# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREFACE</td>
<td>1</td>
</tr>
<tr>
<td>PURPOSE OF THIS HANDBOOK</td>
<td>1</td>
</tr>
<tr>
<td>INTENDED AUDIENCE</td>
<td>1</td>
</tr>
<tr>
<td>HANDBOOK CONTENTS</td>
<td>1</td>
</tr>
<tr>
<td>ORACLE CODASYL DBMS</td>
<td>2</td>
</tr>
<tr>
<td>OPERATING SYSTEM INFORMATION</td>
<td>2</td>
</tr>
<tr>
<td>RELATED MANUALS</td>
<td>2</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>2</td>
</tr>
<tr>
<td>AUTHOR</td>
<td>3</td>
</tr>
<tr>
<td>RMU SHOW STATISTIC IO STATISTICS (BY FILE) SCREENS</td>
<td>4</td>
</tr>
<tr>
<td>FILE IO OVERVIEW SCREEN</td>
<td>5</td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>5</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>5</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>6</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>7</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>8</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>9</td>
</tr>
<tr>
<td>DEVICE IO OVERVIEW SCREEN</td>
<td>12</td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>12</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>12</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>12</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>13</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>13</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>14</td>
</tr>
<tr>
<td>DEVICE INFORMATION SCREEN</td>
<td>16</td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>16</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>16</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>16</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>17</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>18</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>18</td>
</tr>
<tr>
<td>FILE IO STATISTICS SCREEN</td>
<td>19</td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>19</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>20</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>21</td>
</tr>
<tr>
<td>Screen Name</td>
<td>Section Details</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>Screen Information</td>
<td>Configuration Options</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN INFORMATION</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>SCREEN LOCATION</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>SCREEN EXAMPLE</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>SCREEN FIELDS</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>ON-SCREEN MENU OPTIONS</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN INFORMATION</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN LOCATION</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN EXAMPLE</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN FIELDS</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN LOCATION</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN EXAMPLE</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN FIELDS</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN LOCATION</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN EXAMPLE</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN FIELDS</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN LOCATION</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN EXAMPLE</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN FIELDS</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN LOCATION</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN EXAMPLE</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN FIELDS</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN LOCATION</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>SCREEN EXAMPLE</td>
</tr>
<tr>
<td>Chapter</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>85</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>85</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>88</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>88</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>88</td>
</tr>
<tr>
<td><strong>ACTIVE LOCKS SCREEN</strong></td>
<td><strong>89</strong></td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>89</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>89</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>89</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>92</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>92</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>92</td>
</tr>
<tr>
<td><strong>MEMBIT LOCKS SCREEN</strong></td>
<td><strong>93</strong></td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>93</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>93</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>93</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>96</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>96</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>96</td>
</tr>
<tr>
<td><strong>AIJ LOCKS SCREEN</strong></td>
<td><strong>97</strong></td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>97</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>97</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>98</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>100</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>100</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>101</td>
</tr>
<tr>
<td><strong>SNAPSHOT LOCKS SCREEN</strong></td>
<td><strong>102</strong></td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>102</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>102</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>102</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>105</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>105</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>105</td>
</tr>
<tr>
<td><strong>FREEZE LOCKS SCREEN</strong></td>
<td><strong>106</strong></td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>106</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>106</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>106</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>109</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>109</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>109</td>
</tr>
<tr>
<td><strong>QUIET POINT LOCKS SCREEN</strong></td>
<td><strong>110</strong></td>
</tr>
<tr>
<td>SCREEN LOCATION</td>
<td>110</td>
</tr>
<tr>
<td>SCREEN EXAMPLE</td>
<td>110</td>
</tr>
<tr>
<td>SCREEN FIELDS</td>
<td>111</td>
</tr>
<tr>
<td>ON-SCREEN MENU OPTIONS</td>
<td>113</td>
</tr>
<tr>
<td>SCREEN INFORMATION</td>
<td>113</td>
</tr>
<tr>
<td>CONFIGURATION OPTIONS</td>
<td>114</td>
</tr>
<tr>
<td>Screen Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Stall Time X100 Screen</td>
<td>176</td>
</tr>
<tr>
<td>File Locking Overview Screen</td>
<td>181</td>
</tr>
<tr>
<td>Lock Statistics (by file) Screen</td>
<td>185</td>
</tr>
<tr>
<td>RMU Show Statistic Locking Statistics (by file) Screens</td>
<td>180</td>
</tr>
<tr>
<td>General Information Screen</td>
<td>190</td>
</tr>
<tr>
<td>Buffer Information Screen</td>
<td>195</td>
</tr>
<tr>
<td>Lock Information Screen</td>
<td>201</td>
</tr>
</tbody>
</table>
# RMU SHOW STATISTIC HANDBOOK

<table>
<thead>
<tr>
<th>Screen Information</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AUDIT INFORMATION SCREEN</strong></td>
<td>233</td>
</tr>
<tr>
<td>Screen Location</td>
<td>233</td>
</tr>
<tr>
<td>Screen Example</td>
<td>233</td>
</tr>
<tr>
<td>Screen Fields</td>
<td>233</td>
</tr>
<tr>
<td>On-Screen Menu Options</td>
<td>235</td>
</tr>
<tr>
<td>Screen Information</td>
<td>235</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>235</td>
</tr>
<tr>
<td><strong>ACTIVE USER INFORMATION SCREEN</strong></td>
<td>236</td>
</tr>
<tr>
<td>Screen Location</td>
<td>236</td>
</tr>
<tr>
<td>Screen Example</td>
<td>236</td>
</tr>
<tr>
<td>Screen Fields</td>
<td>236</td>
</tr>
<tr>
<td>On-Screen Menu Options</td>
<td>238</td>
</tr>
<tr>
<td>Screen Information</td>
<td>238</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>238</td>
</tr>
<tr>
<td><strong>STATISTICS EVENT INFORMATION SCREEN</strong></td>
<td>239</td>
</tr>
<tr>
<td>Screen Location</td>
<td>239</td>
</tr>
<tr>
<td>Screen Example</td>
<td>239</td>
</tr>
<tr>
<td>Screen Fields</td>
<td>239</td>
</tr>
<tr>
<td>On-Screen Menu Options</td>
<td>241</td>
</tr>
<tr>
<td>Screen Information</td>
<td>242</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>243</td>
</tr>
<tr>
<td><strong>OPENVMS SYSGEN PARAMETERS SCREEN</strong></td>
<td>244</td>
</tr>
<tr>
<td>Screen Location</td>
<td>244</td>
</tr>
<tr>
<td>Screen Example</td>
<td>244</td>
</tr>
<tr>
<td>Screen Fields</td>
<td>245</td>
</tr>
<tr>
<td>On-Screen Menu Options</td>
<td>247</td>
</tr>
<tr>
<td>Screen Information</td>
<td>248</td>
</tr>
<tr>
<td>Configuration Options</td>
<td>248</td>
</tr>
</tbody>
</table>
Oracle Rdb is a general-purpose high-end database management system based on the relational data model. Applications operate in a client-server environment when accessing the database. The RMU Show Statistic utility is used by DBAs to monitor the operation of the application and analyze performance characteristics of the database.

This handbook describes the various screens available for the RMU Show Statistic utility. The information contained in this handbook is an amalgamation of various sources of information on the utility, combined into a common frame of reference.

This handbook is not intended to be a reference manual nor should the handbook be construed as a tutorial on how to use the utility. Rather, the information contained in this handbook should be useful for anyone interested in detecting, analyzing and correcting performance problems with the database or application.

This handbook is intended for experienced database administrators familiar with the RMU Show Statistic utility. You should also be familiar with database management procedures and terminology.

The RMU Show Statistic “DBA Handbook” is comprised of five volumes. These are the following:

- Volume 1 - Methods and Internals.
- Volume 2 - Screen Chapters 1 through 7.
- Volume 3 - Screen Chapters 8 Through 13.
• Volume 4 - Screen Chapters 14 Through 22.
• Volume 5 - Appendix Information.

This handbook discusses the Oracle Rdb product. However, the RMU Show Statistic utility is almost identical for the Oracle CODASYL DBMS product DBO Show Statistic utility; in fact, it uses a common same source code. Therefore, the few statistic screens that are unique to the Oracle DBMS product are included in this handbook.

The information contained in this handbook is, for the most part, operating system independent. However, on the rare occasion when an operating system specific item is discussed, the section will be appropriately high-lighted.

For more information on the RMU Show Statistic utility, and database performance tuning information in general, please refer to the following Oracle Rdb documentation:
• Oracle Rdb RMU Reference Manual
• Oracle Rdb Guide to Database Maintenance
• Oracle Rdb Guide to Database Performance and Tuning

In addition, there are several “white papers” discussing RMU Show Statistic issues available in MetaLink. For information on the MetaLink system, please contact Oracle Client Relations or visit the Oracle Support webpage at

http://www.oracle.com/support/

A special word of thanks is extended to Simon Pickering for writing the initial version of the Handbook back in 1995, upon which this new version is loosely based.

A word of thanks is also extended to Kathy Oakey, Bill Gettys, Lilian Hobbs and all the other Oracle Rdb consultants who begged and pleaded for more detailed performance analysis screens to make their job easier. Without these suggestions and ideas, the RMU Show Statistic utility would not be as powerful as it is today.

Finally, I have to thank Anna Logan, who wrote the Guide to Database Performance and Tuning manual. She diligently kept after me to keep the “help” documentation up-to-

ACKNOWLEDGMENTS
date, even when I was adding screens to the RMU Show Statistic was utility on a daily basis.

If you have any comments, suggestions or corrections for this handbook, please forward them to the author at the following email address:

rjanders@us.oracle.com
IO Statistics (By File) Screens

RMU Show Statistic IO Statistics (By File) Screens
The “IO Statistics (By File)” menu screens display information about specific storage area I/O operations.
File IO Overview Screen

This screen shows a summary comparison of I/O activity for all database files, including the rootfile, after-image journals, RUJ journals, ACE journals, data and snapshot storage areas. The types of database files can be filtered by name and file type, and the statistic information displayed on this screen can be sorted using various criteria.

This screen displays comparison information about I/O activities that are specific to storage areas and snapshot files. This information is vital in determining which storage areas have the most I/O activity, and analyzing the validity of storage area partitioning.

Note that, unlike the “File Locking Overview” screen, the “File IO Overview” screen does display information about after-image journals, RUJ journals and ACE journals.

You cannot use the information contained on the “File IO Overview” screen on the “Custom Statistics” screen. Also, information on the statistic information files (.RDS) is not available.

This screen resides in the “IO Statistics (by file)” menu.

The following is an example of the “File IO Overview” screen:
Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor 30-JAN-1998 15:35:32.88
Rate: 1.00 Second    File IO Overview (Unsorted total I/O) Elapsed: 02:04:10.09
Page: 1 of 1    DISK$: [DB_HOMEDIR.WORK.STATS] MF_PERSONNEL.RDB; 1 Mode: Online

<table>
<thead>
<tr>
<th>File/Storage.Area.Name........</th>
<th>Sync.Reads</th>
<th>SyncWrites</th>
<th>AsyncReads</th>
<th>AsyncWrits</th>
<th>PgDis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Root</td>
<td>26</td>
<td>6206</td>
<td>0</td>
<td>29578</td>
<td>0</td>
</tr>
<tr>
<td>AIJ (After-Image Journal)</td>
<td>489</td>
<td>27368</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>RUJ (Recovery-Unit Journal)</td>
<td>3</td>
<td>2791</td>
<td>0</td>
<td>20271</td>
<td>0</td>
</tr>
<tr>
<td>ACE (AIJ Cache Electronic)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>All data/snap files</td>
<td>491</td>
<td>34773</td>
<td>0</td>
<td>57344</td>
<td>296k</td>
</tr>
<tr>
<td>data MF_PERS_DEFAULT</td>
<td>66</td>
<td>9467</td>
<td>0</td>
<td>16534</td>
<td>1113</td>
</tr>
<tr>
<td>data DEPARTMENTS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data EMPIDS_LOW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data EMPIDS_MID</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data EMPIDS_OVER</td>
<td>420</td>
<td>25306</td>
<td>0</td>
<td>40810</td>
<td>295k</td>
</tr>
<tr>
<td>data EMP_INFO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data JOBS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data MF_PERS_SEGSTR</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data SALARY_HISTORY</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap MF_PERS_DEFAULT</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap DEPARTMENTS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap EMPIDS_LOW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap EMPIDS_MID</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap EMPIDS_OVER</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap EMP_INFO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap JOBS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap MF_PERS_SEGSTR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap SALARY_HISTORY</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Config Exit Filter Help Menu >next_page <prev_page Options Reset Set_rate Write

**SCREEN FIELDS**

- **File/ Storage Area Name**: This field identifies the particular database rootfile, live or snapshot storage area, or the after-image journal, RUJ journal, and ACE journal. Note that the summation information for all live and snapshot storage areas is identified as “All data/snap files”.

- **Sync Reads**: This field gives the number of synchronous read QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases and snapshot files. This operation reads database pages synchronously from the database.

- **Sync Writes**: This field gives the number of synchronous write QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP). This operation writes modified database pages synchronously back to the database.

- **Async Reads**: This field gives the number of asynchronous read QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snap-
shot (.SNP) files. This operation reads database pages asynchronously from the database.

Async Writs This field gives the number of asynchronous write QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP). This operation writes modified database pages asynchronously back to the database.

PgCkd: This field indicates the number of pages checked in order to store a record. Ideally, very few candidate pages need to be checked when storing a record. However in certain cases, depending on record size, access method, locked space on a page, and SPAM thresholds, storing a record requires a number of page fetches.

PgD is This field identifies the number of pages checked but discarded because the actual free space on that page did not meet the physical requirements needed to store a new record. A discarded page is an indication of wasted resources and decreased throughput and should be more closely examined.

Ideally, this field should always display the value “0”. The screen example above identifies the EMPIDS_OVER storage area as being exceedingly bad and in immediate need of further analysis.

---

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Filter By typing “F” to select the filter on-screen menu option, you can control the display of storage areas by filtering out unwanted storage area names.

The filter option allows you to enter a search string which is used to filter the names of selected storage areas. Only those storage areas whose name contains the specified search string are displayed. The search string may contain one or both of the wildcard characters. The asterisk (*) wildcard character is mapped to zero or more characters and the percent (%) wildcard character is mapped to exactly one character. Note that the search string is not case sensitive.

The filter menu option is highlighted when a search string is actively filtering storage areas. To disable filtering, press the Return key at the search string prompt.

Options This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

---

1 The PgCkd field was replaced by the PgD is field starting with Oracle Rdb 7.0.3.0. The number of pages discarded is more relevant to the detecting excessive pages being accessed to store a new record.
This section discusses screen-specific issues.

The “File IO Overview” screen shows the synchronous and asynchronous read and write I/O counts for all storage areas, the after-image journal (.AIJ), the recovery-unit journal (.RUJ), the AIJ cache on electronic disk (.ACE), the database root file and, finally, all data and snapshot areas combined.

In the default display configuration of the File IO Overview display, the first five rows displayed on the first page are the database root (.RDB) file, the after-image journal (.AIJ) file, the recovery-unit journal (.RUJ) file, the AIJ cache on electronic disk journal (.ACE), and the total read and write statistics for all the database storage (.RDA) and snapshot (.SNP) areas. Note that the statistics for all the storage and snapshot areas include values from the storage and snapshot areas displayed on other pages. These five rows are not repeated on subsequent display pages.

Following the first five rows are the list of database storage areas, identified by the prefix ‘data’, followed by snapshot areas, identified by the prefix ‘snap’. Storage areas that are added to the database are automatically shown in the display.

Using the “File IO Overview” screen makes it easier for you to identify the set of storage areas that are performing an excessive number of synchronous I/O operations. For example, a large number of synchronous write I/O operations should cause you to examine the storage area thresholds for this area to determine the cause of the problem.

When Oracle Rdb is attempting to store a record, it sometimes reads a target page based on an acceptable threshold on the SPAM page, but determines after reading the data page that not enough space is available on the page to store the record.

This behaviour causes a high ratio of pages checked to records stored on the RMU Show Statistics utility “Record Statistics” screen. This behaviour can potentially cause very slow insert performance if a large number of pages needs to be examined for each record stored.

After you detect this problem (that Oracle Rdb is checking excessive pages while attempting to insert a record), it is very difficult to determine which storage area is exhibiting the behaviour and the exact cause of the problem, which could be locked free space, incorrect SPAM thresholds, unique indexes, or estimated record sizes.

For high volume transaction processing applications or applications with a large number of storage areas, it is not practical to manually examine the “IO Statistics (by file)” screen for each storage area in the database, trying to identify the particular storage area with excessive read I/O operations.

To help you more quickly identify the storage area or areas in which excessive page checks are occurring, or, of course, any other type of excessive I/O, use the “File IO
Overview" screen via the "IO Statistics (by file)" menu. When you select the "IO Statistics (by file)" screen, you can type "A" to select the "File IO Overview" screen.

Page Navigation

If there is more information than can fit on one page, the notation "Page 1 of n" appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key. To display a previous page, press the left angle bracket “<” key or the ↑ Previous Screen key.

Binary File Support

This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support

This screen is integrated into the "Cluster Statistic Collection" facility.

Zoom Screen

The zoom screen displays storage area information on its zoom screen.

To simplify the use of the display when a database contains a large number of storage areas, you can configure the "File IO Overview" screen to sort the storage area information in several different ways. Type "C" to select the Config option from the display's on-screen menu, which displays the configuration menu:

Configuration Options

Unsorted totals display

This configuration option displays the by-file I/O count information in unsorted order; this is the default display. With this display, the storage areas are shown in the same order as the RMU Dump utility displays them; that is, all data areas are displayed first, followed by snapshot areas.

Note how this configuration option differs from the "Unsorted current rate display" option below.

Sort by total synchronous reads

This configuration option displays the by-file information sorted by descending synchronous read I/O counts. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort by total synchronous writes

This configuration option displays the by-file information sorted by descending synchronous write I/O counts. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort by total asynchronous reads

This configuration option displays the by-file information sorted by descending asynchronous read I/O counts. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort by total asynchronous writes

This configuration option displays the by-file information sorted by descending asynchronous write I/O counts. Storage areas with duplicate sort criteria are displayed alphabetically.
Sort by total reads & writes: This configuration option displays the by-file information sorted by descending total read and write I/O counts, both synchronous and asynchronous. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort by total pages discarded: This configuration option displays the by-file information sorted by descending pages discarded counts. Storage areas with duplicate sort criteria are displayed alphabetically.

Unsorted current rate display: This configuration option displays the by-file I/O rate information in unsorted order; this is the default display. With this display, the storage areas are shown in the same order as the RMU Dump utility displays them; that is, all data areas are displayed first, followed by snapshot areas.

Note how this configuration option differs from the “Unsorted totals display” option above.

Sort by synchronous read rate: This configuration option displays the by-file information sorted by descending synchronous read I/O occurrence-per-second rate. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort by synchronous write rate: This configuration option displays the by-file information sorted by descending synchronous write I/O occurrence-per-second rate. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort by asynchronous read rate: This configuration option displays the by-file information sorted by descending asynchronous read I/O occurrence-per-second rate. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort by asynchronous write rate: This configuration option displays the by-file information sorted by descending asynchronous write I/O occurrence-per-second rate. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort by total current I/O rates: This configuration option displays the by-file information sorted by descending total read and write I/O occurrence-per-second rate, both synchronous and asynchronous. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort by pages discarded rates: This configuration option displays the by-file information sorted by descending pages discarded occurrence-per-second rate. Storage areas with duplicate sort criteria are displayed alphabetically.

Sort alphabetically: This configuration option displays the by-file information sorted in ascending alphabetical order. This option has the affect of grouping data and snapshot areas together, since their names are the same.

Sort alphabetically by type: This configuration option displays the by-file information sorted in ascending alphabetical order, grouped by storage area type. This option has the affect of grouping all data storage areas together, followed by all snapshot areas together.

Display all storage areas: This configuration option allows the “File IO Overview” screen to display both live and snapshot storage areas. This is the default configuration choice.
This configuration option allows the “File IO Overview” screen to display live storage areas only. This option can be used in conjunction with the “sort” options to filter the number of areas displayed.

This configuration option allows the “File IO Overview” screen to display snapshot storage areas only. This option can be used in conjunction with the “sort” options to filter the number of areas displayed.
**Device IO Overview Screen**

This screen shows a summary of I/O activity for all devices that contain database storage areas.

The “Device IO Overview” screen has the following capabilities:

- It displays real-time information about all devices accessed by the database.
- It displays information for devices on which the database root file and storage areas (live and snapshot) reside.
- It displays information about all database root files and storage areas except for AIJ, ACE, or RUJ devices.

This screen resides in the “IO Statistics (by file)” menu.

The following is an example of the “Device IO Overview” screen:

```plaintext
Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor 30-JAN-1998 15:35:37.18
Rate: 1.00 Second   Device IO Overview (Unsorted total I/O) Elapsed: 02:04:14.39
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

$111$DUA347:           19         517       40998             0        86952

--------------------------------------------------------------------------------
Config Exit Help Menu >next_page <prev_page Options Reset Set_rate Write!
```

**SCREEN LOCATION**

**SCREEN EXAMPLE**

**SCREEN FIELDS**

- **Device Name**
  - This field identifies the particular device on which live or snapshot storage area and database rootfile reside.
  - Note that this device name does not include devices where AIJ or RUJ journals may reside.

- **Num**
  - This field identifies the number of storage areas that reside in the particular device. It is often desirable to balance the number of storage areas across all devices.
Sync Reads  This field gives the number of synchronous read QIOs (queued I/O requests) issued to the device. This operation reads database pages synchronously from the database.

Sync Writes  This field gives the number of synchronous write QIOs (queued I/O requests) issued to the device. This operation writes modified database pages synchronously back to the database.

Async Reads  This field gives the number of asynchronous read QIOs (queued I/O requests) issued to the device. This operation reads database pages asynchronously from the database.

Async Writes  This field gives the number of asynchronous write QIOs (queued I/O requests) issued to the device. This operation writes modified database pages asynchronously back to the database.

ON-SCREEN MENU OPTIONS

Options  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

Reset  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

SCREEN INFORMATION

Page Navigation  If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key. To display a previous page, press the left angle bracket “<” key or the ↑ Previous Screen key.

Binary File Support  This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support  This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  This screen does not have a zoom screen.
To simplify the use of the display when a database contains a large number of storage areas, you can configure the "Device IO Overview" screen to sort the storage area information in several different ways. Type "C" to select the Config option from the display's on-screen menu, which displays the configuration menu:

**Unsorted totals display**

This configuration option displays the device I/O count information in unsorted order; this is the default display. With this display, the devices are shown in an arbitrary order.

Note how this configuration option differs from the “Unsorted current rate display” option below.

**Sort by total synchronous reads**

This configuration option displays the device information sorted by descending synchronous read I/O counts. Devices with duplicate sort criteria are displayed alphabetically.

**Sort by total synchronous writes**

This configuration option displays the device information sorted by descending synchronous write I/O counts. Devices with duplicate sort criteria are displayed alphabetically.

**Sort by total asynchronous reads**

This configuration option displays the device information sorted by descending asynchronous read I/O counts. Devices with duplicate sort criteria are displayed alphabetically.

**Sort by total asynchronous writes**

This configuration option displays the device information sorted by descending asynchronous write I/O counts. Devices with duplicate sort criteria are displayed alphabetically.

**Sort by total reads & writes**

This configuration option displays the device information sorted by descending total read and write I/O counts, both synchronous and asynchronous. Devices with duplicate sort criteria are displayed alphabetically.

**Sort by total pages checked**

This configuration option displays the device information sorted by descending pages checked counts. Devices with duplicate sort criteria are displayed alphabetically.

**Unsorted current rate display**

This configuration option displays the device I/O rate information in unsorted order; this is the default display. With this display, the storage areas are shown in an arbitrary order.

Note how this configuration option differs from the “Unsorted totals display” option above.

**Sort by synchronous read rate**

This configuration option displays the device information sorted by descending synchronous read I/O occurrence-per-second rate. Devices with duplicate sort criteria are displayed alphabetically.

**Sort by synchronous write rate**

This configuration option displays the device information sorted by descending synchronous write I/O occurrence-per-second rate. Devices with duplicate sort criteria are displayed alphabetically.
<table>
<thead>
<tr>
<th>Sort by asynchronous read rate</th>
<th>This configuration option displays the device information sorted by descending asynchronous read I/O occurrence-per-second rate. Devices with duplicate sort criteria are displayed alphabetically.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sort by asynchronous write rate</td>
<td>This configuration option displays the device information sorted by descending asynchronous write I/O occurrence-per-second rate. Devices with duplicate sort criteria are displayed alphabetically.</td>
</tr>
<tr>
<td>Sort by total current I/O rates</td>
<td>This configuration option displays the device information sorted by descending total read and write I/O occurrence-per-second rate, both synchronous and asynchronous. Devices with duplicate sort criteria are displayed alphabetically.</td>
</tr>
<tr>
<td>Sort by pages checked rates</td>
<td>This configuration option displays the device information sorted by descending pages checked occurrence-per-second rate. Devices with duplicate sort criteria are displayed alphabetically.</td>
</tr>
</tbody>
</table>
Device Information Screen

This screen provides an online view of the storage-area device information local to a particular database.

The “Device Information” screen has the following capabilities:

- It displays real-time information about all devices accessed by the database.
- It displays information for devices on which the database root file and storage areas (live and snapshot) reside.
- It displays information about all database root files and storage areas except for AIJ, ACE, or RUJ devices.

Note that the “Device Information” screen is available in replay mode only if OPTIONS=AREA qualifier allowed device information to be recorded in the binary output file. However, because the information is displayed in real time, it does not reflect the time when the device information was recorded.

---

SCREEN LOCATION

This screen resides in the “IO Statistics (by file)” menu.

---

SCREEN EXAMPLE

The following is an example of the “Device Information” screen:

```
Rate: 1.00 Second        Device Information           Elapsed: 02:04:16.56
Page: 1 of 1              DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online
--------------------------------------------------------------------------------
Device.Name......... Status.  #Err Volume.Label FreeBlocks Max#Blocks %Full
$11$DUA347:         Mounted    0 KODA_USER3      2700076    4109470  34.2
--------------------------------------------------------------------------------
```

---

SCREEN FIELDS

Exit Help Menu >next_page <prev_page Options Set_rate Write !
Device Name  This field identifies the name of the device.

Status  This field identifies the status of the device. Any of the following status types might be displayed in this column:

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mounted</td>
<td>Device is mounted and active</td>
</tr>
<tr>
<td>Valid</td>
<td>Device is software valid</td>
</tr>
<tr>
<td>Busy</td>
<td>Device is busy</td>
</tr>
<tr>
<td>MntVrfy</td>
<td>Device is undergoing Mount Verification</td>
</tr>
<tr>
<td>TimeOut</td>
<td>Device has timed out</td>
</tr>
<tr>
<td>PowerFl</td>
<td>Device has experienced power failure</td>
</tr>
<tr>
<td>Unknown</td>
<td>Device is in some unexpected state</td>
</tr>
</tbody>
</table>

#Err  The number of hardware errors that have occurred on the device since it was mounted.

Volume Label  This field indicates the name of the device specified when it was mounted.

Free Block  This field indicates the number of free (available) blocks on the device. This number approaches zero (0) as the device becomes full. This is one of the most important field in the screen.

Max #Blocks  This field indicates the total number of blocks in the device.

%Full  This field indicates the percentage of the disk that is full.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Options  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
This section discusses screen-specific issues.

If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key. To display a previous page, press the left angle bracket “<” key or the ↑ Previous Screen key.

Binary File Support

This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support

This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen

This screen does not have a zoom screen.

This screen does not have any configuration options.
File IO Statistics Screen

This screen allows you to display I/O statistics for each individual file in the database.

When you select “IO Statistics (by file)” from the display menu, Oracle Rdb displays a list of files that comprise the database and for which you can choose to view statistics. With the exception of the all data/snap files screen, each screen shows the I/O activity for a specific database file. The all data/snap files screen shows a summation of I/O activity for all data and snapshot files.

The information in this screen applies from the time that your RMU Show Statistic utility session began, or since the accumulators were last reset (using the “Reset” on-screen menu option).

Note that the accumulators on this screen can be reset using the “Reset” on-screen menu option.

This screen resides in the “IO Statistics (by file)” menu.
The following is an example of the “File IO Statistics” screen:

Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor 30-JAN-1998 15:35:45.64
Rate: 1.00 Second              File IO Statistics           Elapsed: 02:04:22.85
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1    Mode: Online

For File: Database Root

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>max</td>
<td>cur</td>
<td>avg</td>
</tr>
<tr>
<td>total I/O rates</td>
<td>1</td>
<td>35845</td>
<td>1.1</td>
</tr>
<tr>
<td>(Synch. reads)</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>(Synch. writes)</td>
<td>5</td>
<td>0</td>
<td>0.8</td>
</tr>
<tr>
<td>(Extends)</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>(Asynch. reads)</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>(Asynch. writes)</td>
<td>16</td>
<td>1</td>
<td>3.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>statistic</th>
<th>blocks.transferred</th>
<th>stall.time.(x100)</th>
<th>name</th>
<th>avg.per.I/O</th>
<th>total</th>
<th>avg.per.I/O</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>total I/O sizes</td>
<td>1.0</td>
<td>36004</td>
<td>0.6</td>
<td>22477</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Synch. reads)</td>
<td>1.8</td>
<td>48</td>
<td>0.3</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Synch. writes)</td>
<td>1.0</td>
<td>6221</td>
<td>3.3</td>
<td>20914</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Extends)</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Asynch. reads)</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Asynch. writes)</td>
<td>1.0</td>
<td>29735</td>
<td>0.0</td>
<td>1554</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Exit Help Menu Options Reset Set_rate Write!
The following example shows the “File IO Statistics” screen for a specific storage area:

Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor 30-JAN-1998 15:35:53.23
Rate: 1.00 Second    File IO Statistics    Elapsed: 02:04:30.44
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1    Mode: Online

--------------------------------------------------------------------------------
For File: DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERS_DEFAULT.RDA;1

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate per second</th>
<th>total</th>
<th>average</th>
<th>count</th>
<th>per trans.</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>max</td>
<td>cur</td>
<td>avg</td>
<td>count</td>
<td>per trans.</td>
</tr>
<tr>
<td>total I/O rates</td>
<td>1</td>
<td>26118</td>
<td>0.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Synch. reads)</td>
<td>1</td>
<td>0</td>
<td>0.0</td>
<td>66</td>
<td>0.0</td>
</tr>
<tr>
<td>(Synch. writes)</td>
<td>12</td>
<td>0</td>
<td>1.2</td>
<td>9478</td>
<td>0.3</td>
</tr>
<tr>
<td>(Extends)</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>(Asynch. reads)</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>(Asynch. writes)</td>
<td>18</td>
<td>1</td>
<td>2.2</td>
<td>16574</td>
<td>0.5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>statistic</th>
<th>blocks transferred</th>
<th>stall time (x100)</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>avg per I/O</td>
<td>total</td>
</tr>
<tr>
<td>total I/O sizes</td>
<td>2.0</td>
<td>54294</td>
</tr>
<tr>
<td>(Synch. reads)</td>
<td>10.0</td>
<td>662</td>
</tr>
<tr>
<td>(Synch. writes)</td>
<td>2.0</td>
<td>19744</td>
</tr>
<tr>
<td>(Extends)</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>(Asynch. reads)</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>(Asynch. writes)</td>
<td>2.0</td>
<td>33888</td>
</tr>
</tbody>
</table>

--------------------------------------------------------------------------------
Exit Help Menu Options Reset Set_rate Write !

**SCREEN FIELDS**

- **total I/O rates**: The total number of I/O operations to the file being displayed, broken down to the number of synchronous read, synchronous write, extend, asynchronous read, and asynchronous write operations. This number does not include creation or truncation operations.

- **Synch reads**: This field gives the occurrence-per-second rate of synchronous read QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP). This operation reads database pages synchronously from the database.

- **Synch writes**: This field gives the occurrence-per-second rate of synchronous write QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP). This operation writes modified database pages synchronously back to the database.
This field gives the occurrence-per-second rate of file extension operations issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP).

This field gives the occurrence-per-second rate of asynchronous read QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP). This operation reads database pages asynchronously from the database.

This field gives the occurrence-per-second rate of asynchronous write QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP). This operation writes modified database pages asynchronously back to the database.

The total number of I/O operations to the file being displayed, broken down to the number of synchronous read, synchronous write, extend, asynchronous read, and asynchronous write operations. This number does not include creation or truncation operations.

This field gives the blocks-transferred information for synchronous read QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP).

This field gives the blocks-transferred information for synchronous write QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP).

This field gives the blocks-transferred information for file extension operations QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP).

This field gives the blocks-transferred information for asynchronous read QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP).

This field gives the blocks-transferred information for asynchronous write QIOs (queued I/O requests) issued to the database storage area for single-file and multi-file databases (.RDA) and snapshot files (.SNP).

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
Reset  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

**SCREEN INFORMATION**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Page Navigation</td>
<td>This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.</td>
</tr>
<tr>
<td>Binary File Support</td>
<td>This screen is available during replay of a binary file specified by the INPUT qualifier.</td>
</tr>
<tr>
<td>Cluster Statistic Collection Support</td>
<td>This screen is integrated into the “Cluster Statistic Collection” facility.</td>
</tr>
<tr>
<td>Zoom Screen</td>
<td>This screen does not have a zoom screen.</td>
</tr>
</tbody>
</table>

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
Logical Area Information Screens

**RMU Show Statistic Logical Area Statistics Screens**

The “Logical Area Information” menu screens display information on specific logical areas. A logical area is a table, btree index, hash index or binary large object.
Logical Area Overview Screen

This screen displays comparative information for all logical areas of a particular type, on a single screen.

By default, the screen displays the four most useful statistic fields for the respective logical area type. However, you may configure both the statistic fields displayed as well as the type of statistic information displayed.

The “system” logical area information can be filtered from the display, which will result in application logical areas being displayed only.

There is no way to “mix and match” different logical areas on the same screen display. This is impossible because of the different statistic information collected for different logical area types.

It is possible to zoom on a specific logical area and display its relevant statistic information. This drill-down capability makes the “Logical Area Overview” an extremely useful analytical tool for identifying performance bottlenecks or potential problems.

The “Logical Area Overview” screen is not available if the NOLOGICAL_AREA qualifier is specified, or if the INPUT qualifier is specified.

This screen resides in the “Logical Area Information” menu.

SCREEN LOCATION

---

2 Available in Oracle Rdb 7.0.2.0.
The following is an example of the “Logical Area Overview” screen for a table logical area:

```
Node: MYNODE (1/1/16)   Oracle Rdb X7.0-00 Perf. Monitor 18-MAR-1998 14:20:54.98
Rate: 1.00 Second        Logical Area Overview (Tables)     Elapsed: 03:28:56.70
Page: 1 of 1    DISK$:[DB_HOMEDIR WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

<table>
<thead>
<tr>
<th>Logical.Area.Name...</th>
<th>record</th>
<th>fetch</th>
<th>record</th>
<th>store</th>
<th>record</th>
<th>erase</th>
<th>discarded</th>
<th>CurTot</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDB$RELATIONS.RDB$SY</td>
<td>29</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$FIELD_VERSIONS.R</td>
<td>217</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$INDICES.RDB$SYST</td>
<td>35</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$INDEX_SEGMENTS.R</td>
<td>35</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$FIELDS.RDB$SYSTE</td>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$RELATION_FIELDS.</td>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$DATABASE.RDB$SYS</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$VIEW_RELATIONS.R</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$CONSTRAINT_RELAT</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$CONSTRAINTS.RDB$</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$STORAGE_MAPS.RDB</td>
<td>9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$STORAGE_MAP_AREA</td>
<td>17</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$INTERRELATIONS.R</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$COLLATIONS.RDB$S</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$TRIGGERS.RDB$SYS</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$RELATION_CONSRA</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$RELATION_CONSRA</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$PRIVILEGES.RDB$S</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$MODULES.RDB$SYST</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$ROUTINES.RDB$SYS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$PARAMETERS.RDB$S</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$QUERY_OUTLINES.R</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RDB$WORKLOAD.RDB$SYS</td>
<td>37</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>CANDIDATES.RDB$SYSTE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>COLLEGES.EMP_INFO</td>
<td>495</td>
<td>0</td>
<td>165</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DEGREES.EMP_INFO</td>
<td>2626</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DEPARTMENTS.DEPARTME</td>
<td>148</td>
<td>0</td>
<td>37</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>EMPLOYEES.EMPIDS_LOW</td>
<td>228</td>
<td>0</td>
<td>57</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>EMPLOYEES.EMPIDS_OVE</td>
<td>24</td>
<td>0</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>JOBS.JOBS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>JOB_HISTORY.EMPIDS_L</td>
<td>306</td>
<td>0</td>
<td>102</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>JOB_HISTORY.EMPIDS_M</td>
<td>450</td>
<td>0</td>
<td>150</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>JOB_HISTORY.EMPIDS_O</td>
<td>66</td>
<td>0</td>
<td>22</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RESUMES_EMP_INFO</td>
<td>58600</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SALARY_HISTORY.SALAR</td>
<td>2187</td>
<td>0</td>
<td>729</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>WORK_STATUS.EMP_INFO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
```

---

**SCREEN FIELDS**

The following is an example of the “Logical Area Overview” screen for a table logical area:
This column displays the name of the logical area, followed by a period “.”, followed by the name of the physical area (storage area) in which the logical area partition resides.

A maximum of 20 characters is displayed, which typically results in the storage area name being partially truncated. To display the entire storage area name, it may be necessary to zoom on the logical area using the “Zoom” on-screen menu option.

For performance reasons, the logical area names are not sorted in any particular order by default. You can configure the screen to display the logical areas in alphabetical order.

Each logical area displayed represents a single partition of that logical area. There is no method available to display the logical area’s aggregate statistic information.

This column displays a user-selectable statistic field appropriate for the logical area type. The default statistic field is the following:

<table>
<thead>
<tr>
<th>Logical Area Type</th>
<th>Default Statistic Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>record fetch</td>
</tr>
<tr>
<td>Btree Index</td>
<td>leaf fetches</td>
</tr>
<tr>
<td>Hash Index</td>
<td>hash index fetched</td>
</tr>
<tr>
<td>Blob</td>
<td>blob fetched</td>
</tr>
</tbody>
</table>

This column displays a user-selectable statistic field appropriate for the logical area type. The default statistic field is the following:

<table>
<thead>
<tr>
<th>Logical Area Type</th>
<th>Default Statistic Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>record stored</td>
</tr>
<tr>
<td>Btree Index</td>
<td>leaf insertion</td>
</tr>
<tr>
<td>Hash Index</td>
<td>hash insertion</td>
</tr>
<tr>
<td>Blob</td>
<td>blob stored</td>
</tr>
</tbody>
</table>
Statistic Field #3  This column displays a user-selectable statistic field appropriate for the logical area type. The default statistic field is the following:

<table>
<thead>
<tr>
<th>Logical Area Type</th>
<th>Default Statistic Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>record erased</td>
</tr>
<tr>
<td>Btree Index</td>
<td>leaf removal</td>
</tr>
<tr>
<td>Hash Index</td>
<td>hash deletion</td>
</tr>
<tr>
<td>Blob</td>
<td>blob erased</td>
</tr>
</tbody>
</table>

Statistic Field #4  This column displays a user-selectable statistic field appropriate for the logical area type. The default statistic field is the following:

<table>
<thead>
<tr>
<th>Logical Area Type</th>
<th>Default Statistic Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>pages discarded</td>
</tr>
<tr>
<td>Btree Index</td>
<td>pages discarded</td>
</tr>
<tr>
<td>Hash Index</td>
<td>pages discarded</td>
</tr>
<tr>
<td>Blob</td>
<td>pages discarded</td>
</tr>
</tbody>
</table>

Statistic Type.  This column identifies the type of statistic information being displayed. The following types are available:

<table>
<thead>
<tr>
<th>Statistic Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CurTot</td>
<td>This column indicates the total number of occurrences of the statistic since the database was opened or the statistic collection information was reset; you can reset this information using the “Reset” on-screen menu option.</td>
</tr>
<tr>
<td>CurRate</td>
<td>This column indicates the current occurrence-per-second rate of the statistic during the last sample interval (screen refresh); you can change the sample interval using the “Set_rate” on-screen menu option.</td>
</tr>
<tr>
<td>MaxRate</td>
<td>This column indicates the maximum occurrence-per-second rate of the statistic since the RMU Show Statistic utility was started or the statistic collection information was reset; you can reset</td>
</tr>
<tr>
<td>Statistic Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>this information using the “Reset” on-screen menu option.</td>
<td></td>
</tr>
<tr>
<td>AvgRate</td>
<td>This column indicates the average occurrence-per-second rate of the statistic since the database was opened or the statistic collection information was reset; you can reset this information using the “Reset” on-screen menu option.</td>
</tr>
<tr>
<td>PerTrans</td>
<td>This column indicates the transaction average of the total-count column and the total number of completed transactions.</td>
</tr>
</tbody>
</table>

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Options**

- **Pause**: This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**: This on-screen menu option temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Unreset**: This on-screen menu option restores the original statistic values after using “Reset” option.

The “system” logical areas can be filtered from the “Logical Area Overview” screen by selecting the “Display application logical areas” option of the “Tools” menu. System logical areas can be included on the screen by selecting the “Display all logical areas” option of the “Tools” menu. Note that these options are not available using the “Config” on-screen menu option.

The “Logical Area Overview” screen can also be configured to display application logical areas only (no “system” logical areas) using the configuration variable `SYSTEM_LOGICAL_AREAS` with the keyword FALSE. Specifying the configuration variable with the keyword TRUE, the default, will display all logical areas, including “system” logical areas.
The “Logical Area Overview” screen statistic type can be specified using the configuration variable LOGICAL_OVERVIEW_STAT with one of the following keywords: CUR_TOTAL, CUR_RATE, MAX_RATE, AVG_RATE or PER_TRANS.

The “Logical Area Overview” screen logical area type can be specified using the configuration variable LOGICAL_OVERVIEW_TYPE with one of the following keywords: TABLE, BTREE, HASH or BLOB.

When selecting statistic fields for the various columns using the “Config” on-screen menu option, no validation is performed to eliminate duplicate selections. This means you can display the same statistic field in one or more columns at the same time, if you so desire.

The following is an example of the “Logical Area Overview” screen for btree index logical areas:

Node: MYNODE (1/1/16)  Oracle Rdb X7.0-00 Perf. Monitor 18-MAR-1998 15:10:40.79
Rate: 1.00 Second    Logical Area Overview (Btree Indexes)  Elapsed: 04:18:42.51
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

Logical.Area.Name... leaf fetches leaf inserti leaf removal    discarded CurTot
COLL_COLLEGE_CODE.RD            0            0            0            0
DEG_EMP_ID.RDB$SYSTE          103            0            5            0
DEPARTMENTS_INDEX.DE          100            0            0            0
EMPLOYEE_ID.RDB$            5            0            3            0
SH_EMPLOYEE_ID.RDB$          103            0            3            0

The following is an example of the “Logical Area Overview” screen for hash index logical areas:

Node: MYNODE (1/1/16)  Oracle Rdb X7.0-00 Perf. Monitor 18-MAR-1998 15:11:09.68
Rate: 1.00 Second     Logical Area Overview (Hash Indexes)  Elapsed: 04:19:11.40
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

Logical.Area.Name... hash index f hash inserti hash deletio    discarded CurTot
EMPLOYEES_HASH.EMPID           37            0           37            0
EMPLOYEES_HASH.EMPID           57            0           57            0
EMPLOYEES_HASH.EMPID            6            0            6            0
JOB_HISTORY_HASH.EMP          235            0           37            0
JOB_HISTORY_HASH.EMP          343            0           57            0
JOB_HISTORY_HASH.EMP           50            0            6            0

Config Exit Help Menu >next_page <prev_page Options Pause Reset Set_rate Write
The following is an example of the “Logical Area Overview” screen for blob logical areas:

Node: MYNODE (1/1/16)  Oracle Rdb X7.0-00 Perf. Monitor 18-MAR-1998 15:11:38.15
Rate: 1.00 Second   Logical Area Overview (Blobs)   Elapsed: 04:19:39.87
Page: 1 of 1   DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1   Mode: Online

Logical.Area.Name... blob fetched blob stored blob erased discarded CurTot
RDB$SEGMENTED_STRING  73  0  0  0

If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key. To display a previous page, press the left angle bracket “<” key or the ↑ Previous Screen key.

This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

This screen is integrated into the “Cluster Statistic Collection” facility.

This screen displays the “logical area” zoom screen.

The “Logical Area Overview” screen can be easily customized and configured to display a variety of information. From the “Logical Area Overview” screen, pressing “C” will display the configuration menu, which consists of the following options:

- Modify column #1: This option allows you to choose a different statistic field for column number 1. You will be presented with a sub-menu containing the valid statistic fields for the selected logical area type.

- Modify column #2: This option allows you to choose a different statistic field for column number 2. You will be presented with a sub-menu containing the valid statistic fields for the selected logical area type.

- Modify column #3: This option allows you to choose a different statistic field for column number 3. You will be presented with a sub-menu containing the valid statistic fields for the selected logical area type.

- Modify column #4: This option allows you to choose a different statistic field for column number 4. You will be presented with a sub-menu containing the valid statistic fields for the selected logical area type.
Change logical area type
This option allows you to choose a different logical area type to be displayed on the screen. Selecting a new logical area type will reset the statistic fields to the default fields for that logical area type.

Logical area types are: table, btree index, hash index and blob.

The selected logical area type will be displayed on the screen name in the header region.

Change statistic type
This option allows you to choose a different statistic type to be displayed on the screen. The selected statistic type applies to all statistic fields on the screen.

Statistic types are: current total, current rate, maximum rate, average rate and average per transaction rate.

The selected statistic type will be displayed in the heading line of the screen display.

Unsorted Display
This option displays the logical areas on the “Logical Area Overview” screen in an unsorted order. This is the default presentation. For performance reasons, this option is recommended if you have a large number of logical areas, or you are using a very frequent screen refresh rate, such as one second.

Sort alphabetically
This option displays the logical areas on the “Logical Area Overview” screen in alphabetical order.
Logical Area Statistics Screen (Tables) Screen

This screen displays statistics about a specific logical area. A “logical area” is a table, btree index or hash index. This screen essentially provides the means to “drill down” to a specific table’s or index’s statistic information.

It is even possible to display a particular segment of a partitioned logical area.

The “Logical Area Statistics” screen is actually comprised of several different screens, one for each logical area type. The following logical types are available:

- table
- btree index, both regular and pseudo-ranked
- hash index
- blob (binary large object)

The “Logical Area” screens are selected using the “Logical Area Information” option of the main menu. Currently, there is only 1 logical area screen, the “Logical Area Statistics” screen. Selecting this option will display a sub-menu containing all available logical areas, listed in alphabetical order. The choice of logical areas in this sub-menu can be highly configured and filtered.

When you have selected a logical area, the RMU Show Statistic utility will identify the type of the selected logical area. If the type cannot be determined, you will be prompted to enter the logical area type.

The “Logical Area” statistics are not written to the binary output file. Conversely, the “Logical Area” statistics screens are not available during binary input file replay.

A “summary information” screen of all logical areas is not currently available.

The number of lines of statistics information displayed depends on the size of the terminal or window. In particular, the btree index displays up to 24 lines of statistics information. This means that on a standard 24x80 terminal, this means that only the first 15 lines of statistics would be displayed.

Any table or index created prior to Oracle Rdb 7.0.1.1 will require that the user enter the appropriate logical area type.

WARNING: If you specify an incorrect logical area type (i.e. specify a table as being a hash index), inappropriate statistics fields will be displayed.

The RMU Repair utility can be used to update the logical area type in the database. For more information on how to use this utility to update the logical area information, please refer to Appendix E.
The output from the “Logical Area Statistics” screen is not written to the output file; therefore, the output cannot be replayed using an input file.

The “Logical Area Statistics” screen is integrated into the “Cluster Statistic Collection” facility.

This screen resides in the “Logical Area Information” menu.

The following is an example of the “Logical Area Statistics” screen for a table logical area:

Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor  1-FEB-1998 07:31:22.54
Rate: 1.00 Second        Logical Area Statistics         Elapsed: 00:00:16.03
Page: 1 of 1 DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1    Mode: Online
-------------------------------------------------------------------------
Table EMPLOYEES in EMPIDS_LOW

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>record marked</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record fetched</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>fragmented</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

| record stored   | 0               | 0         | 0.0     |
| fragmented      | 0               | 0         | 0.0     |
| pages checked   | 0               | 0         | 0.0     |
| saved IO        | 0               | 0         | 0.0     |
| discarded       | 0               | 0         | 0.0     |
| record erased   | 0               | 0         | 0.0     |
| fragmented      | 0               | 0         | 0.0     |

-------------------------------------------------------------------------
Config Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot

 record marked  This field gives the number of records marked. A record is marked when it is modified or it is erased, but not when it is stored.

 record fetched This field gives the number of records, including snapshot records, fetched.

Note that this value may be more than the actual number of records returned by a query. The reason is that queries may fetch records during the search phase, and then re-fetch the selected records so that they may be returned to the user.
This sub-field indicates the number of record fragments that Oracle Rdb had to fetch. A record is fragmented if it is too large to fit on one page. A fragmented record requires more CPU time and more virtual memory, and often requires additional I/O operations because each record fragment must be fetched.

If this value is high compared to the number of records fetched, Oracle Corporation recommends that you use the RMU Analyze Areas command to further analyze the problem to see how many records are actually fragmented.

This field gives the number of records stored in the database.

This sub-field indicates the number of rows stored as fragmented records in the database. This number indicates that a page size is smaller than a record's uncompressed size (including overhead). You should use the RMU Analyze Areas utility to further analyze the problem and then increase the page size for the storage area that has the problem.

This field indicates the number of pages checked in order to store a record. Ideally, very few candidate pages need to be checked when storing a record. However in certain cases, depending on record size, access method, locked space on a page, and SPAM thresholds, storing a record requires a number of page fetches.

This field gives the number of pages checked that did not result in an I/O because the page was already in the buffer. It is essentially a "for free" pages checked.

This field identifies the number of pages checked but discarded because the actual free space on that page did not meet the physical requirements needed to store a new record.

This field gives the number of records erased from the database.

This sub-field indicates the number of fragmented records erased from the database.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Graph This on-screen menu option displays a histogram graph instead of numbers.

Numbers This on-screen menu option displays numeric statistics instead of histogram graph.

Options This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

This on-screen menu option plots a specific field's value by time.

This on-screen menu option restores the original statistic values after using “Reset” option.

This on-screen menu option plots a specific field's value using scatter-based display.

Appendix G contains a SQL script that will automatically generate the options file for the RMU Repair utility.

On average, each logical area requires approximately 13k bytes of VM. If you do not wish to view the logical area statistic screens, then using the NOLOGICAL_AREA qualifier is recommended. Also, this is based on the largest logical area identifier, not the actual number of logical areas. Therefore, a database containing 50 logical areas whose largest logical area identifier is 1234 would require approximately 16Mb of VM to present the logical area information.

This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

This screen is integrated into the “Cluster Statistic Collection” facility.

This screen does not have a zoom screen.

For partitioned logical areas such as tables and indexes, the logical area screens can be configured using the “Config” on-screen menu option to display either the “individual” partition statistics (default) or the “aggregated” statistics.

From the “Logical Area Statistics” screen, pressing “C” will display the configuration menu, which consists of the following options:
<table>
<thead>
<tr>
<th>Configuration Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Individual Logical Area</td>
<td>This configuration option indicates that the “Logical Area Statistics” screen is to display information on a per-segment for logical area partitioned over multiple storage areas.</td>
</tr>
<tr>
<td>Display Aggregate Logical Area</td>
<td>This configuration option indicates that the “Logical Area Statistics” screen is to display aggregated, or cumulative, information for logical areas partitioned over multiple storage areas.</td>
</tr>
<tr>
<td>Change Logical Area Type</td>
<td>This configuration option allows you to dynamically change a logical area’s type. Changing the logical area’s type affects the current RMU Show Statistic utility session only, and any changes are not recorded in the database. This configuration option is primarily used when the logical area type is not defined in the database, because the database was created prior to Oracle Rdb v7.0.1.0, and you entered an inappropriate type.</td>
</tr>
</tbody>
</table>
Logical Area Statistics Screen (Hash Indexes) Screen

This screen displays statistics about a specific logical area. A “logical area” is a table, btree index or hash index. This screen essentially provides the means to drill down to a specific table’s or index’s statistic information.

It is even possible to display a particular segment of a partitioned logical area.

The “Logical Area Statistics” screen is actually comprised of several different screens, one for each logical area type. The following logical types are available:

- table
- btree index, both regular and pseudo-ranked
- hash index
- blob (binary large object)

The “Logical Area” screens are selected using the “Logical Area Information” option of the main menu. Currently, there is only 1 logical area screen, the “Logical Area Statistics” screen. Selecting this option will display a sub-menu containing all available logical areas, listed in alphabetical order. The choice of logical areas in this sub-menu can be highly configured and filtered.

When you have selected a logical area, the RMU Show Statistic utility will identify the type of the selected logical area. If the type cannot be determined, you will be prompted to enter the logical area type.

The “Logical Area” statistics are not written to the binary output file. Conversely, the “Logical Area” statistics screens are not available during binary input file replay.

A summary information screen of all logical areas is not currently available.

The number of lines of statistics information displayed depends on the size of the terminal or window. In particular, the btree index displays up to 24 lines of statistics information. This means that on a standard 24x80 terminal, this means that only the first 15 lines of statistics would be displayed.

Any table or index created prior to Oracle Rdb 7.0.1.1 will require that the user enter the appropriate logical area type.

WARNING: If you specify an incorrect logical area type (i.e. specify a table as being a hash index), inappropriate statistics fields will be displayed.

The RMU Repair utility can be used to update the logical area type in the database. For more information on how to use this utility to update the logical area information, please refer to Appendix E.
The output from the “Logical Area Statistics” screen is not written to the output file; therefore, the output cannot be replayed using an input file.

The “Logical Area Statistics” screen is integrated into the “Cluster Statistic Collection” facility.

This screen resides in the “Logical Area Information” menu.

The following is an example of the “Logical Area Statistics” screen for a hash index logical area:

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
<th>count</th>
<th>per.trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>bucket marked</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>bucket fetched</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>bucket stored</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>pages checked</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>saved IO</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>discarded</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>hash insertions</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>duplicates</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>hash deletions</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>duplicates</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>hash scans</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>hash index fetches</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>bucket fragments</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>duplicate nodes</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

This field gives the number of buckets marked. A bucket is marked when it is modified or it is erased, but not when it is stored.
bucket fetched  This field gives the number of buckets fetched. The value of this field includes fetching the system record resident on each page, necessary to identify index-specific buckets. This field is incremented when buckets are updated.

Note the difference between this field and the “hashed index fetches” field described later in this section.

fragmented  This sub-field indicates the number of bucket fragments that Oracle Rdb had to fetch. A bucket is fragmented if it is too large to fit on one page. A fragmented bucket requires more CPU time and more virtual memory, and often requires additional I/O operations because each bucket fragment must be fetched. If this value is high compared to the number of buckets fetched, Oracle Corporation recommends that you use the RMU Analyze Areas utility to further analyze the problem to see how many buckets are actually fragmented.

bucket stored  This field gives the number of buckets stored in the database.

fragmented  This sub-field indicates the number of buckets stored as fragmented buckets in the database. This number indicates that a page size is smaller than a bucket's uncompressed size (including overhead). You should use the RMU Analyze Areas utility to further analyze the problem and then increase the page size for the storage area that has the problem.

hash insertions  This field gives the number of hash key insertions in the database's hashed indexes. It includes unique key insertions as well as duplicate key insertions.

duplicates  This field gives the number of duplicate key updates in the database's hashed indexes.

hash deletions  This field gives the number of hash key deletions from the database's hashed indexes. It includes unique key deletions as well as duplicate key deletions.

duplicates  This field gives the number of duplicate key deletions in the database's hashed indexes.

hash scans  This field gives the number of hashed index scans, including both retrieval and update scans, that were opened on the database's hashed indexes. A scan is defined as the sequential processing of the records that meet the search criteria of a query. Hashed scans then refer to the case where duplicate records are returned that meet the search criteria of a query from a scan of the hashed index.

hash index fetches  This field gives the number of hashed index nodes that were fetched on a successful search of the database's hashed indexes. This includes fetches of duplicate nodes as well as bucket fragment nodes.

Note the difference between this field and the “bucket fetched” field. This field identifies the search of a bucket for a particular key value, whereas the “bucket fetched” field identifies possible node fetches, including system records.

bucket fragments  This field gives the number of bucket fragments that were fetched on a successful search of the database's hashed indexes.
duplicate nodes

This field gives the number of duplicate nodes that were fetched on a successful search of the database's hashed indexes.

ON-SCREEN MENU OPTIONS

Graph This on-screen menu option displays a histogram graph instead of numbers.

Numbers This on-screen menu option displays numeric statistics instead of histogram graph.

Options This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

Pause This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

Reset This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

Time_plot This on-screen menu option plots a specific field's value by time.

Unreset This on-screen menu option restores the original statistic values after using “Reset” option.

X_plot This on-screen menu option plots a specific field's value using scatter-based display.

SCREEN INFORMATION

Appendix G contains a SQL script that will automatically generate the options file for the RMU Repair utility.

On average, each logical area requires approximately 13k bytes of VM. If you do not wish to view the logical area statistic screens, then using the NOLOGICAL_AREA qualifier is recommended. Also, this is based on the largest logical area identifier, not the actual number of logical areas. Therefore, a database containing 50 logical areas whose largest logical area identifier is 1234 would require approximately 16Mb of VM to present the logical area information.

Page Navigation This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
Binary File Support  This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

Cluster Statistic Collection Support  This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  This screen does not have a zoom screen.

**CONFIGURATION OPTIONS**

For partitioned logical areas such as tables and indexes, the logical area screens can be configured using the “Config” on-screen menu option to display either the “individual” partition statistics (default) or the “aggregated” statistics.

From the “Logical Area Statistics” screen, pressing “C” will display the configuration menu, which consists of the following options:

**Display Individual Logical Area**  This configuration option indicates that the “Logical Area Statistics” screen is to display information on a per-segment for logical area partitioned over multiple storage areas.

**Display Aggregate Logical Area**  This configuration option indicates that the “Logical Area Statistics” screen is to display aggregated, or cumulative, information for logical areas partitioned over multiple storage areas.

**Change Logical Area Type**  This configuration option allows you to dynamically change a logical area’s type. Changing the logical area’s type affects the current RMU Show Statistic utility session only, and any changes are not recorded in the database. This configuration option is primarily used when the logical area type is not defined in the database, because the database was created prior to Oracle Rdb v7.0.1.0, and you entered an inappropriate type.
Logical Area Statistics Screen (Btree Indexes) Screen

This screen displays statistics about a specific logical area. A “logical area” is a table, btree index or hash index. This screen essentially provides the means to drill down to a specific table’s or index’s statistic information.

It is even possible to display a particular segment of a partitioned logical area.

The “Logical Area Statistics” screen is actually comprised of several different screens, one for each logical area type. The following logical types are available:

- table
- btree index, both regular and pseudo-ranked
- hash index
- blob (binary large object)

The “Logical Area” screens are selected using the “Logical Area Information” option of the main menu. Currently, there is only 1 logical area screen, the “Logical Area Statistics” screen. Selecting this option will display a sub-menu containing all available logical areas, listed in alphabetical order. The choice of logical areas in this sub-menu can be highly configured and filtered.

When you have selected a logical area, the RMU Show Statistic utility will identify the type of the selected logical area. If the type cannot be determined, you will be prompted to enter the logical area type.

The “Logical Area” statistics are not written to the binary output file. Conversely, the “Logical Area” statistics screens are not available during binary input file replay.

A summary information screen of all logical areas is not currently available.

The number of lines of statistics information displayed depends on the size of the terminal or window. In particular, the btree index displays up to 24 lines of statistics information. This means that on a standard 24x80 terminal, this means that only the first 15 lines of statistics would be displayed.

Any table or index created prior to Oracle Rdb 7.0.1.1 will require that the user enter the appropriate logical area type.

WARNING: If you specify an incorrect logical area type (i.e. specify a table as being a hash index), inappropriate statistics fields will be displayed.

The RMU Repair utility can be used to update the logical area type in the database. For more information on how to use this utility to update the logical area information, please refer to Appendix E.
The output from the “Logical Area Statistics” screen is not written to the output file; therefore, the output cannot be replayed using an input file.

The “Logical Area Statistics” screen is integrated into the “Cluster Statistic Collection” facility.

This screen resides in the “Logical Area Information” menu.

The following is an example of the “Logical Area Statistics” screen for a b-tree index logical area:

---

Rate: 1.00 Second  Logical Area Statistics  Elapsed: 00:00:27.70  
Page: 1 of 1  DISK$: [DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1  Mode: Online  
---

Btree DEPARTMENTS_INDEX in DEPARTMENTS

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
<th>max</th>
<th>cur</th>
<th>avg</th>
<th>count</th>
<th>per.trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>node fetches</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>leaf fetches</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>dup. fetches</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>index lookups</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>index scans</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>primary entries</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>dup. entries</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>node insertions</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>root insertions</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>leaf insertions</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>dup. insertions</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>node creations</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>root splits</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>leaf creations</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>dup. creations</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>index creations</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>node removals</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>root removals</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>leaf removals</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>dup. removals</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>node deletions</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>leaf deletions</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>pages checked</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>saved IO</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>discarded</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

---

Config Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot
<table>
<thead>
<tr>
<th>Screen Fields</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>node fetches</td>
<td>This field gives the number of times Oracle Rdb fetched an index node during index retrievals. This number includes the number of leaf nodes and duplicate nodes fetched. Therefore, the calculation for the number of upper-level index nodes accessed is: this “node fetches” field minus the sum of the leaf and duplicate node fetches. The result can indicate the depth of the database indexes.</td>
</tr>
<tr>
<td>leaf fetches</td>
<td>This field gives the number of times Oracle Rdb fetched bottom level (leaf) nodes during index retrievals. This number, along with the “index scans” field, can indicate the length of scans in terms of index nodes accessed. There is one leaf node fetch for each “index lookup” retrieval.</td>
</tr>
<tr>
<td>dup fetches</td>
<td>This field gives the number of times Oracle Rdb fetched a duplicate node (as opposed to a leaf node) during index retrievals. This number can indicate the lengths of duplicate node chains in the database indexes. When a duplicate node is retrieved, the operation always includes one leaf fetch.</td>
</tr>
<tr>
<td>index lookups</td>
<td>This field gives the number of direct single-key retrievals performed on the database indexes. This statistic shows up only on unique key retrievals and not on duplicate key retrievals.</td>
</tr>
<tr>
<td>index scans</td>
<td>This field gives the number of scans, or range retrievals, performed on the database indexes. In an index scan, Oracle Rdb searches an index from top to bottom to find the starting point (low value) of the retrieval. Oracle Rdb then searches the bottom level nodes of the index, including duplicate nodes, until the scan’s end condition is met. This statistic is only incremented at the start of a particular scan.</td>
</tr>
<tr>
<td>primary entries</td>
<td>This field gives the number of unique keys found during the index scan. This statistic is incremented for each node accessed as part of an index scan.</td>
</tr>
<tr>
<td>dup entries</td>
<td>This field gives the number of duplicate keys found during the index scans. If an index has two entries with the same key value, the first one is a primary entry and the second is a duplicate entry.</td>
</tr>
<tr>
<td>node insertions</td>
<td>This field gives the number of index entries inserted into all index nodes. This number includes root, leaf, and duplicate entries within user-and system-defined indexes. This number is greater than the number of records being stored in the database because it usually takes one to two insertions into an index for each record for each index. The calculation of node insertions minus the sum of the root, leaf, and duplicate insertions yields the number of entries inserted into mid-level nodes. This number and the “root insertions” field indicate sorted balancing activity.</td>
</tr>
</tbody>
</table>
root insertions
This field gives the number of entries inserted into the root (top-level) index nodes. The number of insertions should be small except for when you load the database. Also, 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.

leaf insertions
This field gives the number of unique keys inserted into the database's indexes. This field indicates the number of entries inserted into the leaf (bottom-level) index nodes.

dup insertions
This field gives 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.

node creations
This field gives the total number of index nodes created during insertion of index entries into the index trees. This includes root, leaf, and duplicate nodes created within user- and system-defined indexes. Nodes are created three ways:

- When an index is first defined; this is the "root" node.
- When a node cannot accommodate an insertion, causing it to overflow into a new node (node splitting).
- When the first duplicate for a particular key is inserted into an index, causing a duplicate node to be created.

The total number of nodes created and the associated fields should be relatively small, except for an initial load of the database with indexes already defined, or for creation of indexes on already-stored data.

root splits
This field gives the number of times the root nodes have split because they overflowed after an insertion. A root node split causes the index to grow by one level--a parent node must be created to point to the two "halves" of the overflowed root node. Therefore, two nodes are created--the parent node and the node for the second half of the root node. Increasing the number of tree levels means Oracle Rdb must search more index nodes to access a data row; this can result in additional I/O operations.

leaf creations
This field gives the number of times a leaf (bottom level) node was created because an existing leaf node had become full and needed to accommodate another unique index key entry.

dup creations
This field gives the number of times a duplicate node was created to accommodate more duplicated entries within the duplicate index node or on the first store of a duplicate key entry.

This field gives the number of times an index was created on a particular table. This count is the number of CREATE INDEX statements. Also, if an index is partitioned over three areas, for example, there will be a count of three index creations.
index creations This field gives the number of times an index was created on a particular table of the database. This count is the number of CREATE INDEX statements executed. Also, if an index is partitioned across three storage areas, for example, there will be a count of three index creations.

node removals This field gives the total number of index entries within the root, leaf, and duplicate nodes that have been removed. This removal can be triggered by erasing rows, deleting tables, or deleting indexes. The calculation of node removals minus the sum of the root, leaf, and duplicate node removals yields the number of entries removed from mid-level nodes. A node is not deleted until all its entries are removed.

root removals This field gives the number of index entries removed from a root node due to 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.

leaf removals This field gives the number of unique index keys removed from the leaf nodes during an SQL DELETE operation.

dup removals This field gives the number of duplicate index keys removed from duplicate nodes due to the deletion of duplicate records. This should be a one-to-one correspondence to the number of erased duplicate records within the database.

node deletions This field gives the total number of index nodes deleted due to an SQL DROP INDEX statement or when the nodes become empty (except for the root node, which remains even when it is empty). When an index is deleted, this number should be equal to the total number of index nodes within the index. This field minus the sum of leaf and duplicate node deletions yields the number of mid-level node deletions.

leaf deletions This field gives the number of leaf (bottom level) nodes deleted from the database’s indexes. A leaf node is deleted only when it becomes empty.

dup deletions This field gives the number of duplicate node deletions within the indexes.

index destructions This field gives the number of indexes deleted with an SQL DROP INDEX statement. This count will be 1 if the index is not partitioned. If an index that is partitioned over three areas is deleted, for example, then the count will be 3. This count also gives the number of root node deletions.

pages checked This field indicates the number of pages checked in order to store a btree index node. Ideally, very few candidate pages need to be checked when storing a btree index node. However in certain cases, depending on the size of the segment, locked space on a page, and SPAM thresholds, storing a btree index node requires a number of page fetches.

saved IO This field gives the number of pages checked that did not result in an I/O because the page was already in the buffer. It is essentially a “for free” pages checked.
This field identifies the number of pages checked but discarded because the actual free space on that page did not meet the physical requirements needed to store a new btree index node.

---

**ON-SCREEN MENU OPTIONS**

- **Graph**: This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**: This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**: This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause**: This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**: This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Time_plot**: This on-screen menu option plots a specific field's value by time.
- **Unreset**: This on-screen menu option restores the original statistic values after using “Reset” option.
- **X_plot**: This on-screen menu option plots a specific field's value using scatter-based display.

---

**SCREEN INFORMATION**

- Appendix G contains a SQL script that will automatically generate the options file for the RMU Repair utility.

- On average, each logical area requires approximately 13k bytes of VM. If you do not wish to view the logical area statistic screens, then using the NOLOGICAL_AREA qualifier is recommended. Also, this is based on the largest logical area identifier, not the actual number of logical areas. Therefore, a database containing 50 logical areas whose largest logical area identifier is 1234 would require approximately 16Mb of VM to present the logical area information.

**Page Navigation**

This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

This screen is integrated into the "Cluster Statistic Collection" facility.

This screen does not have a zoom screen.

For partitioned logical areas such as tables and indexes, the logical area screens can be configured using the "Config" on-screen menu option to display either the "individual" partition statistics (default) or the "aggregated" statistics.

From the "Logical Area Statistics" screen, pressing "C" will display the configuration menu, which consists of the following options:

- **Display Individual Logical Area**
  - This configuration option indicates that the "Logical Area Statistics" screen is to display information on a per-segment for logical area partitioned over multiple storage areas.

- **Display Aggregate Logical Area**
  - This configuration option indicates that the "Logical Area Statistics" screen is to display aggregated, or cumulative, information for logical areas partitioned over multiple storage areas.

- **Change Logical Area Type**
  - This configuration option allows you to dynamically change a logical area's type. Changing the logical area's type affects the current RMU Show Statistic utility session only, and any changes are not recorded in the database. This configuration option is primarily used when the logical area type is not defined in the database, because the database was created prior to Oracle Rdb v7.0.1.0, and you entered an inappropriate type.
Logical Area Statistics Screen (Blobs) Screen

This screen displays statistics about a specific logical area. A “logical area” is a table, btree index or hash index. This screen essentially provides the means to “drill down” to a specific table’s or index’s statistic information.

It is even possible to display a particular segment of a partitioned logical area.

The “Logical Area Statistics” screen is actually comprised of several different screens, one for each logical area type. The following logical types are available:

- table
- btree index, both regular and pseudo-ranked
- hash index
- blob (binary large object)

The “Logical Area” screens are selected using the “Logical Area Information” option of the main menu. Currently, there is only 1 logical area screen, the “Logical Area Statistics” screen. Selecting this option will display a sub-menu containing all available logical areas, listed in alphabetical order. The choice of logical areas in this sub-menu can be highly configured and filtered.

When you have selected a logical area, the RMU Show Statistic utility will identify the type of the selected logical area. If the type cannot be determined, you will be prompted to enter the logical area type.

The “Logical Area” statistics are not written to the binary output file. Conversely, the “Logical Area” statistics screens are not available during binary input file replay.

A “summary information” screen of all logical areas is not currently available.

The number of lines of statistics information displayed depends on the size of the terminal or window. In particular, the btree index displays up to 24 lines of statistics information. This means that on a standard 24x80 terminal, this means that only the first 15 lines of statistics would be displayed.

Any table or index created prior to Oracle Rdb 7.0.1.1 will require that the user enter the appropriate logical area type.

WARNING: If you specify an incorrect logical area type (i.e. specify a table as being a hash index), inappropriate statistics fields will be displayed.

The RMU Repair utility can be used to update the logical area type in the database. For more information on how to use this utility to update the logical area information, please refer to Appendix E.
The output from the “Logical Area Statistics” screen is not written to the output file; therefore, the output cannot be replayed using an input file.

The “Logical Area Statistics” screen is integrated into the “Cluster Statistic Collection” facility.

This screen resides in the “Logical Area Information” menu.

The following is an example of the “Logical Area Statistics” screen for a blob logical area:

```
Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor  1-FEB-1998 07:31:22.54
Rate: 1.00 Second       Logical Area Statistics    Elapsed: 00:00:16.03
Page: 1 of 1  DISK$:[DB_HOMEDIR.WORK.STATS\]MF_PERSONNEL.RDB;1 Mode: Online

-----------------------------------------------------------------------------
Blob RDB$SEGMENTED_STRINGS in MF_PERS_SEG
statistic........... rate.per.second............. total....... average...... count...... per.trans....
name.............. max..... cur..... avg....... count....... per.trans....

  blob marked                    0        0        0.0            0           0.0
  blob fetched                  0        0        0.0            0           0.0
  fragmented               0        0        0.0            0           0.0
  blob stored                   0        0        0.0            0           0.0
  fragmented               0        0        0.0            0           0.0
  pages checked                  0        0        0.0            0           0.0
  saved IO                         0        0        0.0            0           0.0
  discarded              0        0        0.0            0           0.0
  blob erased                   0        0        0.0            0           0.0
  fragmented               0        0        0.0            0           0.0

-----------------------------------------------------------------------------
```

**SCREEN LOCATION**

This field gives the number of blob segments marked. A blob segment is marked when it is erased or reused, but not when it is stored.

**SCREEN EXAMPLE**

This field gives the number of blob segments fetched. This number includes pointer segments in addition to actual user data.

**SCREEN FIELDS**
This sub-field indicates the number of blob segment fragments that Oracle Rdb had to fetch. A blob segment is fragmented if it is too large to fit on one page. A fragmented blob segment requires more CPU time and more virtual memory, and often requires additional I/O operations because each blob segment fragment must be fetched.

This number refers only to actual user data as pointer segments are unlikely to fragment.

This field gives the number of blob segments stored in the database. This number includes pointer segments in addition to actual user data.

This sub-field indicates the number of rows stored as fragmented blob segments in the database. This number indicates that a page size is smaller than a blob segment.

This number refers only to actual user data as pointer segments are unlikely to fragment.

This field indicates the number of pages checked in order to store a blob segment. Ideally, very few candidate pages need to be checked when storing a blob segment. However in certain cases, depending on the size of the segment, locked space on a page, and SPAM thresholds, storing a blob segment requires a number of page fetches.

This field gives the number of pages checked that did not result in an I/O because the page was already in the buffer. It is essentially a “for free” pages checked.

This field identifies the number of pages checked but discarded because the actual free space on that page did not meet the physical requirements needed to store a new blob segment.

This field indicates the number of pages checked in order to store a hash index node. Ideally, very few candidate pages need to be checked when storing a hash index node. However in certain cases, depending on the size of the segment, locked space on a page, and SPAM thresholds, storing a hash index node requires a number of page fetches.

This field gives the number of pages checked that did not result in an I/O because the page was already in the buffer. It is essentially a “for free” pages checked.

This field identifies the number of pages checked but discarded because the actual free space on that page did not meet the physical requirements needed to store a new btree index node.

This field gives the number of blob segments erased from the database.
Note that the actual erase of blob segments is deferred until COMMIT time.

This sub-field indicates the number of fragmented blob segments erased from the database. This number refers only to actual user data as pointer segments are unlikely to fragment.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Graph**
This on-screen menu option displays a histogram graph instead of numbers.

**Numbers**
This on-screen menu option displays numeric statistics instead of histogram graph.

**Options**
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Pause**
This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

**Reset**
This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

**Time_plot**
This on-screen menu option plots a specific field's value by time.

**Unreset**
This on-screen menu option restores the original statistic values after using “Reset” option.

**X_plot**
This on-screen menu option plots a specific field's value using scatter-based display.

Appendix G contains a SQL script that will automatically generate the options file for the RMU Repair utility.

On average, each logical area requires approximately 13k bytes of VM. If you do not wish to view the logical area statistic screens, then using the NOLOGICAL_AREA qualifier is recommended. Also, this is based on the largest logical area identifier, not the actual number of logical areas. Therefore, a database containing 50 logical areas whose largest logical area identifier is 1234 would require approximately 16Mb of VM to present the logical area information.
Page Navigation  
This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support  
This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

Cluster Statistic Collection Support  
This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  
This screen does not have a zoom screen.

---

**Configuration Options**

For partitioned logical areas such as tables and indexes, the logical area screens can be configured using the “Config” on-screen menu option to display either the “individual” partition statistics (default) or the “aggregated” statistics.

From the “Logical Area Statistics” screen, pressing “C” will display the configuration menu, which consists of the following options:

- **Display Individual Logical Area**  
  This configuration option indicates that the “Logical Area Statistics” screen is to display information on a per-segment for logical area partitioned over multiple storage areas.

- **Display Aggregate Logical Area**  
  This configuration option indicates that the “Logical Area Statistics” screen is to display aggregated, or cumulative, information for logical areas partitioned over multiple storage areas.

- **Change Logical Area Type**  
  This configuration option allows you to dynamically change a logical area’s type. Changing the logical area’s type affects the current RMU Show Statistic utility session only, and any changes are not recorded in the database. This configuration option is primarily used when the logical area type is not defined in the database, because the database was created prior to Oracle Rdb v7.0.1.0, and you entered an inappropriate type.
Locking (One Lock Type) Screens

_RMU Show Statistic Locking (One Lock Type) Screens_

The “Locking (One Lock Type)” menu screens display locking information for a specific lock type.
**Total Locks Screen**

This screen monitors the total database locking activity. The statistics in this screen are the totals for all types of database locks.

Note that this screen includes locks that may have their own screen, such as the AIJ switch-over or Hot Standby locks.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (total locks)” screen:

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
<th>count</th>
<th>per.trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>max..... cur..... avg....... count....... per.trans....</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>locks requested</td>
<td>5</td>
<td>0</td>
<td>0.5</td>
<td>19</td>
<td>0.0</td>
</tr>
<tr>
<td>rqsts not queued</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>rqsts stalled</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>rqst timeouts</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>rqst deadlocks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>locks promoted</td>
<td>7</td>
<td>0</td>
<td>0.8</td>
<td>30</td>
<td>0.0</td>
</tr>
<tr>
<td>proms not queued</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>proms stalled</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>prom timeouts</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>prom deadlocks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>locks demoted</td>
<td>3</td>
<td>0</td>
<td>0.4</td>
<td>15</td>
<td>0.0</td>
</tr>
<tr>
<td>locks released</td>
<td>3</td>
<td>0</td>
<td>0.3</td>
<td>14</td>
<td>0.0</td>
</tr>
<tr>
<td>blocking ASTs</td>
<td>3</td>
<td>0</td>
<td>0.4</td>
<td>15</td>
<td>0.0</td>
</tr>
<tr>
<td>stall time x100</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!
This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count. The "rqsts not queued", "rqsts stalled", and "rqst deadlocks" counts provide further detail for enqueue lock requests statistics.

This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the "rqst timeouts" field is also reported in the "rqsts stalled" field. This is because each timed out request is also a stalled request.

This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the "rqst deadlocks" field is also reported in the "rqsts stalled" field. This is because each deadlocked request is also a stalled request.

This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The "proms not queued," "proms stalled," and "prom deadlocks" counts provide further detail for the locks promotion statistics.

This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.
Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.

prom deadlocks This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

locks demoted This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\].

blocking AST's This field gives the number of blocking ASTs, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking AST's reported is actually comprised of two different types of blocking AST's, those blocking AST's externally generated and those blocking AST's internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking AST's.

stall time x100 This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total
can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**ON-SCREEN MENU OPTIONS**

- **Graph**: This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**: This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**: This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause**: This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**: This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Unreset**: This on-screen menu option restores the original statistic values after using “Reset” option.

**SCREEN INFORMATION**

- **Page Navigation**: This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
- **Binary File Support**: This screen is available during replay of a binary file specified by the INPUT qualifier.
- **Cluster Statistic Collection Support**: This screen is integrated into the “Cluster Statistic Collection” facility.
- **Zoom Screen**: This screen does not have a zoom screen.

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
Area Locks Screen

This screen monitors the database storage area locks. Physical areas are simply another name for “storage areas”.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (area locks)” screen:

Rate: 1.00 Second        Locking (area locks)           Elapsed: 00:00:36.88
Page: 1 of 1 DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1 Mode: Online

<table>
<thead>
<tr>
<th>statistic........</th>
<th>rate.per.second........</th>
<th>total........</th>
<th>ave rage.......</th>
</tr>
</thead>
<tbody>
<tr>
<td>locks requested</td>
<td></td>
<td>max....</td>
<td>cur....</td>
</tr>
<tr>
<td>rqsts not queued</td>
<td></td>
<td>avg.......</td>
<td>count.......</td>
</tr>
<tr>
<td>rqsts stalled</td>
<td></td>
<td>count........</td>
<td>per.trans......</td>
</tr>
<tr>
<td>rqst timeouts</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rqst deadlocks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>locks promoted</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>proms not queued</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>proms stalled</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>prom timeouts</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>prom deadlocks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>locks demoted</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>locks released</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>blocking ASTs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stall time x100</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!

SCREEN FIELDS

locks requested: This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count.
The "rqsts not queued", "rqsts stalled", and "rqst deadlocks" counts provide further
detail for enqueue lock requests statistics.

rqsts not queued
This field gives the number of enqueue lock requests for new locks that were rejected
immediately because of a lock conflict. Oracle Rdb often requests a lock without
waiting and, when a conflict is detected, resorts to a secondary locking protocol to
avoid unnecessary deadlocks. This number is one measure of lock contention.

rqsts stalled
This field gives the number of enqueue lock requests for new locks that were stalled
because of a lock conflict. Whether or not the lock request ultimately succeeds, it is
included in this count. This number is one measure of lock contention.

rqst timeouts
This field shows the total number of lock requests that could not be granted because
they timed out. These are typically logical areas.

Each lock timeout reported in the "rqst timeouts" field is also reported in the "rqsts
stalled" field. This is because each timed out request is also a stalled request.

rqst deadlocks
This field gives the number of stalled enqueue lock requests for new locks that ulti-
mately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle
Rdb without the application program ever knowing there was a deadlock. Therefore,
the number shown in this field does not necessarily reflect the number of deadlocks
reported to the application program.

Each lock deadlock reported in the "rqst deadlocks" field is also reported in the "rqsts
stalled" field. This is because each deadlocked request is also a stalled request.

locks promoted
This field gives the number of enqueue lock requests to promote an existing lock to a
higher lock mode. Whether or not the lock request succeeds, it is included in this
count. The "proms not queued," "proms stalled," and "prom deadlocks" counts
provide further detail for the locks promotion statistics.

proms not queued
This field gives the number of enqueue lock requests to promote an existing lock that
were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock
without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary lock-
ning protocol to avoid unnecessary deadlocks. This number is one measure of lock
contention.

proms stalled
This field gives the number of enqueue lock requests to promote an existing lock to a
higher lock mode that were stalled because of a lock conflict. Whether or not the lock
request ultimately succeeds, it is included in this count. This number is one measure of
lock contention.

prom timeouts
This field shows the total number of lock promotions that could not be granted be-
cause they timed out. These are typically logical areas.

Each promotion timeout reported in the "prom timeouts" field is also reported in the
"proms stalled" field. This is because each timed out request is also a stalled request.
This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[
\text{(locks requested)} - \text{(rqsts not queued)} - \text{(rqst deadlocks)} - \text{(locks released)}.
\]

This field gives the number of blocking ASTs, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**ON-SCREEN MENU OPTIONS**

- **Graph** This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers** This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options** This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause** This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset** This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Time_plot** This on-screen menu option plots a specific field's value by time.
- **Unreset** This on-screen menu option restores the original statistic values after using “Reset” option.
- **X_plot** This on-screen menu option plots a specific field's value using scatter-based display.
- **Yank** Places a selected statistic on the “Custom Statistics” screen.

**SCREEN INFORMATION**

- **Page Navigation** This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
- **Binary File Support** This screen is available during replay of a binary file specified by the INPUT qualifier.
- **Cluster Statistic Collection Support** This screen is integrated into the “Cluster Statistic Collection” facility.
- **Zoom Screen** This screen does not have a zoom screen.

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
Buffer/Page Locks Screen

This screen monitors the database page locks. Page locks are used to manage the database page buffer pool.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (buffer locks)” screen:

```
Rate: 1.00 Second       Locking (buffer locks)         Elapsed: 00:00:36.88
Page: 1 of 1            DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1      Mode: Online

statistic........      rate.per.second............. total....... average...... count....... per.trans....
name...............      max...... cur...... avg...... count....... per.trans....
locks requested                5        0        0.5           19           0.0
rqsts not queued              0        0        0.0            0           0.0
rqsts stalled                0        0        0.0            0           0.0
rqst timeouts                0        0        0.0            0           0.0
rqst deadlocks               0        0        0.0            0           0.0
locks promoted               7        0        0.8           30           0.0
proms not queued            0        0        0.0            0           0.0
proms stalled               0        0        0.0            0           0.0
prom timeouts               0        0        0.0            0           0.0
prom deadlocks              0        0        0.0            0           0.0
locks demoted               3        0        0.4           15           0.0
locks released              3        0        0.3           14           0.0
blocking ASTs               3        0        0.4           15           0.0
stall time x100             0        0        0.0            0           0.0
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

SCREEN FIELDS

- **locks requested**: This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count.
The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

rqsts not queued This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

rqsts stalled This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

rqst timeouts This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

rqst deadlocks This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

locks promoted This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued,” “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.

proms not queued This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

proms stalled This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

prom timeouts This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.
prom deadlocks: This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

locks demoted: This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released: This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\].

blocking AST's: This field gives the number of blocking AST's, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking AST's reported is actually comprised of two different types of blocking AST's, those blocking AST's externally generated and those blocking AST's internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking AST's.

stall time x100: This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Graph**  This on-screen menu option displays a histogram graph instead of numbers.

**Numbers**  This on-screen menu option displays numeric statistics instead of histogram graph.

**Options**  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Pause**  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

**Reset**  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

**Time_plot**  This on-screen menu option plots a specific field's value by time.

**Unreset**  This on-screen menu option restores the original statistic values after using “Reset” option.

**X_plot**  This on-screen menu option plots a specific field's value using scatter-based display.

**Yank**  Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

**Page Navigation**  This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

**Binary File Support**  This screen is available during replay of a binary file specified by the INPUT qualifier.

**Cluster Statistic Collection Support**  This screen is integrated into the “Cluster Statistic Collection” facility.

**Zoom Screen**  This screen does not have a zoom screen.

This screen does not have any configuration options.
Record Locks Screen

This screen monitors the database record locks. Record locks are used to maintain the logical consistency of the database. All record locks in the adjustable lock granularity tree are included here.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (record locks)” screen:

Rate: 1.00 Second          Locking (record locks)    Elapsed: 00:00:36.88
Page: 1 of 1    DISK$: [DB_HOMEDIR.WORK.STATS]MP_PERSONNEL.RDB;1 Mode: Online

| statistic.......... | rate.per.second............. total....... average...... count....... per.trans.... |
|--------------------|----------------------------|-------------|-----------------|-----------------|-----------------|
| locks requested     | 5                          | 0           | 0.5             | 19              | 0.0             |
| rqsts not queued   | 0                          | 0           | 0.0             | 0               | 0.0             |
| rqsts stalled      | 0                          | 0           | 0.0             | 0               | 0.0             |
| rqst timeouts      | 0                          | 0           | 0.0             | 0               | 0.0             |
| rqst deadlocks     | 0                          | 0           | 0.0             | 0               | 0.0             |
| locks promoted     | 7                          | 0           | 0.8             | 30              | 0.0             |
| proms not queued   | 0                          | 0           | 0.0             | 0               | 0.0             |
| proms stalled      | 0                          | 0           | 0.0             | 0               | 0.0             |
| prom timeouts      | 0                          | 0           | 0.0             | 0               | 0.0             |
| prom deadlocks     | 0                          | 0           | 0.0             | 0               | 0.0             |
| locks demoted      | 3                          | 0           | 0.4             | 15              | 0.0             |
| locks released     | 3                          | 0           | 0.3             | 14              | 0.0             |
| blocking ASTs      | 3                          | 0           | 0.4             | 15              | 0.0             |
| stall time x100    | 0                          | 0           | 0.0             | 0               | 0.0             |

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!

SCREEn FIELDS

- **locks requested**: This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count.
The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

rqsts not queued This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

rqsts stalled This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

rqst timeouts This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

rqst deadlocks This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

locks promoted This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued,” “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.

proms not queued This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

proms stalled This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

prom timeouts This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.
prom deadlocks: This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

locks demoted: This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released: This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\]

blocking AST's: This field gives the number of blocking AST’s, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking AST’s reported is actually comprised of two different types of blocking AST’s, those blocking AST’s externally generated and those blocking AST’s internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking AST’s.

stall time x100: This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
### On-Screen Menu Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graph</td>
<td>This on-screen menu option displays a histogram graph instead of numbers.</td>
</tr>
<tr>
<td>Numbers</td>
<td>This on-screen menu option displays numeric statistics instead of histogram graph.</td>
</tr>
<tr>
<td>Options</td>
<td>This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.</td>
</tr>
<tr>
<td>Pause</td>
<td>This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.</td>
</tr>
<tr>
<td>Reset</td>
<td>This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.</td>
</tr>
<tr>
<td>Time_plot</td>
<td>This on-screen menu option plots a specific field's value by time.</td>
</tr>
<tr>
<td>Unreset</td>
<td>This on-screen menu option restores the original statistic values after using “Reset” option.</td>
</tr>
<tr>
<td>X_plot</td>
<td>This on-screen menu option plots a specific field's value using scatter-based display.</td>
</tr>
<tr>
<td>Yank</td>
<td>Places a selected statistic on the “Custom Statistics” screen.</td>
</tr>
</tbody>
</table>

### Screen Information

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Page Navigation</td>
<td>This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.</td>
</tr>
<tr>
<td>Binary File Support</td>
<td>This screen is available during replay of a binary file specified by the INPUT qualifier.</td>
</tr>
<tr>
<td>Cluster Statistic Collection Support</td>
<td>This screen is integrated into the “Cluster Statistic Collection” facility.</td>
</tr>
<tr>
<td>Zoom Screen</td>
<td>This screen does not have a zoom screen.</td>
</tr>
</tbody>
</table>

### Configuration Options

This screen does not have any configuration options.
SEQBLK Locks Screen

This screen monitors the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global transaction sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (SEQBLK locks)” screen:

```
Rate: 1.00 Second            Locking (SEQBLK locks)         Elapsed: 00:00:36.88
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online
--------------------------------------------------------------------------------
statistic.........      rate.per.second............. total....... average......
name..............      max..... cur..... avg....... count....... per.trans....
--------------------------------------------------------------------------------
locks requested                5        0        0.5           19           0.0
rqsts not queued              0        0        0.0            0           0.0
rqsts stalled                 0        0        0.0            0           0.0
rqst timeouts                 0        0        0.0            0           0.0
rqst deadlocks                0        0        0.0            0           0.0
locks promoted                 7        0        0.8           30           0.0
proms not queued              0        0        0.0            0           0.0
proms stalled                 0        0        0.0            0           0.0
prom timeouts                 0        0        0.0            0           0.0
prom deadlocks                0        0        0.0            0           0.0
locks demoted                  3        0        0.4           15           0.0
locks released                 3        0        0.3           14           0.0
blocking ASTs                  3        0        0.4           15           0.0
stall time x100                0        0        0.0            0           0.0
--------------------------------------------------------------------------------
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

**SCREEN FIELDS**

- **locks requested**: This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count.
The "rqsts not queued", "rqsts stalled", and "rqst deadlocks" counts provide further detail for enqueue lock requests statistics.

rqsts not queued  This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

rqsts stalled  This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

rqst timeouts  This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the "rqst timeouts" field is also reported in the "rqsts stalled" field. This is because each timed out request is also a stalled request.

rqst deadlocks  This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the "rqst deadlocks" field is also reported in the "rqsts stalled" field. This is because each deadlocked request is also a stalled request.

locks promoted  This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The "proms not queued," "proms stalled," and "prom deadlocks" counts provide further detail for the locks promotion statistics.

proms not queued  This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

proms stalled  This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

prom timeouts  This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the "prom timeouts" field is also reported in the "proms stalled" field. This is because each timed out request is also a stalled request.
prom deadlocks This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

locks demoted This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\].

blocking AST's This field gives the number of blocking AST’s, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking AST’s reported is actually comprised of two different types of blocking AST’s, those blocking AST’s externally generated and those blocking AST’s internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking AST’s.

stall time x100 This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Graph**
This on-screen menu option displays a histogram graph instead of numbers.

**Numbers**
This on-screen menu option displays numeric statistics instead of histogram graph.

**Options**
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Pause**
This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

**Reset**
This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

**Time_plot**
This on-screen menu option plots a specific field's value by time.

**Unreset**
This on-screen menu option restores the original statistic values after using “Reset” option.

**X_plot**
This on-screen menu option plots a specific field's value using scatter-based display.

**Yank**
Places a selected statistic on the “Custom Statistics” screen.

---

**SCREEN INFORMATION**

This section discusses screen-specific issues.

**Page Navigation**
This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

**Binary File Support**
This screen is available during replay of a binary file specified by the INPUT qualifier.

**Cluster Statistic Collection Support**
This screen is integrated into the “Cluster Statistic Collection” facility.

**Zoom Screen**
This screen does not have a zoom screen.

---

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
FILID Locks Screen

This screen monitors the database file identification (FILID) locks. The FILID locks are used to maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (FILID locks)” screen:

Rate: 1.00 Second          Locking (FILID locks)          Elapsed: 00:00:36.88
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MP_PERSONNEL.RDB;1    Mode: Online
--------------------------------------------------------------------------------
statistic........      rate.per.second............. total....... average...... count....... per.trans....
name..............      max..... cur..... avg.......
locks requested                5        0        0.5           19           0.0
rqsts not queued              0        0        0.0            0           0.0
rqsts stalled                 0        0        0.0            0           0.0
rqst timeouts                 0        0        0.0            0           0.0
rqst deadlocks                0        0        0.0            0           0.0
locks promoted                 7        0        0.8           30           0.0
proms not queued              0        0        0.0            0           0.0
proms stalled                 0        0        0.0            0           0.0
prom timeouts                 0        0        0.0            0           0.0
prom deadlocks                0        0        0.0            0           0.0
locks demoted                  3        0        0.4           15           0.0
locks released                 3        0        0.3           14           0.0
blocking ASTs                  3        0        0.4           15           0.0
stall time x100               0        0        0.0            0           0.0
--------------------------------------------------------------------------------

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

SCREEN FIELDS

locks requested This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count.
The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

**rqsts not queued**  
This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

**rqsts stalled**  
This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

**rqst timeouts**  
This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

**rqst deadlocks**  
This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

**locks promoted**  
This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued,” “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.

**proms not queued**  
This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

**proms stalled**  
This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

**prom timeouts**  
This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.
prom deadlocks  This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

locks demoted  This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released  This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[
\text{locks requested} - \text{rqsts not queued} - \text{rqst deadlocks} - \text{locks released}.
\]

blocking ASTs  This field gives the number of blocking ASTs, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

stall time x100  This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Graph**
This on-screen menu option displays a histogram graph instead of numbers.

**Numbers**
This on-screen menu option displays numeric statistics instead of histogram graph.

**Options**
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Pause**
This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

**Reset**
This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

**Time_plot**
This on-screen menu option plots a specific field’s value by time.

**Unreset**
This on-screen menu option restores the original statistic values after using “Reset” option.

**X_plot**
This on-screen menu option plots a specific field’s value using scatter-based display.

**Yank**
Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

**Page Navigation**
This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

**Binary File Support**
This screen is available during replay of a binary file specified by the INPUT qualifier.

**Cluster Statistic Collection Support**
This screen is integrated into the “Cluster Statistic Collection” facility.

**Zoom Screen**
This screen does not have a zoom screen.

This screen does not have any configuration options.
**TSNBLK Locks Screen**

This screen monitors the database transaction block (TSNBLK) locks. The TSNBLK locks are used to control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (TSNBLK locks)” screen:

```
Rate: 1.00 Second            Locking (TSNBLK locks)         Elapsed: 00:00:36.88
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online
--------------------------------------------------------------------------------
statistic.......      rate.per.second............. total....... average......
name..............      max..... cur..... avg....... count....... per.trans....
locks requested                  5        0        0.5           19           0.0
rqsts not queued                 0        0        0.0            0           0.0
rqsts stalled                   0        0        0.0            0           0.0
rqst timeouts                   0        0        0.0            0           0.0
rqst deadlocks                  0        0        0.0            0           0.0
locks promoted                  7        0        0.8           30           0.0
proms not queued                0        0        0.0            0           0.0
proms stalled                  0        0        0.0            0           0.0
prom timeouts                   0        0        0.0            0           0.0
prom deadlocks                  0        0        0.0            0           0.0
locks demoted                  3        0        0.4           15           0.0
locks released                  3        0        0.3           14           0.0
blocking ASTs                   3        0        0.4           15           0.0
stall time x100                0        0        0.0            0           0.0
--------------------------------------------------------------------------------
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !
This field gives the number of enqueue lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count. The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued”, “proms stalled”, and “prom deadlocks” counts provide further detail for the locks promotion statistics.

This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.
Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.

**prom deadlocks**

This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

**locks demoted**

This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

**locks released**

This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\].

**blocking ASTs**

This field gives the number of blocking ASTs, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

**stall time x100**

This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total
can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

ON-SCREEN MENU OPTIONS

- **Graph**
  This on-screen menu option displays a histogram graph instead of numbers.

- **Numbers**
  This on-screen menu option displays numeric statistics instead of histogram graph.

- **Options**
  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

- **Pause**
  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the "Pause" on-screen menu option will resume the statistic presentation.

- **Reset**
  This on-screen menu option resets temporarily resets the statistic information. Use the "Unreset" on-screen menu option to restore the original statistic information.

- **Time_plot**
  This on-screen menu option plots a specific field's value by time.

- **Unreset**
  This on-screen menu option restores the original statistic values after using "Reset" option.

- **X_plot**
  This on-screen menu option plots a specific field's value using scatter-based display.

- **Yank**
  Places a selected statistic on the "Custom Statistics" screen.

SCREEN INFORMATION

This section discusses screen-specific issues.

- **Page Navigation**
  This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

- **Binary File Support**
  This screen is available during replay of a binary file specified by the INPUT qualifier.

- **Cluster Statistic Collection Support**
  This screen is integrated into the "Cluster Statistic Collection" facility.

- **Zoom Screen**
  This screen does not have a zoom screen.
This screen does not have any configuration options.
RTUPB Locks Screen

This screen monitors the database runtime user process block (RTUPB) lock. The RTUPB lock is used to maintain a consistent list of the users who are attached to the database.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (RTUPB locks)” screen:

Rate: 1.00 Second            Locking (RTUPB locks)          Elapsed: 00:00:36.88
Page: 1 of 1    DISK$: [DB_HOMEDIR.WORK.STATS]MP_PERSONNEL.RDB;1     Mode: Online

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
<th>count</th>
<th>per.trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>locks requested</td>
<td>5</td>
<td>0</td>
<td>0.5</td>
<td>19</td>
<td>0</td>
</tr>
<tr>
<td>rqsts not queued</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rqsts stalled</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rqst timeouts</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>rqst deadlocks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>locks promoted</td>
<td>7</td>
<td>0</td>
<td>0.8</td>
<td>30</td>
<td>0</td>
</tr>
<tr>
<td>proms not queued</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>proms stalled</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>prom timeouts</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>prom deadlocks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>locks demoted</td>
<td>3</td>
<td>0</td>
<td>0.4</td>
<td>15</td>
<td>0</td>
</tr>
<tr>
<td>locks released</td>
<td>3</td>
<td>0</td>
<td>0.3</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>blocking ASTs</td>
<td>3</td>
<td>0</td>
<td>0.4</td>
<td>15</td>
<td>0</td>
</tr>
<tr>
<td>stall time x100</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

- **locks requested**: This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count.
The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

rqsts not queued
This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

rqsts stalled
This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

rqst timeouts
This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

rqst deadlocks
This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

locks promoted
This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued,” “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.

proms not queued
This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

proms stalled
This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

prom timeouts
This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.
This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\].

This field gives the number of blocking ASTs, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

### ON-SCREEN MENU OPTIONS

- **Graph**: This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**: This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**: This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause**: This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**: This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Time_plot**: This on-screen menu option plots a specific field's value by time.
- **Unreset**: This on-screen menu option restores the original statistic values after using “Reset” option.
- **X_plot**: This on-screen menu option plots a specific field's value using scatter-based display.
- **Yank**: Places a selected statistic on the “Custom Statistics” screen.

### SCREEN INFORMATION

- **Page Navigation**: This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
- **Binary File Support**: This screen is available during replay of a binary file specified by the INPUT qualifier.
- **Cluster Statistic Collection Support**: This screen is integrated into the “Cluster Statistic Collection” facility.
- **Zoom Screen**: This screen does not have a zoom screen.

### CONFIGURATION OPTIONS

This screen does not have any configuration options.
**ACTIVE Locks Screen**

This screen monitors the database active user bit map (ACTIVE) lock. The ACTIVE lock is used to maintain a consistent list (in bit map form) of the users who are attached to the database.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (ACTIVE locks)” screen:

```
Rate: 1.00 Second   Locking (ACTIVE locks)   Elapsed: 00:00:36.88
Page: 1 of 1   DISK$: [DB_HOMEDIR.WORK.STATS]MP_PERSONNEL.RDB;1   Mode: Online

<table>
<thead>
<tr>
<th>statistic ..........</th>
<th>rate.per.second ..........</th>
<th>total ..........</th>
<th>ave rage ..........</th>
<th>max .... cur .... avg .... count .... per.trans ....</th>
</tr>
</thead>
<tbody>
<tr>
<td>locks requested .....</td>
<td>5            0          0.5       19          0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rqsts not queued</td>
<td>0            0          0.0       0           0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rqsts stalled</td>
<td>0            0          0.0       0           0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rqst timeouts</td>
<td>0            0          0.0       0           0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rqst deadlocks</td>
<td>0            0          0.0       0           0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>locks promoted</td>
<td>7            0          0.8       30          0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>proms not queued</td>
<td>0            0          0.0       0           0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>proms stalled</td>
<td>0            0          0.0       0           0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>prom timeouts</td>
<td>0            0          0.0       0           0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>prom deadlocks</td>
<td>0            0          0.0       0           0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>locks demoted</td>
<td>3            0          0.4       15          0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>locks released</td>
<td>3            0          0.3       14          0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>blocking ASTs</td>
<td>3            0          0.4       15          0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stall time x100</td>
<td>0            0          0.0       0           0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!
The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

**rqsts not queued**
This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

**rqsts stalled**
This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

**rqst timeouts**
This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

**rqst deadlocks**
This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

**locks promoted**
This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued,” “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.

**proms not queued**
This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

**proms stalled**
This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

**prom timeouts**
This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.
prom deadlocks: This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the "prom deadlocks" field is also reported in the "proms stalled" field. This is because each deadlocked request is also a stalled request.

locks demoted: This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released: This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqst not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\]

blocking ASTs: This field gives the number of blocking ASTs, sometimes referred to as "blasts", delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

stall time x100: This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

<table>
<thead>
<tr>
<th>ON-SCREEN MENU OPTIONS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graph</td>
<td>This on-screen menu option displays a histogram graph instead of numbers.</td>
</tr>
<tr>
<td>Numbers</td>
<td>This on-screen menu option displays numeric statistics instead of histogram graph.</td>
</tr>
<tr>
<td>Options</td>
<td>This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.</td>
</tr>
<tr>
<td>Pause</td>
<td>This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.</td>
</tr>
<tr>
<td>Reset</td>
<td>This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.</td>
</tr>
<tr>
<td>Time_plot</td>
<td>This on-screen menu option plots a specific field's value by time.</td>
</tr>
<tr>
<td>Unreset</td>
<td>This on-screen menu option restores the original statistic values after using “Reset” option.</td>
</tr>
<tr>
<td>X_plot</td>
<td>This on-screen menu option plots a specific field’s value using scatter-based display.</td>
</tr>
<tr>
<td>Yank</td>
<td>Places a selected statistic on the “Custom Statistics” screen.</td>
</tr>
</tbody>
</table>

This section discusses screen-specific issues.

<table>
<thead>
<tr>
<th>SCREEN INFORMATION</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Page Navigation</td>
<td>This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.</td>
</tr>
<tr>
<td>Binary File Support</td>
<td>This screen is available during replay of a binary file specified by the INPUT qualifier.</td>
</tr>
<tr>
<td>Cluster Statistic Collection Support</td>
<td>This screen is integrated into the “Cluster Statistic Collection” facility.</td>
</tr>
<tr>
<td>Zoom Screen</td>
<td>This screen does not have a zoom screen.</td>
</tr>
</tbody>
</table>

This screen does not have any configuration options.
MEMBIT Locks Screen

This screen monitors the database membership node bit map (MEMBIT) lock. The MEMBIT lock is used to maintain a consistent list (in bit map form) of the nodes on which the database is currently accessed.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (MEMBIT locks)” screen:

Rate: 1.00 Second       Locking (MEMBIT locks)     Elapsed: 00:00:36.88
Page: 1 of 1            DISK$: [DB_HOMEDIR.WORK.STATS]MP_PERSONNEL.RDB;1   Mode: Online
--------------------------------------------------------------------------------
statistic........      rate.per.second............. total....... average......
name..............      max..... cur..... avg....... count....... per.trans....
locks requested                5        0        0.5           19           0.0
rqsts not queued              0        0        0.0            0           0.0
rqsts stalled                 0        0        0.0            0           0.0
rqst timeouts                 0        0        0.0            0           0.0
rqst deadlocks                0        0        0.0            0           0.0
locks promoted                 7        0        0.8           30           0.0
proms not queued              0        0        0.0            0           0.0
proms stalled                 0        0        0.0            0           0.0
prom timeouts                 0        0        0.0            0           0.0
prom deadlock                 0        0        0.0            0           0.0
locks demoted                 3        0        0.4           15           0.0
locks released                3        0        0.3           14           0.0
blocking ASTs                 3        0        0.4           15           0.0
stall time x100               0        0        0.0            0           0.0
--------------------------------------------------------------------------------
Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

SCREEN FIELDS

locks requested     This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count.
The "rqsts not queued", "rqsts stalled", and "rqst deadlocks" counts provide further detail for enqueue lock requests statistics.

rqsts not queued This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

rqsts stalled This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

rqst timeouts This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the "rqst timeouts" field is also reported in the "rqsts stalled" field. This is because each timed out request is also a stalled request.

rqst deadlocks This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the "rqst deadlocks" field is also reported in the "rqsts stalled" field. This is because each deadlocked request is also a stalled request.

locks promoted This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The "proms not queued," "proms stalled," and "prom deadlocks" counts provide further detail for the locks promotion statistics.

proms not queued This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

proms stalled This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

prom timeouts This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the "prom timeouts" field is also reported in the "proms stalled" field. This is because each timed out request is also a stalled request.
prom deadlocks  This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the "prom deadlocks" field is also reported in the "proms stalled" field. This is because each deadlock request is also a stalled request.

locks demoted  This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released  This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\].

blocking AST's  This field gives the number of blocking ASTs, sometimes referred to as "blasts", delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

stall time x100  This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

<table>
<thead>
<tr>
<th>ON-SCREEN MENU OPTIONS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graph</td>
<td>This on-screen menu option displays a histogram graph instead of numbers.</td>
</tr>
<tr>
<td>Numbers</td>
<td>This on-screen menu option displays numeric statistics instead of histogram graph.</td>
</tr>
<tr>
<td>Options</td>
<td>This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.</td>
</tr>
<tr>
<td>Pause</td>
<td>This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.</td>
</tr>
<tr>
<td>Reset</td>
<td>This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.</td>
</tr>
<tr>
<td>Time_plot</td>
<td>This on-screen menu option plots a specific field’s value by time.</td>
</tr>
<tr>
<td>Unreset</td>
<td>This on-screen menu option restores the original statistic values after using “Reset” option.</td>
</tr>
<tr>
<td>X_plot</td>
<td>This on-screen menu option plots a specific field’s value using scatter-based display.</td>
</tr>
<tr>
<td>Yank</td>
<td>Places a selected statistic on the “Custom Statistics” screen.</td>
</tr>
</tbody>
</table>

This section discusses screen-specific issues.

<table>
<thead>
<tr>
<th>SCREEN INFORMATION</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Page Navigation</td>
<td>This screen does <strong>not</strong> contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.</td>
</tr>
<tr>
<td>Binary File Support</td>
<td>This screen is available during replay of a binary file specified by the INPUT qualifier.</td>
</tr>
<tr>
<td>Cluster Statistic Collection Support</td>
<td>This screen is integrated into the “Cluster Statistic Collection” facility.</td>
</tr>
<tr>
<td>Zoom Screen</td>
<td>This screen does <strong>not</strong> have a zoom screen.</td>
</tr>
</tbody>
</table>

This screen does **not** have any configuration options.
AIJ Locks Screen

This screen monitors the after-image journal (AIJ) locks. The AIJ locks are used to control reading and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, one local AIJ lock on each cluster node manages the global AIJ buffers on that node.

Note that there are actually two separate AIJ locks monitored by this screen. The “global” AIJ lock serializes access to the AIJ end-of-file, while the “local” AIJ lock serializes access to the node-specific AIJ cache located in the database global section.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (AIJ locks)” screen:

```
Rate: 1.00 Second       Locking (AIJ locks)     Elapsed: 00:00:36.88
Page: 1 of 1            DISK$: [DB_HOMEDIR.WORK.STATS] MF_PERSONNEL.RDB;1     Mode: Online
--------------------------------------------------------------------------------
statistic............. rate.per.second............. total....... average...... count....... per.trans....
name..............      max..... cur..... avg....... count....... per.trans....
locks requested                5        0        0.5           19           0.0
rqsts not queued              0        0        0.0            0           0.0
rqsts stalled                 0        0        0.0            0           0.0
rqst timeouts                 0        0        0.0            0           0.0
rqst deadlocks                0        0        0.0            0           0.0
locks promoted                7        0        0.8           30           0.0
proms not queued              0        0        0.0            0           0.0
proms stalled                 0        0        0.0            0           0.0
prom timeouts                 0        0        0.0            0           0.0
prom deadlocks                0        0        0.0            0           0.0
locks demoted                 3        0        0.4           15           0.0
locks released                3        0        0.3           14           0.0
blocking ASTs                 3        0        0.4           15           0.0
stall time x100               0        0        0.0            0           0.0
--------------------------------------------------------------------------------
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!
 locks requested  This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count. The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

 rqsts not queued  This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

 rqsts stalled  This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

 rqst timeouts  This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

 Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

 rqst deadlocks  This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

 Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

 locks promoted  This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued,” “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.

 proms not queued  This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

 proms stalled  This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.
prom timeouts This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.

prom deadlocks This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

locks demoted This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\].

blocking AST’s This field gives the number of blocking AST’s, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking AST’s reported is actually comprised of two different types of blocking AST’s, those blocking AST’s externally generated and those blocking AST’s internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking AST’s.
This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.

### ON-SCREEN MENU OPTIONS

- **Graph**: This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**: This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**: This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause**: This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**: This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Time_plot**: This on-screen menu option plots a specific field's value by time.
- **Unreset**: This on-screen menu option restores the original statistic values after using “Reset” option.
- **X_plot**: This on-screen menu option plots a specific field's value using scatter-based display.
- **Yank**: Places a selected statistic on the “Custom Statistics” screen.

### SCREEN INFORMATION

- **Page Navigation**: This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
- **Binary File Support**: This screen is available during replay of a binary file specified by the INPUT qualifier.
- **Cluster Statistic Collection Support**: This screen is integrated into the “Cluster Statistic Collection” facility.
- **Zoom Screen**: This screen does not have a zoom screen.
This screen does not have any configuration options.
Snapshot Locks Screen

This screen monitors the database snapshot area cursor (SAC) locks. The snapshot locks are used to manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (snapshot locks)” screen:

```
Rate: 1.00 Second           Locking (snapshot locks)        Elapsed: 00:00:36.88
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online
--------------------------------------------------------------------------------
statistic.........      rate.per.second............. total....... average......
name..............      max..... cur..... avg....... count....... per.trans....
locks requested                5        0        0.5           19           0.0
rqsts not queued              0        0        0.0            0           0.0
rqsts stalled                 0        0        0.0            0           0.0
rqst timeouts                0        0        0.0            0           0.0
rqst deadlocks               0        0        0.0            0           0.0
locks promoted                7        0        0.8           30           0.0
proms not queued             0        0        0.0            0           0.0
proms stalled                0        0        0.0            0           0.0
prom timeouts                0        0        0.0            0           0.0
prom deadlocks               0        0        0.0            0           0.0
locks demoted                 3        0        0.4           15           0.0
locks released                3        0        0.3           14           0.0
blocking ASTs                 3        0        0.4           15           0.0
stall time x100               0        0        0.0            0           0.0
--------------------------------------------------------------------------------
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

**SCREEN LOCATION**

**SCREEN EXAMPLE**

**SCREEN FIELDS**

- **locks requested**: This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count.
The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

**rqsts not queued**  This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

**rqsts stalled**  This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

**rqst timeouts**  This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

**rqst deadlocks**  This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

**locks promoted**  This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued,” “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.

**proms not queued**  This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

**proms stalled**  This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

**prom timeouts**  This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.
This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the "prom deadlocks" field is also reported in the "proms stalled" field. This is because each deadlocked request is also a stalled request.

This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\].

This field gives the number of blocking ASTs, sometimes referred to as "blasts", delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Graph**
This on-screen menu option displays a histogram graph instead of numbers.

**Numbers**
This on-screen menu option displays numeric statistics instead of histogram graph.

**Options**
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Pause**
This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

**Reset**
This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

**Time_plot**
This on-screen menu option plots a specific field's value by time.

**Unreset**
This on-screen menu option restores the original statistic values after using “Reset” option.

**X_plot**
This on-screen menu option plots a specific field's value using scatter-based display.

**Yank**
Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

**Page Navigation**
This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

**Binary File Support**
This screen is available during replay of a binary file specified by the INPUT qualifier.

**Cluster Statistic Collection Support**
This screen is integrated into the “Cluster Statistic Collection” facility.

**Zoom Screen**
This screen does not have a zoom screen.

This screen does not have any configuration options.
Freeze Locks Screen

This screen monitors the database freeze lock. The freeze lock is used to suspend database activity on all nodes of the cluster while a database recovery process is running.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (freeze locks)” screen:

---

Rate: 1.00 Second    Locking (freeze locks)    Elapsed: 00:00:36.88
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1    Mode: Online

<table>
<thead>
<tr>
<th>statistic...........</th>
<th>rate.per.second.........</th>
<th>total.......</th>
<th>average......</th>
</tr>
</thead>
<tbody>
<tr>
<td>locks requested</td>
<td>5</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>rqsts not queued</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>rqsts stalled</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>rqst timeouts</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>rqst deadlocks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>locks promoted</td>
<td>7</td>
<td>0</td>
<td>0.8</td>
</tr>
<tr>
<td>proms not queued</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>proms stalled</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>prom timeouts</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>prom deadlocks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>locks demoted</td>
<td>3</td>
<td>0</td>
<td>0.4</td>
</tr>
<tr>
<td>locks released</td>
<td>3</td>
<td>0</td>
<td>0.3</td>
</tr>
<tr>
<td>blocking ASTs</td>
<td>3</td>
<td>0</td>
<td>0.4</td>
</tr>
<tr>
<td>stall time x100</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

---

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

screen fields

| locks requested | This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count. |
The "rqsts not queued", "rqsts stalled", and "rqst deadlocks" counts provide further detail for enqueue lock requests statistics.

rqsts not queued  This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

rqsts stalled  This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

rqst timeouts  This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the "rqst timeouts" field is also reported in the "rqsts stalled" field. This is because each timed out request is also a stalled request.

rqst deadlocks  This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the "rqst deadlocks" field is also reported in the "rqsts stalled" field. This is because each deadlocked request is also a stalled request.

locks promoted  This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The "proms not queued, " "proms stalled, " and "prom deadlocks" counts provide further detail for the locks promotion statistics.

proms not queued  This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

proms stalled  This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

prom timeouts  This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the "prom timeouts" field is also reported in the "proms stalled" field. This is because each timed out request is also a stalled request.
prom deadlocks This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

locks demoted This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[ (\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released}) \]

blocking AST's This field gives the number of blocking ASTs, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

stall time x100 This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**ON-SCREEN MENU OPTIONS**

- **Graph**: This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**: This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**: This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause**: This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**: This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Time_plot**: This on-screen menu option plots a specific field's value by time.
- **Unreset**: This on-screen menu option restores the original statistic values after using “Reset” option.
- **X_plot**: This on-screen menu option plots a specific field's value using scatter-based display.
- **Yank**: Places a selected statistic on the “Custom Statistics” screen.

**SCREEN INFORMATION**

- **Page Navigation**: This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
- **Binary File Support**: This screen is available during replay of a binary file specified by the/INPUT qualifier.
- **Cluster Statistic Collection Support**: This screen is integrated into the “Cluster Statistic Collection” facility.
- **Zoom Screen**: This screen does not have a zoom screen.

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
Quiet Point Locks Screen

This screen monitors the database quiet-point lock.

The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or AIJ Backup Server (ABS). The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup utility. The Oracle Rdb Hot Standby facility also uses the quiet-point lock during startup.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (quiet point locks)” screen:

```
Rate: 1.00 Second      Locking (quiet point locks)   Elapsed: 00:00:36.88
Page: 1 of 1           DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1   Mode: Online
--------------------------------------------------------------------------------
statistic........    rate.per.second............ total....... average......
name..............    max..... cur..... avg....... count....... per.trans....
locks requested    5        0        0.5           19           0.0
rqsts not queued   0        0        0.0            0           0.0
rqsts stalled      0        0        0.0            0           0.0
rqst timeouts      0        0        0.0            0           0.0
rqst deadlocks     0        0        0.0            0           0.0
locks promoted     7        0        0.8           30           0.0
proms not queued   0        0        0.0            0           0.0
proms stalled      0        0        0.0            0           0.0
prom timeouts       0        0        0.0            0           0.0
prom deadlocks     0        0        0.0            0           0.0
locks demoted      3        0        0.4           15           0.0
locks released     3        0        0.3           14           0.0
blocking ASTs      3        0        0.4           15           0.0
stall time x100    0        0        0.0            0           0.0
--------------------------------------------------------------------------------
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !
### Screen Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>locks requested</strong></td>
<td>This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count. The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.</td>
</tr>
<tr>
<td><strong>rqsts not queued</strong></td>
<td>This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.</td>
</tr>
<tr>
<td><strong>rqsts stalled</strong></td>
<td>This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.</td>
</tr>
<tr>
<td><strong>rqst timeouts</strong></td>
<td>This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.</td>
</tr>
<tr>
<td><strong>rqst deadlocks</strong></td>
<td>This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program. Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.</td>
</tr>
<tr>
<td><strong>locks promoted</strong></td>
<td>This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued”, “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.</td>
</tr>
<tr>
<td><strong>proms not queued</strong></td>
<td>This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.</td>
</tr>
<tr>
<td><strong>proms stalled</strong></td>
<td>This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.</td>
</tr>
</tbody>
</table>
prom timeouts  This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.

prom deadlocks  This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

locks demoted  This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released  This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[
\text{locks requested} - \text{rqsts not queued} - \text{rqst deadlocks} - \text{locks released}
\]

blocking AST's  This field gives the number of blocking AST’s, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking AST’s reported is actually comprised of two different types of blocking AST's, those blocking AST’s externally generated and those blocking AST’s internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.
stall time x100 This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Graph This on-screen menu option displays a histogram graph instead of numbers.

Numbers This on-screen menu option displays numeric statistics instead of histogram graph.

Options This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

Pause This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

Reset This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

Time_plot This on-screen menu option plots a specific field’s value by time.

Unreset This on-screen menu option restores the original statistic values after using “Reset” option.

X_plot This on-screen menu option plots a specific field’s value using scatter-based display.

Yank Places a selected statistic on the “Custom Statistics” screen.

Using the RDM$BIND_SNAP_QUIET_POINT logical with a value of “0” in almost all cases improves the quiet-point lock acquisition, by allowing snapshot transactions to not be affected by the lock.

Page Navigation This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen This screen does not have a zoom screen.
This screen does not have any configuration options.
Logical Area Locks Screen

This screen monitors database logical area locks. Logical area locks are obtained when Oracle Rdb readies tables. Lock carryover can help reduce the number of logical area locks.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (logical area locks)” screen:

Rate: 1.00 Second            Locking (logical area locks)          Elapsed: 00:00:36.88
Page: 1 of 1  DISK$:[DB_HOMEDIR.WORK.STATS]MP_PERSONNEL.RDB;1  Mode: Online

<table>
<thead>
<tr>
<th>statistic.........</th>
<th>rate.per.second.............</th>
<th>total.......</th>
<th>average......</th>
<th>count.......</th>
<th>per.trans....</th>
</tr>
</thead>
<tbody>
<tr>
<td>locks requested</td>
<td>5</td>
<td>0.5</td>
<td>19</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>rqsts not queued</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>rqsts stalled</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>rqst timeouts</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>rqst deadlocks</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>locks promoted</td>
<td>7</td>
<td>0.8</td>
<td>30</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>proms not queued</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>proms stalled</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>prom timeouts</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>prom deadlocks</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>locks demoted</td>
<td>3</td>
<td>0.4</td>
<td>15</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>locks released</td>
<td>3</td>
<td>0.3</td>
<td>14</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>blocking ASTs</td>
<td>3</td>
<td>0.4</td>
<td>15</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>stall time x100</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td></td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

**SCREEN LOCATION**

**SCREEN EXAMPLE**

**SCREEN FIELDS**

locks requested  This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count.
The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

rqsts not queued
This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

rqsts stalled
This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

rqst timeouts
This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

rqst deadlocks
This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

rqst timeouts
This field gives the total number of enqueue lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

rqst deadlocks
This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

locks promoted
This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued,” “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.

proms not queued
This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

proms stalled
This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

prom timeouts
This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

This field shows the total number of enqueue lock requests that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.
This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\].

This field gives the number of blocking ASTs, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**ON-SCREEN MENU OPTIONS**

- **Graph**: This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**: This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**: This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause**: This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**: This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Time_plot**: This on-screen menu option plots a specific field's value by time.
- **Unreset**: This on-screen menu option restores the original statistic values after using “Reset” option.
- **X_plot**: This on-screen menu option plots a specific field’s value using scatter-based display.
- **Yank**: Places a selected statistic on the “Custom Statistics” screen.

**SCREEN INFORMATION**

- **Page Navigation**: This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
- **Binary File Support**: This screen is available during replay of a binary file specified by the INPUT qualifier.
- **Cluster Statistic Collection Support**: This screen is integrated into the “Cluster Statistic Collection” facility.
- **Zoom Screen**: This screen does not have a zoom screen.

**CONFIGURATION OPTIONS**

- **This screen does not have any configuration options.**
Nowait Transaction Locks Screen

This screen monitors the database nowait transaction lock.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (nowait locks)” screen:

```
Rate: 1.00 Second            Locking (nowait locks)         Elapsed: 00:00:36.88
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online
--------------------------------------------------------------------------------
statistic.........      rate.per.second............. total....... average......
name..............      max..... cur..... avg....... count....... per.trans....
locks requested      5        0        0.5           19           0.0
rqsts not queued              0        0        0.0            0           0.0
rqsts stalled                 0        0        0.0            0           0.0
rqst timeouts                 0        0        0.0            0           0.0
rqst deadlocks                0        0        0.0            0           0.0
locks promoted                 7        0        0.8           30           0.0
proms not queued              0        0        0.0            0           0.0
proms stalled                 0        0        0.0            0           0.0
prom timeouts                 0        0        0.0            0           0.0
prom deadlocks                0        0        0.0            0           0.0
locks demoted                  3        0        0.4           15           0.0
locks released                 3        0        0.3           14           0.0
blocking ASTs                  3        0        0.4           15           0.0
stall time x100                0        0        0.0            0           0.0
--------------------------------------------------------------------------------
Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!
```

SCREEn FIELDS

- **locks requested**: This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count. The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.
rqsts not queued This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

rqsts stalled This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

rqst timeouts This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

rqst deadlocks This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

locks promoted This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued,” “proms stalled,” and “prom deadlocks” counts provide further detail for the locks promotion statistics.

proms not queued This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

proms stalled This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

prom timeouts This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.

prom deadlocks This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved...
by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

- **locks demoted**: This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

- **locks released**: This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[
(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released}).
\]

- **blocking AST's**: This field gives the number of blocking ASTs, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking AST's reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking AST's.

- **stall time x100**: This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Graph**
This on-screen menu option displays a histogram graph instead of numbers.

**Numbers**
This on-screen menu option displays numeric statistics instead of histogram graph.

**Options**
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Pause**
This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

**Reset**
This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

**Time_plot**
This on-screen menu option plots a specific field's value by time.

**Unreset**
This on-screen menu option restores the original statistic values after using “Reset” option.

**X_plot**
This on-screen menu option plots a specific field's value using scatter-based display.

**Yank**
Places a selected statistic on the “Custom Statistics” screen.

---

**NOWAIT transactions do not wait for locks. If a lock requested by a NOWAIT transaction cannot be granted immediately, Oracle Rdb issues an error message and the transaction aborts. As part of carry-over lock optimization, a NOWAIT transaction requests, acquires, and holds a NOWAIT lock. This signals other processes accessing the database that a NOWAIT transaction exists and results in the release of all carry-over locks. If carry-over locks were not released, a NOWAIT transaction could not access an area held by a WAIT transaction’s carry-over lock until the WAIT transaction’s process detached from the database.**

Every NOWAIT transaction requests the NOWAIT lock at transaction start in CW mode and waits until this lock is granted. When a transaction acquires the NOWAIT lock in CW mode, this indicates that all other users know that a NOWAIT transaction is running and indicates that all carry-over locks have been released, thus reducing the possibility of lock contention.

All transactions request the NOWAIT lock in PR mode at commit time. If the NOWAIT lock is granted in PR mode, it indicates that there are no NOWAIT transactions attached to the database and carry-over locks are permitted. If the NOWAIT
lock request is not granted (because a NOWAIT transaction holds the lock in CW mode), carry-over locks are not permitted.

However, a NOWAIT transaction can experience a delay in acquiring the NOWAIT lock if another transaction is holding the lock. This can result in the following RMU Show Statistic utility stall message: “waiting for NOWAIT signal (CW)”.

The Oracle Rdb “Carry-over Lock Optimization” feature allows transactions to avoid some logical area and storage area lock request overhead at commit time. This section provides a general description of how carry-over lock optimization works, and then describes its particular effects on WAIT and NOWAIT transactions.

An area lock requested by a current transaction is called an active lock. At commit time, Oracle Rdb tries to avoid demoting area locks. Those area locks that are not demoted at commit time are called “carry-over” locks.

While attached to the database, a process can have some active locks (locks used by the current transaction) and some carry-over locks (locks requested in earlier transactions that have not been demoted). If a transaction needs a lock that it has currently marked as carry-over, it can reuse the lock by changing it to an active lock. Thus, the same lock can go from active to carry-over to active multiple times without paying the cost of lock request and demotion. This substantially reduces the number of lock requests if a process accesses the same set of areas repeatedly.

Whenever a WAIT transaction requests an area lock, Oracle Rdb must distinguish between the following two cases in which process A has a lock on area X and process B wants to access the same area:

- If the lock that process A has on area X is a carry-over lock, A gives up the lock on demand, process B gets it, and B continues to process.
- If the lock that process A has on area X is an active lock, A cannot give up the lock before its transaction has completed. In this case, Oracle Rdb sets a flag to indicate that this lock must be demoted when A’s transaction commits, so that B can acquire it. Because process B cannot get the lock on demand, B must wait.

For WAIT transactions, the reduced number of locks associated with carry-over lock optimization can result in an increase in blocking ASTs. You can see an increase in blocking ASTs by using the various RMU Show Statistic utility Locking screens.

Carry-over lock optimization works well when applications are designed so that each transaction accesses its own set of data; that is, transactions do not randomly access data in all partitions, thereby increasing contention. For example, consider the EMPLOYEE_ID column, which partitions the EMPLOYEES table to three areas. Applications that access the EMPLOYEES table should be designed so that transactions access a particular area or set of areas instead of randomly selecting any area. Furthermore, carry-over lock optimization works best if transactions repeatedly access
the same area or set of areas. The partitioning and placement features available in Oracle Rdb should help in this regard.

If NOWAIT transactions are noticeably slow in executing, you can disable carry-over lock optimization by using the CARRY OVER LOCKS ARE [ENABLED | DISABLED] clause with either the SQL CREATE DATABASE or SQL ALTER DATABASE statements. By default, carry-over locks are enabled.

The following example shows how to disable carry-over locks that have been enabled by default.

```
SQL> ALTER DATABASE FILENAME test1
cont> CARRY OVER LOCKS ARE DISABLED;
```

Page Navigation  This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support  This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support  This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  This screen does not have a zoom screen.

This screen does not have any configuration options.
CHAPTER 10 - LOCKING (ONE LOCK TYPE)

CLIENT Locks Screen

This screen monitors the database client information (CLIENT) locks. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

The name in parenthesis in the header region of the screen reflects your screen selection.

This screen resides in the “Locking (One Lock Type)” menu.

The following is an example of the “Locking (Client locks)” screen:

Rate: 1.00 Second        Locking (CLIENT locks)         Elapsed: 00:00:36.88
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
<th>count</th>
<th>per.trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>locks requested</td>
<td>5</td>
<td>0.5</td>
<td>0</td>
<td>19</td>
<td>0.0</td>
</tr>
<tr>
<td>rqsts not queued</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>rqsts stalled</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>rqst timeouts</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>rqst deadlocks</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>locks promoted</td>
<td>7</td>
<td>0.8</td>
<td>30</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>proms not queued</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>proms stalled</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>prom timeouts</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>prom deadlocks</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>locks demoted</td>
<td>3</td>
<td>0.4</td>
<td>15</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>locks released</td>
<td>3</td>
<td>0.3</td>
<td>14</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>blocking ASTs</td>
<td>3</td>
<td>0.4</td>
<td>15</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>stall time x100</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

Available starting with Oracle Rdb 7.0.2.0.

SCREEN FIELDS
This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count. The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued”, “proms stalled”, and “prom deadlocks” counts provide further detail for the locks promotion statistics.

This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.
**prom timeouts**

This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.

**prom deadlocks**

This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

**locks demoted**

This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

**locks released**

This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[(\text{locks requested}) - (\text{rqsts not queued}) - (\text{rqst deadlocks}) - (\text{locks released})\]

**blocking AST’s**

This field gives the number of blocking AST’s, sometimes referred to as “blasts”, delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking AST’s reported is actually comprised of two different types of blocking AST’s, those blocking AST’s externally generated and those blocking AST’s internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.
This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.

---

**ON-SCREEN MENU OPTIONS**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graph</td>
<td>This on-screen menu option displays a histogram graph instead of numbers.</td>
</tr>
<tr>
<td>Numbers</td>
<td>This on-screen menu option displays numeric statistics instead of histogram graph.</td>
</tr>
<tr>
<td>Options</td>
<td>This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.</td>
</tr>
<tr>
<td>Pause</td>
<td>This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.</td>
</tr>
<tr>
<td>Reset</td>
<td>This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.</td>
</tr>
<tr>
<td>Time_plot</td>
<td>This on-screen menu option plots a specific field's value by time.</td>
</tr>
<tr>
<td>Unreset</td>
<td>This on-screen menu option restores the original statistic values after using “Reset” option.</td>
</tr>
<tr>
<td>X_plot</td>
<td>This on-screen menu option plots a specific field's value using scatter-based display.</td>
</tr>
<tr>
<td>Yank</td>
<td>Places a selected statistic on the “Custom Statistics” screen.</td>
</tr>
</tbody>
</table>

---

**SCREEN INFORMATION**

- **Page Navigation**: This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
- **Binary File Support**: This screen is available during replay of a binary file specified by the INPUT qualifier.
- **Cluster Statistic Collection Support**: This screen is integrated into the “Cluster Statistic Collection” facility.
- **Zoom Screen**: This screen does not have a zoom screen.
This screen does not have any configuration options.
Locking (One Stat Field) Screens

RMU Show Statistic Locking (One Stat Field) Screens

The “Locking (One Stat Field)” menu screens display locking information for a specific lock statistic, compared to the various lock types.
Locks Requested Screen

This screen monitors the number of enqueue lock requests for new locks. Whether the lock request succeeds or fails, it is included in these counts.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (locks requested)” screen:

```plaintext
Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor  1-FEB-1998 16:24:25.67
Rate: 1.00 Second          Locking (locks requested)        Elapsed: 00:00:46.33
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
<th>count</th>
<th>per.trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>total locks</td>
<td>5</td>
<td>0</td>
<td>0.4</td>
<td>19</td>
<td>0.0</td>
</tr>
<tr>
<td>area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>FILID locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>AIJ locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>freeze lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>logical area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

total locks  This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.
This field gives statistics for database storage area locks.

buffer page locks This field gives statistics for database page locks. Page locks manage the database page buffer pool.

record locks This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

SEQBLK lock This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

FILID locks This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

TSNBLK locks This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

RTUPB lock This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

ACTIVE lock This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

MEMBIT lock This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

AIJ locks This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.

snapshot locks This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

freeze lock This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.

For more information on transaction recovery, please refer to the "Transaction Recovery Duration Estimate" screen and the "Recovery Statistics" screen.
quiet point lock  This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

logical area locks  Logical area locks are obtained when Oracle Rdb readies tables. Lock carryover can help reduce the number of logical area locks.

nowait transaction  This field gives statistics for the database nowait transaction lock.

CLIENT locks  This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Graph  This on-screen menu option displays a histogram graph instead of numbers.

Numbers  This on-screen menu option displays numeric statistics instead of histogram graph.

Options  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

Pause  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

Reset  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

Time_plot  This on-screen menu option plots a specific field's value by time.

Unreset  This on-screen menu option restores the original statistic values after using “Reset” option.

X_plot  This on-screen menu option plots a specific field’s value using scatter-based display.

Yank  Places a selected statistic on the “Custom Statistics” screen.
This section discusses screen-specific issues.

### Screen Information

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Page Navigation</td>
<td>This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.</td>
</tr>
<tr>
<td>Binary File Support</td>
<td>This screen is available during replay of a binary file specified by the INPUT qualifier.</td>
</tr>
<tr>
<td>Cluster Statistic Collection Support</td>
<td>This screen is integrated into the “Cluster Statistic Collection” facility.</td>
</tr>
<tr>
<td>Zoom Screen</td>
<td>This screen does not have a zoom screen.</td>
</tr>
</tbody>
</table>

### Configuration Options

This screen does not have any configuration options.
**Rqsts Not Queued Screen**

This screen monitors the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting, and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. These numbers are one measure of lock contention.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (rqsts not queued)” screen:

<table>
<thead>
<tr>
<th>statistic ..........</th>
<th>rate.per.second..........</th>
<th>total.......</th>
<th>ave rage......</th>
</tr>
</thead>
<tbody>
<tr>
<td>name ...............</td>
<td>max.....</td>
<td>cur.....</td>
<td>avg.......</td>
</tr>
<tr>
<td>total locks</td>
<td>5</td>
<td>0</td>
<td>0.4</td>
</tr>
<tr>
<td>area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>FILID locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>AIJ locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>freeze lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>logical area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!
This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.

This field gives statistics for database storage area locks.

This field gives statistics for database page locks. Page locks manage the database page buffer pool.

This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.

This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.
For more information on transaction recovery, please refer to the “Transaction Recovery Duration Estimate” screen and the “Recovery Statistics” screen.

**quiet point lock**

This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

**logical area locks**

Logical area locks are obtained when Oracle Rdb reads tables. Lock carryover can help reduce the number of logical area locks.

**nowait transaction**

This field gives statistics for the database nowait transaction lock.

**CLIENT locks**

This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

---

**ON-SCREEN MENU OPTIONS**

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

- **Graph**: This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**: This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**: This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause**: This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**: This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Time_plot**: This on-screen menu option plots a specific field’s value by time.
- **Unreset**: This on-screen menu option restores the original statistic values after using “Reset” option.
- **X_plot**: This on-screen menu option plots a specific field’s value using scatter-based display.
Yank

Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

Page Navigation

This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support

This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support

This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen

This screen does not have a zoom screen.

This screen does not have any configuration options.
Rqsts Stalled Screen

This screen monitors the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting, and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. These numbers are one measure of lock contention.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (rqsts stalled)” screen:

```
Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor 1-FEB-1998 16:24:25.67
Rate: 1.00 Second        Locking (rqsts stalled)       Elapsed: 00:00:46.33
Page: 1 of 1           DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1    Mode: Online

<table>
<thead>
<tr>
<th>statistic........</th>
<th>rate.per.second........</th>
<th>total.......</th>
<th>average......</th>
<th>count.......</th>
<th>per.trans....</th>
</tr>
</thead>
<tbody>
<tr>
<td>total locks</td>
<td>5</td>
<td>0</td>
<td>0.4</td>
<td>19</td>
<td>0.0</td>
</tr>
<tr>
<td>area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>FILID locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>AJI locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>freeze lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>logical area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!
This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.

This field gives statistics for database storage area locks.

This field gives statistics for database page locks. Page locks manage the database page buffer pool.

This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.

This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.
For more information on transaction recovery, please refer to the “Transaction Recovery Duration Estimate” screen and the “Recovery Statistics” screen.

**quiet point lock**

This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

**logical area locks**

Logical area locks are obtained when Oracle Rdb reads tables. Lock carryover can help reduce the number of logical area locks.

**nowait transaction**

This field gives statistics for the database nowait transaction lock.

**CLIENT locks**

This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

---

### ON-SCREEN MENU OPTIONS

- **Graph**
  
  This on-screen menu option displays a histogram graph instead of numbers.

- **Numbers**
  
  This on-screen menu option displays numeric statistics instead of histogram graph.

- **Options**
  
  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

- **Pause**
  
  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

- **Reset**
  
  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

- **Time_plot**
  
  This on-screen menu option plots a specific field’s value by time.

- **Unreset**
  
  This on-screen menu option restores the original statistic values after using “Reset” option.

- **X_plot**
  
  This on-screen menu option plots a specific field’s value using scatter-based display.
**Yank**  Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

<table>
<thead>
<tr>
<th>Screen Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Page Navigation</td>
</tr>
<tr>
<td>Binary File Support</td>
</tr>
<tr>
<td>Cluster Statistic Collection Support</td>
</tr>
<tr>
<td>Zoom Screen</td>
</tr>
</tbody>
</table>

**SCREEN INFORMATION**

This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

This screen is available during replay of a binary file specified by the INPUT qualifier.

This screen is integrated into the “Cluster Statistic Collection” facility.

This screen does not have a zoom screen.

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
**Rqst Timeouts Screen**

This screen monitors the total number of lock requests that could not be granted because they timed out.

These are typically logical areas or record locks.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (rqst timeout)” screen:

```
Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor 1-FEB-1998 16:24:25.67
Rate: 1.00 Second        Locking (rqsts timeouts)        Elapsed: 00:00:46.33
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1    Mode: Online

statistic............ rate.per.second............. total....... average...... count....... per.trans....
name..............  max..... cur..... avg....... count....... per.trans....
total locks                    5        0        0.4           19           0.0
area locks                     0        0        0.0            0           0.0
buffer/page locks               0        0        0.0            0           0.0
record locks                    0        0        0.0            0           0.0
SEQBLK lock                     0        0        0.0            1           0.0
FILID locks                     0        0        0.0            0           0.0
TSNBLK locks                    0        0        0.0            2           0.0
RTUPB lock                      0        0        0.0            0           0.0
ACTIVE lock                     0        0        0.0            0           0.0
MEMBIT lock                     0        0        0.0            0           0.0
AIJ locks                       0        0        0.0            0           0.0
snapshot locks                  0        0        0.0            0           0.0
freeze lock                     0        0        0.0            0           0.0
quiet point lock                0        0        0.0            0           0.0
logical area locks              0        0        0.0            0           0.0
nowait transaction             0        0        0.0            0           0.0
CLIENT locks                    0        0        0.0            0           0.0
```
This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.

This field gives statistics for database storage area locks.

This field gives statistics for database page locks. Page locks manage the database page buffer pool.

This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.

This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.
freeze lock  This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.

For more information on transaction recovery, please refer to the “Transaction Recovery Duration Estimate” screen and the “Recovery Statistics” screen.

quiet point lock  This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

logical area locks  Logical area locks are obtained when Oracle Rdb readies tables. Lock carryover can help reduce the number of logical area locks.

nowait transaction  This field gives statistics for the database nowait transaction lock.

CLIENT locks  This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**ON-SCREEN MENU OPTIONS**

Graph  This on-screen menu option displays a histogram graph instead of numbers.

Numbers  This on-screen menu option displays numeric statistics instead of histogram graph.

Options  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

Pause  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

Reset  This on-screen menu option temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

Time_plot  This on-screen menu option plots a specific field's value by time.
Unreset  This on-screen menu option restores the original statistic values after using “Reset” option.

X_plot  This on-screen menu option plots a specific field’s value using scatter-based display.

Yank  Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

Page Navigation  This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support  This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support  This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  This screen does not have a zoom screen.

This screen does not have any configuration options.
**Rqst Deadlocks Screen**

This screen monitors the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are resolved and retried by Oracle Rdb transparently to the application program, therefore these numbers do not necessarily reflect the number of deadlocks reported to the application program.

These are typically page locks.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (rqst deadlocks)” screen:

```
Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor  1-FEB-1998 16:24:25.67
Rate: 1.00 Second          Locking (rqsts deadlocks)        Elapsed: 00:00:46.33
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

<table>
<thead>
<tr>
<th>statistic ..........</th>
<th>rate.per.second.........</th>
<th>total.......</th>
<th>average......</th>
</tr>
</thead>
<tbody>
<tr>
<td>total locks</td>
<td>5</td>
<td>0</td>
<td>0.4</td>
</tr>
<tr>
<td>area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>FILID locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>AIJ locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>freeze lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>logical area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !
CHAPTER 11 - LOCKING (ONE STAT FIELD)

RMU Show Statistic Handbook

This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.

area locks This field gives statistics for database storage area locks.

buffer page locks This field gives statistics for database page locks. Page locks manage the database page buffer pool.

record locks This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

SEQBLK lock This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

FILID locks This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

TSNBLK locks This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

RTUPB lock This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

ACTIVE lock This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

MEMBIT lock This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

AIJ locks This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.
This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.

For more information on transaction recovery, please refer to the “Transaction Recovery Duration Estimate” screen and the “Recovery Statistics” screen.

This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

Logical area locks are obtained when Oracle Rdb readies tables. Lock carryover can help reduce the number of logical area locks.

This field gives statistics for the database nowait transaction lock.

This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Graph
This on-screen menu option displays a histogram graph instead of numbers.

Numbers
This on-screen menu option displays numeric statistics instead of histogram graph.

Options
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

Pause
This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

Reset
This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
Time_plot  This on-screen menu option plots a specific field's value by time.

Unreset   This on-screen menu option restores the original statistic values after using "Reset" option.

X_plot    This on-screen menu option plots a specific field's value using scatter-based display.

Yank      Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

Page Navigation  This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support  This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support  This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  This screen does not have a zoom screen.

This screen does not have any configuration options.
Locks Promoted Screen

This screen monitors the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether the lock requests succeed or fail, they are included in these counts.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (lock promoted)” screen:

```
Node: MYNODE (1/1/2)  Oracle Rdb X7.0-00 Perf. Monitor  1-FEB-1998 16:24:25.67
Rate: 1.00 Second     Locking (locks promoted)   Elapsed: 00:00:46.33
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1   Mode: Online

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
<th>count</th>
<th>per.trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>total locks</td>
<td>5</td>
<td>0</td>
<td>0.4</td>
<td>19</td>
<td>0.0</td>
</tr>
<tr>
<td>area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>FILID locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>A1J locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>freeze lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>logical area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!

**SCREEN FIELDS**

- **total locks**: This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.
### RMU SHOW STATISTIC HANDBOOK
#### CHAPTER 11 - LOCKING (ONE STAT FIELD)

**area locks**  
This field gives statistics for database storage area locks.

**buffer page locks**  
This field gives statistics for database page locks. Page locks manage the database page buffer pool.

**record locks**  
This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

**SEQBLK lock**  
This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

**FILID locks**  
This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

**TSNBLK locks**  
This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

**RTUPB lock**  
This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

**ACTIVE lock**  
This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

**MEMBIT lock**  
This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

**AIJ locks**  
This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.

**snapshot locks**  
This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

**freeze lock**  
This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.

For more information on transaction recovery, please refer to the "Transaction Recovery Duration Estimate" screen and the "Recovery Statistics" screen.
quiet point lock  This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

logical area locks  Logical area locks are obtained when Oracle Rdb readies tables. Lock carryover can help reduce the number of logical area locks.

nowait transaction  This field gives statistics for the database nowait transaction lock.

CLIENT locks  This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Graph  This on-screen menu option displays a histogram graph instead of numbers.

Numbers  This on-screen menu option displays numeric statistics instead of histogram graph.

Options  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

Pause  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

Reset  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

Time_plot  This on-screen menu option plots a specific field's value by time.

Unreset  This on-screen menu option restores the original statistic values after using “Reset” option.

X_plot  This on-screen menu option plots a specific field’s value using scatter-based display.

Yank  Places a selected statistic on the “Custom Statistics” screen.
This section discusses screen-specific issues.

**SCREEN INFORMATION**

- **Page Navigation**: This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

- **Binary File Support**: This screen is available during replay of a binary file specified by the INPUT qualifier.

- **Cluster Statistic Collection Support**: This screen is integrated into the “Cluster Statistic Collection” facility.

- **Zoom Screen**: This screen does not have a zoom screen.

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
Proms Not Queued Screen

This screen monitors the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting, and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. These numbers are one measure of lock contention.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (proms not queued)” screen:

```
Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor  1-FEB-1998 16:24:25.67
Rate: 1.00 Second       Locking (proms not queued)       Elapsed: 00:00:46.33
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

---

statistic...........      rate.per.second............. total....... average...... count....... per.trans....
name..............      max..... cur..... avg....... count....... per.trans....

  total locks                    5        0        0.4           19           0.0
  area locks                     0        0        0.0            0           0.0
  buffer/page locks              0        0        0.0            0           0.0
  record locks                   0        0        0.0            0           0.0
  SEQBLK lock                    0        0        0.0            1           0.0
  FILID locks                    0        0        0.0            0           0.0
  TSNBLK locks                   0        0        0.0            2           0.0
  RTUPB lock                     0        0        0.0            0           0.0
  ACTIVE lock                    0        0        0.0            0           0.0
  MEMBIT lock                    0        0        0.0            0           0.0
  AIJ locks                      0        0        0.0            0           0.0
  snapshot locks                 0        0        0.0            0           0.0
  freeze lock                    0        0        0.0            0           0.0
  quiet point lock               0        0        0.0            0           0.0
  logical area locks             0        0        0.0            0           0.0
  nowait transaction             0        0        0.0            0           0.0
  CLIENT locks                   0        0        0.0            0           0.0

---

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

---

SCREEN LOCATION
SCREEN EXAMPLE
SCREEN FIELDS
total locks  This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.

area locks  This field gives statistics for database storage area locks.

buffer page locks  This field gives statistics for database page locks. Page locks manage the database page buffer pool.

record locks  This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

SEQBLK lock  This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

FILID locks  This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

TSNBLK locks  This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

RTUPB lock  This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

ACTIVE lock  This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

MEMBIT lock  This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

AIJ locks  This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.

snapshot locks  This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

freeze lock  This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.
For more information on transaction recovery, please refer to the “Transaction Recovery Duration Estimate” screen and the “Recovery Statistics” screen.

**quiet point lock**
This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

**logical area locks**
Logical area locks are obtained when Oracle Rdb reads tables. Lock carryover can help reduce the number of logical area locks.

**nowait transaction**
This field gives statistics for the database nowait transaction lock.

**CLIENT locks**
This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

- **Graph**
  This on-screen menu option displays a histogram graph instead of numbers.

- **Numbers**
  This on-screen menu option displays numeric statistics instead of histogram graph.

- **Options**
  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

- **Pause**
  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

- **Reset**
  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

- **Time_plot**
  This on-screen menu option plots a specific field’s value by time.

- **Unreset**
  This on-screen menu option restores the original statistic values after using “Reset” option.

- **X_plot**
  This on-screen menu option plots a specific field’s value using scatter-based display.
Yank  Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

SCREEN INFORMATION

Page Navigation  This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support  This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support  This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  This screen does not have a zoom screen.

CONFIGURATION OPTIONS

This screen does not have any configuration options.
Proms Stalled Screen

This screen monitors the number of enqueue lock requests to promote an existing lock to a higher lock mode that were stalled because of a lock conflict. Whether the lock requests ultimately succeed or fail (resulting in a deadlock), they are included in these counts. These numbers are one measure of lock contention.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (proms stalled)” screen:

Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor 1-FEB-1998 16:24:25.67
Rate: 1.00 Second Locking (proms stalled) Elapsed: 00:00:46.33
Page: 1 of 1 DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1 Mode: Online

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
<th>count</th>
<th>per.trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>total locks</td>
<td>5</td>
<td>0</td>
<td>0.4</td>
<td>19</td>
<td>0.0</td>
</tr>
<tr>
<td>area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>FILID locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>AIJ locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>freeze lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>logical area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!
This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.

**Area locks**
This field gives statistics for database storage area locks.

**Buffer page locks**
This field gives statistics for database page locks. Page locks manage the database page buffer pool.

**Record locks**
This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

**SEQBLK lock**
This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

**FILID locks**
This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

**TSNBLK locks**
This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

**RTUPB lock**
This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

**ACTIVE lock**
This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

**MEMBIT lock**
This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

**AIJ locks**
This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.

**Snapshot locks**
This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

**Freeze lock**
This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.
For more information on transaction recovery, please refer to the “Transaction Recovery Duration Estimate” screen and the “Recovery Statistics” screen.

**quiet point lock**
This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

**logical area locks**
Logical area locks are obtained when Oracle Rdb reads tables. Lock carryover can help reduce the number of logical area locks.

**nowait transaction**
This field gives statistics for the database nowait transaction lock.

**CLIENT locks**
This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

---

**ON-SCREEN MENU OPTIONS**

- **Graph**
  This on-screen menu option displays a histogram graph instead of numbers.

- **Numbers**
  This on-screen menu option displays numeric statistics instead of histogram graph.

- **Options**
  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

- **Pause**
  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

- **Reset**
  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.

- **Time_plot**
  This on-screen menu option plots a specific field’s value by time.

- **Unreset**
  This on-screen menu option restores the original statistic values after using “Reset” option.

- **X_plot**
  This on-screen menu option plots a specific field’s value using scatter-based display.
Yank Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

<table>
<thead>
<tr>
<th>SCREEN INFORMATION</th>
</tr>
</thead>
</table>

Page Navigation This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen This screen does not have a zoom screen.

<table>
<thead>
<tr>
<th>CONFIGURATION OPTIONS</th>
</tr>
</thead>
</table>

This screen does not have any configuration options.
**Prom Timeouts Screen**

This screen monitors the number of lock promotions that could not be granted because they timed out.

These are typically logical areas or record locks.

Each promotion timeout reported in the “prom timeouts” field is also reported in the “proms stalled” field. This is because each timed out request is also a stalled request.

This screen resides in the “Locking (One Stat Field)” menu.

---

**Screen Example**

The following is an example of the “Locking (prom timeouts)” screen:

```
Node: MYNODE (1/1/2)    Oracle Rdb X7.0-00 Perf. Monitor 1-FEB-1998 16:24:25.67
Rate: 1.00 Second        Locking (prom timeouts)        Elapsed: 00:00:46.33
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1  Mode: Online

<table>
<thead>
<tr>
<th>statistic.........</th>
<th>rate.per.second.............</th>
<th>total.......</th>
<th>average......</th>
<th>count.......</th>
<th>per.trans....</th>
</tr>
</thead>
<tbody>
<tr>
<td>total locks</td>
<td>5</td>
<td>0</td>
<td>0.4</td>
<td>19</td>
<td>0.0</td>
</tr>
<tr>
<td>area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>FILID locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>AIJ locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>freeze lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>logical area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
```

---

**Screen Fields**

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !
total locks  This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.

area locks  This field gives statistics for database storage area locks.

buffer page locks  This field gives statistics for database page locks. Page locks manage the database page buffer pool.

record locks  This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

SEQBLK lock  This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

FILID locks  This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

TSNBLK locks  This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

RTUPB lock  This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

ACTIVE lock  This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

MEMBIT lock  This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

AIJ locks  This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.

snapshot locks  This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.
freeze lock: This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.

For more information on transaction recovery, please refer to the “Transaction Recovery Duration Estimate” screen and the “Recovery Statistics” screen.

quiet point lock: This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

logical area locks: Logical area locks are obtained when Oracle Rdb readies tables. Lock carryover can help reduce the number of logical area locks.

nowait transaction: This field gives statistics for the database nowait transaction lock.

CLIENT locks: This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

- **Graph**: This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**: This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**: This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause**: This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**: This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Time_plot**: This on-screen menu option plots a specific field's value by time.
Unreset  This on-screen menu option restores the original statistic values after using “Reset” option.

X_plot  This on-screen menu option plots a specific field’s value using scatter-based display.

Yank    Places a selected statistic on the “Custom Statistics” screen.

This section discusses screen-specific issues.

Page Navigation  This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support  This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support  This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  This screen does not have a zoom screen.

This screen does not have any configuration options.
Prom Deadlocks Screen

This screen monitors the number of stalled enqueue lock requests to promote an existing lock to a higher lock mode that ultimately resulted in a deadlock. Most deadlocks are resolved and retried by Oracle Rdb transparently to the application program, therefore these numbers do not necessarily reflect the number of deadlocks reported to the application program.

These are typically page locks.

Each promotion deadlock reported in the “prom deadlocks” field is also reported in the “proms stalled” field. This is because each deadlocked request is also a stalled request.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (prom deadlocks)” screen:

```
Node: MYNODE (1/1/2) Oracle Rdb X7.0-00 Perf. Monitor 1-FEB-1998 16:24:25.67
Rate: 1.00 Second Locking (prom deadlocks) Elapsed: 00:00:46.33
Page: 1 of 1 DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1 Mode: Online

<table>
<thead>
<tr>
<th>statistic ...........</th>
<th>rate.per.second ............</th>
<th>total.......</th>
<th>average......</th>
</tr>
</thead>
<tbody>
<tr>
<td>name .................</td>
<td>max..... cur..... avg...... count....... per.trans....</td>
<td></td>
<td></td>
</tr>
<tr>
<td>total locks ..........</td>
<td>5           0        0.4      19           0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>area locks</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>record locks</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>0           0        0.0      1            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FILID locks</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0           0        0.0      2            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AIJ locks</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>freeze lock</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>logical area locks</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0           0        0.0      0            0.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank!
This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.

This field gives statistics for database storage area locks.

This field gives statistics for database page locks. Page locks manage the database page buffer pool.

This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.

This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.

This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.
This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.

For more information on transaction recovery, please refer to the “Transaction Recovery Duration Estimate” screen and the “Recovery Statistics” screen.

This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

Logical area locks are obtained when Oracle Rdb reads tables. Lock carryover can help reduce the number of logical area locks.

This field gives statistics for the database nowait transaction lock.

This field gives statistics for the database CLIENT lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Graph**
This on-screen menu option displays a histogram graph instead of numbers.

**Numbers**
This on-screen menu option displays numeric statistics instead of histogram graph.

**Options**
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Pause**
This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

**Reset**
This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
Time_plot  This on-screen menu option plots a specific field's value by time.

Unreset  This on-screen menu option restores the original statistic values after using “Reset” option.

X_plot  This on-screen menu option plots a specific field’s value using scatter-based display.

Yank  Places a selected statistic on the “Custom Statistics” screen.

---

Screen Information

This section discusses screen-specific issues.

Page Navigation  This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support  This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support  This screen is integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  This screen does not have a zoom screen.

---

Configuration Options

This screen does not have any configuration options.
Blocking ASTs Screen

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

The number of blocking ASTs reported is actually comprised of two different types of blocking ASTs, those blocking ASTs externally generated and those blocking ASTs internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.

This screen resides in the “Locking (One Stat Field)” menu.
The following is an example of the “Locking (blocking ASTs)” screen:

Node: MYNODE (1/1/2)  Oracle Rdb X7.0-00 Perf. Monitor  1-FEB-1998 16:24:25.67
Rate: 1.00 Second  Locking (blocking ASTs)  Elapsed: 00:00:46.33
Page: 1 of 1  DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1  Mode: Online

<table>
<thead>
<tr>
<th>statistic</th>
<th>rate.per.second</th>
<th>total</th>
<th>average</th>
<th>count</th>
<th>per.trans</th>
</tr>
</thead>
<tbody>
<tr>
<td>total locks</td>
<td>5</td>
<td>0</td>
<td>0.4</td>
<td>19</td>
<td>0.0</td>
</tr>
<tr>
<td>area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>SEQBBLK lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>FILID locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>AIJ locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>freeze lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>logical area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !

**SCREEN FIELDS**

- **total locks**: This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.
- **area locks**: This field gives statistics for database storage area locks.
- **buffer page locks**: This field gives statistics for database page locks. Page locks manage the database page buffer pool.
- **record locks**: This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.
- **SEQBLK lock**: This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.
FILID locks This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).

TSNBLK locks This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.

RTUPB lock This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.

ACTIVE lock This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.

MEMBIT lock This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.

AIJ locks This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.

snapshot locks This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

freeze lock This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.

quiet point lock This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

logical area locks Logical area locks are obtained when Oracle Rdb reads tables. Lock carryover can help reduce the number of logical area locks.

nowait transaction This field gives statistics for the database nowait transaction lock.
CLIENT locks  This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

**ON-SCREEN MENU OPTIONS**

- **Graph**  This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**  This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Pause**  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.
- **Reset**  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Time_plot**  This on-screen menu option plots a specific field's value by time.
- **Unreset**  This on-screen menu option restores the original statistic values after using “Reset” option.
- **X_plot**  This on-screen menu option plots a specific field's value using scatter-based display.
- **Yank**  Places a selected statistic on the “Custom Statistics” screen.

**SCREEN INFORMATION**

**Page Navigation**  This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

**Binary File Support**  This screen is available during replay of a binary file specified by the INPUT qualifier.

**Cluster Statistic Collection Support**  This screen is integrated into the “Cluster Statistic Collection” facility.

**Zoom Screen**  This screen does not have a zoom screen.
This screen does not have any configuration options.
Stall Time x100 Screen

This screen monitors the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. These statistics give a relative measure of work lost due to lock conflicts.

This is probably the most best locking screen to examine first, as it presents an overall identification of lost work. Eliminating stalls from this screen almost always directly improves application throughput.

This screen resides in the “Locking (One Stat Field)” menu.

The following is an example of the “Locking (stall time x100)” screen:

Node: MYNODE (1/1/2)  Oracle Rdb X7.0-00 Perf. Monitor 1-FEB-1998 16:24:25.67
Rate: 1.00 Second    Locking (stall time x100)    Elapsed: 00:00:46.33
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1    Mode: Online

<table>
<thead>
<tr>
<th>statistic ...............</th>
<th>rate.per.second .............</th>
<th>total.......</th>
<th>average......</th>
</tr>
</thead>
<tbody>
<tr>
<td>name ....................</td>
<td>max.....</td>
<td>cur.....</td>
<td>avg.......</td>
</tr>
<tr>
<td>total locks</td>
<td>5</td>
<td>0</td>
<td>0.4</td>
</tr>
<tr>
<td>area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>buffer/page locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>record locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>FILID locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>AIJ locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>snapshot locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>freeze lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>quiet point lock</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>logical area locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>nowait transaction</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>CLIENT locks</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Exit Graph Help Menu Options Pause Reset Set_rate Time_plot Write X_plot Yank !
### SCREEN FIELDS

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>total locks</td>
<td>This field gives statistics for all types of database locks. Note that this count includes all locks, not just those listed below.</td>
</tr>
<tr>
<td>area locks</td>
<td>This field gives statistics for database storage area locks.</td>
</tr>
<tr>
<td>buffer page locks</td>
<td>This field gives statistics for database page locks. Page locks manage the database page buffer pool.</td>
</tr>
<tr>
<td>record locks</td>
<td>This field gives statistics for database record locks. Record locks maintain the logical consistency of the database. This set of statistics includes all record locks in the adjustable lock granularity tree.</td>
</tr>
<tr>
<td>SEQBLK lock</td>
<td>This field gives statistics for the database sequence block (SEQBLK) locks. The SEQBLK locks maintain global sequence numbers or transaction and commit sequence numbers and control COMMIT and ROLLBACK operations.</td>
</tr>
<tr>
<td>FILID locks</td>
<td>This field gives statistics for the database file identification (FILID) locks. The FILID locks maintain consistent end-of-file information for the database rootfile (.RDB), live storage areas (.RDA) and snapshot storage areas (.SNP).</td>
</tr>
<tr>
<td>TSNBLK locks</td>
<td>This field gives statistics for the database transaction block (TSNBLK) locks. The TSNBLK locks control the COMMIT and ROLLBACK operations on each cluster node. TSNBLK locks are also used to control SQL SET TRANSACTION statements for read-only transactions.</td>
</tr>
<tr>
<td>RTUPB lock</td>
<td>This field gives statistics for the database runtime user process block (RTUPB) lock. The RTUPB locks maintain a consistent list of the users who are attached to the database. They also maintain the process checkpoint information when the Fast Commit feature is enabled.</td>
</tr>
<tr>
<td>ACTIVE lock</td>
<td>This field gives statistics for the database active user bit map (ACTIVE) lock. The ACTIVE lock maintains a consistent list (in bit map form) of the users who are attached to the database.</td>
</tr>
<tr>
<td>MEMBIT lock</td>
<td>This field gives statistics for the database membership node bit map (MEMBIT) lock. The MEMBIT locks maintain a consistent list (in bit map form) of the cluster nodes on which the database is currently accessed.</td>
</tr>
<tr>
<td>AIJ locks</td>
<td>This field gives statistics for the after-image journal (AIJ) locks. AIJ locks control reading from and writing to the after-image journal. One global AIJ lock maintains current end-of-file information. In addition, there is one local AIJ lock on each cluster node that manages the global AIJ buffer on that node.</td>
</tr>
</tbody>
</table>
**Snapshot locks**
This field gives statistics for the database snapshot area cursor (SAC) locks. Snapshot locks manage the allocation of snapshot pages to users who are updating the database. Snapshot locks are only used if snapshots are enabled for a storage area.

**Freeze lock**
This field gives statistics for the database freeze lock. The freeze lock suspends database activity while a database recovery process is running.

For more information on transaction recovery, please refer to the “Transaction Recovery Duration Estimate” screen and the “Recovery Statistics” screen.

**Quiet point lock**
This field gives statistics for the database quiet-point lock. The quiet-point lock suspends starting new transactions while the AIJ backup utility is trying to finish backing up the contents of the after-image journal when you use the RMU Backup After_Journal utility or the AIJ Backup Server (ABS) process. The quiet-point lock also suspends starting new transactions during the startup of an online RMU Backup command.

**Logical area locks**
Logical area locks are obtained when Oracle Rdb readies tables. Lock carryover can help reduce the number of logical area locks.

**Nowait transaction**
This field gives statistics for the database nowait transaction lock.

**CLIENT locks**
This field monitors the database client information (CLIENT) lock. The CLIENT locks are used to provide serialized access to the database metadata stored in the system relations. The CLIENT locks are also used to serialize operations such as creating tables and indices.

Note: this field is only displayed on terminal displays containing more than 24 lines.

---

**On-screen menu options**

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

- **Graph**
  This on-screen menu option displays a histogram graph instead of numbers.

- **Numbers**
  This on-screen menu option displays numeric statistics instead of histogram graph.

- **Options**
  This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

- **Pause**
  This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

- **Reset**
  This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
Time_plot: This on-screen menu option plots a specific field's value by time.

Unreset: This on-screen menu option restores the original statistic values after using "Reset" option.

X_plot: This on-screen menu option plots a specific field's value using scatter-based display.

Yank: Places a selected statistic on the "Custom Statistics" screen.

This section discusses screen-specific issues.

Page Navigation: This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.

Binary File Support: This screen is available during replay of a binary file specified by the INPUT qualifier.

Cluster Statistic Collection Support: This screen is integrated into the "Cluster Statistic Collection" facility.

Zoom Screen: This screen does not have a zoom screen.

This screen does not have any configuration options.
Lock Statistics (By File) Screens

RMU Show Statistic Locking Statistics (By File) Screens

The “Locking Statistics (By File)” menu screens display locking information for specific storage areas. The lock types are primarily page and record locks.
File Locking Overview Screen

This screen shows a summary comparison of locking activity for all database files, including the rootfile, data and snapshot storage areas. The types of database files can be filtered by name and file type, and the statistic information displayed on this screen can be sorted using various criteria.

This screen displays comparison information about locks that are specific to storage areas and snapshot files. This information is vital in determining which storage areas have the most locking activity, and analyzing the validity of storage area partitioning.

Note that, unlike the "File IO Overview" screen, the "File Locking Overview" screen does not display information about after-image journals, RUJ journals and ACE journals.

You cannot use the information contained on the "File Lock Overview" screens on the Custom Statistics screen.

This screen resides in the "Lock Statistics (by file)" menu.
The following is an example of the “File Lock Overview” screen:

Rate: 1.00 Second        File Lock Overview (Unsorted)      Elapsed: 00:26:26.90
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1    Mode: Online

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Root</td>
<td>37307</td>
<td>12351</td>
<td>20442</td>
<td>27775</td>
</tr>
<tr>
<td>data MF_PERS_DEFAULT</td>
<td>2564</td>
<td>3919</td>
<td>4611</td>
<td>2024</td>
</tr>
<tr>
<td>data DEPARTMENTS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data EMPIDS_LOW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data EMPIDS_MID</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data EMPIDS_OVER</td>
<td>31692</td>
<td>4699</td>
<td>11399</td>
<td>23164</td>
</tr>
<tr>
<td>data EMP_INFO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data JOBS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>data MF_PERS_SEGSTR</td>
<td>107</td>
<td>3</td>
<td>17</td>
<td>89</td>
</tr>
<tr>
<td>data SALARY_HISTORY</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap MF_PERS_DEFAULT</td>
<td>890</td>
<td>1925</td>
<td>2141</td>
<td>724</td>
</tr>
<tr>
<td>snap DEPARTMENTS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap EMPIDS_LOW</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap EMPIDS_MID</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap EMPIDS_OVER</td>
<td>2051</td>
<td>1801</td>
<td>2275</td>
<td>1774</td>
</tr>
<tr>
<td>snap EMP_INFO</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap JOBS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap MF_PERS_SEGSTR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>snap SALARY_HISTORY</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

This field identifies the particular database rootfile, live or snapshot storage area. Note that the summation information for all live and snapshot storage areas is identified as “All data/snap files”.

This field gives the number of lock requests (also referred to as enqueue lock requests) for new locks. Whether the lock request succeeds or fails, it is included in this count.

This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count.

These requests always succeed.

This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Filter**
By typing “F” to select the filter on-screen menu option, you can control the display of storage areas by filtering out unwanted storage area names.

The filter option allows you to enter a search string which is used to filter the names of selected storage areas. Only those storage areas whose name contains the specified search string are displayed. The search string may contain one or both of the wildcard characters. The asterisk “*” wildcard character is mapped to zero or more characters and the percent “%” wildcard character is mapped to exactly one character. Note that the search string is not case sensitive.

The filter menu option is highlighted when a search string is actively filtering storage areas. To disable filtering, press the Return key at the search string prompt.

**Options**
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

---

**SCREEN INFORMATION**

- The “File Lock Overview” screen shows the lock acquisition and lock release counts for all storage areas, the database root file and, finally, all data and snapshot areas combined.

Following the database rootfile row are the list of database storage areas, identified by the prefix ‘data’, followed by snapshot areas, identified by the prefix ‘snap’. Storage areas that are added to the database are automatically shown in the display.

- Using the “File Lock Overview” screen makes it easier for you to identify the set of storage areas that are performing an excessive number of locking operations.

- Page Navigation: If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the Next Screen key. To display a previous page, press the left angle bracket “<” key or the Previous Screen key.

- Binary File Support: This screen is available during replay of a binary file specified by the INPUT qualifier.

- Cluster Statistic Collection Support: This screen is integrated into the “Cluster Statistic Collection” facility.

- Zoom Screen: The zoom screen displays storage area information on its zoom screen.

---

(locks requested) - (rqsts not queued) - (rqst deadlocks) - (locks released).
To simplify the use of the display when a database contains a large number of storage areas, you can configure the “File Lock Overview” screen to sort the storage area information in several different ways. Type “C” to select the Config option from the display’s on-screen menu, which displays the configuration menu:

**Configuration Options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unsorted totals display</td>
<td>This configuration option displays the by-file lock count information in unsorted order; this is the default display. With this display, the storage areas are shown in the same order as the RMU Dump utility displays them; that is, all data areas are displayed first, followed by snapshot areas.</td>
</tr>
<tr>
<td>Sort by lock requests</td>
<td>This configuration option displays the by-file information sorted by descending lock request counts. Storage areas with duplicate sort criteria are displayed alphabetically.</td>
</tr>
<tr>
<td>Sort by promote requests</td>
<td>This configuration option displays the by-file information sorted by descending lock promotion counts. Storage areas with duplicate sort criteria are displayed alphabetically.</td>
</tr>
<tr>
<td>Sort by lock &amp; promote</td>
<td>This configuration option displays the by-file information sorted by descending lock request and promotion counts. Storage areas with duplicate sort criteria are displayed alphabetically.</td>
</tr>
<tr>
<td>Sort by demotes</td>
<td>This configuration option displays the by-file information sorted by descending lock demote counts. Storage areas with duplicate sort criteria are displayed alphabetically.</td>
</tr>
<tr>
<td>Sort by unlocks</td>
<td>This configuration option displays the by-file information sorted by descending lock release counts. Storage areas with duplicate sort criteria are displayed alphabetically.</td>
</tr>
<tr>
<td>Sort by total requests</td>
<td>This configuration option displays the by-file information sorted by descending total lock operation counts. Storage areas with duplicate sort criteria are displayed alphabetically.</td>
</tr>
<tr>
<td>Sort alphabetically by type</td>
<td>This configuration option displays the by-file information sorted in ascending alphabetical order, grouped by storage area type. This option has the affect of grouping all data storage areas together, followed by all snapshot areas together.</td>
</tr>
<tr>
<td>Display all storage areas</td>
<td>This configuration option allows the “File Lock Overview” screen to display both live and snapshot storage areas. This is the default configuration choice.</td>
</tr>
<tr>
<td>Display data storage areas only</td>
<td>This configuration option allows the “File Lock Overview” screen to display live storage areas only. This option can be used in conjunction with the “sort” options to filter the number of areas displayed.</td>
</tr>
<tr>
<td>Display snap storage areas only</td>
<td>This configuration option allows the “File Lock Overview” screen to display snapshot storage areas only. This option can be used in conjunction with the “sort” options to filter the number of areas displayed.</td>
</tr>
</tbody>
</table>
Lock Statistics (by file) Screen

This screen displays information about page locks that are specific to storage areas and snapshot files. This information is vital in determining which storage areas have the most locking activity, and analyzing the validity of storage area partitioning.

You cannot use the information contained on the “Lock Statistics (by file)” screens on the “Custom Statistics” screen.

The Lock Statistics screens are recorded in the binary output file produced using the OUTPUT qualifier. Consequently, the screens are available when you replay a binary file using the INPUT qualifier.

This screen resides in the “Lock Statistics (by file)” menu.

The following is an example of the “File Locking Statistics” screen:

```
Node: MYNODE (1/1/16)  Oracle Rdb X7.0-00 Perf. Monitor  2-FEB-1998 14:40:26.95
Rate: 1.00 Second       File Locking Statistics         Elapsed: 03:33:26.93
Page: 1 of 1            DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1   Mode: Online
--------------------------------------------------------------------------------
For File: All data/snap files
statistic........... rate.per.second............. total....... average......
name................ max..... cur..... avg....... count....... per.trans....
locks requested     626        0       39.8       510220          25.3
rqsts not queued    0          0        0.0            0           0.0
rqsts stalled       40         0       2.6         33534           1.6
rqst timeouts       0          0        0.0            0           0.0
rqst deadlocks      2          0       0.0          623           0.0
locks promoted      83         0       4.3         55163           2.7
proms not queued    0          0        0.0            0           0.0
proms stalled       23         0       0.5         6786            0.3
prom timeouts       0          0        0.0            0           0.0
prom deadlocks      1          0       0.0          654           0.0
locks demoted       169        0       8.7         111524          5.5
locks released      539        0      31.8         408513          20.2
blocking ASTs       57         0       3.3          43335           2.1
stall time x100     7753       0      345.3        4424651         219.8
--------------------------------------------------------------------------------
```

Exit Graph Help Menu Options Reset Set_rate Write !
This field gives the number of lock requests, also referred to as enqueue lock requests, for new locks. Whether the lock request succeeds or fails, it is included in this count. The “rqsts not queued”, “rqsts stalled”, and “rqst deadlocks” counts provide further detail for enqueue lock requests statistics.

This field gives the number of enqueue lock requests for new locks that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting and, when a conflict is detected, resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

This field gives the number of enqueue lock requests for new locks that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.

This field shows the total number of lock requests that could not be granted because they timed out. These are typically logical areas.

Each lock timeout reported in the “rqst timeouts” field is also reported in the “rqsts stalled” field. This is because each timed out request is also a stalled request.

This field gives the number of stalled enqueue lock requests for new locks that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, the number shown in this field does not necessarily reflect the number of deadlocks reported to the application program.

Each lock deadlock reported in the “rqst deadlocks” field is also reported in the “rqsts stalled” field. This is because each deadlocked request is also a stalled request.

This field gives the number of enqueue lock requests to promote an existing lock to a higher lock mode. Whether or not the lock request succeeds, it is included in this count. The “proms not queued”, “proms stalled”, and “prom deadlocks” counts provide further detail for the locks promotion statistics.

This field gives the number of enqueue lock requests to promote an existing lock that were rejected immediately because of a lock conflict. Oracle Rdb often requests a lock without waiting. When a conflict is detected, Oracle Rdb resorts to a secondary locking protocol to avoid unnecessary deadlocks. This number is one measure of lock contention.

This field gives the number of enqueue lock requests to promote an existing lock that were stalled because of a lock conflict. Whether or not the lock request ultimately succeeds, it is included in this count. This number is one measure of lock contention.
prom timeouts This field shows the total number of lock promotions that could not be granted because they timed out. These are typically logical areas.

Each promotion timeout reported in the "prom timeouts" field is also reported in the "proms stalled" field. This is because each timed out request is also a stalled request.

prom deadlocks This field gives the number of stalled enqueue lock requests to promote an existing lock that ultimately resulted in a deadlock. Most deadlocks are tried again and resolved by Oracle Rdb without the application program ever knowing there was a deadlock. Therefore, this number does not necessarily reflect the number of deadlocks reported to the application program.

Each promotion deadlock reported in the "prom deadlocks" field is also reported in the "proms stalled" field. This is because each deadlocked request is also a stalled request.

locks demoted This field gives the number of enqueue lock requests to demote an existing lock to a lower lock mode. These requests always succeed.

locks released This field gives the number of deallocating lock requests to release an existing lock. These requests always succeed. The number of outstanding locks can be determined by the formula:

\[
\text{locks requested} - \text{rqsts not queued} - \text{rqst deadlocks} - \text{locks released}.
\]

blocking AST's This field gives the number of blocking AST's, sometimes referred to as "blasts", delivered to Oracle Rdb by the OpenVMS lock manager. A blocking AST is delivered to the holder of a lock when a lock conflict is detected, which is a good indication of contention problems. When Oracle Rdb receives a blocking AST, it often demotes or releases a lock in an attempt to avoid unnecessary deadlocks.

The number of blocking AST's reported is actually comprised of two different types of blocking AST's, those blocking AST's externally generated and those blocking AST's internally generated.

An externally generated blocking AST occurs when a blocking AST is actually received by the process from the operating system in response to some lock conflict with another process. A blocking AST routine is executed and the RMU Show Statistic utility records the blocking AST activity.

An internally generated blocking AST occurs when a lock blocking AST routine is executed by the process in anticipation that the same work would have to be performed anyway if a blocking AST were to be received from the operating system, even when no blocking AST from the operating system actually occurred. This algorithm serves as an optimistic code optimization; it is assumed that the process would eventually receive a blocking AST for the particular lock, so it optimistically executes the blocking AST routine. The RMU Show Statistic utility does not differentiate between these two types of blocking ASTs.
This field gives the total time (in hundredths of a second) spent by all users waiting for a lock. Since more than one user can be waiting for a lock at the same time, this total can be greater than the actual elapsed clock time. This statistic gives a relative measure of work lost due to lock conflicts.

**ON-SCREEN MENU OPTIONS**

- **Graph**: This on-screen menu option displays a histogram graph instead of numbers.
- **Numbers**: This on-screen menu option displays numeric statistics instead of histogram graph.
- **Options**: This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- **Reset**: This on-screen menu option resets temporarily resets the statistic information. Use the “Unreset” on-screen menu option to restore the original statistic information.
- **Unreset**: This on-screen menu option restores the original statistic values after using “Reset” option.

**SCREEN INFORMATION**

- **Page Navigation**: This screen does not contain multiple pages of information and, therefore, does not contain any special navigation keystrokes.
- **Binary File Support**: This screen is available during replay of a binary file specified by the INPUT qualifier.
- **Cluster Statistic Collection Support**: This screen is integrated into the “Cluster Statistic Collection” facility.
- **Zoom Screen**: This screen does not have a zoom screen.

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
Database Parameter Information Screens

**RMU Show Statistic Database Parameter Information Screens**

The “Database Parameter Information” menu screens display information about specific categories of database attributes and parameters.
General Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “General Information” screen reflects information stored in the KROOT portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “General Information” screen:

Rate: 1.00 Second General Information Elapsed: 00:00:13.18
Page: 1 of 1 DISK$: [DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB; 1 Mode: Online
--------------------------------------------------------------------------------
Database created at 1-DEC-1992 12:46:08.86
Maximum user count is 75
Maximum node count is 16
Database open mode is Automatic
Database close mode is Automatic
Snapshot mode is Automatic
Statistics collection is enabled
Active storage area count is 9
Reserved storage area count is 20
Default recovery-unit journal filename is “Not Specified”
Date of last backup is 2-FEB-1998 11:00:40.39
Fast incremental backup is disabled
--------------------------------------------------------------------------------
Exit Help Menu Options Refresh Set_rate Write !
### Screen Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database created at date/time</td>
<td>This field indicates the date and time when the database was originally created. This field is preserved when the database is restored to a different location.</td>
</tr>
<tr>
<td>Maximum user count is number</td>
<td>This field indicates the maximum number of users that can attach to the database. Note that it may not be possible for the maximum number of users to attach to a single node, depending on the number of nodes specified.</td>
</tr>
<tr>
<td>Maximum node count is number</td>
<td>This field indicates the maximum number of cluster nodes on which the database may be opened simultaneously.</td>
</tr>
<tr>
<td>Database open mode is keyword</td>
<td>This field indicates the database open mode. The keyword keywords are the following:</td>
</tr>
<tr>
<td><strong>Keyword</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>AUTOMATIC</td>
<td>the database is automatically opened by the first user to attach to the database on the node.</td>
</tr>
<tr>
<td>MANUAL</td>
<td>the DBA must explicitly open the database using the RMU Open utility.</td>
</tr>
<tr>
<td>Database close mode is keyword</td>
<td>This field indicates the database close mode. The keyword keywords are the following:</td>
</tr>
<tr>
<td><strong>Keyword</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>AUTOMATIC</td>
<td>the database is automatically closed by the last user to detach from the database on the node.</td>
</tr>
<tr>
<td>AUTOMATIC (STORED AS MANUAL)</td>
<td>the database is automatically closed by the last user to detach from the database on the node, even though the database is designated as being closed manually. This case occurs when the database open mode is AUTOMATIC but the database close mode is MANUAL.</td>
</tr>
<tr>
<td>MANUAL</td>
<td>the DBA must explicitly close the database using the RMU Close utility.</td>
</tr>
<tr>
<td>MANUAL (STORED AS AUTOMATIC)</td>
<td>the DBA must explicitly close the database using the RMU Close utility, even though the database is designated as being closed automatically. This case occurs when the database open mode is MANUAL but the database close mode is AUTOMATIC.</td>
</tr>
<tr>
<td>TIMED AUTOMATIC</td>
<td>the database is automatically closed by the database monitor after a user-specified interval, if no user subsequently attaches to the</td>
</tr>
</tbody>
</table>
Snapshot mode is keyword

This field indicates the snapshot mode. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMMEDIATE</td>
<td>the record before-images are always written to the snapshot file.</td>
</tr>
<tr>
<td>DEFERRED</td>
<td>the record images are written to the snapshot file only if one or more read-only transactions are active</td>
</tr>
</tbody>
</table>

Statistics collection is keyword

This field indicates whether or not statistics collection is enabled on the database. Since you are in the RMU Show Statistic utility, hopefully statistic collection is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>statistic collection is enabled.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>statistic collection is disabled and you should not be here.</td>
</tr>
</tbody>
</table>

Active storage area count is number

This field indicates the total number of active storage areas in the database. This field does not include snapshot storage areas.

Reserved storage area count is number

This field indicates the total number of reserved storage area slots in the database. This is the number of storage areas that can be added while the database is open.

Fault recovery-unit journal filename is “filespec”

This field indicates the default name of the .RUJ journal for each user. The filespec keyword is either the default RUJ file name or “not specified”.

Date of last backup is date/time

This field indicates the date and time of the last database backup operation using the RMU Backup utility. Hopefully, you have performed a database backup operation.
This field indicates whether or not the “Fast Incremental Backup” feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The “Fast Incremental Backup” feature is enabled. Watch the “SPAM Access” screen to monitor the affect of this feature on the database performance.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The “Fast Incremental Backup” feature is disabled.</td>
</tr>
</tbody>
</table>

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Options**
- This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
- This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.

This section discusses screen-specific issues.

**Page Navigation**
- If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ‹ Next Screen key. To display a previous page, press the left angle bracket “<” key or the ↑ Previous Screen key.

**Binary File Support**
- This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

**Cluster Statistic Collection Support**
- This screen is not integrated into the “Cluster Statistic Collection” facility.

**Zoom Screen**
- This screen does not have a zoom screen.
This screen does not have any configuration options.
Buffer Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “Buffer Information” screen reflects information stored in the KROOT portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Buffer Information” screen:
Node: MYNODE (1/1/16)   Oracle Rdb X7.0-00 Perf. Monitor  3-FEB-1998 05:39:34.73
Rate: 1.00 Second              Buffer Information           Elapsed: 00:00:16.33
Page: 1 of 1    DISK$: [DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

Default user buffer count is 20
Default recovery buffer count is 20
Buffer size is 6 blocks
Maximum pages per buffer is 3
Global Buffers are disabled
  - Global buffer count is 250
  - Maximum global buffer count per user is 5
  - Page transfer via memory is disabled
Global section size with global buffers disabled is 1131069 bytes
  - With global buffers enabled is 2061787 bytes
Asynchronous pre-fetch is enabled
  - Maximum pre-fetch depth is 5 buffers
Detected asynchronous pre-fetch is enabled
  - Maximum pre-fetch depth is 4 buffers
  - Pre-fetch threshold is 4 pages
Asynchronous batch-write is enabled
  - Clean buffer count is 5
  - Maximum batch size is 4 buffers

Screen Fields

Default user buffer count is number

This field indicates the default user buffer count.

Default recovery buffer count is number (stored in root as number)

This field indicates the database recovery process (DBR) buffer count; note that the number of buffers used is larger than the value stored in the database rootfile. The DBR buffer count is always the larger of the default user buffer count or the DBR recovery buffer count.

Default recovery buffer count is number

This field indicates the Database Recovery process (DBR) buffer count.

Buffer size is number blocks

This field indicates that the user buffer size contains the specified number of blocks.
This field indicates the maximum number of database pages that can reside in one buffer.

**Global Buffers are**

This field indicates whether the database uses local buffers or global buffers. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>the database uses global buffers.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>the database uses local buffers.</td>
</tr>
</tbody>
</table>

**Global buffer count is**

If global buffers are enabled, this field indicates the total number of available buffers. This message is displayed even if global buffers are disabled so that you can review the value.

**Maximum global buffer count per user is**

If global buffers are enabled, this field indicates the number of buffers exclusively available to each user. This message is displayed even if global buffers are disabled so that you can review the value.

**Page transfer via memory is**

This field indicates whether or not the “Page Transfer Via Memory” feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The “Page Transfer Via Memory” feature is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The “Page Transfer Via Memory” feature is not used.</td>
</tr>
</tbody>
</table>

**Bal section size with global buffers disabled is**

number bytes
This field indicates the size of the database global section when global buffers are disabled.

Global buffers enabled is number bytes

This field indicates the size of the database global section when global buffers are enabled.

Asynchronous pre-fetch is keyword

This field indicates whether or not the “Asynchronous Pre-Fetch” (APF) feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The “Asynchronous Pre-Fetch” feature is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The “Asynchronous Pre-Fetch” feature is not used</td>
</tr>
</tbody>
</table>

Maximum pre-fetch depth is number buffers

This field indicates the maximum number of buffers that are eligible to be pre-fetched.

Detected asynchronous pre-fetch is keyword

This field indicates whether or not the “Detected Asynchronous Pre-Fetch” (DAPF) feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The “Detected Asynchronous Pre-Fetch” feature is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The “Detected Asynchronous Pre-Fetch” feature is not used</td>
</tr>
</tbody>
</table>

Maximum pre-fetch depth is number buffers

This field indicates the maximum number of buffers that are eligible to be pre-fetched.
This field indicates the number of pages that must be fetched prior to starting the detected asynchronous pre-fetch operation.

This field indicates whether or not the "Asynchronous Batch Write" (ABW) feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The &quot;Asynchronous batch Write&quot; feature is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The &quot;Asynchronous Batch Write&quot; feature is not used.</td>
</tr>
</tbody>
</table>

This field indicates the minimum number of buffers that must remain unmodified.

This field indicates the maximum number of buffers that can be flushed to disk as part of an asynchronous batch write operation.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Options**

This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Refresh**

This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.

This section discusses screen-specific issues.
Page Navigation  
If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the \( \downarrow \) Next Screen key. To display a previous page, press the left angle bracket “<” key or the \( \uparrow \) Previous Screen key.

Binary File Support  
This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

Cluster Statistic Collection Support  
This screen is not integrated into the “Cluster Statistic Collection” facility.

Zoom Screen  
This screen does not have a zoom screen.

This screen does not have any configuration options.
Lock Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “Lock Information” screen reflects information stored in the KROOT portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Lock Information” screen:

Node: MYNODE (1/1/16)   Oracle Rdb X7.0-00 Perf. Monitor 3-FEB-1998 05:39:35.78
Rate: 1.00 Second               Lock Information            Elapsed: 00:00:17.38
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

Adjustable record locking granularity is enabled
- Fanout factor 1 is 10 (10 pages)
- Fanout factor 2 is 10 (100 pages)
- Fanout factor 3 is 10 (1000 pages)
Carryover lock optimization is enabled
Lock Tree Partitioning is disabled
Lock timeout is disabled

Exit Help Menu Options Refresh Set_rate Write !
This field indicates whether or not the “Adjustable Lock Granularity” (ALG) feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The “Adjustable Lock Granularity” feature is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The “Adjustable Lock Granularity” feature is not used.</td>
</tr>
</tbody>
</table>

This field indicates the ALG fanout factor at each level.

This field indicates whether or not the “Carryover Lock Optimization” feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The “Carryover Lock Optimization” feature is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The “Carryover Lock Optimization” feature is not used.</td>
</tr>
</tbody>
</table>

This field indicates whether or not the “Partitioned Lock Tree” (PLT) feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The “Partitioned Lock Tree” feature is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The “Partitioned Lock Tree” feature is not used.</td>
</tr>
</tbody>
</table>

Lock timeout is disabled

This field indicates that the “Lock Timeout” feature is disabled.
timeout interval is number seconds

This field indicates that the “Lock Timeout” feature is enabled with the specified number of seconds.

---

**ON-SCREEN MENU OPTIONS**

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refresh</td>
<td>This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.</td>
</tr>
</tbody>
</table>

---

**SCREEN INFORMATION**

<table>
<thead>
<tr>
<th>Page Navigation</th>
<th>If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “&gt;” key or the Next Screen key. To display a previous page, press the left angle bracket “&lt;” key or the Previous Screen key.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary File Support</td>
<td>This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.</td>
</tr>
<tr>
<td>Cluster Statistic Collection Support</td>
<td>This screen is not integrated into the “Cluster Statistic Collection” facility.</td>
</tr>
<tr>
<td>Zoom Screen</td>
<td>This screen does not have a zoom screen.</td>
</tr>
</tbody>
</table>

---

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
Storage Area Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “Storage Area Information” screen reflects information stored in the FILID portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Storage Area Information” screen:

Node: MYNODE (1/1/16) Oracle Rdb X7.0-00 Perf. Monitor 3-FEB-1998 05:39:37.88
Rate: 1.00 Second Storage Area Information Elapsed: 00:00:19.48
Page: 1 of 18 DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1 Mode: Online

-----------------------------------------------------------------------------

Storage area "RDB$SYSTEM"
Area ID number is 1
Filename is "DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERS_DEFAULT.RDA;1"
Access mode is read/write
Page format is uniform
Page size is 2 blocks
- Current physical page count is 1505
Row level locking is enabled
Row caching is enabled
No row cache is defined for this area
Extends are enabled
- Extend area by 20%, minimum of 99 pages, maximum of 9999 pages
- Area has been extended 4 times
Volume set spreading is enabled
Snapshot area ID number is 30
SPAMs are enabled
- Interval is 1089 data pages
- Current SPAM page count is 2
Area last backed up at 2-FEB-1998 11:00:40.39
Area has never been incrementally restored

-----------------------------------------------------------------------------

Exit Help Menu >next_page <prev_page Options Refresh Set_rate Write!
SCREEN FIELDS

Storage area “name” This field identifies the storage area name. Note that this is not the storage area filename.

Area ID number is number This field identifies the storage area number.

Filename is “filename” This field identifies the storage area file name.

Access mode is keyword This field identifies the storage area access mode. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>READ/WRITE</td>
<td>The storage area can be modified.</td>
</tr>
<tr>
<td>READ-ONLY</td>
<td>The storage area cannot be modified.</td>
</tr>
<tr>
<td>WORM</td>
<td>The storage area is a “Write Once Read Many” device.</td>
</tr>
</tbody>
</table>

Page format is keyword This field identifies the storage type page format. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIXED</td>
<td>The storage area is mixed-format.</td>
</tr>
<tr>
<td>UNIFORM</td>
<td>The storage area is uniform-format.</td>
</tr>
</tbody>
</table>

Page size is number blocks This field indicates the storage area page size, in blocks.

rent initialized page count is number

If the storage area resides on a WORM device, this field identifies the number of initialized pages.

rent physical page count is pages
If the storage area resides on a WORM device, this field identifies the end-of-file (last written) page. Otherwise, this field identifies the size of the storage area.

This field indicates whether row-level locking or page-level locking is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROW</td>
<td>Row-level locking is enabled.</td>
</tr>
<tr>
<td>PAGE</td>
<td>Page-level locking is enabled.</td>
</tr>
</tbody>
</table>

This field indicates whether or not row caching is enabled for this storage area. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>Row caching is enabled for this storage area.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>Row caching is disabled for this storage area.</td>
</tr>
</tbody>
</table>

This field indicates that there are no row caches defined for this storage area.

This field identifies the row cache associated with this storage area.

This field indicates whether or not the storage area can be extended. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The storage area is allowed to be extended.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The storage area cannot be extended.</td>
</tr>
</tbody>
</table>

This field indicates the number of number of pages to extend the storage area.

This field identifies the storage area extension parameters.
Area has never been extended

This field indicates that the storage area has never been extended.

A has been extended number times

This field indicates the total number of times the storage area has been extended.

Volume set spreading is keyword

This field indicates whether or not the storage area can be spread over virtual volumes. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The storage area is allowed to be spread across virtual volumes.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The storage area cannot be spread across virtual volumes.</td>
</tr>
</tbody>
</table>

Snapshot area ID number is number

This field identifies the snapshot storage area that corresponds to the live storage area.

Snapshots were last suppressed by TSN number: number

This field identifies the TSN that last suppressed snapshots.

SPAMs are keyword

This field indicates whether or not SPAMs are enabled for the storage area. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The storage area contains SPAM pages.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The storage area does not contain SPAM pages.</td>
</tr>
</tbody>
</table>

Interval is number data pages

This field identifies the SPAM interval, which is the number of data pages controlled by each SPAM page.

Thresholds are percent, percent, and percent.
This field identifies the SPAM thresholds for the storage area.

This field indicates the total number of SPAM pages that reside in the storage area.

This field indicates that the WORM device is full.

This field indicates that AIJ information is not written for modifications to the storage area.

This field indicates that the WORM device is corrupt and cannot be accessed.

This field indicates that the storage area is corrupt and cannot be accessed. The storage area must be restored from a database backup file and recovered from after-image journals before it can be subsequently used.

This field indicates that the storage area has been restored from a database backup file and needs to be recovered from after-image journals.

This field indicates the current recover sequence number for the storage area.

This field indicates that the storage area has never been backed up.

This field indicates the date and time the storage area was last backed up.

This field indicates that the storage area has never been restored as part of an incremental database restore operation.
This field identifies the date and time when the storage area was last restored from an incremental restore operation.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Options
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

Refresh
This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.

This section discusses screen-specific issues.

Page Navigation
If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ‡ Next Screen key. To display a previous page, press the left angle bracket “<” key or the † Previous Screen key.

Binary File Support
This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

Cluster Statistic Collection Support
This screen is not integrated into the “Cluster Statistic Collection” facility.

Zoom Screen
This screen does not have a zoom screen.

This screen does not have any configuration options.
Row Cache Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “Row Cache Information” screen reflects information stored in the RCACHE portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Row Cache Information” screen:

Node: MYNODE (1/1/1) Oracle Rdb X7.0-00 Perf. Monitor 4-FEB-1998 06:10:12.85
Rate: 1.00 Second Row Cache Information Elapsed: 00:00:17.02
Page: 4 of 24 DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1 Mode: Online

Row cache “EMPIDS_OVER”
Cache ID number is 4
Cache filespec is “$_S111$DUA347:[DB_HOMEDIR.WORK.STATS]”
Row slot allocation is 3000 for maximum record length 300
Maximum working set count is 10
Row replacement is enabled
Cache will be mapped in process space
Dynamic mapping is disabled
Dynamic mapping window count is 100
No checkpoint for sequence 0

Exit Help Menu >next_page <prev_page Options Refresh Set_rate Write !
Row cache “name”  This field identifies the row cache name.

Cache ID number is number  This field identifies the row cache identifier.

Cache filespec is “filespec”  This field identifies the filename template used to create the row cache database backing store files (.RDC).

Rows slot allocation is number for maximum record length number  This field identifies the number of rows in the row cache and specifies the maximum row size that can fit into a row cache row.

Maximum working set count is number  This field identifies the row cache maximum working set count. Do not confuse this field with the OpenVMS “working set” parameter.

Row replacement is keyword  This field indicates whether or not row replacement in the cache is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>Row replacement is allowed.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>Row replacement is not allowed.</td>
</tr>
</tbody>
</table>

Cache will be mapped in keyword  This field indicates where the row cache “maps” will be created. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSTEM SPACE</td>
<td>The row cache resides in OpenVMS “system space” (S1 space).</td>
</tr>
<tr>
<td>(AXP ONLY)</td>
<td></td>
</tr>
<tr>
<td>PROCESS SPACE</td>
<td>The row cache resides in traditional OpenVMS global section (S0 space).</td>
</tr>
</tbody>
</table>

Dynamic mapping is keyword  This field indicates whether or not row cache dynamic mapping is enabled. The keyword keywords are the following:
### Keyword | Description
--- | ---
ENABLED | Row cache dynamic mapping is allowed.
DISABLED | Row cache dynamic mapping is not allowed.

| Dynamic mapping window count is number |
| This field identifies the row cache window count used when dynamic mapping is enabled. |

| Checkpoint for sequence number |
| This field indicates that the row cache has not yet checkpointed. |

| Checkpoint is number: number for sequence number |
| This field identifies the latest checkpoint for the row cache. The first number identifies the after-image journal sequence number and the second number identifies the block number within that after-image journal. The third number sequence indicates the current checkpoint iteration. |

### ON SCREEN MENU OPTIONS

- **Options**
  - This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

- **Refresh**
  - This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.

### SCREEN INFORMATION

- **Page Navigation**
  - If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key.
To display a previous page, press the left angle bracket “<” key or the Previous Screen key.

<table>
<thead>
<tr>
<th>Configuration Options</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Binary File Support</strong></td>
</tr>
<tr>
<td>This screen is <strong>not</strong> recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.</td>
</tr>
<tr>
<td><strong>Cluster Statistic Collection Support</strong></td>
</tr>
<tr>
<td>This screen is <strong>not</strong> integrated into the “Cluster Statistic Collection” facility.</td>
</tr>
<tr>
<td><strong>Zoom Screen</strong></td>
</tr>
<tr>
<td>This screen does <strong>not</strong> have a zoom screen.</td>
</tr>
</tbody>
</table>

This screen does **not** have any configuration options.
Journaling Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “Journaling Information” screen reflects information stored in the AIJDB portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Journaling Information” screen:

Node: MYNODE (1/16)   Oracle Rdb X7.0-00 Perf. Monitor  3-FEB-1998 05:39:41.01  
Rate: 1.00 Second       Journaling Information         Elapsed: 00:00:22.61  
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online  

after-image journaling is enabled  
Database is configured for 11 journals  
Reserved journal count is 11  
Available journal count is 5  
Journal “RICK4” is current  
Shutdown time is 60 minutes  
Backup server (spooler) is enabled  
- Backup server uses no-quiet-point  
Default backup filename edits are disabled  
Log server is Manual  
AIJ cache on “electronic disk” is disabled  
Journal overwrite is disabled  
Default journal allocation is 1024 blocks  
Default journal extension is 1024 blocks  
Current roll-forward sequence number is 39  
Current backup sequence number is 39  
Default backup filename is “KODA_TEST:[DB_HOMEDIR.TMP]BACKUP.AIJ;”

Exit Help Menu Options Refresh Set_rate Write !
This field indicates whether or not the AIJ Journaling facility is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The AIJ Journaling facility is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The database does not use the AIJ Journaling facility.</td>
</tr>
</tbody>
</table>

This field identifies the total number of after-image journals that can be created for the database.

This field identifies the total number of reserved after-image journal slots that remain in the database. This field represents the number of after-image journals that can be created while the database is open.

This field identifies the total number of created after-image journals for the database.

This field identifies the name of the “current” after-image journal.

This field identifies the AIJ Journaling facility shutdown time. The shutdown interval is primarily used during a suspended AIJ switch-over operation.

This field indicates whether or not the AIJ Backup Server (ABS) is enabled. The keyword keywords are the following:
Backup server uses keyword

This field indicates whether or not the AIJ Backup Server (ABS) uses a quiet-point during its backup operation. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUIET-POINT</td>
<td>The ABS acquires the quiet-point lock</td>
</tr>
<tr>
<td>NO-QUIET-POINT</td>
<td>The ABS does not acquire the quiet-point lock.</td>
</tr>
</tbody>
</table>

Backup filename edits are keyword

This field indicates whether or not the AIJ backup filename edits are enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The AIJ backup filename edits are enabled; use the RMU Dump Header utility to display the actual filename edits.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The AIJ backup filename edits are not used.</td>
</tr>
</tbody>
</table>

Log server is keyword

This field indicates whether or not the AIJ Log Server (ALS) is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The AIJ Log Server is enabled.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The AIJ Log Server is not used.</td>
</tr>
</tbody>
</table>
This field indicates whether or not the “AIJ Cache on Electronic disk” (ACE) feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The “AIJ Cache on Electronic disk” feature is enabled.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The “AIJ Cache on Electronic disk” feature is disabled.</td>
</tr>
</tbody>
</table>

Filename is “name”

This field identifies the “AIJ Cache on Electronic disk” (ACE) feature filename.

Journal overwrite is keyword

This field indicates whether or not the after-image journals may be over-written if they have not yet been backed up. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The after-image journals may be overwritten</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The after-image journals must be backed up.</td>
</tr>
</tbody>
</table>

After journal allocation is number blocks

This field identifies the default after-image journal allocation size, in blocks.

After journal extension is number blocks

This field identifies the default after-image journal extension size, in blocks.

Current roll-forward sequence number is number

This field identifies the roll-forward sequence number for the next after-image journal to be recovered.

Current backup sequence number is number

This field identifies the backup sequence number for the “current” after-image journal. This is the sequence number recorded in the “open” header record of the “current” after-image journal.
This field indicates that a new version of the after-image journal should be created and to use the previous after-image journal as the backup file.

ault backup filename is “same as journal”

This field identifies the default after-image journal backup filename.

ault backup filename is “filespec”

This field indicates that there is no default after-image journal backup filename.

ault backup filename is “not specified”

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**ON-SCREEN MENU OPTIONS**

<table>
<thead>
<tr>
<th>Options</th>
<th>This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refresh</td>
<td>This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.</td>
</tr>
</tbody>
</table>

This section discusses screen-specific issues.

**SCREEN INFORMATION**

<table>
<thead>
<tr>
<th>Page Navigation</th>
<th>If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “&gt;” key or the Next Screen key. To display a previous page, press the left angle bracket “&lt;” key or the Previous Screen key.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary File Support</td>
<td>This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.</td>
</tr>
<tr>
<td>Cluster Statistic Collection Support</td>
<td>This screen is not integrated into the “Cluster Statistic Collection” facility.</td>
</tr>
<tr>
<td>Zoom Screen</td>
<td>This screen does not have a zoom screen.</td>
</tr>
</tbody>
</table>
This screen does not have any configuration options.
Journal Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “Journal Information” screen reflects information stored in the AIJFB portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Journal Information” screen:

Rate: 1.00 Second             Journal Information           Elapsed: 00:00:24.72
Page: 1 of 5    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online
--------------------------------------------------------------------------------
AIJ Journal “RICK1”
Filename is “KODA_TEST:[DB_HOMEDIR.TMP]RICK1.AIJ;1”
Default AIJ filename is “KODA_TEST:[DB_HOMEDIR.TMP]RICK1.AIJ”
Backup sequence number is -1
Allocation is specified as 1024 blocks
Extension is defaulted to 1024 blocks
Backup filename is defaulted to “KODA_TEST:[DB_HOMEDIR.TMP]BACKUP.AIJ;”
Backup filename edits are disabled
--------------------------------------------------------------------------------
Exit Help Menu >next_page <prev_page Options Refresh Set_rate Write !
AIJ Journal "name" This field identifies the after-image journal name.

Filename is "filespec" This field identifies the after-image journal filename actually created.

Default AIJ filename is "filespec" This field identifies the after-image journal filename template used to create the actual after-image journal.

kup sequence number is number This field identifies the AIJ sequence number for the after-image journal; this is the sequence number written in the “open” header record of the after-image journal.

Journal was activated date/time This field identifies the date and time when the after-image journal was selected to be modified.

This field indicates that the after-image journal was created during emergency AIJ switch-over situation

This field indicates that the after-image journal was created by the “Emergency AIJ Journal” facility to alleviate a suspended AIJ switch-over operation. This is a normal after-image journal in all respects.

Allocation is number blocks This field identifies the allocation size of the after-image journal.

Extension is number blocks This field identifies the extension size of the after-image journal.

kup filename is “same as journal” This field indicates that a new version of the after-image journal should be created and to use the previous after-image journal as the backup file.

Backup filename is "filespec" This field identifies the after-image journal backup filename.

kup filename is defaulted to “same as journal” This field indicates that the default attribute is that a new version of the after-image journal should be created and to use the previous after-image journal as the backup file.

kup filename is defaulted to “filespec”
This field identifies that the default after-image journal backup filename.

Keyword: backup filename is "not specified"

This field indicates that the after-image journal does not specify an AIJ backup filename.

Keyword: backup filename edits are enabled

This field indicates whether or not the AIJ backup filename edits are enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The AIJ backup filename edits are enabled; use the RMU Dump Header utility to display the actual filename edits.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The AIJ backup filename edits are not used.</td>
</tr>
</tbody>
</table>

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Options**

This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Refresh**

This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.

This section discusses screen-specific issues.

**Page Navigation**

If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key. To display a previous page, press the left angle bracket “<” key or the ↑ Previous Screen key.
This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

This screen is not integrated into the “Cluster Statistic Collection” facility.

This screen does not have a zoom screen.

This screen does not have any configuration options.
Fast Commit Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “Fast Commit Information” screen reflects information stored in the KROOT and AIJD B portions of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Fast Commit Information” screen:

---

Node: MYNODE (1/1/16)  Oracle Rdb X7.0-00 Perf. Monitor  3-FEB-1998 05:39:45.22
Rate: 1.00 Second       Fast Commit Information       Elapsed: 00:00:26.82
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

AIJ Fast Commit is enabled
- Checkpointing AIJ interval is 1250 blocks
- Checkpointing time interval is 600 seconds
Commit to AIJ optimization is disabled
- Transaction interval is 256

---

Exit Help Menu Options Refresh Set_rate Write !

---

AIJ Fast Commit is keyword
This field indicates whether or not the “Fast Commit” feature is enabled. The keyword keywords are the following:
**Keyword** | **Description**
--- | ---
ENABLED | The “Fast Commit” feature is enabled.
DISABLED | The “Fast Commit” feature is not used.

This field indicates that there is no AIJ-growth checkpoint threshold.

**Checkpointing AIJ interval is specified**

This field identifies the AIJ-growth checkpoint threshold.

**Checkpointing time interval is specified**

This field indicates that there is no elapsed-time checkpoint threshold.

**Checkpointing time interval is specified**

This field identifies the elapsed-time checkpoint threshold.

**Commit to AIJ optimization is set to “enabled” but is disabled**

This field indicates that the “Commit To Journal” (CTJ) feature is enabled but cannot be used because either after-image journaling is disabled or the “Fast Commit” feature is disabled.

**Commit to AIJ optimization is keyword**

This field indicates whether or not the “Commit To Journal” (CTJ) feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The “Commit To Journal optimization” feature is enabled.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The “Commit To Journal optimization” feature is not used.</td>
</tr>
</tbody>
</table>

**Transaction interval is number**

This field identifies the “Commit To Journal” (CTJ) feature transaction threshold.
This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Options**

This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Refresh**

This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.

---

This section discusses screen-specific issues.

**Page Navigation**

If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key. To display a previous page, press the left angle bracket “<” key or the ↑ Previous Screen key.

**Binary File Support**

This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

**Cluster Statistic Collection Support**

This screen is not integrated into the “Cluster Statistic Collection” facility.

**Zoom Screen**

This screen does not have a zoom screen.

**Configuration Options**

This screen does not have any configuration options.
Hot Standby Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “General Information” screen reflects information stored in the AIJDB portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Hot Standby Information” screen:

Node: MYNODE (1/1/16) Oracle Rdb X7.0-00 Perf. Monitor 3-FEB-1998 13:14:38.18
Rate: 1.00 Second Hot Standby Information Elapsed: 00:13:43.69
Page: 1 of 1 KODA_TEST:[DB_HOMEDIR.REGTESTS]MF_PERSONNEL.RDB;1 Mode: Online
--------------------------------------------------------------------------------
after-image journaling is enabled
Standby state: Active
Database: _DPA48:[DB_HOMEDIR.REGTESTS]T_PERSONNEL.RDB;1
Server name: 000038344150445F060000005B02AD
Checkpoint interval: 1000
Connection timeout interval: 1
Sequenced Message timeout interval: 0
Buffer count: 0
Data synchronization mode: Cold
Auto synchronization mode: Cold
Number of clients: 1
Quiet-point: disabled
--------------------------------------------------------------------------------
Exit Help Menu Options Refresh Set_rate Write !

SCREEN LOCATION

SCREEN EXAMPLE

SCREEN FIELDS
This field indicates whether or not the AIJ Journaling facility is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The AIJ Journaling facility is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The database does not use the AIJ Journaling facility.</td>
</tr>
</tbody>
</table>

This field identifies the hot standby replication state. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inactive</td>
<td>This keyword indicates that the Hot Standby feature is not being used.</td>
</tr>
<tr>
<td>DB Bind</td>
<td>This keyword indicates that the database is being set up for replication.</td>
</tr>
<tr>
<td>Net Bind</td>
<td>This keyword indicates that the network connection is being established.</td>
</tr>
<tr>
<td>Restart</td>
<td>This keyword indicates that the standby database is performing replication “restart” processing.</td>
</tr>
<tr>
<td>Connecting</td>
<td>This keyword indicates that master and standby databases are connecting together to establish the proper replication modes.</td>
</tr>
<tr>
<td>DB Synch.</td>
<td>This keyword indicates that the master and standby databases are being synchronized.</td>
</tr>
<tr>
<td>Activating</td>
<td>This keyword indicates that the AIJ Log Servers (ALS) on the master database are being activated. That is, replication is commencing.</td>
</tr>
<tr>
<td>SyncCmpltn</td>
<td>This keyword indicates that the master and standby databases have been synchronized and that the two databases are being verified for transactional consistency.</td>
</tr>
<tr>
<td>DECnet</td>
<td>This keyword indicates that the Hot Standby feature is active and using the DECnet communications protocol.</td>
</tr>
<tr>
<td>TCP/ IP</td>
<td>This keyword indicates that the Hot Standby feature is active and using the TCP/ IP communications protocol.</td>
</tr>
<tr>
<td>Active</td>
<td>This keyword indicates that the Hot Standby feature is active and being replicated locally. Local replication means that the master and standby</td>
</tr>
</tbody>
</table>
### Keyword | Description
--- | ---
databases are on the same node of the Cluster. |  
Suspended | This keyword indicates that the Hot Standby feature is temporarily suspended.  
Resumption | This keyword indicates that a temporarily suspended master database is in the process of being resumed. Transactional activity that occurred while the Hot Standby feature was suspended is being transferred to the standby database.  
Completion | This keyword indicates that the Hot Standby feature is being shut down, and that final completion processing is being performed.  
Shutdown | This keyword indicates that the Hot Standby feature is being shut down.  
Net Unbind | This keyword indicates that the network connection between the master and standby databases is being disconnected.  
DB Unbind | This keyword indicates that the database is being removed from Hot Standby replication mode.  
Recovery | This keyword indicates that the database is being recovered following a premature failure of the Hot Standby feature, usually as a result of network failure or server failure.

**Database:** filespec | This field identifies the standby database filename if the current database is the master. Conversely, this field displays the master database filename if the current database is the standby.  
**Server name:** name | This field identifies the server name, which is essentially a serial number of the database.  
**Checkpoint interval:** number | This field identifies the “Hot Standby” feature checkpoint interval. Do not confuse this field with the “Fast Commit” feature checkpoint.  
**Connection timeout interval:** number | This field identifies the initial connection timeout interval, in seconds. This field is important during replication startup.  
**Encoded Message timeout interval:** number |  

---

229
This field identifies the network timeout interval, in seconds.

Buffer count: number

This field identifies the standby database buffer count.

a synchronization mode: keyword

This field identifies the user-specified database synchronization mode. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLD</td>
<td>no synchronization.</td>
</tr>
<tr>
<td>WARM</td>
<td>network synchronization.</td>
</tr>
<tr>
<td>HOT</td>
<td>AIJ synchronization.</td>
</tr>
<tr>
<td>COMMIT</td>
<td>Transaction synchronization.</td>
</tr>
</tbody>
</table>

Replication governor: Disabled

This field indicates that the “Replication Governor” feature is disabled.

Replication governor: keyword

This field identifies the replication governor database synchronization mode. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLD</td>
<td>no synchronization.</td>
</tr>
<tr>
<td>WARM</td>
<td>network synchronization.</td>
</tr>
<tr>
<td>HOT</td>
<td>AIJ synchronization.</td>
</tr>
<tr>
<td>COMMIT</td>
<td>Transaction synchronization.</td>
</tr>
</tbody>
</table>

---
Number of clients: number

This field identifies the number of clients actively participating in the “Hot Standby” feature. A client is an AIJ Log Server (ALS) on the master database.

Quiet-point: keyword

This field indicates whether or not the “Hot Standby” feature uses a quiet-point during startup. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The quiet-point is used at startup.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>No quiet-point is used at startup.</td>
</tr>
</tbody>
</table>

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Options

This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

Refresh

This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.

This section discusses screen-specific issues.

Page Navigation

If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key. To display a previous page, press the left angle bracket “<” key or the ↑ Previous Screen key.

Binary File Support

This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

Cluster Statistic Collection Support

This screen is not integrated into the “Cluster Statistic Collection” facility.

Zoom Screen

This screen does not have a zoom screen.
This screen does **not** have any configuration options.
Audit Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “Audit Information” screen reflects information stored in the KROOT portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Audit Information” screen:

Node: MYNODE (1/1/16)   Oracle Rdb X7.0-00 Perf. Monitor  3-FEB-1998 05:39:52.56  
Rate: 1.00 Second              Audit Information            Elapsed: 00:00:34.16  
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online  

--------------------------------------------------------------------------------
Security Auditing is disabled
Security alarm is disabled
Audit journal filename is “Not Specified”
Alarm name is “Not Specified”
Synchronous audit record flushing is disabled
Audit every access

--------------------------------------------------------------------------------
Exit Help Menu Options Refresh Set_rate Write !

Screen Fields

Security Auditing is keyword
This field indicates whether or not the “Security Auditing” feature is enabled. The keyword keywords are the following:
**Security auditing**

This field indicates whether or not the "Security Auditing" feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th><strong>Keyword</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The &quot;Security Auditing&quot; feature is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The &quot;Security Auditing&quot; feature is not used.</td>
</tr>
</tbody>
</table>

**Security alarm is enabled**

This field indicates whether or not the "Security Alarm" feature is enabled. The keyword keywords are the following:

<table>
<thead>
<tr>
<th><strong>Keyword</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>The &quot;Security Alarm&quot; feature is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>The &quot;Security Alarm&quot; feature is not used.</td>
</tr>
</tbody>
</table>

**Audit journal filename is "filespec"**

This field identifies the "Security Auditing" feature journal filename. If no journal filename is available, "Not Specified" is displayed.

**Alarm name is "name"**

This field identifies the "Security Auditing" feature alarm name. If no alarm name is available, "Not Specified" is displayed.

**Asynchronous audit record flushing is keyword**

This field indicates whether or not audit record flushing is synchronous. The keyword keywords are the following:

<table>
<thead>
<tr>
<th><strong>Keyword</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>Flushing is synchronous.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>Flushing is asynchronous.</td>
</tr>
</tbody>
</table>

**Audit keyword access**

This field indicates whether to audit every access or just the first access. The keyword keywords are the following:

<table>
<thead>
<tr>
<th><strong>Keyword</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>FIRST</td>
<td>Audit the first access only.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>EVERY</td>
<td>Audit every access.</td>
</tr>
</tbody>
</table>

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

**Options**

- **Refresh**
  - This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.
  - This option refreshes information that may have been modified on another node of the cluster. Normally, the information refresh is performed automatically by other database users on the current node.

**Screen Information**

- **Page Navigation**
  - If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key. To display a previous page, press the left angle bracket “<” key or the ↑ Previous Screen key.

- **Binary File Support**
  - This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

- **Cluster Statistic Collection Support**
  - This screen is not integrated into the “Cluster Statistic Collection” facility.

- **Zoom Screen**
  - This screen does not have a zoom screen.

**Configuration Options**

- This screen does not have any configuration options.
Active User Information Screen

This screen displays dynamic information that automatically changes to reflect database parameter modifications. This information is similar to that produced by the RMU Dump Header utility.

The “Active User Information” screen reflects information stored in the RTUPB portion of the database rootfile.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to database parameters on other cluster nodes may not be visible immediately on the local node. You can either wait for the screen to eventually refresh itself, or you can use the “Refresh” on-screen menu option to periodically refresh the screen information.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Active User Information” screen:

Node: MYNODE (1/1/16)  Oracle Rdb X7.0-00 Perf. Monitor  3-FEB-1998 05:40:00.89
Rate: 1.00 Second       Active User Information         Elapsed: 00:00:42.49
Page: 1 of 4     DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

--------------------------------------------------------------------------------
Active user with process ID 38203205 (recovery in progress)
Read/write transaction in progress
Transaction sequence number is 0:36282
Monitor ID is 2
Internal Stream ID is 1
Internal Transaction ID is 4
--------------------------------------------------------------------------------
Exit Help Menu >next_page <prev_page Options Refresh Set_rate Write !

Screen Fields

Active user with process ID number keyword
This field identifies an active database user with the specified process identifier. The keywords describe various transient process states, as seen in the example above. The keyword keywords are the following:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECOVERY IN PROGRESS</td>
<td>The process is being recovered by the database recovery process (DBR).</td>
</tr>
<tr>
<td>REMOTE ACCESS SERVER</td>
<td>The process is a remote connection server.</td>
</tr>
<tr>
<td>DATABASE SERVER</td>
<td>The process is a database server, such as the AIJ Log Server (ALS).</td>
</tr>
<tr>
<td>DATABASE UTILITY</td>
<td>The process is a database utility, such as the RMU Backup utility.</td>
</tr>
</tbody>
</table>

| User name is name              | This field identifies the user name.                                       |
| Process name is name           | This field identifies the process name.                                     |
| Image name is name             | This field identifies the image being executed by the process.             |
| keyword transaction in progress| This field identifies the type of transaction in progress, if any. Note that pre-started transactions are not identified as such. The keyword keywords are the following: |
| READ/WRITE                     | The user is using a read/write transaction.                                |
| SNAPSHOT                       | The user is using a snapshot (read-only) transaction.                      |
| DEFERRED SNAPSHOT              | The user is using a deferred (pending) snapshot (read-only) transaction.   |
| NO                             | The user does not have an active transaction.                              |

| Transaction sequence number is number: number | This field identifies the current transaction sequence number (TSN) if any. |
| Monitor ID is number              | This field identifies the monitor identifier for the current node.        |
| Internal Stream ID is number      | This field identifies the internal process stream identifier (SID).        |
This field identifies the internal transaction identifier (TID).

**ON-SCREEN MENU OPTIONS**

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refresh</td>
<td>This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.</td>
</tr>
</tbody>
</table>

**SCREEN INFORMATION**

<table>
<thead>
<tr>
<th>Page Navigation</th>
<th>If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “&gt;” key or the Next Screen key. To display a previous page, press the left angle bracket “&lt;” key or the Previous Screen key.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary File Support</td>
<td>This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.</td>
</tr>
<tr>
<td>Cluster Statistic Collection Support</td>
<td>This screen is not integrated into the “Cluster Statistic Collection” facility.</td>
</tr>
<tr>
<td>Zoom Screen</td>
<td>This screen does not have a zoom screen.</td>
</tr>
</tbody>
</table>

**CONFIGURATION OPTIONS**

This screen does not have any configuration options.
Statistics Event Information Screen

This screen displays information about user-defined events. These are not database parameters in the tradition sense; however, they control the operation of the RMU Show Statistic utility, so they are presented as a database parameter.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

The user-defined event information is presented in no particular order.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “Statistics Event Information” screen:

Rate: 1.00 Second         Statistics Event Information      Elapsed: 00:00:19.10
Page: 1 of 1    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online
--------------------------------------------------------------------------------
Statistic......... Event........ State...     Threshold Every       Current  Cnt
transactions      MAX_RATE      enabled            1.0     2           0.0    0
transactions      MAX_CUR_TOTAL enabled            2.0     2           0.0    0
transactions      MIN_CUR_RATE  enabled           10.0     2           0.0    0
transactions      MAX_CUR_RATE  enabled            2.0     2           0.0    0
transactions      MIN_AVG_RATE  enabled           4294967295.0     0           0.0    0
transactions      MAX_AVG_RATE  enabled            3.0     2           0.0    0
transactions      MIN_PER_TX    enabled   4294967295.0     2           0.0    0
transactions      MAX_PER_TX    enabled            4.0     2           0.0    0
process failures   MAX_CUR_TOTAL enabled            0.0     1           0.0    0
process attaches   MAX_CUR_TOTAL enabled            1.0     1           1.0    1
synch data reads   MAX_CUR_TOTAL enabled            7.0    11           0.0    0
locks requested    MAX_CUR_TOTAL enabled            6.0    10           0.0    0
pages checked      MAX_CUR_TOTAL enabled            3.0     7           0.0    0
pages checked      MAX_CUR_TOTAL enabled            4.0     8           0.0    0
pages checked      MAX_CUR_TOTAL enabled            5.0     9           0.0    0
pages checked      MAX_CUR_TOTAL enabled            1.0     1           0.0    0
--------------------------------------------------------------------------------

Config Exit Full Help Menu >next_page <prev_page Options Pause Set_rate Write !

The following screen fields are available when both the “Brief” and “Full” on-screen menu options are selected:

Statistic This field identifies the name of the statistic for which the event is defined.
This field identifies the name of the event for the statistic.

This field identifies the state of the event. The state field has the following values:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENABLED</td>
<td>the event is active.</td>
</tr>
<tr>
<td>DISABLED</td>
<td>the event is no longer active.</td>
</tr>
</tbody>
</table>

This field defines the “current” event threshold; the default value is “0” for MAX_XXX thresholds and “very big number” for MIN_XXX thresholds. The default value guarantees that at least one event will be signaled, thereby initializing the new “current” threshold value.

This field defines the value by which the initial threshold will be incremented or decremented when an event is signaled. If this value is “0”, the default value, for any event except the MIN_CUR_RATE and MAX_CUR_RATE events, then the event will be signaled just once.

This field identifies the current value of the statistic field. This is presented for reference only.

This field identifies the number of times the user-defined event as been triggered.

Program/ Operator Notification

This field defines the user-supplied program to be invoked for all events defined on the specified statistic. On OpenVMS, the program name is specified as a DCL process global symbol known to the RMU Show Statistic utility. If a program is not specified, this field defines the quoted comma-separated list of operators to be notified for all events defined on the specified statistic. Valid operator keywords are CENTRAL, DISKS, CLUSTER, SECURITY and OPER1 through OPER12.

This field identifies the storage area identifier for events defined on partitioned logical area or specific storage area statistics. The value “0” indicates the storage area is not used.

This field identifies the logical area identifier, if any. The value “0” indicates the logical area is not used.

This field defines the number of event notifications remaining before performing an actual notification. This clause is extremely useful for the MIN_CUR_RATE and
MAX_CUR_RATE events, as the thresholds for these events are not reset upon being signaled. The default value of "0" ensures that all events are notified.

Limit

This field defines the maximum number of times the event may be signaled. If the value is "0", the default value, events may be signaled indefinitely, if the EVERY clause is specified with a non-zero value.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.

Brief

You select brief mode by typing "B". In brief mode, one line per event is displayed, providing the following information: statistic name, event name, event state, threshold value, every value, current value and trigger count.

The following is an example of the "Statistics Event Information" screen in brief mode:

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Event</th>
<th>State</th>
<th>Threshold</th>
<th>Every</th>
<th>Current</th>
<th>Cnt</th>
</tr>
</thead>
<tbody>
<tr>
<td>transactions</td>
<td>MAX_RATE</td>
<td>enabled</td>
<td>1.0</td>
<td>2</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>2.0</td>
<td>2</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MIN_CUR_RATE</td>
<td>enabled</td>
<td>10.0</td>
<td>2</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MAX_CUR_RATE</td>
<td>enabled</td>
<td>2.0</td>
<td>2</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MIN_AVG_RATE</td>
<td>enabled</td>
<td>4294967295.0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MAX_AVG_RATE</td>
<td>enabled</td>
<td>3.0</td>
<td>2</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MIN_PER_TX</td>
<td>enabled</td>
<td>4294967295.0</td>
<td>2</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MAX_PER_TX</td>
<td>enabled</td>
<td>4.0</td>
<td>2</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>process failures</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>0.0</td>
<td>1</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>process attaches</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>1.0</td>
<td>1</td>
<td>1.0</td>
<td>1</td>
</tr>
<tr>
<td>synch data reads</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>7.0</td>
<td>11</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>locks requested</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>6.0</td>
<td>10</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>pages checked</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>3.0</td>
<td>7</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>pages checked</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>4.0</td>
<td>8</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>pages checked</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>5.0</td>
<td>9</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>pages checked</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>1.0</td>
<td>1</td>
<td>0.0</td>
<td>0</td>
</tr>
</tbody>
</table>

Config Exit Full Help Menu >next_page <prev_page Options Pause Set_rate Write!
The following is an example of the “Statistics Event Information” screen in full mode:

```
Node: MYNODE (1/1/16)   Oracle Rdb X7.0-00 Perf. Monitor  3-FEB-1998 17:59:20.82
Rate: 1.00 Second         Statistics Event Information      Elapsed: 00:00:15.79
Page: 1 of 2    DISK$:[DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1     Mode: Online

<table>
<thead>
<tr>
<th>Statistic ..........</th>
<th>Event ..........</th>
<th>State ..........</th>
<th>Threshold Every</th>
<th>Current Cnt</th>
</tr>
</thead>
<tbody>
<tr>
<td>transactions</td>
<td>MAX_RATE</td>
<td>enabled</td>
<td>1.0</td>
<td>0</td>
</tr>
<tr>
<td>EVENT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>2.0</td>
<td>0</td>
</tr>
<tr>
<td>EVENT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MIN_CUR_RATE</td>
<td>enabled</td>
<td>10.0</td>
<td>0</td>
</tr>
<tr>
<td>EVENT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MAX_CUR_RATE</td>
<td>enabled</td>
<td>2.0</td>
<td>0</td>
</tr>
<tr>
<td>EVENT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MIN_AVG_RATE</td>
<td>enabled</td>
<td>4294967295.0</td>
<td>0</td>
</tr>
<tr>
<td>EVENT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MAX_AVG_RATE</td>
<td>enabled</td>
<td>3.0</td>
<td>0</td>
</tr>
<tr>
<td>EVENT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MIN_PER_TX</td>
<td>enabled</td>
<td>4294967295.0</td>
<td>2</td>
</tr>
<tr>
<td>EVENT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>transactions</td>
<td>MAX_PER_TX</td>
<td>enabled</td>
<td>4.0</td>
<td>0</td>
</tr>
<tr>
<td>EVENT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>process failures</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>DBR_LOGGER</td>
<td></td>
<td></td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>process attaches</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>1.0</td>
<td>0</td>
</tr>
<tr>
<td>OPER11 OPER12</td>
<td></td>
<td></td>
<td>1.0</td>
<td>1</td>
</tr>
<tr>
<td>synch data reads</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>7.0</td>
<td>0</td>
</tr>
<tr>
<td>DB_ALERT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>100</td>
</tr>
<tr>
<td>locks requested</td>
<td>MAX_CUR_TOTAL</td>
<td>enabled</td>
<td>6.0</td>
<td>0</td>
</tr>
<tr>
<td>DB_ALERT</td>
<td></td>
<td></td>
<td>0.0</td>
<td>100</td>
</tr>
</tbody>
</table>
```

---

**Options**

This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

**Pause**

This on-screen menu option temporarily pauses the presentation of the statistic information. Pressing the “Pause” on-screen menu option will resume the statistic presentation.

---

**Page Navigation**

If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the ↓ Next Screen key.
To display a previous page, press the left angle bracket “<” key or the Previous Screen key.

**Binary File Support**
This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

**Cluster Statistic Collection Support**
This screen is not integrated into the “Cluster Statistic Collection” facility.

**Zoom Screen**
This screen does not have a zoom screen.

---

**Configuration Options**
This screen does not have any configuration options.
OpenVMS SYSGEN Parameters Screen

It is often necessary to review VMS SYSGEN parameter settings when analyzing database performance characteristics. It is very beneficial that you do not have to exit the RMU Show Statistic utility to monitor the interesting SYSGEN parameters.

The “OpenVMS SYSGEN Parameters” screen reflects information acquired from OpenVMS directly. This information is not stored in the database.

However, because the information is local to the node from which you are running the RMU Show Statistic utility, the modifications you make to OpenVMS SYSGEN parameters on other cluster nodes may not be visible immediately on the local node. However, because the information cannot be readily retrieved from other nodes in the cluster, you cannot refresh their values manually.

You cannot use the information contained in this screen on the “Custom Statistics” screen.

The “OpenVMS SYSGEN Parameters” screen contains a number of interesting OpenVMS SYSGEN utility parameters that have been determined to be of value in analyzing database performance. The screen is divided into two columns, in the belief that eventually there will be sufficient “interesting” parameters that require two columns. The alphabetical VMS SYSGEN parameters are displayed in “directory format”, from top to bottom in each column.

This screen resides in the “Database Parameter Info” menu.

The following is an example of the “OpenVMS SYSGEN Parameters” screen:
Node: MYNODE (1/1/16)   Oracle Rdb X7.0-00 Perf. Monitor 3-FEB-1998 05:40:06.43
Rate: 1.00 Second OpenVMS SYSGEN Parameters  Elapsed: 00:00:48.03
Page: 1 of 1 DISK$: [DB_HOMEDIR.WORK.STATS]MF_PERSONNEL.RDB;1 Mode: Online

Parameter.Name.... Parameter.Value... Parameter.Name..... Parameter.Value...
CHANNELCNT                   1500
CONTIG_GBLPAGES            163686
DEADLOCK_WAIT                  10
FREE_GBLPAGES              170974
GBLPAGES                   350000
GBLPAGFIL                   70000
GBLSECTIONS                  2500
NPAGEDYN                  11999744
NPAGEVIR                  48000000
PAGEDYN                  7875072
PAGEFILE_FREE             2028382
PAGEFILE_PAGE             2097104
SWAPFILE_FREE              159992
SWAPFILE_PAGE              159992
VIRTUALPAGECNT            450000
WSMAX                        150000

Please refer to the appropriate OpenVMS reference manuals for detailed descriptions of the SYSGEN parameters displayed on this screen.

SCREEN FIELDS

CHANNELCNT
CHANNELCNT specifies the number of permanent I/O channels available to the system. You should set this value to a number larger than the largest FILLM value in the database environment. One channel count is used every time a file is opened in your database application. For example, in a single-file database with one user, this value will be over 4: one for the .RDB file, one for the .SNP file, one for the user's .RUJ file, and one for the after-image journal (when enabled). Temporary files such as sort files and files assigned to SYS$OUTPUT and SYS$INPUT also use channel count.

DEADLOCK_WAIT
DEADLOCK_WAIT defines the number of seconds a lock request must wait before the system initiates a deadlock search. Set DEADLOCK_WAIT low for a multi-user application involving a high-contention, high-update work load in either a cluster or non-cluster environment. One second per node in the Cluster +1 is recommended. For example, if there are 10 nodes in the cluster that access the database, then set DEADLOCK_WAIT to 11 seconds. When DEADLOCK_WAIT is set to 11 seconds, a lock request must wait 11 seconds before the system initiates a deadlock search.

Work loads characterized by read-only tasks use fewer lock resources and are not likely to encounter deadlocks. Therefore, a low value for DEADLOCK_WAIT in a read-intensive environment is not as critical and might decrease overall performance by
causing unnecessary checking for deadlocks. In this case, the default value of 10 is recommended.

You may decide to set different values for this parameter and determine which value best meets the performance needs for your particular database environment. Review the lock statistics on your system using MONITOR. This utility provides information on both deadlock search and deadlock find counts. Oracle Corporation recommends that you leave the default value of 10 seconds for DEADLOCK_WAIT unless MONITOR shows a high percentage of deadlocks. Set the value to less than 10 seconds to get deadlocks signaled more frequently. If, at 10 seconds, there are frequently no deadlocks found, set the DEADLOCK_WAIT value to 20 or 30 seconds. Be careful when you use the RMU Show Statistic utility to tune DEADLOCK_WAIT because this command shows you locking statistics for just the database in question, not the entire system.

FREE_GBLPAGES identifies the total number of available global page table entries. This parameter cannot actually be changed by the DBA and is provided for informational purposes only.

GBLPAGES sets the number of global page table entries allocated at startup time. Each global section requires one global page table entry. Every 128 entries add 4 bytes to permanently resident memory in the form of a system page table entry. When global buffers are enabled for a database, the size of the global section increases, which could mean that you need to increase the GBLPAGES value.

GBLSECTIONS sets the number of global section descriptors allocated in the system header at startup time. Each section takes 32 bytes of permanently resident memory.

NPAGEVIR defines the maximum size to which NPAGEDYN can be increased. If this value is too small, systems could hang. To set this parameter, use the default value initially, and then monitor the amount of space actually used with the DCL command SHOW MEMORY/POOL/FULL.

NPAGEDYN defines the size of the non-paged dynamic pool in bytes. This parameter NPAGEDYN establishes the initial setting of the non-paged pool size, but the pool size can be increased dynamically. To set a value for this parameter, use the default value initially, and then monitor the amount of space actually used with the DCL command SHOW MEMORY/POOL/FULL.

PAGEFILE_FREE identifies the total number of available pagefile space. This parameter cannot actually be changed by the DBA and is provided for informational purposes only.

PAGEFILE_PAGE identifies the total number of in-use pagefile space. This parameter cannot actually be changed by the DBA and is provided for informational purposes only.
**SWAPFILE_FREE**

SWAPFILE_FREE identifies the total number of available swapfile space. This parameter cannot actually be changed by the DBA and is provided for informational purposes only.

**SWAPFILE_PAGE**

SWAPFILE_PAGE identifies the total number of available swapfile space. This parameter cannot actually be changed by the DBA and is provided for informational purposes only.

**VIRTUALPAGECNT**

The VIRTUALPAGECNT parameter sets the total number of virtual pages that your process is allowed to map. Because your process maps to the global section, as the global section grows, so should the number of virtual pages that processes are allowed to map. Therefore, if you increase the GBLPAGES parameter, you should increase the VIRTUALPAGECNT parameter by about the same amount.

If you change the VIRTUALPAGECNT parameter, you must reboot your system for the change to take effect.

Oracle Rdb users who have global buffers enabled on the database sometimes encounter the -LIB-F-INSVIRMEM, insufficient virtual memory error. This error message indicates that the VIRTUALPAGECNT or PGFLQ QUOTA quotas for a process are not large enough.

When global buffers are enabled, the monitor process' virtual memory consumption is proportional to the number of global buffers.

In addition, the more global buffers that are defined for the database, the higher the virtual memory requirements are for each user. Each user needs virtual memory to map the buffer pool. Although this section discusses the virtual memory consumed by the Oracle Rdb monitor, keep in mind that virtual memory and pagefile quota limits also apply to user processes. This section discusses the monitor because it is the less obvious place to look for a quota problem.

The monitor does not completely release all virtual memory when a database with global buffers enabled is closed, and this appears to contribute to the frequency of virtual memory errors.

**WSMAX**

WSMAX sets the maximum number of pages on a system-wide basis for any working set. Set the value for WSMAX to the size of the largest working set needed on your system. This is useful in a heterogeneous cluster environment, where memory differs but a common UAF file is used. The default value is appropriate for normal time-sharing operations, while significantly larger values should be used only to reduce page-faulting for programs with very large virtual address spaces.

This section discusses the on-screen menu options available to this screen. Remember that the Exit, Help (?), Menu, Set_rate, Write and (!) on-screen menu options are available on all screens unless otherwise noted.
This on-screen menu option displays a menu containing options for writing all of the screens to a text file named STATISTICS.RPT. This option should be used with care because it generates a very large output file.

This section discusses screen-specific issues.

Page Navigation: If there is more information than can fit on one page, the notation “Page 1 of n” appears in the header region, where n is the total number of pages. You can display successive pages by pressing the right angle bracket “>” key or the \( \downarrow \) Next Screen key. To display a previous page, press the left angle bracket “<” key or the \( \uparrow \) Previous Screen key.

Binary File Support: This screen is not recorded in the binary output file produced using the OUTPUT qualifier. Consequently, this screen is not available when you replay a binary file using the INPUT qualifier.

Cluster Statistic Collection Support: This screen is not integrated into the “Cluster Statistic Collection” facility.

Zoom Screen: This screen does not have a zoom screen.

Configuration Options: This screen does not have any configuration options.
Index

/CLUSTER, 240
/OPTIONS, 16
ACTIVE, 89, 131, 132, 135, 136, 139, 140, 143, 144, 147, 148, 151, 152, 155, 156, 159, 160, 163, 164, 167, 168, 172, 173, 176, 177
AU Backup Server, 110, 133, 137, 141, 145, 149, 153, 157, 161, 165, 169, 173, 178, 215, 216
AU Log Server, 216, 228, 231, 237
AUDB, 214, 224, 227
AUFB, 220
CLIENT, 125, 131, 133, 135, 137, 139, 141, 143, 145, 147, 149, 151, 153, 155, 157, 159, 161, 163, 165, 167, 169, 172, 174, 176, 178
Commit to Journal Optimization, 225
DECnet, 228
Emergency AU facility, 221
Fast Commit, 132, 136, 140, 144, 148, 152, 156, 160, 164, 168, 173, 177, 224, 225, 229
Fast Incremental Backup, 193
FILID, 76, 131, 132, 135, 136, 139, 140, 143, 144, 147, 148, 151, 152, 155, 156, 159, 160, 163, 164, 167, 168, 172, 173, 176, 177
Hot Standby, 56, 110, 227, 228, 229, 231
MEMBIT, 93, 125, 131, 132, 133, 135, 137, 139, 141, 143, 145, 147, 149, 151, 153, 155, 157, 159, 161, 163, 165, 167, 169, 172, 174, 176, 178
RCACHE, 210
RDM$BIND_SNAP_QUIET_POINT, 113
RMU Analyze, 35, 40
RMU Backup, 110, 133, 137, 141, 145, 149, 153, 157, 161, 165, 169, 173, 178, 192, 237
RMU Backup After_Journal, 110, 133, 137, 141, 145, 149, 153, 157, 161, 165, 169, 173, 178
RMU Close, 191
RMU Dump Header, 190, 195, 201, 204, 210, 214, 216, 220, 222, 224, 227, 233, 236
RMU Open, 191
RMU Repair, 33, 36, 38, 41, 43, 48, 50, 53
Rootfile, 6, 12, 76, 132, 136, 140, 144, 148, 152, 156, 160, 164, 168, 173, 177, 182, 183, 190, 195, 196, 201, 204, 210, 214, 220, 224, 227, 233, 236
RTUPB, 85, 131, 132, 135, 136, 139, 140, 143, 144, 147, 148, 151, 152, 155, 156, 159, 160, 163, 164, 167, 168, 172, 173, 176, 177, 236
SNAPSHOT Area Cursor lock, 102, 132, 136, 140, 144, 149, 152, 156, 160, 164, 169, 173, 178
Snapshot Area File, 205
SNAPSHOT File, 5, 6, 7, 19, 21, 22, 181, 185, 192
Storage Area File, 205
TCP/IP, 228
TSNBLK, 80, 131, 132, 135, 136, 139, 140, 143, 144, 147, 148, 151, 152, 155, 156, 159, 160, 163, 164, 167, 168, 172, 173, 176, 177