SHARE in Orlando 2015

SHARE Educate · Network · Influence

17359: Reducing CPU Consumption with Oracle on IBM z Systems for Extreme Consolidation

Speakers: David Simpson and Sam Amsavelu (IBM)

simpson.dave@us.ibm.com

samevlu@us.ibm.com

Southern Hemisphere 3 (Walt Disney World Dolphin)

#SHAREorg

SHARE is an independent volunteer-run information technology association that provides **education**, professional **networking** and industry **influence.**

CELEBRATING 60 YEARS OF SHARE Influencing IT Since 1955

Copyright and Trademark Information

- For IBM – can be found at http://www.ibm.com/legal/us/en/copytrade.shtml
- For Oracle – can be found at

  http://www.oracle.com/us/legal/index.html

- Any performance results/observations in this presentation are purely for education and planning purposes. No Test results should be construed as indicative of any particular customer workload or benchmark result.

Reduce Linux RPM's

- Helps reduces the Disk space & the Number of Linux services created.
- Reduces the software updates/bug/security updates that are required.
- Use the Oracle RPM checker
  - Requirements for Installing Oracle Database 12c on RHEL 6 on IBM: Linux on System z (s390x) (Doc ID 1574413.1)
  - Requirements for Installing Oracle Database 12c on SLES 11 on IBM: Linux on System z (s390x) (Doc ID 1574414.1)

- Oracle 12c database no longer requires the 31-bit s390 libraries
  - Oracle client still requires 31-bit libraries (not typically installed on DB server)

Linux paging / swappiness

- With the default swappiness setting of 60 Linux does proactive paging
- Oracle data / code on a Linux (or VM) paging disk has a performance hit when it's needed
  - Observed long (>10s) waits at swap in
  - Guest was sized correctly
  - Guest was using database on a file system without direct I/O
- Recommendation: set swappiness to zero
  - In /etc/syctl.conf add `vm.swappiness=0`
- Largepages are ineligible for swapping.

Sles 11 SP4 Improvements

- Database engines such as Oracle Berkeley DB use memory mappings mmap(2) to manipulate database files.

- The following settings may be relevant when tuning for database workloads:

vm.dirty_ratio=15 - Maximum percentage of dirty system memory (default 40).

vm.dirty_background_ratio = 3 - Percentage of dirty system memory at which background writeback will start (default 10).

vm.dirty_expire_centisecs = 500 - Duration after which dirty system memory is considered old enough to be eligible for writeback (default 3000)

vm.dirty_writeback_centisecs=100 (default 500)

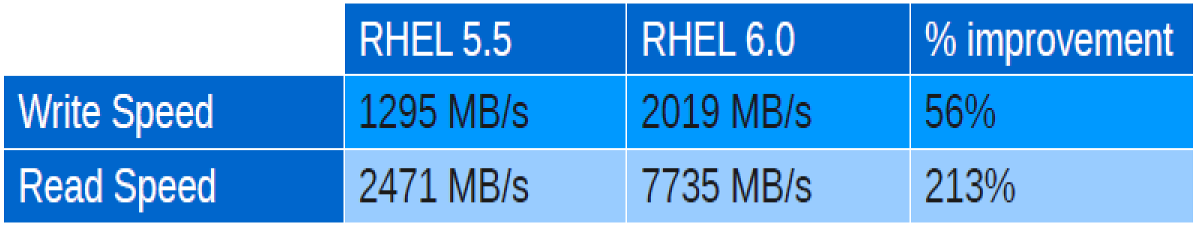
vm.vfs_cache_pressure =200 - Help performance for backups to disk

Source: https://www.suse.com/support/kb/doc.php?id=7010287

OS Level Currency

- **Significant Performance & Security Improvements when upgrading OS Distribution levels:**

Red Hat Memory Performance:



	RHEL 5.5	RHEL 6.0	% improvement
Write Speed	1295 MB/s	2019 MB/s	56%
Read Speed	2471 MB/s	7735 MB/s	213%

Source Red Hat - A Performance Comparison Between RHEL 5 and RHEL 6 on System z

Turning off Unneeded Services

- Keep the golden image as lean as possible in terms of processor usage, some of these services can be turned off with chkconfig command:

Red Hat 6.4+

```
# chkconfig iptables off
# chkconfig ip6tables off
# chkconfig auditd off
# chkconfig abrtd off
# chkconfig atd off
# chkconfig cups off
# chkconfig mdmonitor off
```

Sles 11 sp3+

```
# chkconfig fbset off
# chkconfig network-remotefs off
# chkconfig postfix off
# chkconfig splash off
# chkconfig splash_early off
# chkconfig smartd off
# chkconfig xinetd off
```

Source: http://www.redbooks.ibm.com/abstracts/sg248147.html

The Virtualization Cookbook for IBM z Systems Volume 2: Red Hat Enterprise Linux Server 7.1, SG24-8303

The Virtualization Cookbook for IBM z Systems Volume 3: SUSE Linux Enterprise Server 12, SG24-8890

VDSO – Linux cpu Improvements

- **V**irtual **D**ynamically-linked **S**hared **O**bject (VDSO) is a shared library provided by the kernel. This allows normal programs to do certain system calls without the usual overhead of system calls like switching address spaces.

- Example by using the new VDSO implementation we have seen **six times** reduction in the number of function calls.

- Newer Linux distributions (RHEL 5.9 & 6.x, SLES 11) have this feature and it's enabled by default.

- Oracle calls Linux **gettimeofday()** hundreds of times a second for reporting statistics.

(Less Oracle Oracle products you install the less number of user calls)

- By upgrading Linux, VDSO reduces cpu costs, especially in virtualized environments

Oracle’s VKTM Process

- Oracle’s VKTM timer service centralizes time tracking and offloads multiple timer calls from other clients.

- **VKTM** is responsible for providing a wall-clock time and reference-time counter (updated every 20ms) **even when the database is idle for a long time (CPU Idle).**

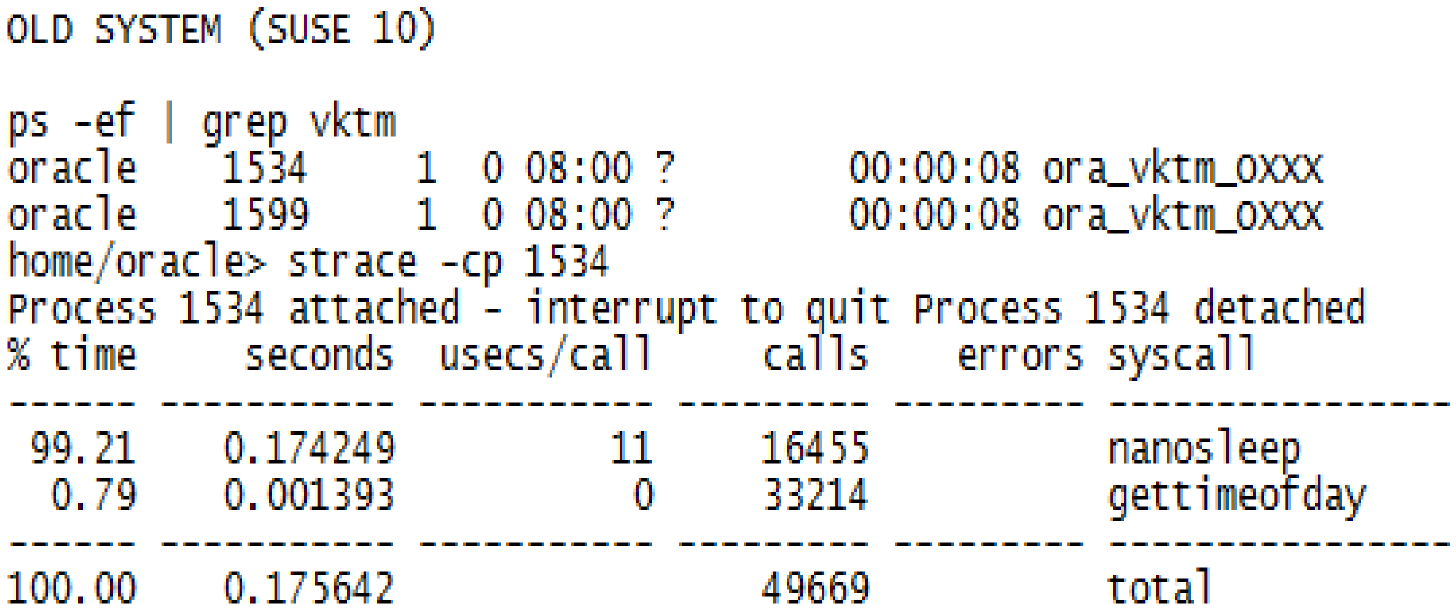
SUSE 10

kernel timer interrupt frequency is approx. 100 Hz

SUSE 11

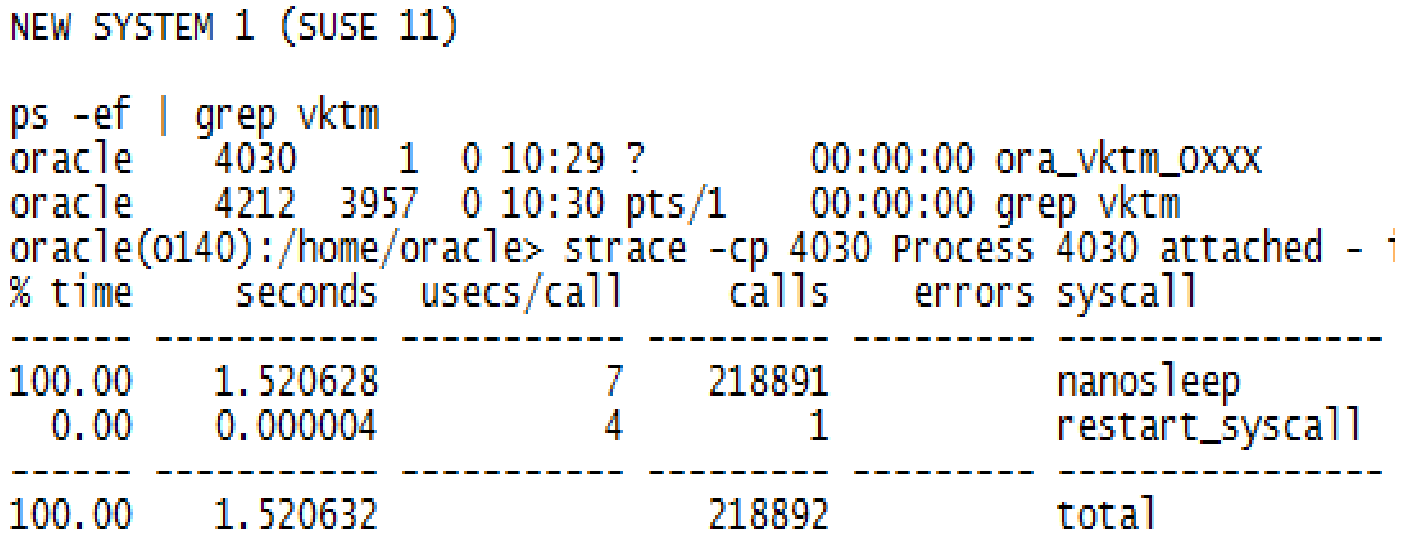
kernel timer interrupt frequency is approx. 4000 Hz or higher

VKTM – OS Upgrade Reduces CPU Usage



```
OLD SYSTEM (SUSE 10)

ps -ef | grep vktm
oracle    1534     1  0 08:00 ?        00:00:08 ora_vktm_OXXX
oracle    1599     1  0 08:00 ?        00:00:08 ora_vktm_OXXX
home/oracle> strace -cp 1534
Process 1534 attached - interrupt to quit Process 1534 detached
% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
 99.21    0.174249          11     16455           nanosleep
  0.79    0.001393           0     33214           gettimeofday
------ ----------- ----------- --------- --------- ----------------
100.00    0.175642                 49669           total
```



```
NEW SYSTEM 1 (SUSE 11)

ps -ef | grep vktm
oracle    4030     1  0 10:29 ?        00:00:00 ora_vktm_OXXX
oracle    4212  3957  0 10:30 pts/1    00:00:00 grep vktm
oracle(0140):/home/oracle> strace -cp 4030 Process 4030 attached - i
% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
100.00    1.520628           7    218891           nanosleep
  0.00    0.000004           4         1           restart_syscall
------ ----------- ----------- --------- --------- ----------------
100.00    1.520632                218892           total
```

VKTM with Oracle 12c & 11gR2

Default Values 11gR2 & 12c:

```
_disable_highres_ticks        False
_timer_precision              10
```

```
% time     seconds  usecs/call   calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
100.00    0.069437          1   125092    nanosleep
  0.00    0.000000          0        1    restart_syscall
------ ----------- ----------- --------- --------- ----------------
100.00   0.069437              125093                  total
```

VKTM Changes to Help Reduce CPU***:

```
_disable_highres_ticks       TRUE

_timer_precision             2000
```

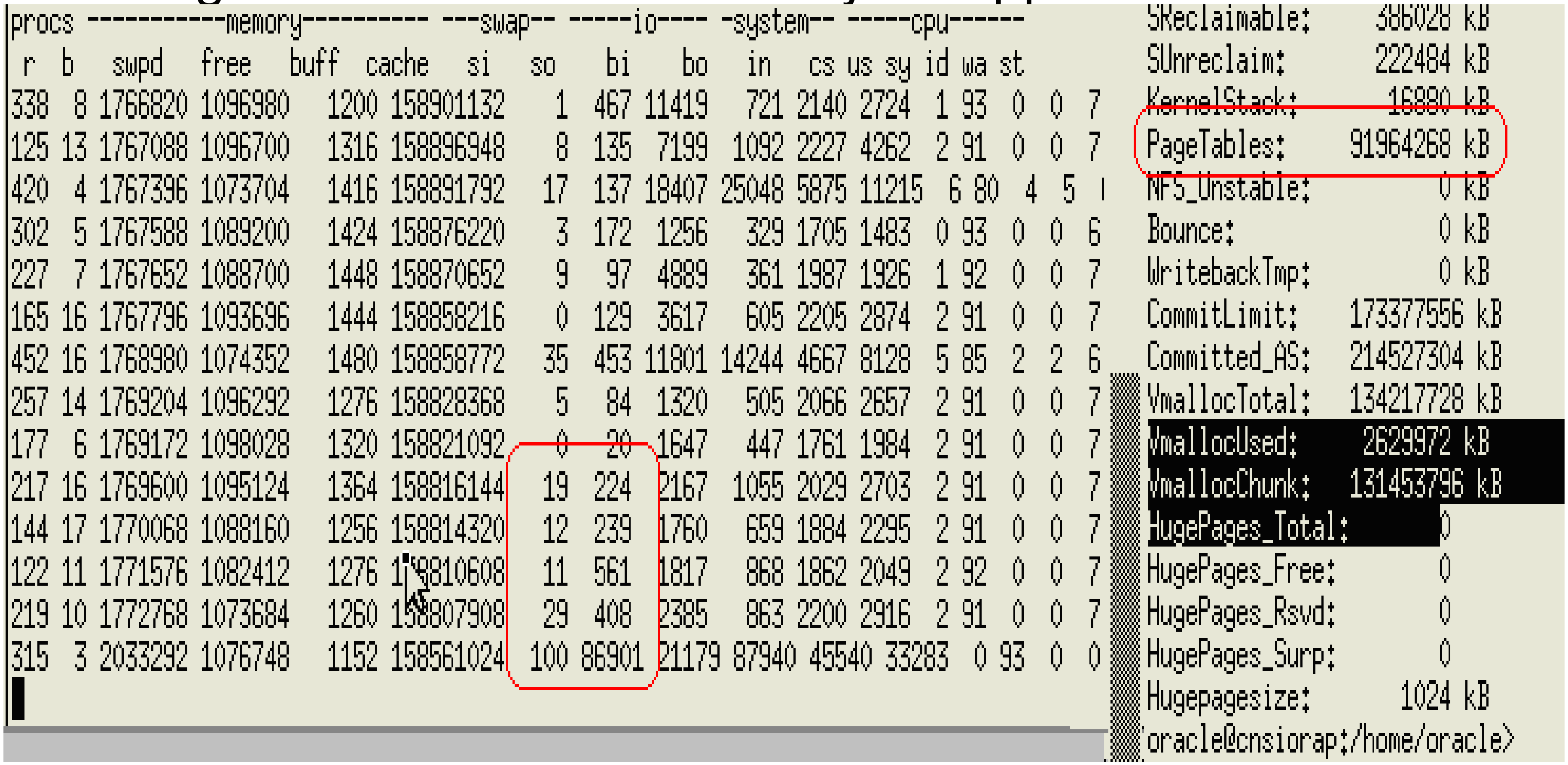
*** **Get Oracle support approval before using.**

```
% time  seconds  usecs/call     calls     errors syscall
------ ----------- ----------- --------- --------- ----------------
 99.81   0.002063          1      1496     nanosleep
  0.19   0.000004          4         1     restart_syscall
------ ----------- ----------- --------- --------- ----------------
100.00    0.002067                1497               total
```

Linux Huge Pages

- **Consider Using Linux Huge Pages for Oracle Database Memory**

  →In general 10-15% can be gained by the reduction in CPU usage as well as more memory for applications that would



```
procs -----------memory---------- ---swap-- -----io---- -system-- -----cpu------
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
338  8 1766820 1096980   1200 158901132    1  467 11419   721 2140 2724  1 93  0  0  7
125 13 1767088 1096700   1316 158896948    8  135  7199  1092 2227 4262  2 91  0  0  7
420  4 1767396 1073704   1416 158891792   17  137 18407 25048 5875 11215  6 80  4  5  1
302  5 1767588 1089200   1424 158876220    3  172  1256   329 1705 1483  0 93  0  0  6
227  7 1767652 1088700   1448 158870652    9   97  4889   361 1987 1926  1 92  0  0  7
165 16 1767796 1093696   1444 158858216    0  129  3617   605 2205 2874  2 91  0  0  7
452 16 1768980 1074352   1480 158858772   35  453 11801 14244 4667 8128  5 85  2  2  6
257 14 1769204 1096292   1276 158828368    5   84  1320   505 2066 2657  2 91  0  0  7
177  6 1769172 1098028   1320 158821092    0   20  1647   447 1761 1984  2 91  0  0  7
217 16 1769600 1095124   1364 158816144   19  224  2167  1055 2029 2703  2 91  0  0  7
144 17 1770068 1088160   1256 158814320   12  239  1760   659 1884 2295  2 91  0  0  7
122 11 1771576 1082412   1276 158810608   11  561  1817   868 1862 2049  2 92  0  0  7
219 10 1772768 1073684   1260 158807908   29  408  2385   863 2200 2916  2 91  0  0  7
315  3 2033292 1076748   1152 158561024  100 86901 21179 87940 45540 33283  0 93  0  0
```

```
SReclaimable:     386028 kB
SUnreclaim:       222484 kB
KernelStack:       16880 kB
PageTables:     91964268 kB
NFS_Unstable:          0 kB
Bounce:                0 kB
WritebackTmp:          0 kB
CommitLimit:   173377556 kB
Committed_AS:  214527304 kB
VmallocTotal:  134217728 kB
VmallocUsed:     2629972 kB
VmallocChunk:  131453796 kB
HugePages_Total:       0
HugePages_Free:        0
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:       1024 kB
oracle@cnsiorap:/home/oracle>
```

Huge Page Considerations:

- Can not use **MEMORY_TARGET** with Huge Pages.
  - Set manually to **SGA_TARGET** not including the **PGA_AGGREGATE_TARGET**.

- Not swappable: Huge Pages are not swappable

- General guideline consider when combined Oracle SGA's are greater than **8 GB** (particularly if a lots of connections)

- Decreased page table overhead; more memory can be freed up for other uses. i.e. more Oracle SGA memory, and less physical I/O's (See also Oracle Note: **361468.1**)

Recommendation: Huge Pages under z/VM

- Under z/VM (which has 4K pages) it’s still recommended to use Huge Pages for SGA’s > 10GB particularly with many connections

- Saves Memory that would otherwise be used for pagetables

- Stability for user process spikes (avoiding swap)

- Less work to manage smaller number of pagetables

Oracle Database 12.1 Support Update for Linux on System z

Linux on System z specifics

- It's Fast
  - Built using PDF (Profile Directed Feedback).
  - Approximately 5% Faster even with all the new features.
- New Features – less resources
- EM agent 12.1. enabled
  - OEM Cloud Control 12cR3 or 12cR4
- IBM Redbook
  - **Experiences with Oracle Database 12c on Linux on System z SG248159** http://www.redbooks.ibm.com/abstracts/sg248159.html?Open

Upgrade 11.2.0.4 -> 12.1.0.1 - CPU

18.9% improvement in response time between 11.2.0.4 & 12.1 (cpu intensive test)

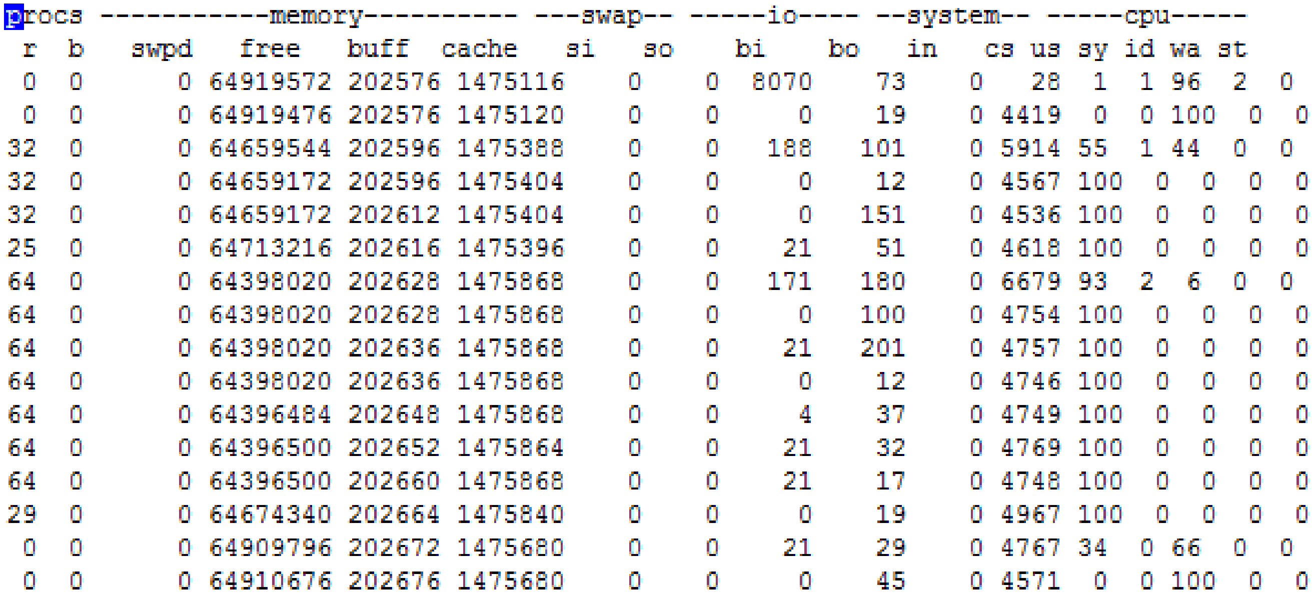
Oracle 11.2.0.4

Running Parallel Processes: 32

real   0m12.01s
user   0m0.20s
sys   0m0.13s

Running Parallel Processes: 64

real   0m23.84s
user   0m0.40s
sys   0m0.26s



```
procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 0  0      0 64919572 202576 1475116    0    0  8070    73    0   28  1  1 96  2  0
 0  0      0 64919476 202576 1475120    0    0     0    19    0 4419  0  0 100  0  0
32  0      0 64659544 202596 1475388    0    0   188   101    0 5914 55  1 44  0  0
32  0      0 64659172 202596 1475404    0    0     0    12    0 4567 100  0  0  0  0
32  0      0 64659172 202612 1475404    0    0     0   151    0 4536 100  0  0  0  0
25  0      0 64713216 202616 1475396    0    0    21    51    0 4618 100  0  0  0  0
64  0      0 64398020 202628 1475868    0    0   171   180    0 6679 93  2  6  0  0
64  0      0 64398020 202628 1475868    0    0     0   100    0 4754 100  0  0  0  0
64  0      0 64398020 202636 1475868    0    0    21   201    0 4757 100  0  0  0  0
64  0      0 64398020 202636 1475868    0    0     0    12    0 4746 100  0  0  0  0
64  0      0 64396484 202648 1475868    0    0     4    37    0 4749 100  0  0  0  0
64  0      0 64396500 202652 1475864    0    0    21    32    0 4769 100  0  0  0  0
64  0      0 64396500 202660 1475868    0    0    21    17    0 4748 100  0  0  0  0
29  0      0 64674340 202664 1475840    0    0     0    19    0 4967 100  0  0  0  0
 0  0      0 64909796 202672 1475680    0    0    21    29    0 4767 34  0 66  0  0
 0  0      0 64910676 202676 1475680    0    0     0    45    0 4571  0  0 100  0  0
```

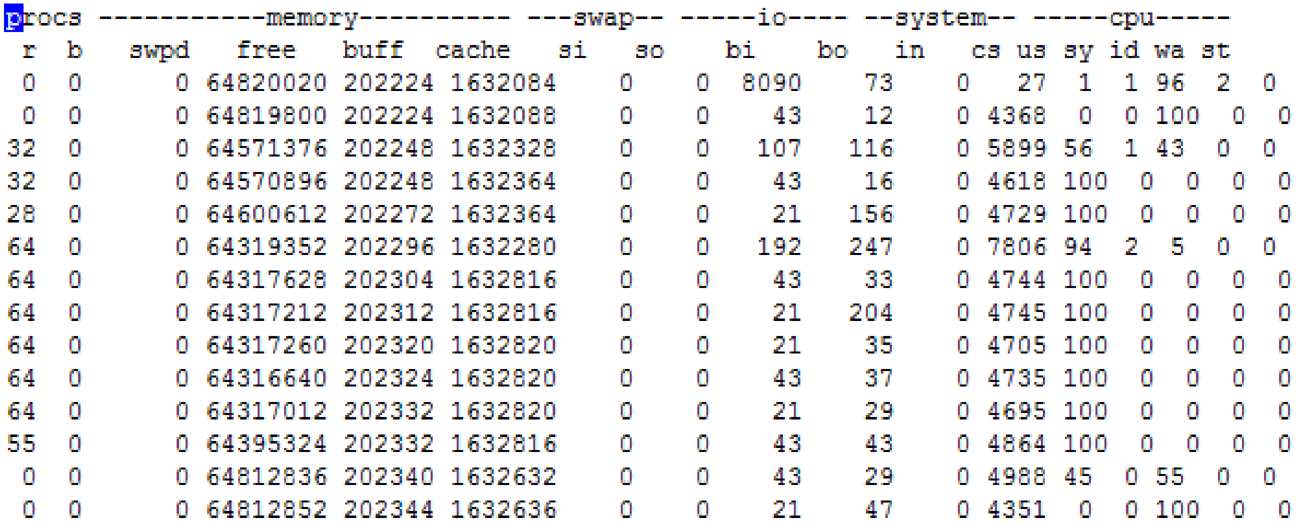
Oracle 12.1.0.1

Running Parallel Processes: 32

real   0m10.12s
user   0m0.16s
sys   0m0.14s

Running Parallel Processes: 64

real   0m20.05s
user   0m0.34s
sys   0m0.27s



```
procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 0  0      0 64820020 202224 1632084    0    0  8090    73    0   27  1  1 96  2  0
 0  0      0 64819800 202224 1632088    0    0    43    12    0 4368  0  0 100  0  0
32  0      0 64571376 202248 1632328    0    0   107   116    0 5899 56  1 43  0  0
32  0      0 64570896 202248 1632364    0    0    43    16    0 4618 100  0  0  0  0
28  0      0 64600612 202272 1632364    0    0    21   156    0 4729 100  0  0  0  0
64  0      0 64319352 202296 1632280    0    0   192   247    0 7806 94  2  5  0  0
64  0      0 64317628 202304 1632816    0    0    43    33    0 4744 100  0  0  0  0
64  0      0 64317212 202312 1632816    0    0    21   204    0 4745 100  0  0  0  0
64  0      0 64317260 202320 1632820    0    0    21    35    0 4705 100  0  0  0  0
64  0      0 64316640 202324 1632820    0    0    43    37    0 4735 100  0  0  0  0
64  0      0 64317012 202332 1632820    0    0    21    29    0 4695 100  0  0  0  0
55  0      0 64395324 202332 1632816    0    0    43    43    0 4864 100  0  0  0  0
 0  0      0 64812836 202340 1632632    0    0    43    29    0 4988 45  0 55  0  0
 0  0      0 64812852 202344 1632636    0    0    21    47    0 4351  0  0 100  0  0
```

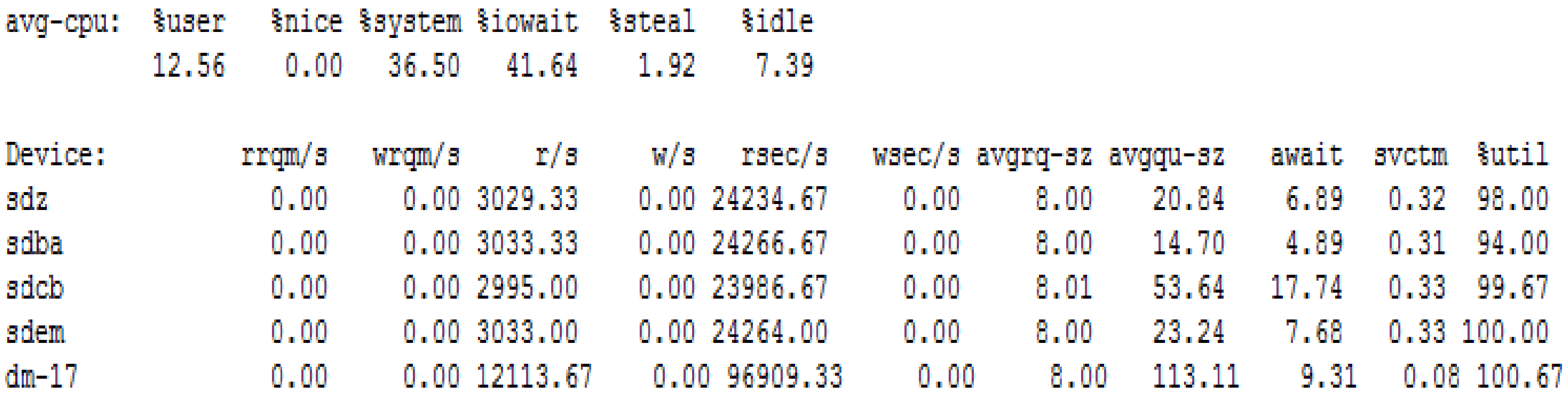
11.2.0.4 -> 12.1.0.1 - I/O Test

Oracle I/O Calibrate (high I/O) Test:

- **Not much change between releases (for this particular I/O test)**

Oracle 11.2.0.4	**Oracle 12.1.0.1**

max_iops = **332989**
latency = 0
max_mbps = 3109
	max_iops = **333576**
	latency = 0
	max_mbps = 3116



```
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          12.56    0.00   36.50   41.64    1.92    7.39

Device:         rrqm/s   wrqm/s     r/s     w/s   rsec/s   wsec/s avgrq-sz avgqu-sz   await  svctm  %util
sdz               0.00     0.00 3029.33    0.00 24234.67     0.00     8.00    20.84    6.89   0.32  98.00
sdba              0.00     0.00 3033.33    0.00 24266.67     0.00     8.00    14.70    4.89   0.31  94.00
sdcb              0.00     0.00 2995.00    0.00 23986.67     0.00     8.01    53.64   17.74   0.33  99.67
sdem              0.00     0.00 3033.00    0.00 24264.00     0.00     8.00    23.24    7.68   0.33 100.00
dm-17             0.00     0.00 12113.67    0.00 96909.33     0.00     8.00   113.11    9.31   0.08 100.67
```

Oracle 12c Trace File Analyzer Disable

Trace File Analyzer Collector (TFA): collects log and trace files from all nodes and products into a single location.

- Written in Java with its own JVM
- Large memory footprint for the heap etc.
- Can be disabled with a single command
- **Note:** next time you run rootcrs.pl (patching for example) it may reinstall itself.

Stop TFA

```
# /etc/init.d/init.tfa stop
```

Start TFA

```
# /etc/init.d/init.tfa  start
```

Stop and removes related inittab entries

```
# /etc/init.d/init.tfa shutdown
```

12c Cluster Verification Utility (CVU) - Disable

Cluster Verification Utility (CVU):

- The CVU tool automatically runs, pointing out configuration issue.
- In Oracle 12.1.0.2, scheduled to run automatically every time the cluster is started and periodically after that.
- The CVU itself and checks use CPU and RAM resources, and are better run manually when such resources are limited.
- It's a quick removal

```
# crs_stat -t
Name          Type          Target   State    Host
------------------------------------------------------------
ora....ER.lsnr ora....er.type ONLINE    ONLINE    clone01
ora....N1.lsnr ora....er.type ONLINE    ONLINE    clone01
ora....N2.lsnr ora....er.type ONLINE    ONLINE    clone01
ora....N3.lsnr ora....er.type ONLINE    ONLINE    clone01
ora.OCR2.dg    ora....up.type ONLINE    ONLINE    clone01
ora.asm        ora.asm.type   ONLINE    ONLINE    clone01
ora....SM1.asm application    ONLINE    ONLINE    clone01
ora....01.lsnr application    ONLINE    ONLINE    clone01
ora....e01.ons application    ONLINE    ONLINE    clone01
ora....e01.vip ora....t1.type ONLINE    ONLINE    clone01
ora.cvu        ora.cvu.type   ONLINE    ONLINE    clone01
ora....network ora....rk.type ONLINE    ONLINE    clone01
ora.oc4j       ora.oc4j.type  OFFLINE   OFFLINE
ora.ons        ora.ons.type   ONLINE    ONLINE    clone01
ora.scan1.vip  ora....ip.type ONLINE    ONLINE    clone01
ora.scan2.vip  ora....ip.type ONLINE    ONLINE    clone01
ora.scan3.vip  ora....ip.type ONLINE    ONLINE    clone01

# srvctl stop cvu -force
```

Source: Marc Fielding http://www.pythian.com/blog/slimming-down-oracle-rac-12cs-resource-footprint/

Oracle 12c OC4J – Ensure Disabled

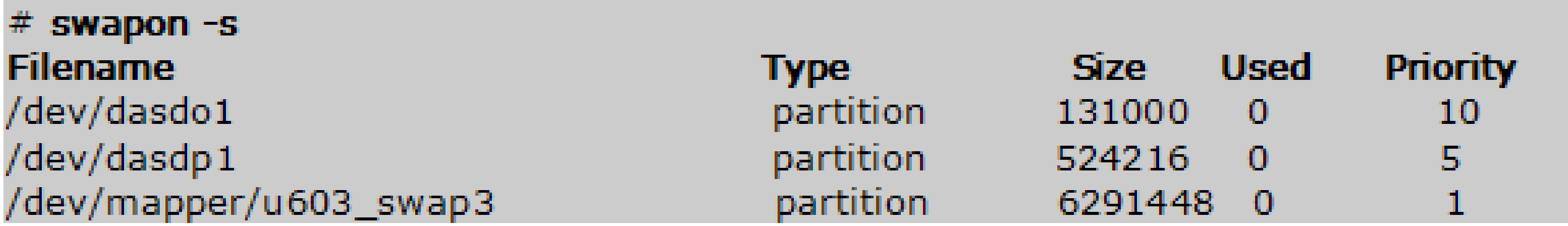
OC4J:

- Every Oracle 12c grid install contains OC4J
- Linux on System z oc4j is disabled by default.
- Ensure oc4j is disabled.

```
# crs_stat -t
Name          Type          Target    State
Host
-----------------------------------------------------------
-
ora....ER.lsnr ora....er.type ONLINE
ONLINE    clone01
ora....N1.lsnr ora....er.type ONLINE
ONLINE    clone01
ora....N2.lsnr ora....er.type ONLINE
ONLINE    clone01
ora....N3.lsnr ora....er.type ONLINE
ONLINE    clone01
ora.OCR2.dg    ora....up.type ONLINE
ONLINE    clone01
ora.asm        ora.asm.type   ONLINE
ONLINE    clone01
ora....SM1.asm application    ONLINE
ONLINE    clone01
ora....01.lsnr application    ONLINE
ONLINE    clone01
ora....e01.ons application    ONLINE
ONLINE    clone01
ora....e01.vip ora....t1.type ONLINE
ONLINE    clone01
```

Swap Sizing Oracle with Linux on System z

- Example of VDISK for **1st** and or **2nd** Level Swap with higher priority and then DASD as a lower priority swap in case of an unexpected memory pattern



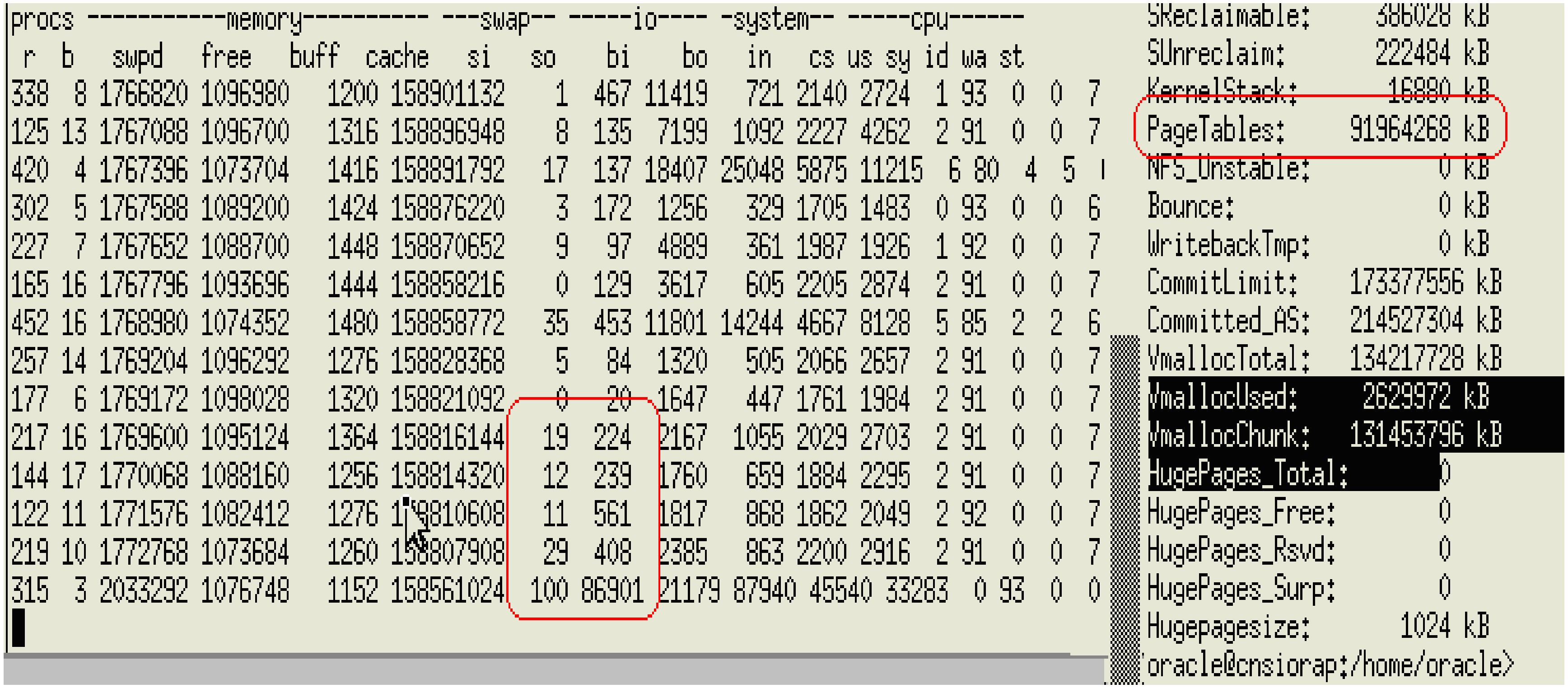
```
# swapon -s
Filename                              Type          Size     Used     Priority
/dev/dasdo1                           partition     131000    0         10
/dev/dasdp1                           partition     524216    0         5
/dev/mapper/u603_swap3                partition     6291448   0         1
```

- May want to recycle the swap from time to time to free swap slots (check swapcache in /proc/meminfo)
  - Ensure there is enough memory (e.g. at night)
  - drop caches
  - swapoff / swapon

Linux Huge Pages

- **Consider Using Linux Huge Pages for Oracle Database Memory**

  →In general 10-15% can be gained by the reduction in CPU usage as well as having a lot more memory for applications that would be consumed in Linux Page Tables…



```
procs -----------memory---------- ---swap-- -----io---- -system-- -----cpu------
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
338  8 1766820 1096980   1200 158901132    1  467 11419   721 2140 2724  1 93  0  0  7
125 13 1767088 1096700   1316 158896948    8  135  7199  1092 2227 4262  2 91  0  0  7
420  4 1767396 1073704   1416 158891792   17  137 18407 25048 5875 11215  6 80  4  5  1
302  5 1767588 1089200   1424 158876220    3  172  1256   329 1705 1483  0 93  0  0  6
227  7 1767652 1088700   1448 158870652    9   97  4889   361 1987 1926  1 92  0  0  7
165 16 1767796 1093696   1444 158858216    0  129  3617   605 2205 2874  2 91  0  0  7
452 16 1768980 1074352   1480 158858772   35  453 11801 14244 4667 8128  5 85  2  2  6
257 14 1769204 1096292   1276 158828368    5   84  1320   505 2066 2657  2 91  0  0  7
177  6 1769172 1098028   1320 158821092    0   20  1647   447 1761 1984  2 91  0  0  7
217 16 1769600 1095124   1364 158816144   19  224  2167  1055 2029 2703  2 91  0  0  7
144 17 1770068 1088160   1256 158814320   12  239  1760   659 1884 2295  2 91  0  0  7
122 11 1771576 1082412   1276 158810608   11  561  1817   868 1862 2049  2 92  0  0  7
219 10 1772768 1073684   1260 158807908   29  408  2385   863 2200 2916  2 91  0  0  7
315  3 2033292 1076748   1152 158561024  100 86901 21179 87940 45540 33283  0 93  0  0
```

```
SReclaimable:     386028 kB
SUnreclaim:       222484 kB
KernelStack:       16880 kB
PageTables:     91964268 kB
NFS_Unstable:          0 kB
Bounce:                0 kB
WritebackTmp:          0 kB
CommitLimit:   173377556 kB
Committed_AS:  214527304 kB
VmallocTotal:  134217728 kB
VmallocUsed:     2629972 kB
VmallocChunk:  131453796 kB
HugePages_Total:     0
HugePages_Free:      0
HugePages_Rsvd:      0
HugePages_Surp:      0
Hugepagesize:       1024 kB
oracle@cnsiorap:/home/oracle>
```

/proc/meminfo – customer example (before)

```
MemTotal: 82371500 kB
MemFree: 371220 kB
Buffers: 4956 kB
Cached: 50274732 kB
SwapCached: 2248480 kB
Active: 53106388 kB
Inactive: 2164644 kB
HighTotal: 0 kB
HighFree: 0 kB
LowTotal: 82371500 kB
LowFree: 371220 kB
SwapTotal: 16408504 kB
SwapFree: 9834092 kB
Dirty: 468 kB
Writeback: 0 kB
AnonPages: 2743884 kB
Mapped: 48976112 kB
Slab: 243944 kB
PageTables: 26095124 kB
NFS_Unstable: 0 kB
Bounce: 0 kB
CommitLimit: 57594252 kB
Committed_AS: 62983256 kB
VmallocTotal: 4211073024 kB
VmallocUsed: 12028 kB
VmallocChunk: 4211060796 kB
HugePages_Total: 0
HugePages_Free: 0
HugePages_Rsvd: 0
Hugepagesize: 2048 kB
```

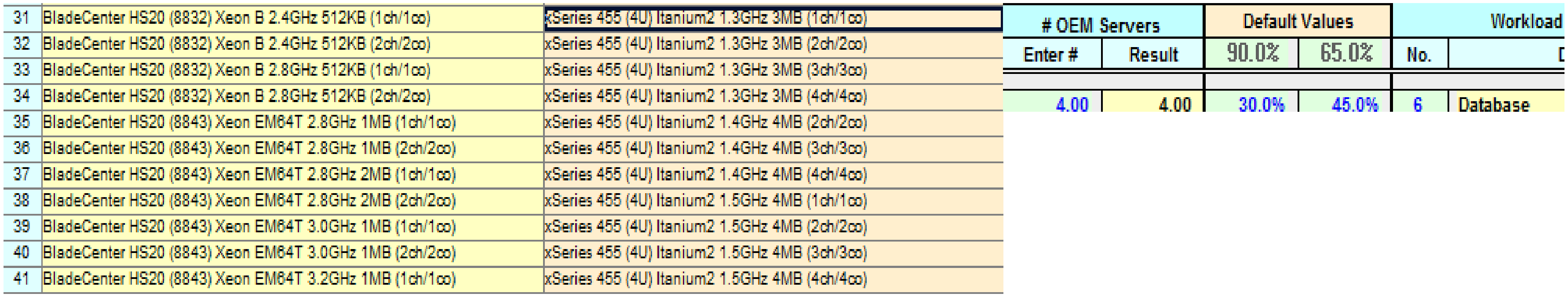
/proc/meminfo – customer example (after)

```
MemTotal:        82371500 kB
MemFree:          7315160 kB
Buffers:           352624 kB
Cached:          12824152 kB
SwapCached:             0 kB
Active:           4000920 kB
Inactive:        12309216 kB
HighTotal:              0 kB
HighFree:               0 kB
LowTotal:        82371500 kB
LowFree:          7315160 kB
SwapTotal:       18456496 kB
SwapFree:        18456496 kB
Dirty:                504 kB
Writeback:            108 kB
AnonPages:        3241568 kB
Mapped:            170176 kB
Slab:              439912 kB
PageTables:        318848 kB
NFS_Unstable:           0 kB
Bounce:                 0 kB
CommitLimit:     30802308 kB
Committed_AS:     6001276 kB
VmallocTotal: 4211073024 kB
VmallocUsed:        13032 kB
VmallocChunk: 4211059808 kB
HugePages_Total:  28164
HugePages_Free:    1208
HugePages_Rsvd:    1205
Hugepagesize:      2048 kB
```

Sizing Linux on System z Workload

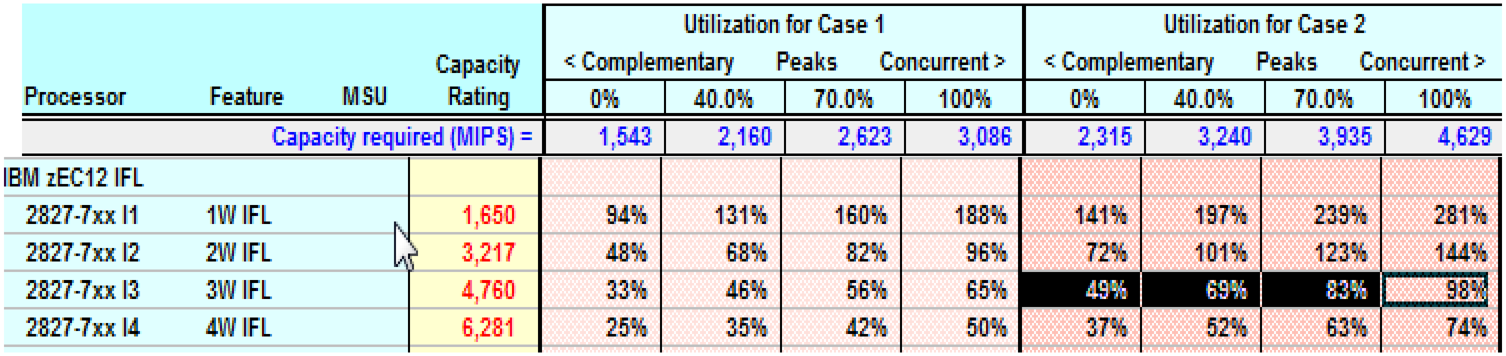
- For existing/running workloads take system utilization data for as long a period as possible or at least peak periods combined with make/model of server.

- If new workload – we use a sizing questionnaire with our TechLine sizing team and used industry standards to size the workload.



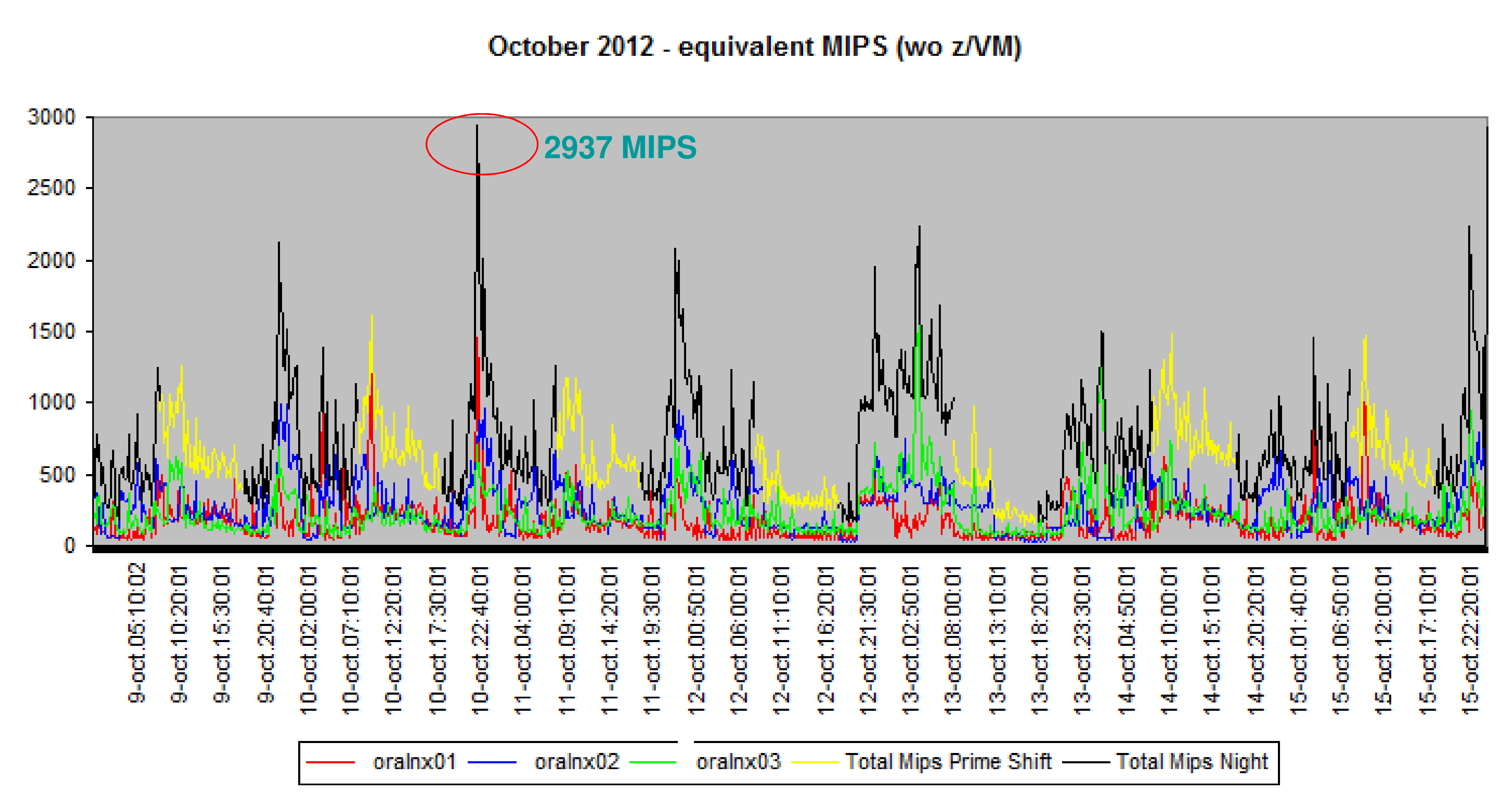
31	BladeCenter HS20 (8832) Xeon B 2.4GHz 512KB (1ch/1co)	xSeries 455 (4U) Itanium2 1.3GHz 3MB (1ch/1co)
32	BladeCenter HS20 (8832) Xeon B 2.4GHz 512KB (2ch/2co)	xSeries 455 (4U) Itanium2 1.3GHz 3MB (2ch/2co)
33	BladeCenter HS20 (8832) Xeon B 2.8GHz 512KB (1ch/1co)	xSeries 455 (4U) Itanium2 1.3GHz 3MB (3ch/3co)
34	BladeCenter HS20 (8832) Xeon B 2.8GHz 512KB (2ch/2co)	xSeries 455 (4U) Itanium2 1.3GHz 3MB (4ch/4co)
35	BladeCenter HS20 (8843) Xeon EM64T 2.8GHz 1MB (1ch/1co)	xSeries 455 (4U) Itanium2 1.4GHz 4MB (2ch/2co)
36	BladeCenter HS20 (8843) Xeon EM64T 2.8GHz 1MB (2ch/2co)	xSeries 455 (4U) Itanium2 1.4GHz 4MB (3ch/3co)
37	BladeCenter HS20 (8843) Xeon EM64T 2.8GHz 2MB (1ch/1co)	xSeries 455 (4U) Itanium2 1.4GHz 4MB (4ch/4co)
38	BladeCenter HS20 (8843) Xeon EM64T 2.8GHz 2MB (2ch/2co)	xSeries 455 (4U) Itanium2 1.5GHz 4MB (1ch/1co)
39	BladeCenter HS20 (8843) Xeon EM64T 3.0GHz 1MB (1ch/1co)	xSeries 455 (4U) Itanium2 1.5GHz 4MB (2ch/2co)
40	BladeCenter HS20 (8843) Xeon EM64T 3.0GHz 1MB (2ch/2co)	xSeries 455 (4U) Itanium2 1.5GHz 4MB (3ch/3co)
41	BladeCenter HS20 (8843) Xeon EM64T 3.2GHz 1MB (1ch/1co)	xSeries 455 (4U) Itanium2 1.5GHz 4MB (4ch/4co)

# OEM Servers		Default Values		Workload	
Enter #	Result	90.0%	65.0%	No.	D
4.00	4.00	30.0%	45.0%	6	Database



				Utilization for Case 1				Utilization for Case 2			
			Capacity	< Complementary		Peaks	Concurrent >	< Complementary		Peaks	Concurrent >
Processor	Feature	MSU	Rating	0%	40.0%	70.0%	100%	0%	40.0%	70.0%	100%
			Capacity required (MIPS) =	1,543	2,160	2,623	3,086	2,315	3,240	3,935	4,629
IBM zEC12 IFL											
2827-7xx I1	1W IFL		1,650	94%	131%	160%	188%	141%	197%	239%	281%
2827-7xx I2	2W IFL		3,217	48%	68%	82%	96%	72%	101%	123%	144%
2827-7xx I3	3W IFL		4,760	33%	46%	56%	65%	49%	69%	83%	98%
2827-7xx I4	4W IFL		6,281	25%	35%	42%	50%	37%	52%	63%	74%

Sizing Consolidated CPU consumption – equivalent MIPS



Memory Sizing Oracle with Linux on System z Linux

- Customer attempted install 11gR2 with 512mb – **could not re-link on install.**
  - Oracle recommends **4GB** for all Linux Platforms, **smallest we would suggest is 2GB of Virtual Memory for a Single Oracle 11g/12c instance.**

- One customer experienced 200 MB more RAM consumption 10gR2 to 11gR2

- **Right Size** the Virtual Memory based on What is needed:
  - **All SGA's (including ASM)** – consider Large Pages
  - **Oracle PGA's** (not eligible for Large Pages – small pages**)**
  - **User Connections** to the database (4.5mb per connection – small pages)
  - **Linux Page Tables** and **Linux Kernel Memory** (small pages)
  - Try NOT to oversize the Linux Guest under z/VM, use VDISKs
  - Leave room (5-10%) such that kswapd and OOM (out of mem mgr) don't kick in,

- Production workloads 1 to 1.5:1 Virtual to Physical Memory, for Test and Dev 2 to 3:1, even 4:1 are possible.

Verify I/O Performance with Oracle Orion

- Oracle ORION Simulates Oracle reads and writes, without having to create a database

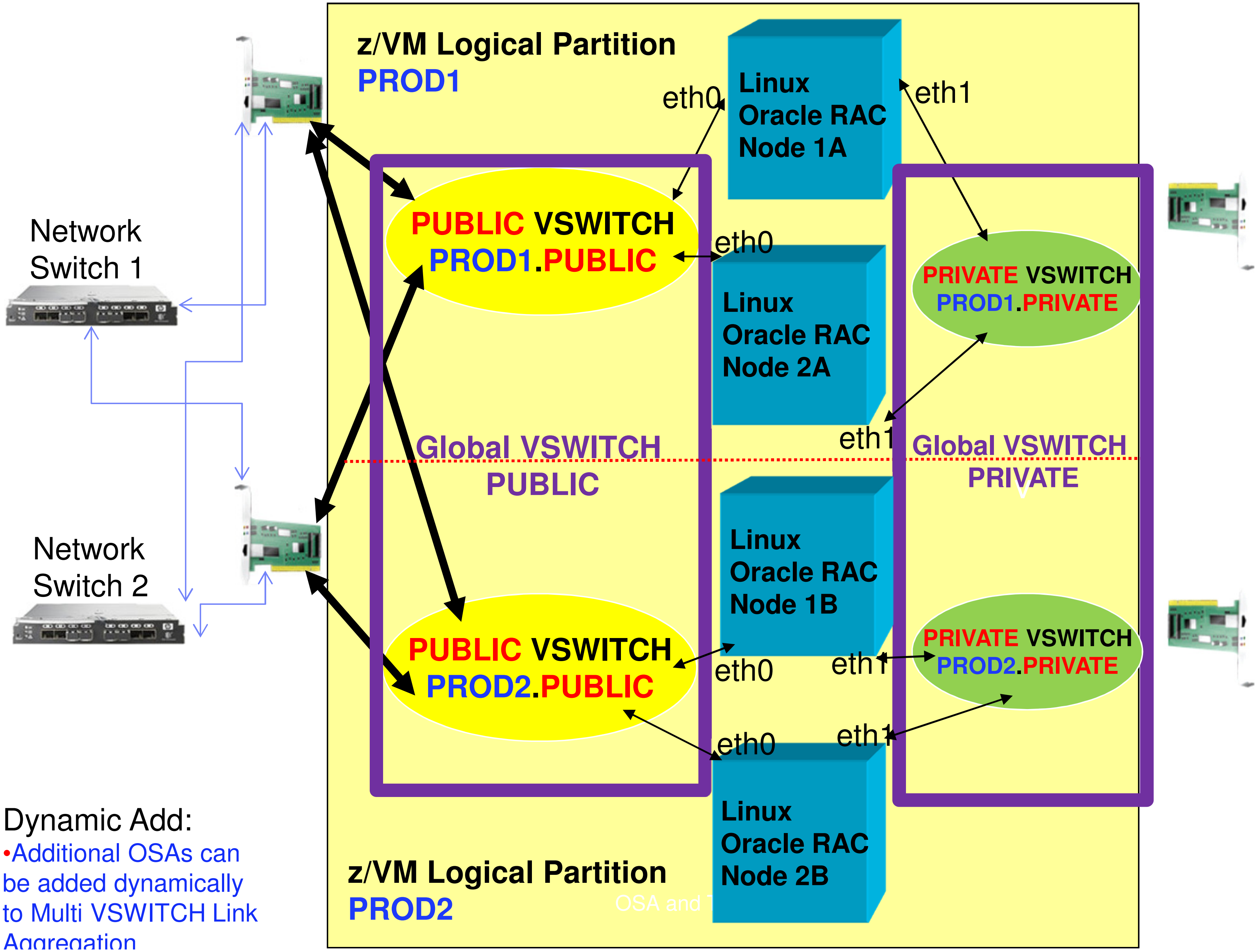
- No Longer Download from Oracle – it is now included with Oracle Code in $ORACLE_HOME/bin/orion

```
./orion_zlinux -run oltp -testname test -num_disks 2 -duration 30 -simulate raid0
ORION VERSION 11.2.0.0.1
Commandline: -run oltp -testname mytest -num_disks 2 -duration 30 -simulate raid0
This maps to this test: Test: mytest
Small IO size: 8 KB Large IO size: 1024 KB
IO Types: Small Random IOs, Large Random IOs
Simulated Array Type: RAID 0   Stripe Depth: 1024 KB
Write: 0% Cache Size: Not Entered
Duration for each Data Point: 30 seconds
Small Columns:,      2,      4,      6,      8,     10,     12,     14,     16,     18,
     20,     22,     24,     26,     28,     30,     32,     34,     36,     38,     40
Large Columns:,      0 Total Data Points: 22
Name: /dev/dasdq1       Size: 2461679616
Name: /dev/dasdr1       Size: 2461679616
2 FILEs found.
Maximum Small IOPS=5035 @ Small=40 and Large=0
Minimum Small Latency=0.55 @ Small=2 and Large=0
```

Oracle High Availability Networking Options:

- **Link Aggregation** – (Active / Active) Allow up to 8 OSA-Express adapters to be aggregated per virtual switch Each OSA-Express feature must be exclusive to the virtual switch (e.g. OSA's can now be shared NEW!
- **Linux Bonding** – create 2 Linux interfaces – e.g. **eth1** & **eth2** and create a bonded interface **bond0** made up of eth1 and eth2.
- **Oracle HAIP** – Oracle 11gR2+ can now have up to 4 Private interconnect interfaces to load balance interconnect traffic.

Oracle RAC with z/VM Multi VSWITCH LAG



Dynamic Add:

- Additional OSAs can be added dynamically to Multi VSWITCH Link Aggregation

Multi VSWITCH Link Aggregation

- z/VM 6.3 with APARS VM65583 and PI21053.
- OSA-Express4S & OSA-Express5s support for Multi-Vswitch Link Aggregation requires IBM z13
- A port group (LAG) can be connected to up to 16 LPARS (single CEC). A port group cannot span multiple CECs.
- *Please See Rick Tarcza's presentation* *http://www.vm.ibm.com/virtualnetwork/63lnkag.pdf* *for more information*

System z & IBM Flash System: Highest Reliability, Maximum Performance

Now you can leverage the "Economies of Scale" of Flash

- Easily added to your existing SAN
- Accelerate Application Performance
- Gain Greater System Utilization
- Lower Software & Hardware Cost
- Save Power / Cooling / Floor Space
- Drive Value Out of Big Data

IBM FlashSystem is certified (reference SSIC) to attach to Linux on System z, with or without an SVC, to meet your business objectives

Would you like to demo this architecture?
You can now demo hardware either in person or virtually.
Demo Location: Benchmark Center in Poughkeepsie, NY

Performance of Linux on System z with FlashSystem

I/O bound relational databases can benefit from IBM FlashSystem over spinning disks.

- **21x reduction in response times***
- **9x improvement in IO wait times***
- **2x improvement in CPU utilization***

*** IBM internal test results**

Why IBM FlashSystem for Linux on System z?

Extreme Performance

Cut IO Wait Time **80%+**

3X increase IOPS

Latency Under 100 Microseconds

Enterprise Reliability

Highest Reliability levels

Purposed-built, Enterprise Architecture

Macro Efficiency

No application Or architecture Changes

Benefits & economics out weigh disk

Reduce floor space, power & cooling

IBM MicroLatency™

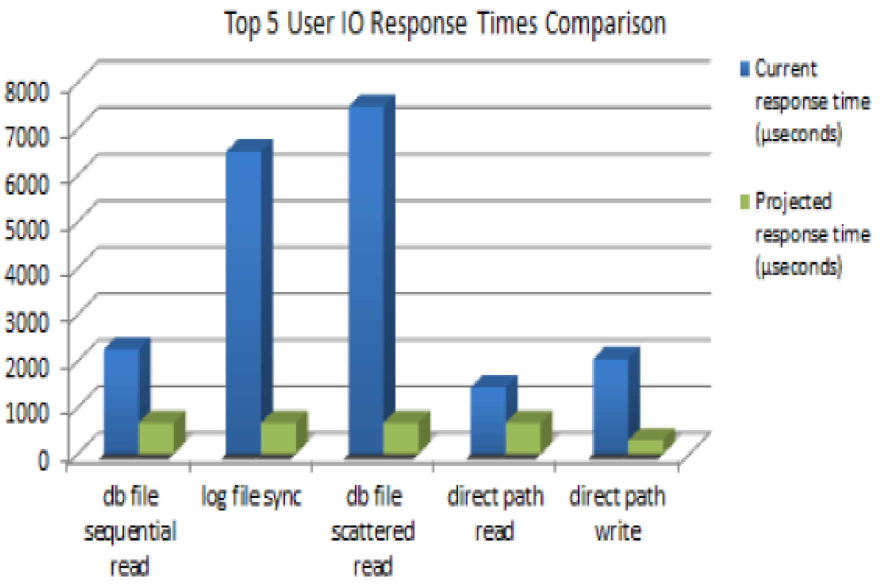
Servers, Applications and Databases are FASTER!

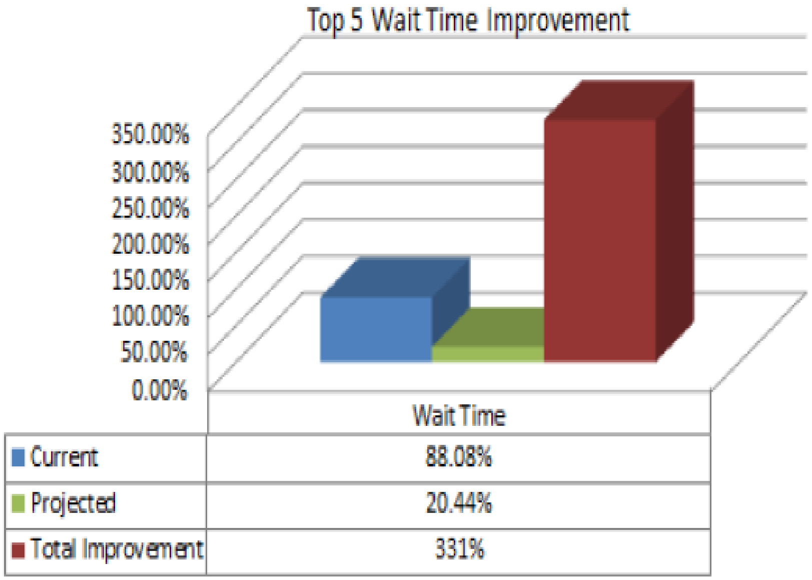
Go FROM 7 milliseconds to 700 microseconds

45X FASTER

Aggregating factors for FlashSystem implementation

- Reduce User IOWait time



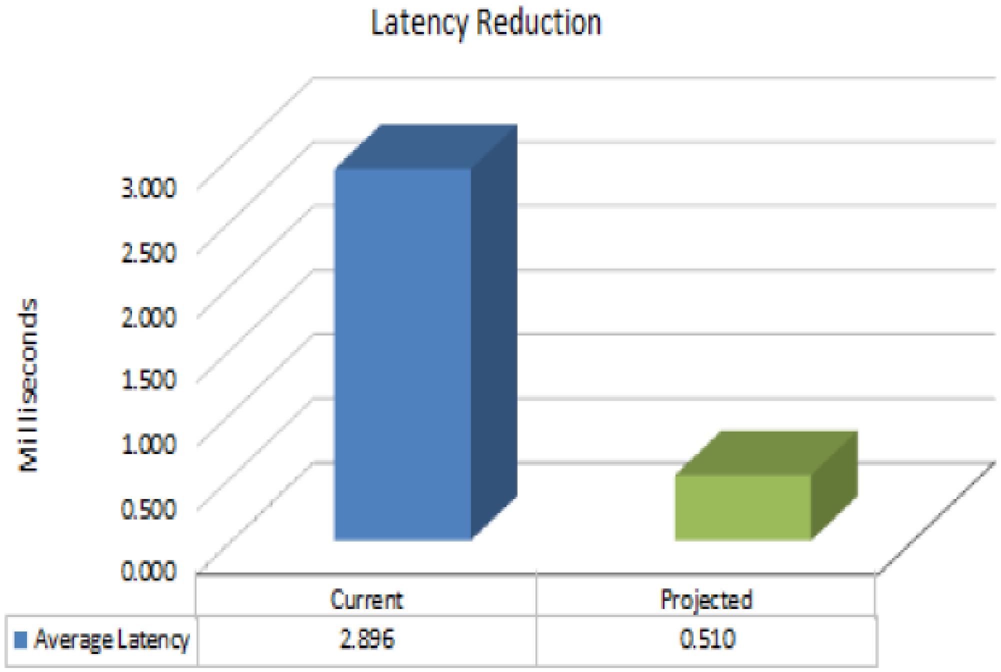


	Wait Time
🟦 Current	88.08%
🟩 Projected	20.44%
🟥 Total Improvement	331%

User IOWait events indicate a reduction in IOWait times are possible with a FlashSystem implementation. Db file sequential read is causing the majority of the disk contention across all three AWRs. The IOWait time would decrease from 88.08% of overall wait time to 22.44%, an improvement of **331%**.

Aggregating factors for FlashSystem implementation

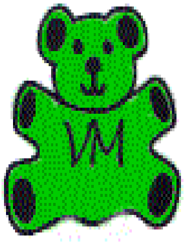
- Reduce Response Time / Latency



The microsecond response times of the FlashSystem would significantly reduce latency while driving higher utilization at the server and application level. Average Latency would decrease from 2,896 microseconds to 510 microseconds.

Oracle Certified Virtualized Platforms

- Oracle VM & IBM z/VM Hypervisors are CERTIFIED to run Oracle workloads. (IBM PowerVM, z PR/SM support LPAR virtualization as well)



- VMWARE supported but NOT certified by Oracle.
- Oracle VM cannot do memory overcommit – maximum recommended overcommit of virtual to real processors is 2:1
- IBM z/VM handles over commitment of Memory and Virtual processors very well. (You still need to conserve resources where possible!)

Source: http://www.oracle.com/technetwork/database/virtualizationmatrix-172995.html

z/VM 6.3 with SMT Enabled

```
# vmcp q mt
Multithreading is enabled.
              Requested    Activated
              Threads      Threads
MAX_THREADS    MAX
2
CP core        MAX           1
IFL core       MAX           2
ICF core       MAX           1
zIIP core      MAX           1
```

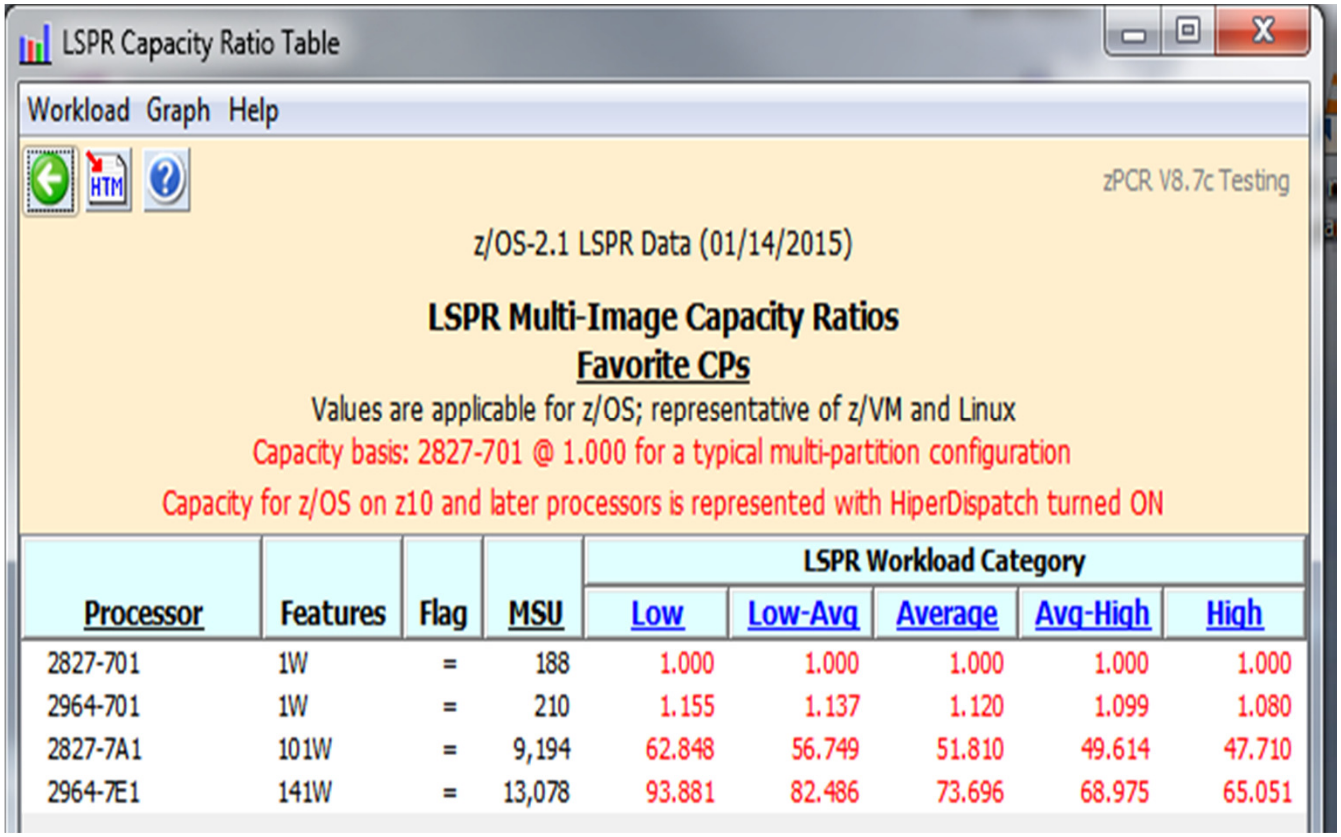
```
 cat /proc/cpuinfo
vendor_id       : IBM/S390
# processors    : 24
bogomips per cpu: 20325.00
features        : esan3 zarch stfle msa ldisp eimm dfp etf3eh highgprs
processor 0: version = FF,  identification = 05DA97,  machine = 2964
processor 1: version = FF,  identification = 05DA97,  machine = 2964
processor 2: version = FF,  identification = 05DA97,  machine = 2964
processor 3: version = FF,  identification = 05DA97,  machine = 2964
…
processor 22: version = FF,  identification = 05DA97,  machine = 2964
processor 23: version = FF,  identification = 05DA97,  machine = 2964
```

- Oracle is licensed by the # of physical CPU Cores (IFLs) in a Hard Partitioned LPAR.

- With z/VM SMT enabled the number of processors will show as the number of virtual processor threads that have been allocated and is not what is licensed on.

New! - IBM z13 CPU Performance

- Published performance improvement with out SMT (threading) is **12%** and **32%** for workloads that can benefit from SMT.
- **SMT** - Pre-install guidance based on internal testing and eventual field experience (20% for IFLs, 25% for zIIPs)
- **.For Oracle workloads were seeing performance gains consistent with these z13 SMT performance guidance.**



z/OS-2.1 LSPR Data (01/14/2015)

LSPR Multi-Image Capacity Ratios

Favorite CPs

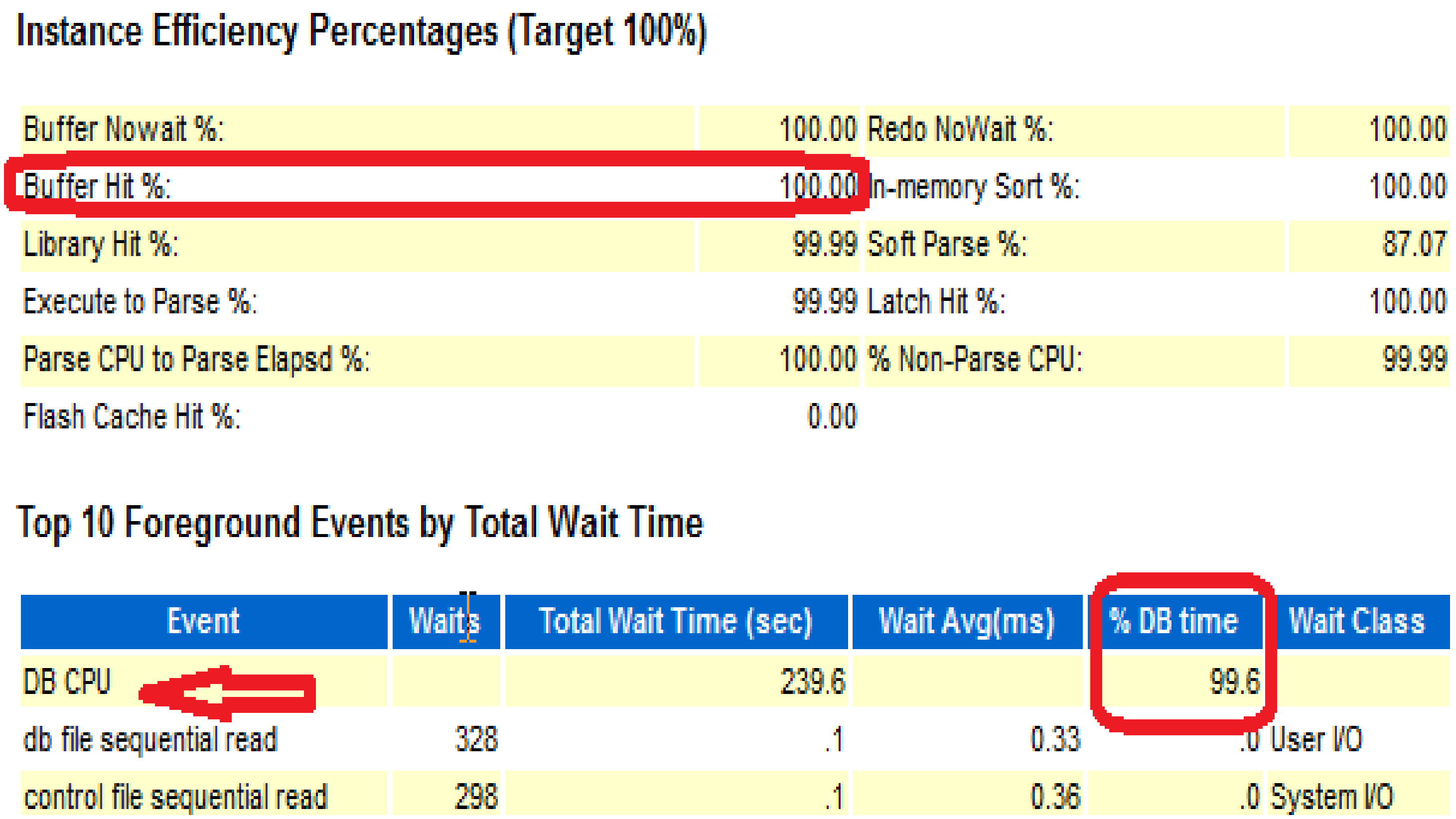
Values are applicable for z/OS; representative of z/VM and Linux

Capacity basis: 2827-701 @ 1.000 for a typical multi-partition configuration

Capacity for z/OS on z10 and later processors is represented with HiperDispatch turned ON

				LSPR Workload Category				
Processor	Features	Flag	MSU	Low	Low-Avg	Average	Avg-High	High
2827-701	1W	=	188	1.000	1.000	1.000	1.000	1.000
2964-701	1W	=	210	1.155	1.137	1.120	1.099	1.080
2827-7A1	101W	=	9,194	62.848	56.749	51.810	49.614	47.710
2964-7E1	141W	=	13,078	93.881	82.486	73.696	68.975	65.051

Testing on New z13 with 2 Dedicated IFLs



Instance Efficiency Percentages (Target 100%)

Buffer Nowait %:	100.00	Redo NoWait %:	100.00
Buffer Hit %:	100.00	In-memory Sort %:	100.00
Library Hit %:	99.99	Soft Parse %:	87.07
Execute to Parse %:	99.99	Latch Hit %:	100.00
Parse CPU to Parse Elapsd %:	100.00	% Non-Parse CPU:	99.99
Flash Cache Hit %:	0.00		

Top 10 Foreground Events by Total Wait Time

Event	Waits	Total Wait Time (sec)	Wait Avg(ms)	% DB time	Wait Class
DB CPU		239.6		99.6	
db file sequential read	328	.1	0.33	.0	User I/O
control file sequential read	298	.1	0.36	.0	System I/O

- Silly Little Oracle Benchmark (SLOB) – (Kevin Closson – author)
- Logical I/O (Random memory access to Oracle SGA)
- Want to have 99% + DB CPU and 100% Buffer Hit Ratio for a clean test from Oracle Automatic Workload Repository (AWR) Report.

zEC12 vs z13 Testing Parameters

- Test results in this presentation are my own for Educational purposes only.

- Test results should not be construed as typical for a particular customer workload.

- z/VM development recommend getting good MONWRITE data BEFORE moving to z13 and initially disable SMT if possible.

- Use the z/VM CPUMF / SMTMET tool to extract SMT metrics

http://www.vm.ibm.com/perf/reports/zvm/html/1q5smt.html

- REALLY Important to be on the recommended z/VM service and Linux kernel levels: Suse 11 SP3+ (3.0.101-0.40.1) / Red Hat 6.6+ (2.6.32-504.16.2.el6) per http://www-03.ibm.com/systems/z/os/linux/resources/testedplatforms.html

Summary

- Performance
  - Oracle runs well on System z for both memory access (Logical I/O)
  - Integration with Flash Systems allows Oracle to run well with Physical I/Os

- Consolidation
  - z/VM can virtualize / overcommit resources well.
  - System z can run Oracle at very high cpu utilization rates with little degradation.
  - System z can dynamically add system resources (memory, network, cpu)

- Highly Available
  - System z runs Oracle workloads highly available (hardware) and in some cases can avoid configuring Oracle RAC for availability.
  - Linux HA solutions can be leveraged to increase application availability.

- Security
  - Oracle on System z can be ran highly secure with FIPs (US Govt.) 140-2 compliance at z/VM and Oracle levels.
  - SSL Crypto card support for Oracle SQL*net network traffic.

Questions?

17359: Reducing CPU Consumption with Oracle on IBM z Systems for Extreme Consolidation