A white paper prepared with Oracle and IBM technical collaboration
Version 1.4

Oracle Real Application Clusters on IBM AIX – Best practices in memory tuning and configuring for system stability

Executive Overview

IBM and Oracle have a long standing history of technical collaboration to enhance Oracle products on IBM Power™ Systems. This joint commitment to support the Power System platform's performance, scalability and clustering capabilities allows customers to effectively deploy Oracle Real Application Clusters (RAC) on a broad range of configurations. To support customers who choose to deploy Oracle RAC on IBM Power Systems servers IBM and Oracle have prepared this recommendation of best practices for managing memory use and setting key system parameters on AIX® systems running Oracle RAC. Implementing these recommendations will provide customers with the best possible availability, scalability and performance of the Oracle Database, and reduce the potential for node evictions due to memory over commitment. Key elements of the best practices are tuning AIX and Oracle RAC, and monitoring the system resources to insure memory is not over committed.

Introduction

Customers who experience Oracle Real Application Clusters (RAC) node evictions due to excessive AIX kernel paging should carefully review and implement these recommended best practices. Testing and experience have found that memory over commitments may cause scheduling delays for Oracle's 'oprocd' process in Oracle RAC versions prior to 11.2 which may result in node evictions. Implementing all of these recommendations will reduce scheduling delays and corresponding oprocd initiated evictions for Oracle RAC versions prior to 11.2. For Oracle RAC versions 11.2 and later, implementing all of these recommendations will ensure optimal performance and scalability.

Problem Validation

This paper addresses the best practices for environments experiencing node evictions caused by critical processes not being able to get scheduled in a timely fashion on AIX due to memory over commitment. To validate that node evections are caused by this situation, the following validation steps should be taken.

Examining the AIX Error Logs

For Oracle RAC 10g Release 2 and Oracle RAC 11g Release 1:

When an Oracle RAC cluster node is rebooted for cluster integrity this can be done by several cluster processes. In Oracle RAC versions prior to 11.2, when a node gets rebooted due do scheduling problems, the process, which would initiate the reboot, is oprocd. The first step in validating that scheduling delays caused a node reboot is to confirm if the oprocd process rebooted the node. When the oprocd process reboots the node there should be only one entry in the output of 'errpt –a' created at the time of the reboot. The error created in the AIX Error Logging subsystem should look similar to the following example:

```
LABEL:          REBOOT_ID
IDENTIFIER:     2BFA76F6
Date/Time:       Thu Apr 23 16:00:58 PDT 2009
Sequence Number: 10
Machine Id:      00C47AAC4C00
Node Id:         racha908
Class:           S
Type:            TEMP
Resource Name:   SYSPROC
Description
SYSTEM SHUTDOWN BY USER
Probable Causes
SYSTEM SHUTDOWN
Detail Data
USER ID
           0
0=SOFT IPL 1=HALT 2=TIME REBOOT
           0
TIME TO REBOOT (FOR TIMED REBOOT ONLY)
           0
```

There should not be a 'SYSDUMP' entry in the 'errpt –a' around the time of the reboot since 'oprocd' does not initiate a 'SYSDUMP'. A 'SYSDUMP' entry is an indication that other problems may be the root cause of node reboots.

For Oracle RAC 11g Release 2:

In Oracle RAC 11g Release 2, severe operating system scheduling issues are detected by the Oracle cssdagent and cssmonitor processes and the node is rebooted. The error created in the AIX Error Logging subsystem should look similar to the following example:

```
LABEL:          REBOOT_ID
IDENTIFIER:     2BFA76F6
Date/Time:       Tue May  8 17:27:14 2012
Sequence Number: 6565
Machine Id:      00F617D74C00
Node Id:         rac94
Class:           S
Type:            TEMP
WPAR:            Global
Resource Name:   SYSPROC
Description
SYSTEM SHUTDOWN BY USER
Probable Causes
SYSTEM SHUTDOWN
Detail Data
USER ID
           0
0=SOFT IPL 1=HALT 2=TIME REBOOT
           0
TIME TO REBOOT (FOR TIMED REBOOT ONLY)
           0
```

Oracle logs indicating a reboot initiated by Oracle

For Oracle RAC 10g Release 2 and Oracle RAC 11g Release 1:

It needs to be noted that to be able to analyze the oprocd log files to identify reboots due to AIX scheduling delays, all required Oracle patches should be in place (recommendation #3 below).

When the required patches are in place, a clear message in the most recent /etc/oracle/oprocd/<node>.oprocd.lgl.<time stamp> file should also show that oprocd rebooted the node.

```
Apr 23 16:00:04.327787 | LASTGASP | AlarmHandler:  timeout(11917 msec) exceeds
interval(1000 msec)+margin(10000 msec).   Rebooting NOW.

Oracle Support Data - Previous intervals (LIFO):
999ms
999ms
1000ms

<< Note: Redundant lines deleted from *lgl* file >>
```

In this example the oprocd margin was already increased from 500 ms to 10 seconds, according to recommendation #2 in this paper. This example shows a system being rebooted due to a scheduling delay of 11.9 seconds. This delay exceeded the sum of the oprocd intervals (1 second) and the oprocd margin (10 seconds). Therefore the reboot was initiated.

For Oracle RAC 11g Release 2:

Either the Oracle cssd agent or cssdmonitor process (or both) will generate a last gasp message showing that a reboot has been initiated. The contents of the files will be similar to the following one: /etc/oracle/lastgasp/cssagent_<node>.lgl

```
OLG01,0308,0000,rac-cluster,5c0fe986ffea4fc4ff8203359c15d48e,rac94,cssagent,L-
2012-05-08-18:40:17.875,Rebooting

after limit 28083 exceeded; disk timeout 27534, network timeout 28083, last
heartbeat from CSSD at epoch seconds

 1336527589.752, 28124 milliseconds ago based on invariant clock value of
1166013910
```

/etc/oracle/lastgasp/cssmonit_<node>.lgl

```
OLG01,0308,0000,rac-cluster,5c0fe986ffea4fc4ff8203359c15d48e,rac94,cssmonit,L-
2012-05-08-18:40:18.011,Rebooting

after limit 28083 exceeded; disk timeout 27534, network timeout 28083, last
heartbeat from CSSD at epoch seconds

 1336527589.752, 28260 milliseconds ago based on invariant clock value of
1166013910
```

Recommendations for System Stability

The following recommendations will help to ensure AIX systems running Oracle RAC will run in a stable and reliable manner. Unless otherwise indicated, all recommendations apply to AIX 5.2, AIX 5.3, AIX 6.1 and AIX 7.1.

1. Implement AIX tuning recommendations for Oracle

The first step in removing scheduling delays due to AIX kernel paging is to make sure a system has the correct AIX vmo parameters set. The following parameters should be verified and set on all Oracle RAC nodes.

```
vmo -p -o maxperm%=90;
vmo -p -o minperm%=3;
vmo -p -o maxclient%=90;
vmo -p -o strict_maxperm=0;
vmo -p -o strict_maxclient=1;
vmo -p -o lru_file_repage=0;
vmo -r -o page_steal_method=1;
chdev –l sys0 –a ‘minpout=4096 maxpout=8193’;
```

(Note: 8193 is the correct value here, although, ‘8192” might look more reasonable.)

These options provide the best system behavior for Oracle workloads, regardless if there are one or more Oracle DB instances active in the LPAR. Appendix B provides an example script to set these parameters.

Enable write/commit behind for remote JFS and JFS2 file systems that are NFS mounted on a RAC node and used for backups. This allows AIX to detect when a file is being written serially, and so the modified pages can be flushed to the paging device in smaller chunks rather than (potentially) all at one time,. This will slow down individual write operations slightly (for that specific file system) but has a system performance benefit similar to I/O pacing. For NFS file systems, the file system should be mounted with the options shown in the following example.

```
mount –o combehind,numclust=128
<remote node>:<remote file system> <local mount point>
```

The "mount" command can then be used to confirm that the options are in use for the file system.

To allow the most efficient utilization of processors and cache for Oracle Database workloads users are advised to use the default dynamic capabilities of the kernel scheduler. With these capabilities enabled, the system will increase and decrease the use of virtual processors in conjunction with the instantaneous load of the partition, as measured by the physical utilization of the partition. This dynamic behavior utilizes processor folding to reduce the number of virtual processors during periods of low utilization, which improves memory locality and virtualization efficiency for better overall system performance. For heavy workloads the kernel scheduler will increase the number of virtual processors up to the maximum set in the LPAR configuration and reported as Online Virtual CPUS.

To confirm that dynamic behavior is enabled, use the following command and verify it returns zero:

```
schedo -o vpm_xvcpus
```

IBM and Oracle recommend all customers permanently reset the schedo tunable parameters to the defaults by entering:

```
schedo -p -D
```

For more information on AIX 7.1, see:

http://publib.boulder.ibm.com/infocenter/aix/v7r1/index.jsp?topic=/com.ibm.aix.prftungd/doc/prftungd/virtual_proc_mngmnt_part.htm

Or for AIX 6.1 see:

http://publib.boulder.ibm.com/infocenter/aix/v6r1/index.jsp?topic=/com.ibm.aix.prftungd/doc/prftungd/virtual_proc_mngmnt_part.htm

Or for AIX 5.3:

http://publib.boulder.ibm.com/infocenter/pseries/v5r3/index.jsp?topic=/com.ibm.aix.prftungd/doc/prftungd/virtual_proc_mngmnt_part.htm

Users should also review all the tuning recommendations in the document
"Tuning IBM AIX 5.3 and AIX 6.1 for Oracle Database" which is available under:
http://public.dhe.ibm.com/partnerworld/pub/whitepaper/162b6.pdf

2. Modify the Oracle ‘diagwait’ parameter

For Oracle RAC 10g Release 2 and Oracle RAC 11g Release 1:

The oprocd process runs with two important parameters: interval and margin. The interval parameter is the time that the process sleeps before trying to get scheduled again. The default interval is one second. The margin parameter is the time, which oprocd is allowed to deviate from the scheduling interval time. The default is 500 milliseconds. To make the oprocd process less sensitive to scheduling delays, the margin can be increased to 10 seconds.

Changing the Clusterware parameter diagwait to 13 is the Oracle supported technique to change the oprocd margin to 10 seconds.

Please note that 13 is the only allowed value for setting the diagwait parameter to. Any value other than 13 (or unset) is not allowed and not supported.

Procedure for changing the Clusterware parameter diagwait

1. Stop Oracle Clusterware on ALL cluster nodes by executing the following command on each node as the root user.

```
#crsctl stop crs

#<CRS_HOME>/bin/oprocd stop
```

2. Ensure that Clusterware stack is down by running the “ps” command. Executing this command should return no processes on any of the cluster nodes.

```
      #ps -ef |egrep "crsd.bin|ocssd.bin|evmd.bin|oprocd"
```

3. From one node of the cluster, change the value of the "diagwait" parameter to 13 seconds by issuing the command as root:

```
      #crsctl set css diagwait 13 -force
```

4. Confirm that diagwait is set successfully by executing the following command. The command should return 13. If diagwait is not set, the following message will be returned "Configuration parameter diagwait is not defined"

```
      #crsctl get css diagwait
```

5. Restart the Oracle Clusterware by running the following command on all the nodes:

```
      #crsctl start crs
```

6. Validate that the node is running by executing:

```
      #crsctl check crs
```

An example of setting diagwait to 13:

```
oratest@racha906 /home/oratest > crsctl set css diagwait 13
Configuration parameter diagwait is now set to 13.

oratest@racha906 /home/oratest > crsctl get css diagwait
13oratest@racha906 /home/oratest >
```

Note that the new value for 'diagwait' is 13, but printed with no carriage return on the line.

When starting up the Oracle Clusterware stack again after we change the diagwait to 13, we will see the following log lines in the oprocd log file /etc/oracle/oprocd/<node>.oprocd.log:

```
Apr 23 16:04:34.413 | INF | monitoring started with timeout(1000), margin(10000),
skewTimeout(250)
Apr 23 16:04:34.495 | INF | fatal mode startup, setting process to fatal mode
Apr 23 16:04:40.385 | INF | enabling fatal mode as per client request
```

Note that now the margin is set to 10000.

For more information on this topic refer to Oracle's MetaLink Docid number 559365.1 "Using Diagwait as a diagnostic to get more information for diagnosing Oracle Clusterware Node evictions."

For Oracle RAC 11g Release 2:

Beginning with Oracle Clusterware 11.2.0.1, the oprocd fencing mechanisms and associated diagwait functionality have been re-architected. As a consequence, setting diagwait will have no effect.

3. Install the required updates and patches

To further reduce the risk of memory over-commitment causing delays in process scheduling which may cause evictions, the following Oracle and AIX software versions should be used.

Required patches and updates for Oracle RAC:

Install the correct patch sets and recommended patch bundles for Oracle Clusterware and Oracle RAC. The following is a list of the minimal required Patch Set levels; higher Patch Set levels can be used instead.

- For Oracle 10*g* Release 2
  - 10*g* Release 2 (10.2.0.4) Patch Set 3 Patch: 6810189
  - Recommended RAC Bundle 3 Patch: 8344348
  - Recommended CRS Bundle 3 Patch: 7715304
  - If running AIX 5.2 ML07 or a higher ML then also install the Oracle patch for Bug 7321562
    - Note, AIX 5.2 entered extended support in April 2009, customers are encouraged to upgrade to AIX 5.3 or AIX 6.1.
  - Fix for Bug 13940331: VALUE FOR SETTING THREAD SCHEDULING IS INCORRECT IN SLTSTSPAWN
  - Fix for Bug 13623902: NODE EVICTIONS ON RAC CLUSTER AFTER EXCESSIVE PAGING

- For Oracle RAC 11*g* Release 1
  - 11*g* Release 1 (11.1.0.7) Patch Set 1 Patch: 6890831
  - Fix for Bug 13940331: VALUE FOR SETTING THREAD SCHEDULING IS INCORRECT IN SLTSTSPAWN
  - Fix for Bug 13623902: NODE EVICTIONS ON RAC CLUSTER AFTER EXCESSIVE PAGING

- Oracle RAC 11*g* Release 2
  - Fix for Bug 13940331: VALUE FOR SETTING THREAD SCHEDULING IS INCORRECT IN SLTSTSPAWN

These patch bundles contain fixes that will:

- Insure that all 'oprocd' messages are being logged and saved in the /etc/oracle/oprocd/*log* and /etc/oracle/oprocd/*lgl* files.
- Allow histogram data of 'oprocd' delays to be collected in /etc/oracle/oprocd/*log* to help in evaluating the severity of the 'oprocd' delay problem.
- Insure that CRS and 'oprocd' are running with the correct priority, scheduling policy, and pinning memory correctly.

Required patch set and updates for AIX:

- For AIX 6.1
    - Follow the AIX requirements in the Oracle Release notes located in the Oracle Database Documentation Library.
        - Oracle 10*g* Release 2 release notes are B19074-11 – or higher
        - Oracle 11*g* Release 1 release notes are B32075-07 – or higher
- For AIX 5.3
    - AIX 5.3 ML06 - or higher ML, in addition to the AIX requirements in the Oracle Release notes.
- For AIX 5.2
    - Follow the AIX requirements in the Oracle Release notes.

In general, please review the following My Oracle Support Notes for the latest Oracle and AIX patch updates:

- DocID 756671.1 – Oracle Recommended Patches -- Oracle Database
- DocID 811293.1 –
  RAC Assurance Support Team: RAC Starter Kit and Best Practices (AIX)

4. Validating the Oracle process configuration

For Oracle RAC 10g Release 2 and Oracle RAC 11g Release 1:

After the installation of the required Oracle patches and AIX patches, verify thread priority:

- oprocd.bin will have a priority (PRI) of '1 or 0' and a scheduling policy (SCH) of '2'.
  - Note, the PRI value for oprocd will depend on the Oracle version and patches applied:
  - for CRS 10.2.0.4 before bundle 6 is applied: 1
  - for CRS 10.2.0.4 after bundle 6 is applied: 0
  - for CRS 11g Release 1: 0
- ocssd.bin should have a priority (PRI) of '0' and a scheduling policy (SCH) of '-'.
- Furthermore, all ocssd.bin threads should have a priority of '0'.
- All 'ora_lms' processes should have a priority (PRI) of '39' and a scheduling policy (SCH) of '--' or '2'.

Priority and scheduling policy for the key processes can be verified as follows:

```
 > ps -ef  -o pid,pri,sched,nice,args |egrep 'oprocd|ora_lms|ocssd|COMMAND' |grep
-v grep |sort +4
    PID PRI SCH NI COMMAND
 503810  40   0  0 /bin/sh -c ulimit -c unlimited; cd
/oratest/Vtest/CRS905/log/racha905/cssd; /oratest/Vtest/CRS905/bin/ocssd  || exit
$?
 974890  60   0 20 /bin/sh /etc/init.cssd oprocd
 602306   0   - -- /oratest/Vtest/CRS905/bin/ocssd.bin
 962684   0   2 -- /oratest/Vtest/CRS905/bin/oprocd.bin run -t 1000 -m 10000 -hsi
5:10:50:75:90 –f
 622646  39   2 -- ora_lms0_swing1
 700516  39   2 -- ora_lms0_tpch5
 577726  39   2 -- ora_lms1_swing1
 688216  39   2 -- ora_lms1_tpch5
 827452  39   2 -- ora_lms2_swing1
 585824  39   2 -- ora_lms2_tpch5
1056948  39   2 -- ora_lms3_swing1
 422014  39   2 -- ora_lms3_tpch5
```

Also note that when all patches are in place, the process names will have changed.

$CRS_HOME/bin/oprocd ➔ $CRS_HOME/bin/oprocd.bin run -t 1000 -m 10000 -hsi 5:10:50:75:90 –f

$CRS_HOME/bin/ocssd ➔ $CRS_HOME/bin/ocssd.bin

All ocssd.bin threads should also have a priority (PRI column) of '0'. Threads with a state (S column) of canceled ('Z') can be ignored. This can be verified by running the following command.

```
> ps -p `ps -ef | grep ocssd.bin | grep -v grep |awk '{print $2}'` -mo THREAD

    USER      PID     PPID       TID S  CP PRI SC    WCHAN        F     TT BND
COMMAND

 grid11g  7667752  8323258         - A   7   0 23        * 10240103      -   -
/grid/bin/ocssd.bin

       -        -        -   9896021 Z   0   0  1        -   c00001      -   - -

       -        -        -  20578455 S   0   0  1 f1000f0a10013a40  8410400      -
- -

       -        -        -  26476641 S   3   0  1        -   418400      -   - -

       -        -        -  27001027 S   0   0  1 f1000f0a10019c40  8410400      -
- -

       -        -        -  31719667 S   0   0  1 f1000f0a1001e440  8410400      -
- -

       -        -        -  33161241 S   0   0  1 f1000f0a1001fa40  8410400      -
- -

       -        -        -  33947683 S   0   0  1 f1000f0a10020640  8410400      -
- -

       -        -        -  34013213 S   0   0  1 f1000f0a10020740  8410400      -
- -

       -        -        -  34078751 S   0   0  1        -   418400      -   - -

       -        -        -  34144289 S   2   0  1        -   418400      -   - -

       -        -        -  34799679 S   0   0  1 f1000f0a10021340  8410400      -
- -

       -        -        -  34865247 S   0   0  1 f1000f0a10021440  8410400      -
- -
```

```
       -        -        -  34930749 S   0   0  1        -   418400      -   - -
       -        -        -  34996289 S   0   0  1 f1000f0a10021640  8410400      -
- -
       -        -        -  38273199 S   0   0  1 f1000f0a10024840  8410400      -
- -
       -        -        -  40304879 S   0   0  1 f1000f0a10026740  8410400      -
- -
       -        -        -  40435939 R   2   0  0        -   400000      -   - -
       -        -        -  51904581 R   0   0  1        -   410400      -   - -
       -        -        -  52494425 S   0   0  1 f1000f0a10032140  8410400      -
- -
       -        -        -  58654829 Z   0  60  1        -   c00001      -   - -
       -        -        -  66519079 Z   0  60  1        -   c00001      -   - -
       -        -        -  69271721 S   0   0  1 f1000f0a10042140  8410400      -
- -
       -        -        -  71500013 S   0   0  1 f1000f0a10044340  8410400      -
- -
       -        -        -  76153005 S   0   0  1 f1000f0a10048a40  8410400      -
- -
```

In addition, oprocd.bin and ocssd.bin should have the majority of their memory pinned. This should be verified using the following command. The "Pin" value should be close to the size of the "Inuse" value.

```
> ps -elf -o "pid,args"|egrep "oprocd|ocssd.bin" |grep -v grep |awk '{ print $1 }'
|xargs svmon -P |egrep "oprocd|ocssd.bin|Command
     Pid Command          Inuse      Pin     Pgsp  Virtual 64-bit Mthrd  16MB
  868420 ocssd.bin        91412    82879        0    91114      Y     Y     N
     Pid Command          Inuse      Pin     Pgsp  Virtual 64-bit Mthrd  16MB
  294970 oprocd.bin       81510    73242        0    81487      Y     N     N
     Pid Command          Inuse      Pin     Pgsp  Virtual 64-bit Mthrd  16MB
```

For Oracle RAC 11g Release 2:

After the installation of the required Oracle patches and AIX patches, verify thread priority:

- ocssd.bin, cssdagent, cssdmonitor, and osysmond.bin should have a priority (PRI) of '0' and a scheduling policy (SCH) of '-'.
- Furthermore, all threads for ocssd.bin, cssdagent, and cssdmonitor should have a priority of '0'.
- All 'ora_lms' processes should have a priority (PRI) of '39' and a scheduling policy (SCH) of '--' or '2'.

Priority and scheduling policy for the key processes can be as follows:

```
 >  ps -ef -o pid,pri,sched,nice,args |egrep
'osysmond|ocssd|ocssd|cssdagent|cssdmonitor|ora_lms|COMMAND' |grep -v grep |sort
+5

     PID PRI SCH NI COMMAND

 3014718   0   - -- /grid/bin/cssdmonitor

 4849720   0   - -- /grid/bin/cssdagent

 6488312   0   - -- /grid/bin/ocssd.bin

11731226  39   2 -- ora_lms1_oastdb_3

12583018  39   2 -- ora_lms0_oastdb_3

12976370   0   - -- /grid/bin/osysmond.bin

15860124   0   0 -- /bin/sh /grid/bin/ocssd
```

All threads for ocssd.bin, cssdagent, cssdmonitor, and osysmond should also have a priority (PRI column) of '0'. Threads with a state (S column) of canceled ('Z') can be ignored. This can be verified by running the following command.

```
> for P in `ps –ef |egrep 'ocssd.bin|cssdagent|cssdmonitor|osysmond' |awk '{print
$2}' ` ; do ps -T $P -mo THREAD ; done

    USER      PID     PPID       TID S  CP PRI SC    WCHAN        F     TT BND
COMMAND

    root  3932300        1         - A   5   0 13        * 10240103      -   -
/grid/b

       -        -        -  28901503 Z   0  60  1        -   c00001      -   - -

       -        -        -  36438111 S   0   0  1 f1000f0a10022c40  8410400      -
- -
```

```
       -        -        -  36503645 S   5   0  1 f1000a0200a58db0   410400      -
- -
       -        -        -  36569183 Z   0  60  1        -   c00001      -   - -
       -        -        -  36634721 S   0   0  1 f1000f0a10022f40  8410400      -
- -
       -        -        -  38207633 S   0   0  1        -   418400      -   - -
       -        -        -  42074117 Z   0   0  1        -   c00001      -   - -
       -        -        -  42139655 S   0   0  1        -   418400      -   - -
       -        -        -  42205193 S   0   0  1 f1000a0200a572b0   410400      -
- -
       -        -        -  42270731 S   0   0  1 f1000a0200a5d9b0   410400      -
- -
       -        -        -  42336269 S   0   0  1        -   418400      -   - -
       -        -        -  42401807 S   0   0  1 f1000f0a10028740  8410400      -
- -
       -        -        -  42467345 S   0   0  1 f1000f0a10028840  8410400      -
- -
    USER      PID     PPID       TID S  CP PRI SC    WCHAN        F     TT BND
COMMAND
 grid11g  6553600  9175098         - A   4   0 22        * 10240103      -   -
/grid/b
       -        -        -  24314099 S   0   0  1 f1000f0a10017340  8410400      -
- -
       -        -        -  24707083 S   0   0  1        -   418400      -   - -
       -        -        -  40698077 S   0   0  1 f1000f0a10026d40  8410400      -
- -
       -        -        -  40829155 S   0   0  1 f1000f0a10026f40  8410400      -
- -
       -        -        -  40894689 S   0   0  1 f1000f0a10027040  8410400      -
- -
       -        -        -  40960227 S   0   0  1        -   418400      -   - -
       -        -        -  41025765 S   0   0  1        -   418400      -   - -
       -        -        -  42532883 S   0   0  1 f1000f0a10028940  8410400      -
- -
       -        -        -  42663959 S   2   0  1        -   418400      -   - -
       -        -        -  42729497 S   1   0  1        -   418400      -   - -
       -        -        -  42795041 S   1   0  1        -   418400      -   - -
       -        -        -  42860577 S   0   0  1 f1000f0a10028e40  8410400      -
- -
       -        -        -  42926113 S   0   0  1 f1000f0a10028f40  8410400      -
- -
       -        -        -  42991651 S   0   0  1 f1000f0a10029040  8410400      -
- -
       -        -        -  43188273 S   0   0  1 f1000f0a10029340  8410400      -
- -
       -        -        -  43384879 S   0   0  1 f1000f0a10029640  8410400      -
- -
       -        -        -  43450417 S   0   0  1 f1000f0a10029740  8410400      -
- -
       -        -        -  43515957 S   0   0  1 f1000f0a10029840  8410400      -
- -
       -        -        -  43581493 S   0   0  1 f1000f0a10029940  8410400      -
- -
       -        -        -  43647037 S   0   0  1 f1000f0a10029a40  8410400      -
- -
       -        -        -  43712569 S   0   0  1 f1000f0a10029b40  8410400      -
- -
       -        -        -  44040259 Z   0  60  1        -   c00001      -   - -
    USER      PID     PPID       TID S  CP PRI SC    WCHAN        F     TT BND
COMMAND
    root  9306214        1         - A   0   0 17        * 10240103      -   -
/grid/b
       -        -        -  36110417 S   0   0  1        -   418400      -   - -
       -        -        -  36765797 S   0   0  1 f1000f0a10023140  8410400      -
- -
```

```
       -        -        -  36831335 S   0   0  1 f1000f0a10023240  8410400      -
- -
       -        -        -  36896873 Z   0  60  1        -   c00001      -   - -
       -        -        -  36962411 S   0   0  1 f1000f0a10023440  8410400      -
- -
       -        -        -  37027949 S   0   0  1 f1000f0a10023540  8410400      -
- -
       -        -        -  37093487 S   0   0  1 f1000f0a10023640  8410400      -
- -
       -        -        -  38404247 S   0   0  1 f1000f0a10024a40  8410400      -
- -
       -        -        -  38469785 S   0   0  1 f1000f0a10024b40  8410400      -
- -
       -        -        -  38863013 S   0   0  1 f1000f0a10025140  8410400      -
- -
       -        -        -  38928551 S   0   0  1 f1000f0a10025240  8410400      -
- -
       -        -        -  41353455 S   0   0  1 f1000f0a10027740  8410400      -
- -
       -        -        -  41418993 S   0   0  1        -   418400      -   - -
       -        -        -  41484531 S   0   0  1 f1000f0a10027940  8410400      -
- -
       -        -        -  41550069 S   0   0  1        -   418400      -   - -
       -        -        -  43778109 S   0   0  1 f1000f0a10029c40  8410400      -
- -
       -        -        -  43909181 S   0   0  1 f1000f0a10029e40  8410400      -
- -
    USER      PID     PPID       TID S  CP PRI SC    WCHAN        F     TT BND
COMMAND
    root  9699498        1         - A   0   0 19        * 10240103      -   -
/grid/b
       -        -        -  37159025 S   0   0  1 f1000f0a10023740  8410400      -
- -
```

```
       -        -        -  38273179 S    0    0  1 f1000f0a10024840  8410400      -
- -
       -        -        -  38994089 S    0    0  1 f1000f0a10025340  8410400      -
- -
       -        -        -  40173773 S    0    0  1 f1000f0a10026540  8410400      -
- -
       -        -        -  40239311 S    0    0  1        -   418400      -   - -
       -        -        -  40304847 S    0    0  1 f1000f0a10026740  8410400      -
- -
       -        -        -  40370385 Z    0  60  1        -   c00001      -   - -
       -        -        -  40435923 S    0    0  1 f1000f0a10026940  8410400      -
- -
       -        -        -  40501461 S    0    0  1 f1000f0a10026a40  8410400      -
- -
       -        -        -  40566999 Z    0  60  1        -   c00001      -   - -
       -        -        -  40632537 S    0    0  1 f1000f0a10026c40  8410400      -
- -
       -        -        -  41091303 S    0    0  1 f1000f0a10027340  8410400      -
- -
       -        -        -  41156841 S    0    0  1        -   418400      -   - -
       -        -        -  41222379 S    0    0  1 f1000f0a10027540  8410400      -
- -
       -        -        -  41287917 S    0    0  1        -   418400      -   - -
       -        -        -  43843643 S    0    0  1 f1000f0a10029d40  8410400      -
- -
       -        -        -  43974719 S    0    0  1 f1000f0a10029f40  8410400      -
- -
       -        -        -  44105797 Z    0  60  1        -   c00001      -   - -
       -        -        -  44171335 S    0    0  1 f1000f0a1002a240  8410400      -
- -
```

In addition, ocssd.bin, cssdagent, cssdmonitor, and osysmond should have the majority of their memory pinned. This should be verified using the following command. The "Pin" value should be close to the size of the "Inuse" value.

```
> ps -elf -o "pid,args"|egrep 'ocssd.bin|cssdagent|cssdmonitor|osysmond' |grep -v
grep |awk '{ print $1 }' |xargs svmon -P |egrep
'ocssd.bin|cssdagent|cssdmonitor|osysmond'

  3014718 cssdmonitor     128065   120938     1788   109308      Y     Y     Y

 4849720 cssdagent       127796   120666     1788   109037      Y     Y     Y

12976370 osysmond.bin    116916   107263     1788   114017      Y     Y     Y

 6488312 ocssd.bin       111163   105930     1788   112086      Y     Y     Y
```

5. Pin the AIX Kernel Memory

For Oracle RAC 10g Release 2 to 11g Release 2:

Beginning with AIX 7.1, the AIX kernel memory is pinned by default. The following example shows how to verify this parameter for AIX 7.1 (vmm_klock_mode) using the "vmo" command. The values for "CUR" "DEF" and "BOOT" should all be 2.

```
> vmo -L vmm_klock_mode

NAME                      CUR    DEF    BOOT   MIN    MAX    UNIT           TYPE

     DEPENDENCIES

--------------------------------------------------------------------------------

vmm_klock_mode            2      2      2      0      3      numeric           B

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

For AIX 6.1, the kernel memory pinning option requires AIX 6.1 TL06 - or higher. The following examples shows how to modify and verify this parameter (vmm_klock_mode=2) in AIX 6.1 using the "vmo" command. Note: Modifying this parameter requires that the "bosboot" command be run and then the partition to be rebooted.

```
# Display default value:

> vmo -L vmm_klock_mode

NAME                      CUR    DEF    BOOT   MIN    MAX    UNIT           TYPE

     DEPENDENCIES

--------------------------------------------------------------------------------

vmm_klock_mode            1      1      1      0      3      numeric           B

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

```
# Modify values to 2:

> vmo -r -o vmm_klock_mode=2;

Modification to restricted tunable vmm_klock_mode, confirmation required yes/no
yes

Setting vmm_klock_mode to 2 in nextboot file

Warning: some changes will take effect only after a bosboot and a reboot

Run bosboot now? yes/no yes

bosboot: Boot image is 45198 512 byte blocks.

Warning: changes will take effect only at next reboot
```

```
# Display values after reboot.
> vmo -L vmm_klock_mode

NAME                      CUR    DEF    BOOT   MIN    MAX    UNIT           TYPE

     DEPENDENCIES

--------------------------------------------------------------------------------

vmm_klock_mode            2      1      2      0      3      numeric           B

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

6. Reduce heavy paging activity

For Oracle RAC 10g Release 2 to 11g Release 2:

As with most operating systems, heavy paging on an AIX system can cause scheduling delays. Very heavy paging can cause longer scheduling delays, which can interfere with the critical Oracle processes like oprocd and cssd.

To prevent delays due to heavy paging, the system should be monitored and tuned to avoid heavy paging. If a system is seeing heavy paging, there are two ways to avoid heavy paging:

- Tune the workload to reduce its memory usage
- Increase the amount of physical memory allocated to the workload

Note: Increasing paging space without any increase in physical memory will not help reduce paging activity.

Monitoring memory usage and paging

Monitoring a workload is important when identifying why a system is heavily paging and to also prevent performance impacts to the workload. Memory utilization can be monitored using many tools and management suites, but some basic monitoring can be achieved using the AIX tools vmstat and svmon.

The vmstat tool can be used to monitor the amount of memory being used as well as the rate of paging on the system. The key metrics to observe are active virtual memory (avm), page-in rate (pi), and page-out rate (po). The avm field will report the total amount of virtual memory in-use in units of 4K pages. When the amount of virtual memory on the system gets close to the memory size of the LPAR, this can be treated as an early indication that the system is getting close to running out of physical memory and may start paging.

It is important to note that the number of free pages (fre) is not a good indication of whether a system is low on free memory. AIX aggressively uses memory to cache file data, and thus, it is not unusual to see a low number of free pages even when the amount of active virtual memory is low. A better indication of whether a system is close to paging is to look at the active virtual memory (avm) field and compare it to the total amount of memory.

Once the amount of active virtual memory (avm) exceeds the amount of memory in the system, AIX will begin paging. The page-in (pi) and page-out (po) fields can be used to monitor the paging activity. These fields report the rate of page-in and page-out operations on a system.

In the following vmstat example, the small amount of free memory (fre) leads to high paging rates for both page in (pi) and page out (po).

The following is an example of a system incurring heavy paging that resulted in an 'oprocd' eviction. These high levels of paging should be avoided.

```
> vmstat –t 2
kthr    memory              page              faults             cpu
time
----- ----------- ----------------------- ------------ ---------------------- --
------
 r  b   avm   fre    re  pi  po  fr   sr  cy  in   sy  cs us sy id wa    pc    ec
hr mi se
 2 17 8056132 12319   0   0 386 512  512   0 395   34 534  0 45  0 54  1.01  50.5
15:11:28
 1 16 8056912 12330   0   0 396 320  320   0 379   39 429  0 46  0 54  1.01  50.7
15:11:30
 2 18 8057768 12568   0   0 548 320  320   0 390   36 571  0 46  0 54  1.01  50.7
15:11:32
 1 17 8067254 12317   0   0 4617 4864 4878   0 538  149 1203  2 48  0 51  1.09
54.3 15:11:34
 1 18 8072089 12492   0   0 2504 2432 2436   0 434   52 691  1 46  0 53  1.04
52.2 15:11:36
 1 16 8084793 12577   0   0 6393 6336 6355   0 616   27 1156  2 48  0 50  1.10
55.2 15:11:38
 1 12 8105868 13164   0   0 10711 10224 141211   0 823   31 1807  3 54  1 42  1.26
63.2 15:11:40
 1 16 8106267 18276   0   4 2766   0    0   0 492   56 191  0 46 16 37  1.03  51.4
15:11:42
 1 16 8106273 20478   0   5 446   0    0   0 422   48 221  0 46 16 38  1.01  50.5
15:11:47
 1  9 8107224 20111   0 179 491 515  515   0 1377 13155 7880  2 52  6 39  1.21
60.5 15:11:49
```

Another tool that can be used to monitor memory utilization of a system is svmon. Starting with AIX 5.3 TL09 and AIX 6.1 TL02 svmon now reports an "available" metric. This metric can be used to more easily determine how much remaining memory is available to applications. The available metric reports the amount additional amount of physical memory that can be used for applications without incurring paging. When the amount of available memory gets low, this is an indication that the system is close to paging.

This can be seen in the following example:

```
 root@racha905 / > svmon -G -O unit=auto
-------------------------------------------------------------------------------
               size       inuse        free         pin     virtual  available
memory        27.2G       8.04G       19.2G       2.45G       7.43G      19.2G
pg space      24.5G       24.4M

               work        pers        clnt       other
pin           1.26G          4K        388K       1.19G
in use        7.43G     311.66M     311.68M
```

Appendix A: Oprocd logging examples

Clean oprocd log files

- For /etc/oracle/oprocd/<node>.oprocd.log

```
Apr 23 16:04:34.413 | INF | monitoring started with timeout(1000), margin(10000),
skewTimeout(250)
Apr 23 16:04:34.495 | INF | fatal mode startup, setting process to fatal mode
Apr 23 16:04:40.385 | INF | enabling fatal mode as per client request
```

- For /etc/oracle/oprocd/<node>.oprocd.lgl

```
Apr 23 16:04:34.405420 | LASTGASP | InitLastGasp:  Initial write/allocate for last
gasp file
```

Oprocd log files showing scheduling delays

- For /etc/oracle/oprocd/<node>.oprocd.log

```
Apr 23 16:04:34.413 | INF | monitoring started with timeout(1000), margin(10000),
skewTimeout(250)
Apr 23 16:04:34.495 | INF | fatal mode startup, setting process to fatal mode
Apr 23 16:04:40.385 | INF | enabling fatal mode as per client request
```

- For /etc/oracle/oprocd/<node>.oprocd.lgl

```
Apr 23 16:04:34.405420 | LASTGASP | InitLastGasp:  Initial write/allocate for last
gasp file
Apr 24 05:20:03.665 | INF | TrackHistoricalTrends:  added first sample 3242554777
in 10 to 50 percentile
Apr 26 05:26:23.593 | INF | TrackHistoricalTrends:  added first sample 2642327278
in 10 to 50 percentile
```

Note that 2 entries have been recorded where the delay in scheduling was approximately 3.24 and 2.64 seconds respectively which are greater than the old 1.5 second limit but less than the new 11 second limit.

Oprocd log files after a node reboot or CRS restart

During CRS restart, the previous files will be renamed with a time stamp appended to the file names:

- For /etc/oracle/oprocd/<node>.oprocd.log.<time stamp>

```
Apr 23 16:04:34.405420 | LASTGASP | InitLastGasp:  Initial write/allocate for last
gasp file
Apr 24 05:20:03.665 | INF | TrackHistoricalTrends:  added first sample 3242554777
in 10 to 50 percentile
Apr 26 05:26:23.593 | INF | TrackHistoricalTrends:  added first sample 2642327278
in 10 to 50 percentile
```

- For /etc/oracle/oprocd/<node>.oprocd.lgl.<time stamp>

```
Apr 28 16:00:04.327787 | LASTGASP | AlarmHandler:  timeout(11917 msec) exceeds
interval(1000 msec)+margin(10000 msec).   Rebooting NOW.

Oracle Support Data - Previous intervals (LIFO):
999ms
999ms
1000ms

<< Note: Redundant lines deleted from *lgl* file >>
```

Appendix B: Example script for setting the correct VMM settings

The following example script can be used to set the correct VMM parameters according to the recommendations in this white paper:

```
#!/usr/bin/ksh

vmo -p -o maxperm%=90;
vmo -p -o minperm%=3;
vmo -p -o maxclient%=90;
vmo -p -o strict_maxperm=0;
vmo -p -o strict_maxclient=1;
vmo -p -o lru_file_repage=0;
vmo -r -o page_steal_method=1;

chdev –l sys0 –a ‘minpout=4096 maxpout=8193’;
```

References

My Oracle Support DocID # 265769.1 “Troubleshooting CRS Reboots”

My Oracle Support DocID # 419312.1 “Node reboots due to oprocd on AIX”

My Oracle Support DocID # 559365.1 “Using Diagwait as a diagnostic to get more information for diagnosing Oracle Clusterware Node evictions”

My Oracle Support DocID # 282036.1 “Minimum Software Versions and Patches Required to Support Oracle Products on IBM Power Systems”

My Oracle Support DocID # 811293.1 RAC Assurance Support Team: RAC Starter Kit and Best Practices (AIX)

ORACLE

Oracle Real Application Clusters on IBM AIX –
Best practices in memory tuning and configuring
for system stability - Version: 1.4, May 2012

Authors: Rick Piasecki (IBM),
Wayne Martin (IBM)

Contributing Authors:
Dennis Massanari (IBM),
John McHugh (Oracle),
Markus Michalewicz (Oracle),
Anil Nair (Oracle)

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
oracle.com

Oracle is committed to developing practices and products that help protect the environment