

Tuning Modern Hardware for Modern Database Versions on Modern OpenVMS

Keith W. Hare

JCC Consulting, Inc.

September, 2010

Abstract

This is a collection of guidelines that JCC Consulting has gathered in 26 years of tuning some of the most demanding information technology situations on the planet.

- Some of the guidelines come from close association with clients; some come from technology audits of systems on the edge;
- some come from what we have learned with our tool which writes to diverse database and hardware platforms.

Topics

- **Increase buffer usage**
- Increase number of recovery buffers
- Specify index node sizes
- Use ranked indexes at least for duplicate sorted indexes
- **Increase buffer usage**
- Check lockidtbl and reshashtbl against locks and resources used
- Use a bunch of AIJs
- Use Fast Commit
- **Increase buffer usage**
- Set number of cluster nodes to 1
- Use global buffers and **increase buffer usage**
- Page Transfer via Memory
- Use row caches

Tuning Goals

- Reduce:
 - I/O
 - Reduce lock contention
- Tradeoffs
 - Increase memory usage
 - Increase CPU usage
 - Increase disk usage

Midrange Computing Environment

	2000	2010
System	ES40	BL870c I2
CPU	4 500 MHz Alpha EV6 CPUs	4 quad-core 1.6 GHz Itanium CPUs
Cache	8KB Instruction, 8KB Data	24MB
Bus/backplane	133 MHz	1333 MHz
Memory	8 GB	64 GB
Disks	Locally Attached SCSI	8GB Fibre to SAN
Random I/Os per second	500	10,000
Sequential I/Os	1500	25,000
Disk Space	700 GB	7,000 GB

Simple Benchmark

- Load 10,100 Employee records into the MF_Personnel employees table
 - Commit every five rows

```
rmu/load mf_personnel employees user:[dba]employee -  
/record=file=user:[dba]employee/commit=5/log
```
- Initial Test – Argo
 - ES40 with 4 500MHz EV56 CPUs, 12GB memory
 - 10 year old computer
 - 2Gb fibre to EVA4400
 - Plain database
- Final Test – Pandora
 - bl870c with 2 1.6GHz dual-core CPUs, 16GB memory
 - 2 year old computer
 - 4Gb fibre to EVA4400
 - Highly Modified database

Initial – Transaction Duration

Node: ARGO (1/1/16) Oracle Rdb V7.2-411 Perf. Monitor 31-AUG-2010 22:07:43.73
Rate: 3.00 Seconds Transaction Duration (Total) Elapsed: 00:18:18.08
Page: 1 of 1 JCC_ROOT:[KEITH.SQL_CLASS.MF_V72]MF_PERSONNEL.RDB;1 Mode: Online

```
-----  
Total transaction count:          2026  
Seconds   Tx.Count:   % #Complete:   % #Incomplete:   %  
0-< 1:      1886  93%      1886  93%      140   7% <- average = 0.4833  
1-< 2:      132   6%      2018  99%       8   1%  
2-< 3:        6   0%      2024  99%       2   1% <- 95th %ile = 2.00  
3-< 4:        1   0%      2025  99%       1   1%  
4-< 5:        1   0%      2026 100%       0   0%  
5-< 6:        0   0%      2026 100%       0   0%  
6-< 7:        0   0%      2026 100%       0   0%  
7-< 8:        0   0%      2026 100%       0   0%  
8-< 9:        0   0%      2026 100%       0   0%  
9-<10:       0   0%      2026 100%       0   0%  
10+++       0   0%      2026 100%       0   0%  
-----
```

Initial – I/Os

Node: ARGO (1/1/16) Oracle Rdb V7.2-411 Perf. Monitor 31-AUG-2010 22:07:48.57
 Rate: 3.00 Seconds Summary IO Statistics Elapsed: 00:18:22.92
 Page: 1 of 1 JCC_ROOT:[KEITH.SQL_CLASS.MF_V72]MF_PERSONNEL.RDB;1 Mode: Online

```

-----
statistic..... rate.per.second..... total..... average.....
name..... max..... cur..... avg..... count..... per.trans....
transactions          46          0          1.8          2026          1.0
verb successes        574          0         22.5         24806         12.2
verb failures          0          0          0.0           0          0.0

synch data reads      3677          0        1896.3        2087921        1030.5
synch data writes      239          0          1.1          1302           0.6
asynch data reads      268          0        114.6        126258         62.3
asynch data writes     521          0         55.5         61176         30.1
RUJ file reads         0          0          0.0           0           0.0
RUJ file writes       116          0         10.6         11772           5.8
AIJ file reads         0          0          0.0           0           0.0
AIJ file writes        0          0          0.0           0           0.0
root file reads        4          0          0.0           22           0.0
root file writes       49          0          1.9         2178           1.0
-----
  
```


Buffer Information

```
argo > rmu/dump/header=buffer mf_personnel
```

```
*-----  
* Oracle Rdb V7.2-411                                30-AUG-2010 16:11:30.07  
* Dump of Database header  
*      Database: JCC_ROOT:[KEITH.SQL_CLASS.MF_V72]MF_PERSONNEL.RDB;1  
*-----
```

Database Parameters:

Root filename is "JCC_ROOT:[KEITH.SQL_CLASS.MF_V72]MF_PERSONNEL.RDB;1"

Buffers...

- **Default user buffer count is 20**
- **Default recovery buffer count is 20**
- Global buffers are disabled
 - Global buffer count is 250**
 - Maximum global buffer count per user is 5**
 - Large memory is disabled
- **Buffer size is 6 blocks**
 - Maximum pages per buffer is 3**

...

Increase Buffer Usage

- Keep data in memory
- RDM\$BIND_BUFFERS
- ALTER DATABASE <db-root>
Number of buffers is <N>;
- Example:

```
SQL> alter database filename mf_personnel  
cont> number of buffers is 1000;
```

Increase Recovery Buffers

- Reduce time for Data Base Recovery (DBR) processes
- While a DBR is executing, all other attaches stalled (Freeze Lock)
- Should not be doing DBRs, but if you do, make them as fast as possible

- Example:

```
SQL> alter database filename mf_personnel  
cont> number of recovery buffers is 2500;
```

Sorted Indexes

- Bad Example:

```
SQL> show index emp_employee_id  
Indexes on table EMPLOYEES:  
EMP_EMPLOYEE_ID with column EMPLOYEE_ID  
  No Duplicates allowed  
  Type is Sorted  
  Key suffix compression is DISABLED  
  Node size 430
```

- Default node size of 430

- No storage map

- Stored in default storage area
- Often Rdb\$System

Specify Node Sizes & Storage

- Always specify an index node size and storage area
- Better Example:

```
SQL> create unique index EMP_EMPLOYEE_ID  
cont>      on EMPLOYEES (EMPLOYEE_ID asc)  
cont>      type is SORTED  
cont>      node size 480  
cont>      store in employees_indexes;
```

Reasonable Index Node Sizes

- Function of Page Size
- Frequency of Updates
 - AIJ and RUJ space
 - Lock Contention
- Number of rows and Depth of Index

Reasonable Index Node Sizes

Page Size	1 Node per Page	2 Nodes per Page	4 Nodes per Page	6 Nodes per Page	12 Nodes per Page
2	964	480			
4	1988	990	490		
6	3012	1500	740	488	
8	4036	2016	1000	664	
12		3040	1512	1000	488

Use Ranked Indexes

- All nodes are the same size
 - Eliminates discarded pages issues
 - Better performance for updates and deletes
- Bitmap AND and OR optimizations
- Example:

```
SQL> create index EMP_LAST_NAME  
cont>      on EMPLOYEES (LAST_NAME, FIRST_NAME asc)  
cont>      type is SORTED RANKED  
cont>      node size 480  
cont>      store in employees_indexes;
```


Increase Buffer Usage

- Memory usage can be increased by using a larger buffer size
- Doubling the buffer size doubles the memory used

- Example:

```
SQL> alter database filename mf_personnel  
cont>    buffer size is 12 blocks  
cont>    ;
```

Check lockidtbl and reshashtbl

Use Monitor Locks to see locks and resources used:

OpenVMS Monitor Utility
LOCK MANAGEMENT STATISTICS
on node JASON
30-AUG-2010 22:54:19.52

	CUR	AVE	MIN	MAX
New ENQ Rate	11128.16	14401.11	11128.16	17674.05
Converted ENQ Rate	373.98	433.07	373.98	492.17
DEQ Rate	11127.26	14397.71	11127.26	17668.15
Blocking AST Rate	0.00	0.00	0.00	0.00
ENQs Forced To Wait Rate	5.59	14.19	5.59	22.79
ENQs Not Queued Rate	0.00	0.19	0.00	0.39
Deadlock Search Rate	0.00	0.00	0.00	0.00
Deadlock Find Rate	0.00	0.00	0.00	0.00
Total Locks	255484.00	255480.50	255477.00	255484.00
Total Resources	238816.00	238810.00	238804.00	238816.00

System Lock Parameters

```
jason > mcr sysgen
```

```
SYSGEN> SHOW LOCKIDTBL
```

Parameter Name	Current	Default	Min.	Max.	Unit
LOCKIDTBL	822048	3840	1792	16776959	Entries

```
SYSGEN> SHOW RESHASHTBL
```

Parameter Name	Current	Default	Min.	Max.	Unit
RESHASHTBL	1048576	64	1	16777216	Entries

```
SYSGEN>
```

Recommended Values

- The values in this example are OK:

	Peak Usage	Sysgen Values	Recommendation
LOCKIDTBL	255,484	822,048	\geq Maximum Locks
RESHASHTBL	238,816	1,048,576	$\geq 2 * \text{Maximum Locks}$

- Changes to ModParams.dat

```
min_lockidtbl    = 500000
min_reshashtbl   = 1000000
```

Deadlock_Wait

Deadlock_Wait controls how long a lock request stalls before the VMS Lock manager searches for a deadlock

- Default of 10 seconds is too long
- 0 disables deadlock searches – never use
- 1 is reasonable for many systems

```
SYSGEN> SHOW DEADLOCK_WAIT
```

Parameter Name	Current	Default	Min.	Max.	Unit	Dynamic
DEADLOCK_WAIT	1	10	0	-1	Seconds	D

```
SYSGEN>
```

Use a Bunch of AIJs

- Enough AIJ space for several days without an AIJ backup
- Separate disk volume(s) from database disk volume(s)
- Large enough for infrequent AIJ switches
- Four 5,000 block AIJs is a good start for the sample personnel database
 - Use lots more AIJ space for a real database

AIJ Example

```
SQL> alter database filename mf_personnel
cont>   reserve 25 journals;
%RDMS-W-DOFULLBCK, full database backup should be done to ensure
future recovery
SQL> alter database filename mf_personnel
cont>   journal is enabled
cont>   (allocation is 5000 blocks )
cont>   add journal aij_01
cont>       filename 'jcc_user:[keith.sql_class.mf_v72.aij]mfp_aij_01'
cont>   add journal aij_02
cont>       filename 'jcc_user:[keith.sql_class.mf_v72.aij]mfp_aij_02'
cont>   add journal aij_03
cont>       filename 'jcc_user:[keith.sql_class.mf_v72.aij]mfp_aij_03'
cont>   add journal aij_04
cont>       filename 'jcc_user:[keith.sql_class.mf_v72.aij]mfp_aij_04'
cont> ;
%RDMS-W-DOFULLBCK, full database backup should be done to ensure
future recovery
```

Use Fast Commit

- Defers & reduces write I/O for most applications
- AIJ is critical
- Pay attention to FC Checkpoint parameters
- Prerequisite for Row Caches & Hot Standby
- Example:

```
SQL> alter database filename mf_personnel  
cont>     journal is enabled  
cont>     (fast commit is enabled)  
cont>     ;  
SQL>
```


Enable ALS

- Set AIJ Log Server to automatic

- Example:

```
SQL> alter database filename mf_personnel  
cont>      journal is enabled  
cont>      (log server is automatic)  
cont>      ;  
SQL>
```

Increase buffer usage

I haven't increased buffers for several slides...

- 12 block I/O used to be large
- Now, a 24 block I/O is a bit small
- Example:

```
SQL> alter database filename mf_personnel  
cont>    buffer size is 24 blocks;  
SQL>
```

- Maximum buffer size is 128 blocks (64K)

Number of Cluster Nodes is 1

- Distributed lock operations more expensive than local lock operations
- Transaction-related data structures cached in memory
- Reduce root file I/O
- Example:

```
SQL> alter database filename mf_personnel  
cont>    number of cluster nodes is 1;  
SQL>
```

Global Buffers and Increase Buffer Usage

- With global buffers, only one copy of a buffer is in memory
- Better memory utilization
- Reduce Read I/O
- Example:

```
SQL> alter database filename mf_personnel  
cont>      global buffers are enabled  
cont>      (number is 10000  
cont>      ,user limit is 500  
cont>      ,large memory is enabled);  
SQL>
```
- 10,000 global buffers is reasonable for the sample personnel database – real databases need more

Increase Minimum Page Size

- With Global Buffers, Rdb must track every possible page in the global buffer pool
- $\text{Number of Locks} = (\text{buffer size} / \text{minimum page size}) * \text{global buffers}$
- Memory used to manage the locks
- Maximum is 1,048,576 global buffers
- Use caution with large number of global buffers

Minimum Page Size Example

- 100,000 Global Buffers
- 24 Blocks per Buffer

Minimum Page Size	Maximum Pages per Buffer	Locks Required
1 (DO NOT USE)	24	2,400,000
2	12	1,200,000
4	6	600,000
6	4	400,000
8	3	300,000
12	2	200,000
24	1	100,000

Large Memory is Enabled

- Global buffer pool is placed in 64bit memory
- Better use of large amounts of memory

Open is Manual

- Manually open the database at application startup
- Do online backups
- Manually close the database at application shutdown
- Example:

```
SQL> alter database filename mf_personnel  
cont>      open is manual;  
SQL>  
jason > rmu/open mf_personnel
```


Shared Memory is System

- Database mapped once in System Address Space (S0S1) rather than once in each process
- VMS\$MEM_RESIDENT_USER rights identifier required

- Example:

```
SQL> alter database filename mf_personnel  
cont>      shared memory is system;  
SQL>
```

Page Transfer via Memory

- Database open on a single node
- Reduces write I/Os
- Example:

```
SQL> alter database filename mf_personnel  
cont>     global buffers are enabled  
cont>     (number is 10000  
cont>     ,user limit is 500  
cont>     ,large memory is enabled  
cont>     ,page transfer via memory) ;  
SQL>
```

Prestarted Transaction Timeouts

Rdb Prestarts read-write transactions when previous read-write transaction committed

- Can cause snapshot growth
- Pick a reasonable timeout
- No affect on read only transactions
- Example:

```
SQL> alter database filename mf_personnel  
cont>      prestared transactions are enabled  
cont>      (wait 15 seconds for timeout);  
SQL>
```

Use Row Caches

- Precision control of what is in memory
 - Additional effort
 - Additional gain
- Reduce I/O
- Reduce page locks
- Snapshots in cache
- Use row cached indexes instead of hashed indexes

Row Cache Example

```
SQL> alter database filename mf_personnel
cont>     reserve 30 cache slots
cont>     ;
%RDMS-W-DOFULLBCK, full database backup should be done to ensure future
recovery
SQL> alter database filename mf_personnel
cont>     add cache employees
cont>         cache size is 11000 rows
cont> --         row length is 121 bytes
cont>         row replacement is enabled
cont>         row snapshot is enabled
cont>         (cache size is 1000 rows) ;
SQL> alter database filename mf_personnel
cont>     add cache emp_employee_id
cont>         cache size is 513 rows
cont>         row length is 488 bytes
cont>         row replacement is enabled
cont>         row snapshot is enabled
cont>         (cache size is 100 rows) ;
SQL>
```

Buffers After Changes

```
pandor > rmu/dump/header=buffer mf_personnel
```

```
*-----  
* Oracle Rdb V7.2-400                                1-SEP-2010 00:06:05.04  
* Dump of Database header  
*      Database: JCC_ROOT:[KEITH.SQL_CLASS.MF_V72]MF_PERSONNEL.RDB;1  
*-----
```

Database Parameters:

Root filename is "JCC_ROOT:[KEITH.SQL_CLASS.MF_V72]MF_PERSONNEL.RDB;1"

Buffers...

- **Default user buffer count is 1000**
- **Default recovery buffer count is 2500**
- Global buffers are enabled
 - Global buffer count is 10000**
 - Maximum global buffer count per user is 500**
 - Large memory is enabled
- **Buffer size is 24 blocks**
 - Maximum pages per buffer is 12**

...

Final – Transaction Duration

Node: PANDOR (1/1/1) Oracle Rdb V7.2-410 Perf. Monitor 1-SEP-2010 10:24:58.49
Rate: 3.00 Seconds Transaction Duration (Total) Elapsed: 00:01:06.92
Page: 1 of 1 JCC_ROOT:[KEITH.SQL_CLASS.MF_V72]MF_PERSONNEL.RDB;1 Mode: Online

```
-----  
Total transaction count:          2024  
Seconds   Tx.Count:   %   #Complete:   %   #Incomplete:   %  
0-< 1:      2024 100%      2024 100%          0   0% <- avg=0.0068 95th=0.0  
1-< 2:         0   0%      2024 100%          0   0%  
2-< 3:         0   0%      2024 100%          0   0%  
3-< 4:         0   0%      2024 100%          0   0%  
4-< 5:         0   0%      2024 100%          0   0%  
5-< 6:         0   0%      2024 100%          0   0%  
6-< 7:         0   0%      2024 100%          0   0%  
7-< 8:         0   0%      2024 100%          0   0%  
8-< 9:         0   0%      2024 100%          0   0%  
9-<10:        0   0%      2024 100%          0   0%  
10+++       0   0%      2024 100%          0   0%  
-----
```

Final – I/Os

Node: PANDOR (1/1/1) Oracle Rdb V7.2-410 Perf. Monitor 1-SEP-2010 10:25:01.28
 Rate: 3.00 Seconds Summary IO Statistics Elapsed: 00:01:09.70
 Page: 1 of 1 JCC_ROOT:[KEITH.SQL_CLASS.MF_V72]MF_PERSONNEL.RDB;1 Mode: Online

```

-----
statistic..... rate.per.second..... total..... average.....
name..... max..... cur..... avg..... count..... per.trans....
transactions          266          0          29.2          2027          1.0
verb successes        3310          0         358.7         24891         12.2
verb failures          0          0          0.0           0          0.0

synch data reads       24          0          2.1          148          0.0
synch data writes      26          0          3.8          266          0.1
asynch data reads       9          0          1.3           92          0.0
asynch data writes     15          0          2.1          148          0.0
RUJ file reads          0          0          0.0           0          0.0
RUJ file writes         0          0          0.0           1          0.0
AIJ file reads          1          0          0.2           15          0.0
AIJ file writes        266          0          29.2         2031          1.0
root file reads         6          0          0.6           48          0.0
root file writes       283          0         31.3         2176          1.0
-----
  
```


Performance Change

	Initial (ES40)	Final (ES40)	Final (b1870c)
Average Transaction Duration	0.4833	0.0244	0.0068
I/O Per Transaction	1,156.3	2.1	2.1

What else is there?

- RMU/Collect Optimizer_Statistics
- Use enough indexes, but not too many
- Use multi-segment sorted indexes to achieve index-only queries
- Use sequences to generate unique IDs
 - Use randomize characteristic to avoid chronological index issue
- Database backup compression & encryption
- Upgrade – new performance benefits in new Rdb versions

Summary

- Systematic attention to indexes, buffers, caches, etc. will:
 - Reduce I/O
 - Reduce Transaction Duration
 - Reduce locking
- Increase System Memory
- Increase buffer usage

Questions?



<http://www.jcc.com/>

Join the worldwide Rdb community.
Send mail to

OracleRdb-request@JCC.com

with “SUBSCRIBE” in the body of
the message.

For more information send mail to info@jcc.com