 51
  - scheduling 8
  - submission 7

**L**

- language identifier
  - books 107
- late-breaking information 108
- listing
  - data sources, using
    - iwm\_cmd 63
  - system parameters, using
    - iwm\_cmd 65
- log file 3, 86
- log file monitoring 86
- log monitor 3, 85
  - configuration file 86
  - example configuration file 86
  - running 85
- LOGFILE 87

**M**

- mail 2, 9, 35, 41, 85, 88
- MAILTO 88
- management threshold 20
- managing jobs 51
- monitoring
  - CPU utilization 9, 64
  - disk space 64
  - jobs, using iwm command 59
  - jobs, using the control
    - command 59
  - log file 85
  - nodes, using iwm\_cmd 62
- monitoring system configuration 45

## N

- navigation
  - report 72
- Netscape browser
  - installing 113
- node administration 25
- node CPU limit 64
- node disk limit 64
- Node Information table 5
- node manager 3, 9
- nodes
  - controlling, using `iwm_cmd` 62
  - listing 25
  - monitoring, using `iwm_cmd` 62
- notification 9, 35
- notifier 2, 9, 35, 82
- NUM filter 89

## O

- online help 110
- online information
  - searching 116
  - viewing 112

## P

- parameters
  - common for `iwm_cmd` 59
  - data source, for `iwm_cmd` 63
  - data source status 64
  - `iwm_submit` 55
  - job control, for `iwm_cmd` 61
  - job cost limit 64, 65
  - job count limit 64, 65
  - job monitoring for `iwm_cmd` 60
  - job purge days 65
  - mode disk limit 64
  - node control and monitoring, for `iwm_cmd` 62
  - node CPU limit 64
  - results purge days 65
  - system, for `iwm_cmd` 65
- PATH 55
- PDF 108
- performance configuration
  - wizard 115
- PGM filter 89
- printing PDF books 108
- problems
  - QueryAdministrator 97
  - QueryEnabler 95
  - server 92
- process failures 92
- processes
  - agent 92
  - server 91

- processing jobs 7
- profile variables 39
  - DB2\_ABORTRESULT 42
  - DB2DBDFT 39, 42
  - DQP\_ABORTRESULT 48
  - DQP\_DISKMON 39
  - DQP\_EXIT\_AN 39
  - DQP\_INTERVAL 40, 50
  - DQP\_LAST\_RESULT\_DEST 42, 49
  - DQP\_LOCAL\_SERVANTS 40
  - DQP\_LOG 40
  - DQP\_LOGMON 85
  - DQP\_MAIL 41
  - DQP\_MAPI\_ACCOUNT 44
  - DQP\_MAPI\_PASSWORD 44
  - DQP\_MAPI\_PROFILE 44
  - DQP\_NET 41, 42
  - DQP\_NO\_CPU 41
  - DQP\_NOEXPLAIN 41
  - DQP\_NTIER 43, 49
  - DQP\_PURGEHOURS 41
  - DQP\_RES\_TBLSPC 41
  - DQP\_RUNTIME 41, 43, 86, 87
  - DQP\_server 42
  - DQP\_SHARE 43, 49
  - DQP\_STARTUP 44
  - DQP\_TRACEFILE 43, 49
  - DQPLOG 86, 87
  - Windows 44
- profiling queries 45
- purging 65
  - jobs 9, 65
  - result tables 9
- purging, result tables 65

## Q

- queries
  - profiling 45
- QueryAdministrator 4
  - common problems 97
  - starting 15
  - stopping 16
- QueryEnabler 4
  - problems 95
  - troubleshooting 95
- QueryMonitor 4
  - common problems 97
  - troubleshooting 96

## R

- release notes 108
- releasing jobs 51
- replacing the database 53
- reports
  - applying filters 75

- reports (*continued*)
  - changing the date and time range 80
  - Columns Hit by User Y Hitting Table X 70
  - Columns Hit in Table X 69
  - customizing columns 76
  - customizing display 74
  - date range 80
  - displaying 68
  - Job Activity over Time 68, 70
  - navigation 72
  - Rows Returned by Table 68
  - sorting data 75
  - time range 80
  - Total Table Hits 68
  - Total Table Hits report 68
  - Users That Hit Table X 69
  - zooming on a time chart 79
- request server 2
- resizing columns' 17
- resolving user profiles 18
- restore wizard 115
- result destination variables 27
- result destinations 6, 26
  - creating 28
  - editing 30
  - listing 28
  - piped 28
  - removing 30
- Result Destinations table 6, 26
- result tables 7, 9
  - purging 9, 65
- results purge days 65
- running the log monitor 85

## S

- sample programs
  - cross-platform 107
  - HTML 107
- searching
  - online information 114, 116
- server 2
  - common problems 92
  - troubleshooting 91
- server, configuring 45
- server problems 92
- server processes 91
- setting up document server 115
- SEV filter 89
- SmartGuides
  - wizards 114
- socket connections 7
- sorting data
  - reports 75

- starting
  - DB2 Query Patroller 12
  - DB2 Query Patroller on Windows 14
  - log monitor 85
  - Tracker 71
- starting QueryAdministrator 15
- stopping
  - DB2 Query Patroller 12
  - DB2 Query Patroller on Windows 14
  - log monitor 85
  - QueryAdministrator 16
- submit command 55
- submitting 55
- syserr.log 91
- system administration 32
- system error log 91
- system parameters 65
  - controlling using iwmm\_cmd 65
  - editing 32
  - job accounting status 65, 82
  - job cost limit 65
  - job count limit 65
  - job purge days 65
  - listing 32
  - listing using iwmm\_cmd 65
  - results purge days 65
- System Parameters table 5

**T**

- tables
  - Data Sources 6
  - DB2 Query Patroller 4
  - Job 5
  - Job Accounting 2, 6, 9, 65, 82
  - Job Queues 6
  - Node Information 5
  - Result Destinations 6, 26
  - Results 7
  - System Parameters 5
  - User Profile 2, 5
- time chart, zooming 79
- time range 80
- Total Table Hits report 68
- trace file
  - generating 92
- Tracker 4
  - common problems 98
  - creating data 67
  - exiting 82
  - starting 71
  - troubleshooting 98
- Tracker problems 98
- Tracker reports 68
- troubleshooting
  - DB2 Query Patroller clients 95
  - DB2 Query Patroller server 91
  - QueryEnabler 95
  - QueryMonitor 96
  - Tracker 98

**U**

- updating dqpnodes.cfg 11
- user administration 17, 18
- user exits 52
- User Profile table 2, 5
- user profiles
  - changing node status 25
  - creating 18
  - editing 21
  - effective 21
  - removing 21
  - resolving 18
- using QueryAdministrator 15

**V**

- variables
  - profile 39
- variables, result destination 27
- viewing
  - online information 112

**W**

- Windows
  - e-mail notification 35
  - profile variables 44
- wizards
  - add database 114, 115
  - backup database 114
  - completing tasks 114
  - configure multisite update 114
  - create database 115
  - create table 115
  - create table space 115
  - index 115
  - performance configuration 115
  - restore database 115

**Z**

- zooming on a time chart 79

---

## Contacting IBM

If you have a technical problem, please review and carry out the actions suggested by the *Troubleshooting Guide* before contacting DB2 Customer Support. This guide suggests information that you can gather to help DB2 Customer Support to serve you better.

For information or to order any of the DB2 Universal Database products contact an IBM representative at a local branch office or contact any authorized IBM software remarketer.

If you live in the U.S.A., then you can call one of the following numbers:

- 1-800-237-5511 for customer support
- 1-888-426-4343 to learn about available service options

---

## Product Information

If you live in the U.S.A., then you can call one of the following numbers:

- 1-800-IBM-CALL (1-800-426-2255) or 1-800-3IBM-OS2 (1-800-342-6672) to order products or get general information.
- 1-800-879-2755 to order publications.

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

The DB2 World Wide Web pages provide current DB2 information about news, product descriptions, education schedules, and more.

**<http://www.ibm.com/software/data/db2/library/>**

The DB2 Product and Service Technical Library provides access to frequently asked questions, fixes, books, and up-to-date DB2 technical information.

**Note:** This information may be in English only.

**<http://www.elink.ibm.com/pbl/pbl/>**

The International Publications ordering Web site provides information on how to order books.

**<http://www.ibm.com/education/certify/>**

The Professional Certification Program from the IBM Web site provides certification test information for a variety of IBM products, including DB2.

**ftp.software.ibm.com**

Log on as anonymous. In the directory /ps/products/db2, you can find demos, fixes, information, and tools relating to DB2 and many other products.

**comp.databases.ibm-db2, bit.listserv.db2-l**

These Internet newsgroups are available for users to discuss their experiences with DB2 products.

**On CompuServe: GO IBMDB2**

Enter this command to access the IBM DB2 Family forums. All DB2 products are supported through these forums.

For information on how to contact IBM outside of the United States, refer to Appendix A of the *IBM Software Support Handbook*. To access this document, go to the following Web page: <http://www.ibm.com/support/>, and then select the IBM Software Support Handbook link near the bottom of the page.

**Note:** In some countries, IBM-authorized dealers should contact their dealer support structure instead of the IBM Support Center.







Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

SC09-2958-00

