

Guide to Performance and Tuning: Managing the AIP Data Structure

A feature of Oracle Rdb

By Ian Smith
Oracle Rdb Relational Technology Group
Oracle Corporation

| | |
|---|----------|
| GUIDE TO PERFORMANCE AND TUNING: MANAGING THE AIP DATA STRUCTURE | 3 |
| OVERVIEW | 3 |
| FINDING A PAGE IN A UNIFORM AREA | 4 |
| CHANGES TO SQL COMMANDS | 5 |
| CHANGES TO ROW SIZE | 6 |
| CHANGES TO RMU COMMANDS | 8 |
| RMU SET AIP COMMAND | 8 |
| <i>Format</i> | 9 |
| <i>Description</i> | 9 |
| <i>Command Parameters</i> | 9 |
| <i>Command Qualifiers</i> | 10 |
| <i>Usage Notes</i> | 11 |
| <i>Examples</i> | 13 |
| Rebuilding SPAM pages for logical areas | 13 |
| Updating the length in the AIP for a table | 13 |
| Updating the length for a table and rebuilding SPAM pages | 13 |
| Updating the thresholds for a SORTED index | 14 |
| RMU SHOW AIP COMMAND | 15 |
| <i>Format</i> | 15 |
| <i>Description</i> | 15 |
| <i>Command Parameters</i> | 15 |
| <i>Command Qualifiers</i> | 16 |
| <i>Usage Notes</i> | 17 |
| <i>Examples</i> | 17 |
| Displaying the AIP entry for the JOBS table..... | 17 |
| Using wildcards and /TYPE qualifier | 18 |
| Locating AIP entries that need rebuilding | 19 |
| Using /BRIEF and /PAREA qualifiers..... | 20 |
| FREQUENTLY ASKED QUESTIONS | 21 |

The Rdb Technical Corner is a regular feature of the Oracle Rdb Web Journal. The examples in this article use SQL language from Oracle Rdb release 7.2.1 and later versions.

Guide to Performance and Tuning: Managing the AIP Data Structure

Starting Oracle Rdb V7.2.1 management of the AIP (Area Inventory Page) data structure has been enhanced. This paper describes the changes to RMU that allow you to examine and modify the AIP data structure, and changes to SQL that allow these structures to be managed automatically.

Overview

An Oracle Rdb database is constructed from several physical files that contain the data as well as special structures describing the database itself. This metadata is maintained in several forms:

1. The database root file (.RDB) contains information about the physical structure, names and locations of storage area files (.RDA) and their associated snapshot files (.SNP), definition of the after image journals (.AIJ), definition of the row caches and their backing store files (.RDC*) and details of the active transactions and their associated recovery journal files (.RUJ).
2. The Rdb system tables reside in the Rdb\$SYSTEM storage area, and possibly within optional DEFAULT STORAGE AREA and LIST STORAGE AREA files. The system tables describe the database structure including the user-defined tables, columns, indices, sequences and routines.
3. The Rdb\$SYSTEM storage area also contains a special data structure called the AIP, or Area Inventory Pages. The AIP is a chain of pages describing the logical areas created by the SQL statements **create table**, **create storage map** and **alter storage map**, **create index** and **alter index**.

There are two types of physical storage areas available:

- MIXED format areas that allows rows from different tables to be stored on the same page, possibly with index nodes and other index data structures.
- UNIFORM format areas that collect rows or index nodes of the same table or index on the same set of pages. Mixing of different record types normally does not occur on a page.

This article is primarily concerned with **uniform** page format.

Finding a Page in a UNIFORM Area

When a row is inserted into a table the database system calculates several pieces of information:

1. Which physical storage should be used? This is determined by the STORAGE MAP associated with the table, or the storage mapping defined by the CREATE INDEX command.

The SHOW STORAGE MAP (PARTITION) or SHOW INDEX (PARTITION) commands of Interactive SQL will give details of this mapping, showing the names of the physical areas and related attributes.

The storage map commands create **logical areas** within a physical area. The logical segmentation of the physical area allow more efficient management of the data during **insert**, **truncate table**, **drop index**, and similar operations. We will examine this later in this article.

2. Which part of the physical area should be used? Each physical area is divided into fixed sized zones, or ranges. Each range is managed by a special data structure called a SPAM, or Space Management page. The database system uses the ABM, or Area Bitmap to locate any SPAM pages to which the **logical area** is assigned. The SPAM page is then accessed to determine which groups of pages in the SPAM range should be referenced for the **insert**.
3. Which pages should be fetched for the **insert**? The database system tries hard to avoid wasted I/O so avoiding pages that are full is of primary concern. To this end the SPAM pages maintain a fullness indicator that can be used to quickly eliminate pages that appear to have too little room for the inserted data. This fullness level is

known as **logical area thresholds**.

Thresholds can be defined for each logical area using the THRESHOLDS clause of the **create storage map**¹, and **create index** statements. There are three threshold values that can be used to establish rough guidelines for the database system relative to the free space on the page and the length of the data being inserted. For example, a UNIFORM area with page size of 4 blocks will have 2006 free octets available on the page. If I am storing fixed sizes rows of 178 octets then I may choose to set the threshold to 92% so that the page is marked full when I no longer have space to store one more row.

4. Those page groups that appear to have free space (sufficient to fill a buffer) are read and examined. The fetched page may still not have sufficient free space due to locked free space by another transaction, or possibly because of imprecise fullness levels that directed us to this page.

The RMU Show AIP command can be used to display the active logical areas in the database. The RMU Dump Larea=RDB\$AIP command can also be used to examine the logical areas in the database. This command displays all the logical area pages that will include areas that are currently active, that were created and later deleted, or entries that have not yet been used.

Changes to SQL Commands

In previous releases of Oracle Rdb the length of a stored row was recorded by the CREATE TABLE command, or the CREATE STORAGE MAP command when each of the logical areas was created. If at a later time the row was altered then the stored row length might change.

The stored row length is calculated as the sum of the items described in the following table.

| Item | Description |
|---------------|--|
| Record header | This is a standard 5 octet header that describes the table. The relation |

¹ Alter storage map and alter index can also be used to establish thresholds for areas. We will discuss this further below.

| | |
|-----------------------|---|
| | identification (Rdb\$RELATION_ID) is stored here and is used by sequential scans to return table rows. |
| Row version number | This is an unsigned word (smallint) that defines the version of the stored data. |
| User data section | This is the column data as defined by the create table or alter table statements. When computing this length note that VARCHAR columns include an unsigned word (smallint) length field, and computed by columns take no space in the stored row. |
| NULL bit vector (NBV) | A single bit is reserved for each column defined for the table. Each column identifier (rdb\$field_id) is used as an index to this bit map. This distinction is important to fully understand the implications of alter table and its effect on the NBV. When alter table adds a new column a new column identifier is assigned, a column identifier of a dropped column is never reused. Therefore, the NBV may grow over time even if the user data stays the same size. The NBV is rounded up to the next octet boundary. |
| Alignment Padding | Some Rdb system tables include additional padding to force alignment at runtime. This is not present in user-defined data. |

The calculated value is used by Oracle Rdb to locate sufficient free space on a page to insert new data.

Changes to Row Size

Various SQL DDL commands can alter the size of the stored row.

- In some cases **create table** will revise the length of the logical area when **automatic as** columns are defined in the table.
- Adding a new column using **alter table ... add column ...** will revise the length to include the new column size as well as any growth in the null bit vector.
- Dropping an existing column using **alter table ... drop column ...** will revise the length to eliminate the dropped columns size. The exception is dropping **computed by** columns, which don't require space in the row.
- Altering a data type of a column using **alter table ... alter column ...** or **alter domain** may cause the size to change. For instance, modifying a numeric column to a different magnitude type. For example, **integer** to **bigint**.
- Altering the length of a character string using **alter table ... alter column ...** or **alter domain** will often cause the size to change. When a **char** type is changed to **varchar** then a **smallint** length field is included.
- Several **drop** and **add** of columns may causing the null bit vector to grow even when the row content remains the same size. This is because old column identifiers are not reused and space will be required for the new column identifier.

In previous versions these changes were not reflected in the AIP and so over time performance of **insert** may degrade. This occurs because the AIP length is used to calculate a fullness level for the target page. Rdb uses this fullness level to find an appropriate page in the SPAM logical area list. Once found the page is fetched only to find that the new revised length does not fit on the page.

Database administrators can overcome this problem by defining thresholds or updating the AIP length using the RMU Repair Initialize command. However, this is an offline tool and the database administrator must calculate the revised length. Starting with Rdb V7.2.1 Rdb will automatically update the AIP record length field when any of these commands is executed:

- **Alter table ... add column**
- **Alter table ... drop column**
- **Alter domain**
- **Alter table ... alter column**
- **Truncate table:** In this case no size change is expected, however, a fixup action is queued for the logical areas that make up the table. This will repair any AIP record length fields that were out of synchronization with the table definition. This is a low cost action because there are no longer any SPAM pages to update.

Typically these commands are grouped within a transaction. Oracle Rdb queues these AIP update actions for processing at **commit** time. If any of these actions shows no net change then the queued update action is ignored. For instance, if the first **alter table** command drops a column of size 8 and the next **alter table** command defines two columns of size 4 then there may be no overall change in the record length.

The default behavior for Rdb is change the AIP record and set a flag informing the database administrator that a rebuild of the SPAM pages should be performed. The RMU Command section describes the various ways to observe this flag and to rebuild the SPAM pages.

The database administrator may desire that the SPAM pages be rebuilt during the DDL changes. In this case the SQL **set flags** command can be used prior to the **commit** statement to enable the rebuild action.

```
SQL> set flags 'rebuild_spams';
```

This will cause the **commit** processing to rebuild any SPAM pages that require modifying. Additional I/O will be required to scan the ABM pages, SPAM pages and the SPAM intervals.

Changes to RMU Commands

RMU Set AIP command

Oracle Rdb provides a command that modifies the contents of the AIP (Area Inventory Pages) structure. The AIP structure provides a mapping for logical areas to physical areas as well describing each of those logical areas. Information such the logical area name, length of the stored record, and storage thresholds can now be modified using this simple command interface. In prior versions the RMU Repair Initialize=Larea_Parameters command was the only RMU command that allowed updates to this information.

Format

```
RMU/SET AIP root-file-spec larea-name
  [/LAREA=(n [, ...])]
  [/LENGTH[=n]]
  [/LOG]
  [/REBUILD_SPAMS]
  [/RENAME_TO=new-name]
  [/THRESHOLD=(p,q,r)]
```

Description

This RMU command is used to modify some attributes of an existing logical area. It cannot be used to add or delete a logical area. This command can be used to correct the record length, thresholds and name of a logical area described by an AIP entry. It can also be used to rebuild the SPAM pages for a logical area stored in UNIFORM page format areas so that threshold settings for a page correctly reflect the definition of the table.

See also the RMU Repair Spam command for information on rebuilding SPAM pages for MIXED areas.

Command Parameters

- root-file-spec

The file specification for the database root file to be processed. The default file extension is .rdb.

- larea-name

An optional parameter that allows the logical areas to be selected by name. Only those AIP entries are processed.

Any partitioned index or table will create multiple logical areas all sharing the same

name. This string may contain standard OpenVMS file card characters (% and *) so that different names can be matched. Therefore, it is possible for many logical areas to match this name.

The value of **larea-name** may be delimited so that mixed case characters, punctuation and various character sets can be used.

Command Qualifiers

- Larea = (n1 [, n2 ...])

Specifies a list of logical area identifiers. The LAREA qualifier and larea-name parameter are mutually exclusive.

- Length [= value]

Sets the length of the logical area. If no value is provided on the RMU Set AIP command then Oracle Rdb will find the matching table and calculate a revised AIP nominal record length and apply it to the AIP.

- Log

Logs the names and identifiers of logical areas modified by this command.

- Rebuild_Spams

Locate each logical area with the "rebuild-spam" flag set and rebuild the SPAM pages.

- Rename_To = new-name

Used to change the logical area name. This qualifier should be used with caution as some RMU commands assume a strict mapping between table/index names and names of the logical area. This command can be used to repair names that were created in older versions of Oracle Rdb where the **rename table** command did not propagate the change to the AIP. The value of **new-name** may be delimited so that mixed case, punctuation and various character sets can be used.

- Threshold = (t1 [,t2 [, t3]])

Changes the threshold on all logical areas specified using the Larea qualifier or the larea-name parameter. RMU accepts THRESHOLD=(0,0,0) as a valid setting to disable logical area thresholds. Values must be in the range 0 through 100. Any missing values default to 100.

Usage Notes

- The database administrator requires RMU\$ALTER privilege to run the command and the Rdb server also requires SELECT and ALTER privilege on the database.
- This command supersedes the RMU Repair Initialize=Larea_Parameters command that can also change the Thresholds and Length for a logical area. This command can be executed online, where as the RMU Repair command must be run offline.
- Wildcard names are not permitted with the following qualifiers to prevent accidental propagation of values to the wrong database objects.
 - LENGTH qualifier with a value is specified,
 - RENAME_TO qualifier,
 - and THRESHOLDS qualifier.
- RMU Set AIP may be used on a master database configured for HOT STANDBY. All AIP changes and SPAM rebuild actions are written to the after image journal and will be applied to the standby database. This command cannot be applied to a STANDBY database.
- THRESHOLDS for MIXED format areas are physical area attributes and are not supported at the logical area (aka AIP) level. Therefore, THRESHOLDS can not be applied to MIXED areas and specifying logical area will cause an exception to be raised.
- The REBUILD_SPAMS qualifier is only applied to logical areas stored in UNIFORM page format storage areas.

- This command will implicitly commit any changes with no opportunity to undo them using rollback. Access to the functionality is controlled by privileges at the RMU and Rdb database level. We suggest that RMU Show AIP be used prior to any change so that you can compare the results and repeat the RMU Set AIP command with corrections if necessary.
- Some wildcard operations are restricted to prevent accidental damage to the database. For instance, a wildcard matching many objects will be rejected if more than one type of object is being changed. If a wild card selects both table and index types then this command will be rejected.
- This command is an online command. Each logical area will be processed within a single transaction and interact with other online users.
- When the AIP entry is changed online any existing users of the table or index will start to use the new values if the logical areas are reloaded.
- Various SQL alter commands will register changes for the AIP and these are applied at COMMIT time. RMU Verify and RMU Show AIP Option=REBUILD_SPAMS will report any logical areas that require SPAM rebuilding. The database administrator can also examine the output from the RMU Dump Larea=RDB\$AIP command.
- How long can the SPAM rebuild be delayed? The fullness of some page will have been calculated using the old AIP length or THRESHOLD values. Therefore, it might appear that a page is full when in fact the revised length will fit on the page, or the page may appear to have sufficient free space to store a row but once accessed the space is not available. By rebuilding SPAM pages you may reduce I/O during insert operations. However, delaying the rebuild to a convenient time will not affect the integrity of the database.
- The amount of I/O required for Rebuild_Spams depends upon the number of pages allocated to the table or index involved. Assuming just one logical area is selected then Oracle Rdb will read the ABM (Area Bitmap) to locate all SPAM pages in that area that reference this logical area. Rdb will then read each page in the SPAM interval for that SPAM page and recalculate the fullness based on the rows stored on each page.

Examples

Rebuilding SPAM pages for logical areas

RMU will call Rdb for each logical area that requires rebuilding.

```
$ RMU/SET AIP/REBUILD_SPAMS MF_PERSONNEL
%RMU-I-AIPSELMOD, Logical area id 86, name ACCOUNT_AUDIT selected for
modification
%RMU-I-AIPSELMOD, Logical area id 94, name DEPARTMENTS_INDEX selected for
modification
```

Updating the length in the AIP for a table

RMU will request that the EMPLOYEES table length be updated in the AIP. Oracle Rdb will use the latest table layout to calculate the length in the AIP and write this back to the AIP. The EMPLOYEES table is partitioned across three storage areas and therefore the Log qualifier shows these three logical areas being updated.

```
$ RMU/SET AIP MF_PERSONNEL EMPLOYEES/LENGTH/LOG
%RMU-I-AIPSELMOD, Logical area id 80, name EMPLOYEES selected for modification
%RMU-I-AIPSELMOD, Logical area id 81, name EMPLOYEES selected for modification
%RMU-I-AIPSELMOD, Logical area id 82, name EMPLOYEES selected for modification
```

Updating the length for a table and rebuilding SPAM pages

RMU will request that the EMPLOYEES table length be updated in the AIP and then the SPAM pages will be rebuilt. This is an ONLINE operation. Note: there is an implied relationship between the logical area name and the name of the object. This example assumes that the EMPLOYEES object is mapped to a UNIFORM page format area.

```
$ RMU/SET AIP MF_PERSONNEL EMPLOYEES/LENGTH/REBUILD_SPAMS
```

Updating the thresholds for a SORTED index

When thresholds for an index are modified they will not be effective until the SPAM pages are updated (rebuilt) to use these new values. The following example shows index maintenance performed by SQL. The SET FLAGS command is used to display information about the change. Note that the change is applied at COMMIT time and that the SPAM rebuild is deferred until a later time. RMU Set AIP is then used to rebuild the SPAM pages.

```
$ SQL$
SQL> set flags 'index_stats';
SQL> alter index candidates_sorted
cont> store in rdb$system (thresholds are (32,56,77));
~Ai alter index "CANDIDATES_SORTED" (hashed=0, ordered=0)
~Ai larea length is 215
~As locking table "CANDIDATES" (PR -> PU)
~Ai: reads: async 0 synch 58, writes: async 8 synch 0
SQL> commit;
%RDMS-I-LOGMODVAL,      modified space management thresholds to (32%, 56%, 77%)
%RDMS-W-REBUILDSPAMS, SPAM pages should be rebuilt for logical area
CANDIDATES_SORTED
$
$ RMU/SET AIP MF_PERSONNEL CANDIDATES_SORTED/REBUILD_SPAMS/LOG
%RMU-I-AIPSELMOD, Logical area id 74, name CANDIDATES_SORTED selected for
modification
```

RMU Show AIP command

Oracle Rdb provides a command that displays the contents of the AIP (Area Inventory Pages) structure. The AIP structure provides a mapping for logical areas to physical areas as well describing each of those logical areas. Information such the logical area name, length of the stored record, storage thresholds and other information can now be displayed using this simple command interface. In prior versions the RMU Dump Larea=RDB\$AIP command was the only RMU command that displayed this information.

Format

```
RMU/SHOW AIP rootfile [ larea-name ]
  [/LAREA=(n [,...]) ]
  [/PAREA=(n [,...]) ]
  [/OPTION=REBUILD_SPAMS]
  [/OUTPUT=output-filename]
  [/TYPE=type-name]
  [/BRIEF]
```

Description

The RMU Show AIP command allow the database administrator to display details of selected logical areas, or all logical areas in the database.

Command Parameters

- root-file-spec

The file specification for the database root file to be processed. The default file extension is .rdb.

- larea-name

An optional parameter that allows the logical area to be selected by name. Only those AIP entries are displayed. This parameter is optional and will default to all logical areas being displayed.

Any partitioned index or table will create multiple logical areas all sharing the same name. This string may contain standard OpenVMS file card characters (% and *) so that different names can be matched. Therefore, it is possible for many logical areas to match this name.

The value of **larea-name** may be delimited so that mixed case characters, punctuation and various character sets can be used.

Command Qualifiers

- Larea

Specifies a list of logical area identifiers. The LAREA qualifier and larea-name parameter are mutually exclusive. The default if neither the LAREA or PAREA qualifiers nor the larea-name parameter is specified is to display all AIP entries.

- Parea

Specifies a list of physical area identifiers. The PAREA qualifier and larea-name parameter are mutually exclusive. The default if neither the PAREA or LAREA qualifiers nor the larea-name parameter is specified is to display all AIP entries.

- Output [= output-filename]

This qualifier is used to capture the output in a named file. If used a standard RMU header is added to identify the command and database being processed. If omitted the output is written to SYS\$OUTPUT and no header is displayed.

- Option = REBUILD_SPAMS

Display only those logical areas that have the REBUILD_SPAMS flag set.

- Type = type-name

Legal values for type-name are TABLE, SORTED_INDEX, HASH_INDEX, LARGE_OBJECT, and SYSTEM_RECORD.

This qualifier is used in conjunction with **larea-name** to select a subset of the AIP entries that may match a name. For instance, it is legal in Rdb to create a table and an index with the name EMPLOYEES. So using EMPLOYEES/TYPE=TABLE will make the selection unambiguous. It also allows simpler wildcarding. Commands using *EMPLOYEE*/TYPE=TABLE will process only those tables that match and not the associated index logical areas.

- Brief

Display a tubular format brief output. The default is to display a more defaulted full output.

Usage Notes

- The database administrator requires RMU\$DUMP privilege as this command is closely related to the RMU DUMP LAREA=RDB\$AIP command.
- Only AIP entries that are in use are displayed. In contrast the RMU Dump command which also displays deleted and unused AIP entries.

Examples

Displaying the AIP entry for the JOBS table

This example uses the name of a known database table to display details for this single logical area.

```
$ RMU/SHOW AIP SQL$DATABASE JOBS
```

```
Logical area name JOBS
```

```
Type: TABLE
Logical area 85 in mixed physical area 7
Physical area name JOBS
Record length 41
Thesholds are (0, 0, 0)
AIP page number: 151
ABM page number: 0
Snapshot Enabled TSN: 64
```

Using wildcards and /TYPE qualifier

The wildcard string `"*EMPLOYEE*` matches both indices and table logical areas, so here we use `/TYPE` to limit the display to just table logical areas. The table `EMPLOYEES` in the `MF_PERSONNEL` database is partitioned across three storage areas and hence there exists three logical areas.

```
$ RMU/SHOW AIP SQL$DATABASE *EMPLOYEE*/TYPE=TABLE

Logical area name EMPLOYEES
Type: TABLE
Logical area 80 in mixed physical area 3
Physical area name EMPIDS_LOW
Record length 126
Thesholds are (0, 0, 0)
AIP page number: 150
ABM page number: 0
Snapshot Enabled TSN: 4800

Logical area name EMPLOYEES
Type: TABLE
Logical area 81 in mixed physical area 4
Physical area name EMPIDS_MID
Record length 126
Thesholds are (0, 0, 0)
AIP page number: 151
ABM page number: 0
Snapshot Enabled TSN: 1504
```

```
Logical area name EMPLOYEES
Type: TABLE
Logical area 82 in mixed physical area 5
Physical area name EMPIDS_OVER
Record length 126
Thesholds are (0, 0, 0)
AIP page number: 151
ABM page number: 0
Snapshot Enabled TSN: 1504
```

Locating AIP entries that need rebuilding

This example show the REBUILD_SPAMS option used to locate logical areas that require SPAM rebuilds. This may occur because the stored row length changed size, or THRESHOLDS were modified for the index or storage map.

```
$ RMU/SHOW AIP/OPTION=REBUILD_SPAMS
_Root: SQL$DATABASE
_Logical area name:

Logical area name ACCOUNT_AUDIT
Type: TABLE
Logical area 86 in uniform physical area 1
Physical area name RDB$SYSTEM
Record length 12
Thesholds are (10, 100, 100)
Flags:
    SPAM pages should be rebuilt
AIP page number: 151
ABM page number: 1004
Snapshot Enabled TSN: 5824

Logical area name DEPARTMENTS_INDEX
Type: SORTED INDEX
Logical area 94 in uniform physical area 10
Physical area name DEPARTMENT_INFO
Record length 430
Thesholds are (30, 65, 72)
```

```
Flags:
    SPAM pages should be rebuilt
AIP page number: 151
ABM page number: 2
Snapshot Enabled TSN: 7585
```

Using /BRIEF and /PAREA qualifiers

The /PAREA qualifier is used here to specify that only logical areas stored in physical areas 4 and 5 are to be displayed. Further, the /BRIEF qualifier specifies that a condensed tabular output format be used.

```
$ RMU /SHOW AIP /BRIEF MF_PERSONNEL /PAREA=(4,5)
*-----*
* Logical Area Name                LArea PArea   Len Type
*-----*
RDB$SYSTEM_RECORD                 60    4    215 SYSTEM RECORD
RDB$SYSTEM_RECORD                 61    5    215 SYSTEM RECORD
EMPLOYEES_HASH                    79    4    215 HASH INDEX
EMPLOYEES                          82    4    121 TABLE
JOB_HISTORY_HASH                  85    4    215 HASH INDEX
JOB_HISTORY                        88    4     42 TABLE
DEPARTMENTS_INDEX                 89    5    430 SORTED INDEX
DEPARTMENTS                       90    5     55 TABLE
```

Frequently asked Questions

Can RMU Set AIP be used on a database configured for HOT STANDBY? Yes. All AIP changes and SPAM rebuild actions are written to the after image journal and will be applied to the standby database.

Does the RMU Set AIP command apply to MIXED format areas? The command can be used to change the length in the AIP, however, THRESHOLDS for MIXED format areas are physical area attributes and are not supported at the logical area (aka AIP) level. Therefore, REBUILD_SPAMS is not applied to MIXED areas.

What happens if I accidentally change an attribute in the wrong AIP entry? This command will implicitly commit any changes with no opportunity to undo them using rollback. This is very powerful command and only an experienced database administrator should be using it. Access to the functionality is controlled by privileges at the RMU and Rdb database level. We suggest that RMU Show AIP be used prior to any change so that you can compare the results. Repair the AIP using the RMU Set AIP command.

Some wildcard operations are restricted to prevent accidental damage to the database. For instance, a wildcard matching many objects will be rejected if more than one type of object is being changed. If a wild card selects both table and index types then this command will be rejected.

Is this an offline command like RMU Repair SPAM? No. This command is an online command. Each logical area will be processed within a single transaction and interact with other online users.

What will happen to existing users of the table or index if I change the AIP length or thresholds online? The current users will start to use the new values if the logical areas are reloaded.

How do I know when to use RMU SET AIP/REBUILD_SPAM? Various ALTER commands will register changes for the AIP and these are applied at COMMIT time. Therefore, you may see a warning message issued from a COMMIT following an ALTER or TRUNCATE TABLE command. You can also use RMU/VERIFY to report any logical areas that require SPAM rebuilding, or examine the output from the RMU/DUMP/LAREA=RDB\$AIP or RMU/SHOW AIP commands.

What if I choose not to rebuild the SPAM pages? The fullness of some page will have been calculated using the old AIP length or THRESHOLD values. Therefore, it might appear that a page is full when in fact the revised length will fit on the page, or the page may appear to have sufficient free space to store a row but once accessed the space is not available. By rebuilding SPAM pages you may reduce I/O during insert operations. However, delaying the rebuild to a convenient time will not affect the integrity of the database.

How much I/O will be performed on the database when I rebuild SPAM pages using RMU Set AIP/REBUILD_SPAMS? Assuming just one logical area is affected then Oracle Rdb will read the ABM (Area Bitmap) to locate any SPAM pages in that area that reference this logical area. Rdb will then read each page and recalculate the fullness based on the rows stored. Therefore, the amount of I/O will depend upon the size of the table.

ORACLE

Oracle Rdb
Guide to Performance and Tuning: Managing the AIP Data Structure
August 2008

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com

Oracle Corporation provides the software
that powers the internet.

Oracle is a registered trademark of Oracle Corporation. Various
product and service names referenced herein may be trademarks
of Oracle Corporation. All other product and service names
mentioned may be trademarks of their respective owners.

Copyright © 2008 Oracle Corporation
All rights reserved.