PATROL Knowledge Module™ for Rdb
User Guide

Version 3.2

September 30, 1999

bmcsoftware

This document is published by the BMC Software, Inc., Technical Publications Department.

BMC Software, Inc.
2101 CityWest Blvd.
Houston, TX 77042-2827
USA

Contacting BMC Software

For information about obtaining *technical support*, see **www.bmc.com/support.html**.

For all other purposes, contact BMC Software as follows:

USA and Canada

BMC Software, Inc.
2101 CityWest Blvd.
Houston TX 77042-2827
USA
Telephone: 800 841 2031
www.bmc.com

Outside USA and Canada

For a complete list of all BMC Software international offices and subsidiaries, see the BMC Software home page at **www.bmc.com**.

USA corporate headquarters—telephone:
(1) 713 918 8800

Contents

Index

About This Book

This book contains detailed information about the applications, commands, and parameters provided by the PATROL Knowledge Module™ for Rdb (also referred to as the PATROL KM for Rdb) and is intended for system administrators and database administrators (DBAs). Use this book with the appropriate user guide for your console.

This book also contains instructions for loading and configuring the PATROL Knowledge Module (PATROL KM) for Rdb. For more details, refer to the online Help for the PATROL KM.

Note

This book assumes that you are familiar with your host operating system. You should know how to perform basic actions in a window environment, such as choosing menu commands and dragging and dropping icons.

How This Book Is Organized

This book is organized as follows.

Chapter/Appendix Number and Title	Description
Chapter 1, “Introduction”	provides an overview of the features and components of the PATROL KM
Chapter 2, “Getting Started”	describes how to set up and access the PATROL KM
Chapter 3, “Menu Summary”	describes the commands that are available on the PATROL KM menus
Chapter 4, “Parameter Summary”	describes the parameters that you can use with the PATROL KM

Related Documentation

BMC Software products offer several types of documentation:

- online and printed books
- online Help
- release notes

Online and Printed Books

The books that accompany BMC Software products are available in online format and printed format. You can view online books with Acrobat Reader from Adobe Systems. The reader is provided at no cost, as explained in “To Access Online Books.” You can also obtain additional printed books from BMC Software, as explained in “To Request Additional Printed Books.”

To Access Online Books

You can access online books from the documentation compact disc (CD) that accompanies your product or from the World Wide Web.

Accessing Books with Acrobat Reader

Online books are formatted as Portable Document Format (PDF) files. You can view them, print them, or copy them to your computer by using Acrobat Reader 3.0 or later.

Note

In some cases, installation of Acrobat Reader and downloading the online books is an optional part of the product-installation process. For information about downloading the free reader from the Web, go to the Adobe Systems site at **http://www.adobe.com**. For information about installing the reader from the documentation CD, see the **readme.txt** file on the CD.

Accessing Books on the Web

To access any online book that BMC Software offers, visit the support page of the BMC Software Web site at **http://www.bmc.com/support.html**.

Note

To use the support page, you must be a registered user or have a temporary user name and a password.

If you are a registered user, proceed as follows:

1. Go to **http://www.bmc.com/support.html**.
2. Click **Product Directory** under the **SUPPORT CENTER - Login Required** heading.
3. Enter your user ID and password when prompted.
4. To view books for a particular product, click the product name in the Product Directory.
5. Select the book that you want to view.

If you are not a registered user, choose one of the following options:

- To register, go to **http://www.bmc.com/support.html**, click **First Time Users**, complete the form, and click **Submit Registration.**

  When you receive confirmation of your registration by e-mail, you can then access books on the Web.

- To request a temporary user name and a password, contact your BMC Software sales representative.

To Request Additional Printed Books

BMC Software provides a core set of printed books with your product order. To request additional books, go to **http://www.bmc.com/support.html**.

Online Help

You can access Help for a product through the product's Help menu. The online Help provides information about the product's graphical user interface (GUI) and provides instructions for completing tasks.

Release Notes

Printed release notes accompany each BMC Software product. Release notes provide up-to-date information such as

- updates to the installation instructions
- last-minute product information

The latest versions of the release notes are also available on the Web at **http://www.bmc.com/support.html**.

Conventions

The following conventions are used in this book:

- This book includes special elements called *notes*, *warnings*, *examples*, and *tips*:

Note

Notes provide additional information about the current subject.

Warning

Warnings alert you to situations that can cause problems, such as loss of data, if you do not follow instructions carefully.

Example

An example clarifies a concept discussed in text.

A tip provides useful information that may improve product performance or make procedures easier to follow.

- `All syntax, operating system terms, and literal examples are presented in this typeface.`

- In instructions, **boldface** type highlights information that you enter. File names, directories, and Web addresses also appear in boldface type.

- The symbol => connects items in a menu sequence. For example, **Actions => Create Test** instructs you to choose the Create Test command from the Actions menu.

- The symbol » denotes one-step instructions.

- In syntax, path names, or system messages, *italic* text represents a variable, as shown in the following examples:

  `The table` *`table_name`* `is not available.`

  system/*instance/file_name*

- In syntax, the following additional conventions apply:

  — A vertical bar (|) separating items indicates that you must choose one item. In the following example, you would choose *a*, *b*, or *c*:

  `a | b | c`

  — An ellipsis (. . .) indicates that you can repeat the preceding item or items as many times as necessary.

  — Square brackets ([ ]) around an item indicate that the item is optional.

- The following table shows equivalent mouse buttons for Unix users and Windows NT users:

Unix Button	Windows NT Button	Description
MB1	left mouse button	Click this button on an icon or menu command to select that icon or command. Click MB1 on a command button to initiate action. Double-click an icon to open its container.
MB2	not applicable	Click this button on an icon to display the InfoBox for the icon. To simulate MB2 on a two-button mouse, simultaneously press the two buttons (MB1 and MB3).
MB3	right mouse button	Click this button on an icon to display its pop-up menu.

Note

If you have a one-button mouse (such as an Apple Macintosh mouse), assign MB1 to that button. You should also define a user-selectable combination of option and arrow keys to simulate MB2 and MB3. For details, refer to the documentation for your emulation software.

1

Introduction

This chapter provides you with an overview of the PATROL Knowledge Module™ for Rdb (also referred to as the PATROL KM for Rdb).

This chapter presents the following topics:

PATROL KM Features

The PATROL Knowledge Module for Rdb contains the knowledge that PATROL uses during system monitoring, analysis, and management activities. A PATROL Knowledge Module (PATROL KM) is a file containing knowledge in the form of command descriptions, application, parameters, and recovery actions that PATROL can use to monitor Rdb events.

The PATROL KM for Rdb parameters allow you to analyze system performance quickly and easily because they can provide a detailed statement of all system activity over time. You can clearly identify peaks, troughs, and trends in the performance of system resources.

By enabling you to do problem detection, system optimization, trend analysis, capacity planning, and simultaneous multiple host management, the PATROL KM for Rdb helps you ensure that your Rdb installations run efficiently 24 hours a day.

The PATROL KM for Rdb supports Rdb versions 6.0-0 through 7.0-0 on VAX and on Alpha.

PATROL KM Components

The PATROL KM for Rdb includes six separate KM files:

- RDBMON60.km
- RDB60.km
- RDBMON61.km
- RDB61.km
- RDBMON70.km
- RDB70.km

Startup KM Files

RDB60.kml, RDB61.kml, and RDB70.kml are the startup (base) PATROL KMs. Each of these PATROL KMs contains a list of the monitor and database PATROL KMs for that particular Rdb version. When you load a startup PATROL KM, the other two PATROL KMs are loaded.

Monitor KM Files

RDBMON60.km, RDBMON61.km, and RDBMON70.km allow you to manage Rdb monitor instances.

Database KM Files

RDB60.km, RDB61.km, and RDB70.km allow you to manage Rdb database instances.

Where to Go from Here

The following table summarizes where to look for more information on using PATROL and the PATROL KM for Rdb.

If you want information on...	See...
How to set up the PATROL KM	Chapter 2, “Getting Started.”
What a certain menu command does	Chapter 3, “Menu Summary,” and the PATROL KM for Rdb online Help.
What a certain parameter does	Chapter 4, “Parameter Summary,” and the PATROL KM for Rdb online Help.

Getting Started

This chapter provides you with information that you will need to get started with the PATROL Knowledge Module™ for Rdb (also referred to as the PATROL KM for Rdb).

The following topics are discussed:

Setting Up the PATROL KM

The PATROL KM files are copied to your machine during the PATROL installation procedure described in the *PATROL Installation Guide* for your platform.

Software Requirements

- You must be running Rdb V6.0 or later.
- The PATROL Console for Unix version 3.2.05 or the PATROL Console for Windows NT version 3.2.06 must be the management software and must reside on the Console machine.
- The PATROL Agent for Unix, for Windows NT, for OS/2, and for OpenVMS version 3.2.05 and the Agent's configuration file must be installed on each monitored machine.
- The VMS_FILESYSTEM.km must be installed on the Console machine. This PATROL KM is a subset of the PATROL Knowledge Module™ for OpenVMS.
- You must use a PATROL Developer Console to install the PATROL KM, but the Developer Console is not required to monitor your Rdb installation.
- The PATROL KM for Rdb distribution files must have been extracted during the installation of PATROL.
- If you want the PATROL Agent to always start the PATROL KM for Rdb, the PATROL KM files must be installed on the same machine as the PATROL Agent and the Agent's configuration file.
- If you want to manually load the PATROL KM for Rdb via the Console and the PATROL KM has not already been installed, the PATROL KM files must reside on the Console machine so that you can load the files during the PATROL KM setup procedure.

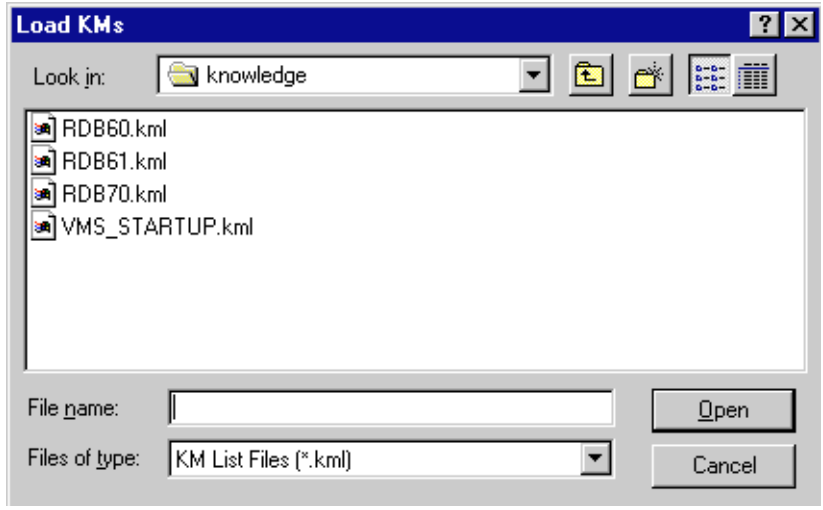
Setup Steps

Step 1 Load the PATROL KM for Rdb.

To load the PATROL KM from the PATROL Console, do the following:

1.A Choose **File => Load...** from the main menu. A list of available PATROL KMs for your site is displayed.

Figure 2-1 Knowledge Module File Selection Dialog Box



1.B Select one of the startup PATROL KMs (RDB60.kml, RDB61.kml, RDB70.kml) to load the two PATROL KM components (monitor and database) for the selected version of Rdb.

1.C Click **OK**.

Step 2 Supply a username and a password for local and remote machines. From the computer icon menu, choose **Customize => User Name/Password**.

Step 3 Save the PATROL KM changes. From the PATROL main menu, choose **File => Save KM**.

Post-Installation Instructions

The following procedures should be done after you have setup the PATROL KM for Rdb:

- Verify command files in sys$library
- Restrict databases to be monitored
- Check the status of applications

Verifying Command Files in sys$library

Check for the existence of the following command procedures in the `sys$library` of your OpenVMS system and make sure that their protections are world read and world execute

- DECRDB$SETVER.COM
- DECRDB$SHOVER.COM
- SQL$SETVER.COM
- SQL$SHOVER.COM

If your OpenVMS system does not have the above command files in `sys$library`, copy these files from `patrol$home:[lib.rdb]` and then set their protections to world read and world execute. Use the following copy procedures:

```
$ copy patrol$home:[lib.rdb]decrdb$setver.com sys$library
$ copy patrol$home:[lib.rdb]decrdb$shover.com sys$library
$ copy patrol$home:[lib.rdb]sql$setver.com sys$library
$ copy patrol$home:[lib.rdb]sql$shover.com sys$library
$ set protection = (w:re) sys$library:decrdb$setver.com
$ set protection = (w:re) sys$library:decrdb$shover.com
$ set protection = (w:re) sys$library:sql$setver.com
$ set protection = (w:re) sys$library:sql$shover.com
```

Restricting Databases to be Monitored

By default, the PATROL KM will discover all Rdb databases on the system. If you are not interested in viewing statistical data for some of the databases, you can turn off the discovery of those databases to save system resources. This is accomplished by creating a catalog file of all the databases you will be interested in. This file is a simple text file with the name of the database followed by a **Y** for "yes, display an icon for this database all the time" or an **N** for "never display this database." If you do not use a catalog file or if a particular database is not listed in the file, then PATROL displays the database only when it is in use.

If you use a catalog file, it must be located in the `SYS$LOGIN` directory of the process you are using on the agent machine. For example, a typical file could include the following:

```
sys$test:mf_personnel.rdb y
disk01:sample2.rdb n
$255$dua11:[garcia.rmu51.v11]rdbx51.rdb y
```

The name of the catalog file must be in the following form:

`RDBCAT60_`*`nodename`*`.CONF`

For example,

`RDBCAT60_OOTOOL.CONF`

where OOTOOL is the node name

Checking the Status of Applications

Check the status of the RDB*xx* and RDBMON*xx* applications in the PATROL KM for Rdb to verify that they are active. The procedures depend on whether you are using the PATROL Console for Unix or the PATROL Console for Windows NT.

Using the PATROL Console for Unix

To check the status of the applications using the PATROL Console for Unix:

Step 1 From the main PATROL Console window, choose **Attributes => Application Classes** from the menu bar. The List of Application Classes window appears.

Step 2 Look at the **Active** column for the RDB*xx* and RDBMON*xx* application classes. If the **Active** column contains **True**, then no further action is necessary. If the **Active** column contains **False**, follow the rest of the steps in this procedure.

Step 3 If the **Active** column for the RDB*xx* application class is **False**, double click the application name. The Application Definition dialog box appears.

Step 4 Click the **Active** button.

Step 5 Click the **OK** button. The dialog box closes.

Step 6 If the **Active** column for the RDBMON*xx* application class is **False**, repeat this procedure for the RDBMON*xx* application class.

Step 7 Save the PATROL KM changes by choosing **File => Save KM**.

The application classes are now active.

To check the status of the applications using the PATROL Console for Windows NT:

Step 1 From the KM tree of the main PATROL Console window, right click the RDB*xx* application class and choose **Properties** from the pop-up menu. The Properties dialog box appears.

Step 2 Click the **General** tab.

Step 3 If the **Active** check box is not selected, click the check box.

Step 4 Click the **OK** button. The dialog box closes.

Step 5 Repeat this procedure for the RDBMON*xx* application class.

Step 6 Save the KM by choosing **File => Save KM**.

The application classes are now active.

PATROL KM Applications

The PATROL KM for Rdb contains six applications, three Database applications and three Monitor applications:

- Rdb V6.0 Monitor
- Rdb V6.0 Database
- Rdb V6.1 Monitor
- Rdb V6.1 Database
- Rdb V7.0 Monitor
- Rdb V7.0 Database

Each application has its own naming convention, InfoBox, and application menu.

Rdb Monitor Applications

An Rdb monitor application instance is instantiated for each Rdb monitor.

Rdb Database Applications

An Rdb database application instance is instantiated for each Rdb database. An instance name is the same as the database name (for example, Production_personnel).

Accessing KM Information

This section contains instructions for obtaining information from the following:

- InfoBoxes
- application menus
- parameters

InfoBoxes

Each Rdb object being monitored has its own InfoBox that displays certain attributes about that particular object. For example, selecting the InfoBox for a database displays information such as the last time the transaction log was dumped or the date on which the database was created.

Application Menus

The PATROL KM menus contain frequently used commands for both monitors and databases. The PATROL KM also provides useful reports that summarize monitor and database data.

Menu command output is sent to the system output window for the computer on which the monitor is running. Double-click on the computer icon to display command output.

To access the application menus, perform one of the following actions:

- **With the PATROL Console for Unix**, click and hold MB3 on the application instance icon.

- **With the PATROL Console for Windows NT**, right-click the application instance icon.

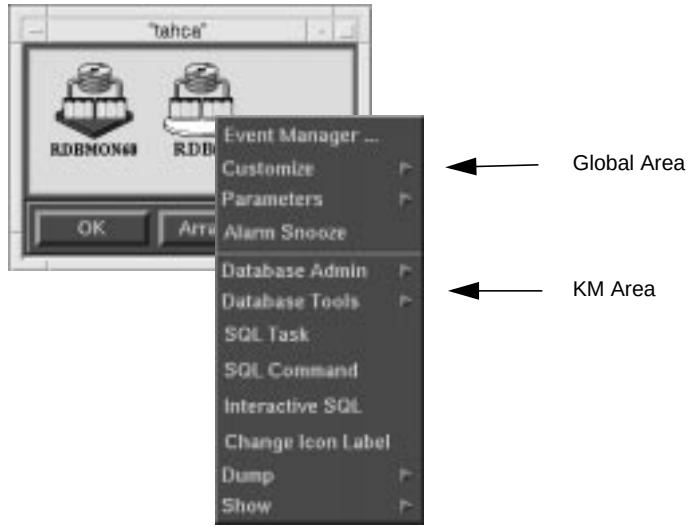


Figure 2-2 Accessing an Application Menu

Chapter 3, “Menu Summary,” briefly describes each menu item.

Parameters

The parameters defined in the PATROL KM for Rdb are activated by default. They continuously monitor key monitor and database resources and warn you of potential problems. All parameters in the PATROL KM are global parameters; that is, they automatically execute against all Rdb instances discovered.

The graphs and gauges do not appear immediately once activated. They appear only after a parameter receives its first data value from the PATROL Agent (that is, after one polling cycle). For example, if the poll time for a particular parameter is set to 10 minutes, then it will take at least 10 minutes for that particular parameter icon to appear in the application window.

Refer to Chapter 4, “Parameter Summary,” for details on parameters and their associated alarms and warnings.

Where to Go from Here

The following table summarizes where to look for more information on using PATROL and the PATROL KM for Rdb.

If you want information on...	See...
What a certain menu command does	Chapter 3, “Menu Summary,” and the PATROL KM for Rdb online Help.
What a certain parameter does	Chapter 4, “Parameter Summary,” and the PATROL KM for Rdb online Help.

Menu Summary

This chapter describes the application menus and menu items for the PATROL KM for Rdb. The menu items are listed alphabetically.

Menu Dictionary

AIP Pages	Displays an after-image journal (`.AIJ`) file or optimized after-image journal (`.OAIJ`) file in ASCII format. You may use this command to examine the contents of your `.AIJ` or `.OAIJ` file. An `.AIJ` file contains header information and data blocks. Header information describes the data blocks, which contain copies of data stored in the database file. Choose **Dump => AIP Pages** from a Database application menu.
Catalogue	Displays information about the specified catalogues. If you do not specify any aliases in the catalogue names, SQL displays information about all attached databases. Choose **Show => Catalogue** from a Database application menu.
Change Icon Label	Allows you to change the label of the icon representing this particular database. It is useful if you have multiple databases with the same, similar, or nondescriptive names. Choose **Change Icon Label** from a Database application menu.
Close DELPRC	Closes the database immediately without allowing current Rdb users to complete active transactions or detach from their databases. Each user process that was attached to an Rdb database is deleted immediately. This menu item is only displayed if the database is already open. Performs the same command as `RMU/CLOSE/ABORT=DELPRC`. Choose **Database Admin => Close DELPRC** from a Database application menu.
Close FORCEX	Closes the database immediately without allowing current Rdb users to complete active transactions or detach from their databases. However, the user processes are not deleted. This menu item is only displayed if the database is already open. Performs the same command as `RMU/CLOSE/ABORT=FORCEX`. Choose **Database Admin => Close FORCEX** from a Database application menu.
Close No Abort	Allows current user processes to continue and complete transactions. This menu item is only displayed if the database is already open. Performs the same command as `RMU/CLOSE/NOABORT`. Choose **Database Admin => Close No Abort** from a Database application menu.

Database Open	Opens a closed database. This menu item is only displayed if the database is closed. Choose **Database Admin => Database Open** from a Database application menu.
Database Tools (Database)	The name of the current database is passed to the tool as part of the startup. The following menu items are displayed: **RMU (Statistics** - Invokes the statistics utility **Graphical Schema Editor (GSE)** - Invokes Oracle Graphical Schema Editor **InstantSQL**- Invokes Oracle InstantSQL for Rdb **RdbExpert** - Invokes Oracle Expert for Rdb Choose **Database Tools** from a Database application menu.
Database Tools (Monitor)	Allows you to invoke the GSE, InstantSQL, or RdbExpert. No database name is passed to the tool. The PATROL Console attempts to direct the output of these GUI tools to the system specified by `PATROL$DISPLAY`. You should define this logical to point to the system where you are running the PATROL Console. With the pointer on the icon for the remote system, choose the **Customize => Environment** option. In this display, add a variable named `PATROL$DISPLAY` with the value *nodename*:0. Alternatively, you could add the following command to the `LOGIN.COM` for the account on the remote system to allow the remote node and username to access your workstation: `$ DEFINE PATROL$DISPLAY "MYNODE::0"` The following menu items are displayed **Graphical Schema Editor (GSE)** -Invokes Oracle Graphical Schema Editor **InstantSQL** - Invokes Oracle InstantSQL for Rdb **RdbExpert** - Invokes Oracle Expert for Rdb If the tools cannot be invoked, a OpenVMS system message is displayed in the system output window. Choose **Database Tools** from a Monitor application menu.
Domain	Displays the names, data types, and character sets of specified domains. If you specify the `SHOW DOMAINS` statement without any arguments, SQL displays names, data types, and character sets of all domains in all attached databases. Choose **Show => Domain** from a Database application menu.

Dump (Database)	Invokes the Rdb Dump utility. Three types of information can be displayed **Header** - Displays header information **Users** - Displays information aboaut current users **AIJ Pages** - Displays AIJ and OAIJ file information Choose **Dump** from a Database application menu.
Function (V6.0 and later)	Displays information about specified external functions. When you enter the `SHOW FUNCTIONS` statement without any arguments, SQL displays the name of the external function only. The following list shows the `SHOW FUNCTION` command options: **DESCRIPTION** - The description of the external function; if none exists, nothing displays. **ID** - The unique identification assigned to the external function **LANGUAGE** - The host language in which the external function is coded **MODULE** - The name of the module in which the function is defined **OWNER** - The owner of the external function **PARAMETER** - Information about the parameters, including the number of arguments, the data type, return type, and how the parameter is passed **SOURCE**- The source definitions for the specified external functions Choose **Show => Function** from a Database application menu.
Graphical Schema Editor (GSE)	Invokes the Oracle Graphical Schema Editor (GSE) if it is installed on your system. GSE allows you to graphically view, create, and modify an Rdb schema. Use windows to enter the information that the Schema Editor needs, and use pull-down menus to perform the various Schema Editor functions. All schemes saved in GSE can be exported to SQL files for database creation. Refer to *DEC Graphical Schema Editor (GSE) for Rdb/VMS User's Guide* for information about using GSE. Choose **Database Tools => Graphical Schema Editor (GSE)** from a Monitor application menu or from a Database application menu.
Header	Lists header information that includes database parameters, storage area parameters, snap-shot area for storage area parameters, and user information such as the number of active users. Header information also includes information on the use of `RMU/ALTER` on the specified database. Choose **Dump => Header** from a Database application menu.

Index	Displays information about specified indexes. SQL displays the name of the index, the associated column and table, the size of the index key, whether the definition allows duplicate values for the column, the type of index (sorted or hashed), and whether index compression is enabled or disabled. If you specify the `SHOW INDEXES` command without any arguments, SQL displays definitions of all indexes in all declared databases. The following list shows the `SHOW INDEXES` command arguments: **alias.*** - An asterisk wildcard, preceded by an optional alias. If you do not precede the wildcard with an alias, SQL displays information about indexes in the default database. If you precede the wildcard with an alias, SQL displays information about indexes in that database. **index-name** - The name of an index whose definition you want to display Choose **Show => Index** from a Database application menu.
InstantSQL	Invokes Oracle InstantSQL for Rdb if it is installed on your system. Using InstantSQL, you can graphically create queries, generate SQL modules and host language interface files, develop prototypes, and work with data and meta data all in the same environment. Refer to *DEC InstantSQL for Rdb/VMS Help Information* for information about using InstantSQL. Choose **Database Tools => InstantSQL** from a Monitor application menu or from a Database application menu.
Interactive SQL	Opens a window and invokes the interactive SQL command environment. You may enter commands and view output as if you were in a regular terminal window. Choose **Interactive SQL** from a Database application menu.
Journal (V6.0 and later)	Displays information about specified journals. SQL displays the name of the file specification and, if created, the backup file specification. Choose **Show => Journal** from a Database application menu.

Module (V6.0 and later)	Displays information about specified modules. If you specify the `SHOW MODULES` command without any options, SQL displays all module information. The following list shows the `SHOW MODULES` command options: **DESCRIPTION** - The description of the external function; if none exists, nothing displays **ID** - The unique identification assigned to the module **NAME** - The name of the module **OWNER** - The owner of the module. If the module is an invoker's rights module, this file is set to `NULL` and does not display anything. If the module is a definer's rights module, the definer's user name displays. **PROCEDURES** - The procedures contained in the module The following defines the `SHOW MODULES` command argument: **alias.*** - An asterisk wildcard, preceded by an optional alias. If you do not precede the wildcard with an alias, SQL displays information about the module in the default database. If you precede the wildcard with an alias, SQL displays information about modules in that database. Choose **Show => Module** from a Database application menu.
Outline (V6.0 and later)	Displays the definition of the specified outline. SQL displays the outline name, ID number, mode, query, compliance, and comment if one exists. If you use the `SHOW OUTLINE` command without the name of a specific outline, the names of all the outlines stored in the database are displayed. However, the invalid outlines are not marked as invalid. Choose **Show => Outline** from a Database application menu.
Procedure (V6.0 and later)	Displays information about stored procedures. If you specify the `SHOW PROCEDURES` command without any options, SQL displays all procedure information. The following table lists the `SHOW PROCEDURES` options. **DESCRIPTION** - The description of the stored procedure; if none exists, nothing displays **ID** - The unique identification assigned to the procedure **MODULE** - The identification number of the module to which a procedure belongs **OWNER** - The owner of the procedure **PARAMETER** - The parameters contained in the module **SOURCE** - The source definitions for the specified modules The following defines the `SHOW PROCEDURES` command argument: **alias.*** - An asterisk wildcard, preceded by an optional alias. If you do not precede the wildcard with an alias, SQL displays information about the procedure in the default database. If you precede the wildcard with an alias, SQL displays information about procedures in that database. Choose **Show => Procedure** from a Database application menu.

RdbExpert	Invokes Oracle Expert for Rdb if it is installed on your system. Oracle Expert is integrated with Rdb and Oracle Trace so that, for existing databases, much of the required information can be captured directly from the database and its associated database applications during normal production activities. Refer to the *DEC RdbExpert for VMS User's Guide* for information about using Oracle Expert. Choose **Database Tools => RdbExpert** from a Monitor application menu or from a Database application menu.
Reopen Log	Closes the current Rdb monitor log file and opens another log file (`SYS$SYSTEM:RDMMON.LOG`) without stopping the monitor. This menu item is only displayed if the monitor is already running. Output is displayed in the system output window. The output tells you whether the operation was successful. You should use this command if the monitor log file gets too large. For example, if you are running out of space on your disk or if database performance slows, you may want to open another log file. If the disk that contains the Rdb monitor log file becomes full, you or your system manager should acquire space on the disk. Once there is sufficient space on this disk, use the Reopen Log command and consider backing up (using the `DCL COPY` command or the OpenVMS Backup utility) the old monitor log file on tape. When the disk that contains the monitor log becomes full, Rdb stops writing to the log file, but the Rdb system does not stop operating. When the disk becomes full, a message is sent to the cluster system operator. Performs the same command as `RMU/MONITOR REOPEN_LOG`. Choose **Monitor => Reopen Log** from a Monitor application menu.
RMU (Statistics)	Invokes the Statistics utility for Rdb. This menu item performs the same function as the DCL command `RMU/SHOW STATISTICS` *databasename.* Choose **Database Tools => RMU** from a Database application menu.
Schema	Displays the names of specified schemas. If you do not specify an alias as part of a schema name, SQL displays schema information for all of the attached databases. For each database that is not multischema, SQL displays the message, “No schemes found.” For each multischema database, SQL displays the alias, followed by a list of schemes contained in that database. Each schema name in the list is preceded by the catalogue and alias names. Choose **Show => Schema** from a Database application menu.

Show (Database)	Displays a variety of database characteristics. These commands function the same as the following SHOW *keyword* command in interactive SQL. **Catalogue** - Displays catalogue information **Domain** - Displays domain information **Function** - Displays external function information **Index** - Displays index information **Journal** - Displays journal information **Module** - Displays module information **Outline** - Displays outline information **Procedure** - Displays information about stored procedures **Schema** - Displays schema information **Storage Area** - Displays information about storage areas **Storage Map** - Displays information about storage maps **Table** - Displays table information **Trigger** - Displays trigger information **View** - Displays view information Choose **Show** from a Database application menu.
SQL Command	Opens a window where you can execute a single SQL command. Output is displayed in the system output window. Choose **SQL Command** from a Database application menu.
SQL Task	Opens a window where you can execute any SQL statement. An icon is displayed in the window that shows the total execution time. Output from the statement is displayed in a task window. This menu item is useful because it allows you to repeat or destroy the task. Choose **SQL Task** from a Database application menu.
Start (Monitor)	Starts the database monitor on the local node. This menu item is only displayed if the monitor is not running. If the monitor cannot be started, an OpenVMS system message is displayed in the system output window. Choose **Monitor => Start** from a Monitor application menu.
Stop DELPRC Abort	Stops the monitor immediately without allowing current Rdb users to complete active transactions or detach from their databases. Each user process that was attached to an Rdb database is deleted immediately. This menu item is only displayed if the monitor is already running. Output is displayed in the system output window. The output tells you whether the operation was successful. Performs the same command as `RMU/MONITOR STOP/ABORT=DELPRC`. Choose **Monitor => Stop DELPRC Abort** from a Monitor application menu.

Stop FORCEX Abort	Stops the monitor immediately without allowing current Rdb users to complete active transactions or detach from their databases. However, the user processes are not deleted. This menu item is only displayed if the monitor is already running. Output is displayed in the system output window. The output tells you whether the operation was successful. Performs the same command as `RMU/MONITOR STOP/ABORT=FORCEX`. Choose **Monitor => Stop FORCEX Abort** from a Monitor application menu.
Stop No Abort	Allows current user processes to continue and complete before stopping. This menu item is only displayed if the monitor is already running. Output is displayed in the system output window. The output tells you whether the operation was successful. New users on the node will not be allowed to attach to any database, but existing database users will be able to complete their sessions normally. Once existing database user processes terminate, the database monitor shuts down. Performs the same command as `RMU/MONITOR STOP`. Choose **Monitor => Stop No Abort** from a Monitor application menu.
Storage Area	Displays information about storage areas. If you do not specify a wildcard or list of storage area names, SQL displays the names of all the storage areas in all attached databases. Choose **Show => Storage Area** from a Database application menu.
Storage Map	Displays information about storage maps. If you do not specify a wildcard or list of storage map names, SQL displays the names of all the storage maps in all attached databases. Choose **Show => Storage Map** from a Database application menu.
System	Displays summary information about all databases accessed on a single node. Choose **Show => Show System** from a Monitor application menu.
Table	Displays information about tables and views. If you do not specify a wildcard or list of table and view names, SQL displays the names of all the tables and views in all attached databases. If you do not specify any of the `SHOW TABLES` options, all table information is displayed including the character set for each column of the specified table. Choose **Show => Table** from a Database application menu.

Trigger	Displays information about triggers. If you do not specify a wildcard or a trigger name, SQL displays the names of all the triggers in all attached databases. The following list shows the `SHOW TRIGGERS` command arguments: **alias.*** - An asterisk wildcard, preceded by an optional alias. If you do not precede the wildcard with an alias, SQL displays information about triggers in the default database. If you precede the wildcard with an alias, SQL displays information about triggers in that database. **trigger-name** - The name of a trigger whose definition you want to display Choose **Show => Trigger** from a Database application menu.
Users (Monitor)	Displays detailed information on all databases accessed on a single node. Choose **Show => Show Users** from a Monitor application menu.
Users (Database)	Lists information about the current users of the database including all users in a VAXcluster or VMScluster environment. Choose **Dump => Users** from a Database application menu.
View	Displays information about views. If you do not specify a wildcard or list of view names, SQL displays the names of all the views in all attached databases. Choose **Show => View** from a Database application menu.

Where to Go from Here

The following table summarizes where to look for more information on using PATROL and the PATROL KM for Rdb.

If you want information on...	See...
What a certain parameter does	Chapter 4, “Parameter Summary,” and the PATROL KM for Rdb online Help.

Parameter Summary

This chapter describes the PATROL KM parameters for Rdb monitors and databases.

The following topics are discussed:

About the PATROL KM Parameters

There are no recovery actions set by default for the PATROL KM for Rdb. You must define the recovery actions.

Currently, the PATROL KM for Rdb has one collector parameter that collects data for all other (consumer) parameters.

Each monitor parameter supports all Monitor applications, and each database parameter supports all Database applications.

All parameter history data is stored in a single file named `param.hist` located in `PATROL$HOME:[LOGHISTORY.`*`nodename.portnumber`*`]`. Refer to the *PATROL User Guide* for detailed information about the parameter history file.

Parameter Descriptions

The following pages describe the attributes and other related information for each parameter. The parameters are organized alphabetically.

The prefix of a consumer parameter's name identifies whether the application class is a monitor or database application. There is one collector parameter with a unique prefix.

Table 4-1 Parameter Prefixes

Prefix	Description
RDB	Rdb Database Consumer
RDBMON	Rdb Monitor Consumer
RMU	Database and Monitor Collector

RDB_aij_reads

Description: Displays the number of read QIOs issued to the database `.AIJ` file. If after-image journaling is not enabled for the database, this parameter will always be zero.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_aij_writes

Description: Displays the total number of write QIOs (queued I/O requests) issued to the database after-image journal (`.AIJ`) file. This operation writes after-image records to the `.AIJ` file to facilitate rollforward recovery using the `RMU/RECOVER` command. If after-image journaling is not enabled for the database, this parameter will always be zero.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_attaches

Description: Displays the number of current attaches to the database.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_blasts

Description: Monitors the number of blocking ASTs, sometimes referred to as blasts, delivered to Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected. When Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_buf_unmark

Description: This parameter is incremented each time a modified buffer is written back to disk. Its value is equal to the sum of the 14 following fields. These fields as well as the SPAM page field provide further detail about buffer unmark activity. Write operations of buffers that contain SPAM pages are included in the total although they are also counted separately by the SPAM page field.

- transaction
- pool overflow
- blocking AST
- lock quota
- lock conflict
- user unbind
- batch rollback
- new area mode
- larea change
- incr backup
- no aij access
- truncate snaps
- checkpoint
- aij backup

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_check_pts

Description: Displays the current number of checkpoints per minute. If most of the checkpoints for a database are triggered by a particular checkpoint limit, that limit may be set too high, or the other two limits may be set too low. You can determine the average interval per checkpoint for each type of checkpoint limit. After you have this information, you can reset the limits so that each type of checkpoint limit triggers approximately the same number of checkpoints, which results in optimal performance.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_df_reads

Description: Displays the number of read QIOs (queued I/O requests) issued to the database storage area for single-file and multifile databases and snapshot files. This operation reads database pages synchronously from the database.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_df_writes

Description: Displays the number of write QIOs (queued I/O requests) issued to the database storage area for single-file and multifile databases and snapshot files. This operation writes modified database pages synchronously back to the database.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_dup_nd_ins

Description: Displays the number of duplicate index keys inserted into the database's indexes. There should be a one-to-one correspondence to the number of duplicate records being stored in the tables.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_fetch_read

Description: Displays the number of synchronous data page requests to the PIO subsystem where only read privileges are being requested for the page. If Rdb reads any area inventory (AIP) pages and area bit map (ABM) pages while fetching the data page, the requests for the AIP and ABM pages are included in the total count field. The sum of the fetch for read and fetch for write fields equals the total number of synchronous data page requests to the PIO subsystem.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_fetch_upd

Description: Displays the number of data page requests to the PIO subsystem where update and read privileges are being requested for the page. If Rdb reads any AIP and ABM pages while fetching the data page, the requests for the AIPs and ABM pages are included in the total count field. The sum of the fetch for read and fetch for write fields equals the total number of data page requests to the PIO subsystem.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_free_global

Description: This parameter displays the current percentage of free global buffers.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Gauge

Unit: Average percentage per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_hash_del

Description: Displays the number of hash key deletions from the database's hashed indexes. It includes unique key deletions and duplicate key deletions.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_hash_dup_ins

Description: Displays the number of duplicate hash key insertions in the database's hashed indexes.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_hash_ins

Description: Displays the number of hash key insertions in the database's hashed indexes. It includes unique key insertions and duplicate key insertions.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_lck_conf_unmark

Description: This parameter is incremented by 1 for each modified buffer that is written back to disk to reduce the possibility of a deadlock when Rdb discovers a lock conflict.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_lock_dem

Description: Displays the number of $ENQ lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_lock_req

Description: Displays the number of lock requests (also referred to as $ENQ lock requests) for new locks. Whether the lock request succeeds or fails, it is included in this count.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_overflow_unmark

Description: This parameter is incremented by 1 for each modified buffer that is written back to disk as a result of a request to read in a new page from disk.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_recoveries

Description: Displays the current number of detached database recovery (DBR) processes acting on this database.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Gauge

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

See Also: RDBMON_recoveries

RDB_rt_nd_rem

Description: Displays the number of index entries removed from a root node because of deletion of entries within lower-level nodes. If an index consists of only one node, removals from this node are not included in this field; but are included in the leaf removals field.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: min =1, max = 10

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_rt_nd_ins

Description: Displays the number of entries inserted into the root (top-level) index nodes. The number of insertions should be small except when you load the database. If an index consists of only one node, insertions into this node are not included in this field but are included in the "leaf insertions field.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_rt_reads

Description: Displays the number of read QIOs (queued I/O requests) issued to the database root (`.Rdb`) file. Rdb reads the `.Rdb` file when a new user attaches to the database and when an `.Rdb` file control block needs to be updated because of database activity on another VMScluster node.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_rt_writes

Description: Displays the number of write QIOs (queued I/O requests) issued to the database root (`.Rdb`) file. Rdb writes to the `.Rdb` file when a user issues a `COMMIT` or `ROLLBACK` statement. Other events also cause updates to the `.Rdb` file.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_ruj_reads

Description: Displays the number of read QIOs (queued I/O requests) issued to the database recovery unit journal (`.RUJ`) files. This operation reads before-image records from the `.RUJ` file to roll back a verb or a transaction.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_ruj_writes

Description: Displays the number of write QIOs (queued I/O requests) issued to the database recovery unit journal (`.RUJ`) files. This operation writes before-image records to the `.RUJ` file in case a verb or transaction must be rolled back. Before-images must be written to the `.RUJ` file before the corresponding database page can be written back to the database.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_trans_cnt

Description: Displays the number of completed database transactions. It is the count of the `COMMIT` and `ROLLBACK` statements that have executed.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDB_txn_unmark

Description: This parameter is incremented by 1 for each modified buffer that is written back to disk as a result of a `COMMIT` or `ROLLBACK` statement.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Recovery Actions: Not applicable

Status: Active

Parameter Type: Consumer

Value Set By: RMU_stats

RDBMON_attaches

Description: Displays the number of current attaches to all databases on this system.

Command Type: Not applicable

Application Class: RDBMON60, RDBMON61, RDBMON70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDBMON_databases

Description: Displays the number of open databases on this system.

Command Type: Not applicable

Application Class: RDBMON60, RDBMON61, RDBMON70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

RDBMON_recoveries

Description: Displays the current number of detached database recovery (DBR) processess on this system.

Command Type: Not applicable

Application Class: RDBMON60, RDBMON61, RDBMON70

Icon: Graph

Unit: Average value per minute

Alert Range: Undefined

Scheduling: Depends on the poll time of the collector

Status: Active

Recovery Actions: Not applicable

Parameter Type: Consumer

Value Set By: RMU_stats

See Also: RDB_recoveries

RMU_stats

Description: This parameter is the collector for all PATROL KM parameters.

Command Type: Not applicable

Application Class: RDB60, RDB61, RDB70

Icon: None

Unit: None

Alert Range: Undefined

Scheduling: 10 minutes

Status: Active

Recovery Actions: Not applicable

Parameter Type: Collector

Value Set By: Not applicable

Where to Go from Here

The following table summarizes where to look for more information on using PATROL and the PATROL KM for Rdb.

If you want information on...	See...
What a certain menu command does	Chapter 3, “Menu Summary,” and the PATROL KM for Rdb online Help.

Index

Symbols

A

B

C

D

F

G

H

I

J

K

L

M

O

P

R

S

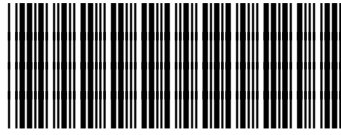
T

U

V

W

Notes



100024905