

ORACLE®

RMU Enhancements

New qualifiers, commands and internal changes

Ian Smith
Oracle Rdb Product Architect
Oracle Rdb Engineering
October, 2014

Remote Speaker Photo



Ian Smith

Rdb Product Architect, Oracle

Program Agenda

- 1 ➤ Introduction
- 2 ➤ NSF Support
- 3 ➤ Review of recent RMU changes
- 4 ➤ Examples to highlight new features
- 5 ➤ Finish with Questions

Program Agenda

- 1 Introduction
- 2 NFS Support
- 3 Review of recent RMU changes
- 4 Examples to highlight new features
- 5 Finish with Questions

Program Agenda

- 1 Introduction
- 2 NFS Support
- 3 Review of recent RMU changes
- 4 Examples to highlight new features
- 5 Finish with Questions

Program Agenda

- 1 Introduction
- 2 NFS Support
- 3 Review of recent RMU changes
- 4 Examples to highlight new features
- 5 Finish with Questions

Program Agenda

- 1 Introduction
- 2 NFS Support
- 3 Review of recent RMU changes
- 4 Examples to highlight new features
- 5 Finish with Questions

Program Agenda

- 1 Introduction
- 2 NFS Support
- 3 Review of recent RMU changes
- 4 Examples to highlight new features
- 5 Finish with Questions

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

NFS – Network File System Support

Rdb Position

NFS

- Network file system supported on OpenVMS
- Presents virtual ODS device mapped to remote system
- Remote system might be a Linux or Windows system
- Limited support by Rdb due to restrictions of NFS

Active System Files

Requires sharing and locking support

- Active files are not support on NFS devices
- Database files actively used by Rdb
 - Database root file (.rdb)
 - Storage area files (.rda)
 - Cache backing store files (.rdc)
 - Snapshot files (.snp)
 - Recovery unit files (.ruj)
 - Active journals (.aij)

Archived Data Files

Supported by Rdb

- These files may reside (be written to) NFS devices
- Archival files
 - Database backup files (.rbf)
 - After Image Journal File backup
 - Export interchange files (.rbr)
 - Unloaded data files (.unl)
- Miscellaneous files
 - For example, RMU Restore Option file, or RMU Load Record Definition file (.rrd)

Rdb V7.3.1.2 or later

Improved support for NFS

- OpenVMS has a severe restriction on the read-ahead RMS attribute (RAH)
- Since the early 1990's Rdb has specified RAH attribute when reading files for better performance
- Inexplicably this attribute causes extreme slow down of all reads from NFS files
- A sample RMU/RESTORE from NFS device
 - Using RAH - 2 hours, 9 minutes and 21 seconds
 - Not using RAH - 5 seconds

Current Status

As of Wednesday, September 17, 2014

- Yes. HP knows about this (we reported it earlier this year)
- No. There is no correction at this time.
- RMU and SQL have been modified to open files without RAH if they reside on NFS devices
- Turns out OpenVMS BACKUP uses this same workaround
 - as we discovered after our own research

Sample TCPIP command for VMS

```
$ tcpip mount –  
    dnfs1:/host="test.company.com"/path="/scratch" –  
    /structure=5/serve=unix/adf/version=2/transport=udp
```

Some things we found during testing

Your own environment might need different settings

- Use /ADF=CREATE
 - This ensures that attributes (such as block size and record length) are preserved on the server
- Use /STRUCTURE=5
 - This will emulate an ODS-5 device and therefore allow the most complete OpenVMS Files-11 On-Disk Structure emulation
- Use /TRANSPORT=UDP
 - We just couldn't get TCP to work consistently
- Use /VERSION=2 or 3

This information is provided to answer customer questions and should not be read as an endorsement or guarantee for NFS systems. Oracle expects configuration, functional testing, performance testing, security and integrity of the NFS data to be performed by our customers.

– *Oracle Rdb Position Paper on NFS Support*

Show Statistics

Using P2 Space

- Moved the larger data structures into P2 space
- Avoids unexpected VASFULL errors

VASFULL Example

```
$ rmu/show statistics test_db
%COSI-F-VASFULL, virtual address space full
-SYSTEM-F-ILLPAGCNT, illegal page count parameter
%RMU-F-FATALOSI, Fatal error from the Operating System Interface.
%RMU-F-FTL_SHOW, Fatal error for SHOW operation at 26-AUG-2013 10:53:36.75
```

```
$ SQL$
SQL> attach 'filename test_db';
SQL> select max(rdb$logical_area_id) from rdb$logical_areas;
```

21987

1 row selected

This many logical areas requires
a lot of virtual memory

Missing Help for Fields

- Recently reported that some (possibly many) fields are missing help text
- RMU provides HELP for various fields on screen
- Menu option “H” selection “C” for **field descriptions** should bring up a help window similar to the following

Help text is shown

statistic.....	rate.per.second.....	total.....	average.....	ns....
----------------	----------------------	------------	--------------	--------

Statistics Help: 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.



fragmented	0	0	0.0	0	0.0
sequential scan	0	0	0.0	0	0.0
record fetched	0	0	0.0	0	0.0

Missing help text

statistic.....	rate.per.second.....			total.....	average.....
name.....	max.....	cur.....	avg.....	count.....	per.trans....
		0	0.0	0	0.0
		0	0.0	0	0.0
		0	0.0	0	0.0
		0	0.0	0	0.0
		0	0.0	0	0.0
		0	0.0	0	0.0
		0	0.0	0	0.0
discarded	0	0	0.0	0	0.0
record erased	0	0	0.0	0	0.0
fragmented	0	0	0.0	0	0.0
sequential scan	0	0	0.0	0	0.0
record fetched	0	0	0.0	0	0.0

Statistics Help: record marked

No help available

Missing Help for Fields

- RMU uses RMUDISPLAY73.HLB to lookup help text using the name of the screen and the name of the field
- An automated key builder uses the on-screen field text replacing spaces with “_” characters
- So in the prior example “record marked” becomes record_marked and “fragmented” becomes _fragmented. Sometimes keys look quite strange
“_____:_failure_field”

Missing Help for Fields

- If the on-screen field layout changes these built HELP keys no longer match the contents of the HELP library and the help is not displayed
- Rdb has fixed the known problems in 7.2.5.6 and 7.3.1.2
- If you observe this behavior Oracle suggests running RMU with the /Options=DEBUG qualifier
- The next slide shows the augmented “No Help available” display

Added details from /OPTION=DEBUG

statistic.....	rate.per.second.....	total.....	average.....
name.....	max..... cur..... avg.....	count.....	per.trans....
<div> <div>Statistics Help: record stored</div> <div> No help available Logical_Area_Statistics_screen record_stored_field </div> </div>			
	0	0.0	0
	0	0.0	0
	0	0.0	0
	0	0.0	0
	0	0.0	0
	0	0.0	0
	0	0.0	0
record erased	0	0	0.0
fragmented	0	0	0.0
sequential scan	0	0	0.0
record fetched	0	0	0.0

Reporting missing help

- Please contact Oracle Support
- Provide either a screen shot or a cut/paste of the No Help available box showing the screen and field names
- This will enable the Rdb team to quickly locate and fix future issues
- Note that this will work with most existing Rdb versions (as far back as I can remember)

Verify Constraints

Verify Constraints

- Constraint verify occurs within the Rdb Server, not RMU
- Same machinery used for /CONSTRAINTS verify as used by **alter table ... add constraint**
- Considerable re-work was done for V7.3.1

PRIMARY KEY and UNIQUE constraints

- Now verified using a rewritten query
- Previously the row by row query was used by verify
- New query avoids table self join
- In the absence of an index I/O performance should be considerably reduced

NOT NULL constraints

- Collect all NOT NULL checks, including that of PRIMARY KEY into single bitmap
- Verify uses a single table scan to check constraints
- Consider that in prior version there was one constraint evaluated for each NOT NULL column
- Should see improved performance (less I/O and reduced CPU)

Order other constraints by TABLE name

- Prior versions would read constraints ordered by constraint name
- This might mean constraints for one table were verified within other table access
- Now we take advantage of any buffered rows from prior constraint verify

Fallback mode

- All existing algorithms are accessible from RMU
- Use /CONSTRAINTS=FALLBACK

```
$ DEFINE/USER RDMS$SET_FLAGS ITEM_LIST
$ RMU/VERIFY/CONSTRAINTS=FALLBACK PERSONNEL
~H Extension (VERIFY CONSTRAINTS) Item List: (len=0)
~H: ...verify constraint "COLLEGE_CODE_REQUIRED"
~H: ...verify constraint "DEPT_CODE_REQUIRED"
~H: ...verify constraint "EMPLOYEE_ID_REQUIRED"
~H: ...verify constraint "JH_EMP_ID_EXISTS"
~H: ...verify constraint "JOB_CODE_REQUIRED"
~H: ...verify constraint "SH_EMP_ID_EXISTS"
~H: 6 tables processed.
$
```

Verify Key_Values

RMU Verify Key_Values

New deeper index verify

- RMU Verify Index
 - Verifies node and hash bucket structure for correctness
- RMU Verify Index Data
 - Additionally collects DBKEY from table and index and ensures a matching set
 - Too many from the table or too many from the index indicates a problem
- RMU Verify Key_Values
 - Reads table columns and compares with index keys

Key_Values

- Does more table and index scanning
- Extra I/O probably makes this a special case command rather than as part of regular maintenance
- Sorted indices use query specific outline to force index-only scan of the table
- Hash indices performed by doing exact key lookup

RMU Verify Index For_Tables

Selective table based index selection

- New FOR_TABLES qualifier used to restrict indices to the listed tables
- Allows smaller verify tasks to be created

```
$ RMU/VERIFY/INDEXES/FOR_TABLE=EMPLOYEES/NOLOG MF_PERSONNEL
```

Backup, Restore and Recover

Progress_Report

- New qualifier to BACKUP and RESTORE
- Shows current progress and for disk based actions shows estimates
- Defaults to 60 seconds
- The estimated completion time is recalculated using the current transfer rate
- For parallel backup each worker generates its own report line

Example

Ctrl/T for interactive process

```
$ rmu/restore/nocdd/dir=USERS2:[TESTING.DATABASE] [-]SAMPLE_DB.RBF/progress
RYEROX::Ian Smith ~? 18:03:08 RMU73   CPU=00:01:19.49 PF=21707 IO=111084 MEM=1160
Read  41 MB ( 6%) at  2 MB/s, estimated completion time 18:08:17.53
Read 278 MB (40%) at  5 MB/s, estimated completion time 18:04:58.26
Read 621 MB (89%) at  5 MB/s, estimated completion time 18:05:00.40
Read 691 MB (99%) at  1 MB/s, estimated completion time 18:05:48.21
%RMU-I-AIJRSTAVL, 0 after-image journals available for use
%RMU-I-AIJSOFF, after-image journaling has been disabled
%RMU-W-USERECCOM, Use the RMU Recover command. The journals are not available.
```

Periodic reports (every *n* seconds)

Set After_Journal /Backups

- Can now define the settings for compression
- Used by ABS (Automatic Backup Server)

```
$ rmu/set after_journal mf_personnel /backup=(automatic,compression=ZLIB:5)  
%RMU-I-LOGMODFLG,   enabled after-image journal spooler  
%RMU-I-LOGMODVAL,   modified AIJ backup compression level to 5
```

New Recover /NoAfter_Journal qualifier

- When an after-image journal is created or dropped a journal record is written to the active AIJ
- When restoring and then recovering the AIJ there may not be sufficient space for the new AIJ
- Requests that RMU ignore AIJ creation and drop during Recover

```
$ rmu/recover/noafter_journal/log mfp2.aij_bck,mfp3.aij_bck  
%RMU-I-LOGRECDB, recovering database file ...
```

Assist qualifier

For tape handling commands

- /Assist – RMU sends requests to the operator
- /NoAssist – these requests are sent to the current user
 - BACKUP
 - BACKUP/AFTER_JOURNAL
 - DUMP/BACKUP_FILE
 - DUMP/AFTER_JOURNAL
 - OPTIMIZE
 - RECOVER
 - RMU/RESTORE

Exit_Error qualifier

- RMU Dump Backup often used to verify the backup media
- New qualifier causes RMU to abort on first error
- The exit status will indicate the reason

```
$ rmu/dump/backup/exit_error MF_PERSONNEL.RBF
%RMU-I-DMPTXT_163, No dump option selected. Performing read check.
%RMU-E-BLOCKLOST, block of DEVICE:[DIRECTORY]MF_PERSONNEL.RBF;
    lost due to unrecoverable error
%RMU-F-FATALERR, fatal error on DUMP_BACKUP
%RMU-F-FTL_DUMP, Fatal error for DUMP operation at 1-MAY-2013 11:32:13.45
$ show symbol $STATUS
$STATUS == "%X12C8821A"
```

Audit, Dump Audit, Show Audit

Improved Management

Supporting more database objects

- Rdb V7.3 supports audit for **routines** and **sequences**
- Also support special classes: **views** and **modules**
- Setting AUDIT or ALARM on a module is a shorthand for auditing all routines (SQL or external) defined by the module
- This means using **alter module ... add** will not only add the new routine, but that routine will inherit the auditing of the module

DACCESS qualifiers

- DATABASE
- TABLE
- VIEW (sub-class of tables)
- COLUMN
- ROUTINE
- SEQUENCE
- MODULE (class of routines)

ENABLE and DISABLE qualifiers

- IDENTIFIERS now allows “*” as a wildcard
 - “*” means all roles and users
 - “[*,*]” means all users, or PUBLIC
 - “[group,*]” means all members of a group
 - “[*,member]” means all groups with this member
- Note: older versions ignore IDENTIFIER=*

RMU Set Audit

- In prior versions all tables, views and columns had to be fully specified
 - This made establishing auditing difficult
- In this release names can now include wildcards
- So all tables can be audited using a simple /TABLE=* qualifier
- Or related columns using /COLUMN=*.BADGE_ID

Wildcard specifications

- Note that wildcard specifications do not select:
 - System tables, system views, system sequences, system or hidden modules or system or hidden routines
 - (hidden objects includes OCI Services Data Dictionary objects)
 - Local routines (see USAGE IS LOCAL clause)

RMU Load Audit

- Loads the data from the audit journal
 - Use DCL SHOW AUDIT/JOURNAL to locate your journal
- RMU supports /BEFORE and /SINCE qualifiers so that a date/time range can be selected

```
$ RMU/LOAD/AUDIT -  
  /SINCE=&start_ts -  
  /BEFORE=&end_ts -  
  TESTDB AUDIT_RECORDS -  
  SYS$COMMON:[SYSMGR]SECURITY.AUDIT$JOURNAL  
%RMU-I-DATRECREAD, 91 data records read from input file.  
%RMU-I-DATRECSTO, 63 data records stored 27-NOV-2013 00:59:33.18.
```

Use symbols to pass range to RMU

RMU Dump Audit

- Audit record contains quite a lot more information that extracted by RMU Load Audit
- New RMU Dump Audit command extracts all Rdb specific audit records
- Output defaults to LIST format – a readable report
- Goal was to provide tool for long-term audit data archival
- The XML format allows loading all the data into a reporting database or other audit facility

Dump Audit Qualifiers

- BEFORE and SINCE to specify date/time range
- TYPE to select AUDIT, DACCESS, PROTECTION, and RMU
- FORMAT to choose between XML and LIST format
- OUTPUT to direct the output to a file

Unload and Load

New qualifiers and matching support

Limit_To

- Allows database administrator to restrict the size of the unloaded file
- Used internally to create sample test tables

Dialect

- Controls the SQL language dialect
- In some cases the format of the delivered text data changes by dialect
- For example, Oracle Database expects INTERVAL types to separate DAY from HOUR with a space

Default dialect

```
$ rmu/unload -  
  /limit_to=3 -  
  /virtual -  
  /record=(format=delimited,nofile) -  
  /field=salary_duration -  
  mf_personnel -  
  salary_history -  
  sys$output:  
" 00240:00"  
"-45348:00"  
" 00203:00"  
%RMU-I-DATRECUNL, 3 data records unloaded 29-SEP-2014 16:12:55.43.
```

Oracle dialect

```
$ rmu/unload -  
  /limit_to=3 -  
  /dialect=oracle -  
  /virtual -  
  /record=(format=delimited,nofile) -  
  /field=salary_duration -  
  mf_personnel -  
  salary_history -  
  sys$output:  
" 00240 00"  
"-45348 00"  
" 00203 00"  
%RMU-I-DATRECUNL,  3 data records unloaded 29-SEP-2014 16:12:55.54.
```

Oracle dialect for Format=Control

```
$ rmu/unload -  
  /limit_to=3 -  
  /dialect=oracle -  
  /virtual -  
  /record=(format=control,nofile) -  
  /field=salary_duration -  
  mf_personnel -  
  salary_history -  
  sys$output:  
000000240 00:00:00.000  
000000000 00:00:00.001  
000000203 00:00:00.000  
%RMU-I-DATRECUNL, 3 data records unloaded 29-SEP-2014 16:12:55.66.
```

Not needed for CONTROL

Format=CONTROL

- Generates control (.ctl) file suitable for use with SQL*Loader (sqlldr)
- Existed for a long time but limited Oracle Database type support
- Previous example demonstrates change for CONTROL
 - DATE VMS is unloaded as TIMESTAMP(2)
 - INTERVAL YEAR, and INTERVAL MONTH types are unloaded as INTERVAL YEAR(9) TO MONTH
 - All other day/time intervals are unloaded as INTERVAL DAY(9) TO SECOND(2)

Unload After_Journal Control

- Similar changes have been made to LogMiner™
- Now support unloading user data of type INTERVAL and made unloading date/time values easier

Load Replace_Rows

- Make use of new REPLACE statement
- If a PRIMARY KEY exists then Rdb will implicitly pre-delete the matching key before inserting the new row

Optimize=BITMAPPED_SCAN

- Allow the database administrator to suggest bitmapped scan when unloading data
- Most useful when unload from a complex view across tables with SORTED RANKED indices

Format=CSV

- Comma Separated Values
- This is a variant for the Format=DELIMITED option
- First row is unloaded as the column name
- Useful when loading data to spreadsheets (Excel, Numbers, etc) or other CSV consuming tools

Format=CSV

- Supported by RMU Load
- Combines Format=DELIMITED and Skip=1

Deletes_First

- RMU Unload After_Image
- All delete operations within each transaction will be returned before add/modify operations

Debug_options=OFFSET

- When creating a record definition (.rrd) file
- The database administrator can annotate the definition with field offset and length information

Example

```
$ rmu/unload-  
    /record=(file=SALARY_HISTORY) -  
    /debug=OFFSET-  
    PERSONNEL -  
    SALARY_HISTORY -  
    SALARY_HISTORY  
%RMU-I-DATRECUNL,    729 data records unloaded  
$ type SALARY_HISTORY.RRD  
DEFINE FIELD EMPLOYEE_ID DATATYPE IS TEXT SIZE IS 5.  
DEFINE FIELD SALARY_AMOUNT DATATYPE IS SIGNED LONGWORD SCALE -2.  
DEFINE FIELD SALARY_START DATATYPE IS DATE.  
DEFINE FIELD SALARY_END DATATYPE IS DATE.  
DEFINE RECORD SALARY_HISTORY.  
    EMPLOYEE_ID .                /* Offset = 0 Length = 5 */  
    SALARY_AMOUNT .              /* Offset = 5 Length = 4 */  
    SALARY_START .               /* Offset = 9 Length = 8 */  
    SALARY_END .                 /* Offset = 17 Length = 8 */  
END SALARY_HISTORY RECORD.      /* Total Length = 25 */  
$
```

Safe Harbor Statement

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.



Any questions?

Oracle Beehive Conferencing Client provides a chat area. Please ask questions there too.

Hardware and Software **Engineered to Work Together**

ORACLE®