

RED HAT :: CHICAGO :: 2009

SUMMIT

FOLLOW US:

TWITTER.COM/REDHATSUMMIT

TWEET ABOUT US:

ADD #SUMMIT AND/OR #JBOSSWORLD TO THE END
OF YOUR EVENT-RELATED TWEET

presented by



RED HAT :: CHICAGO :: 2009

SUMMIT

Performance Analysis & Tuning of Red Hat Enterprise Linux

Larry Woodman / John Shakshober
Kernel/ Consulting Engineer
Red Hat

September 2009

Agenda

Section 1 – System Overview

Section 2 - Analyzing System Performance

Section 3 - Tuning Red Hat Enterprise Linux

Section 4 – Performance Analysis and Tuning Examples

References

Processors Supported/Tested

RHEL4 Limitations

x86 – 32

x86_64 – 8, 64(LargeSMP)

ia64 – 64, 512(SGI)

RHEL5 Limitations

x86 – 32

x86_64 – 255

ia64 – 64, 1024(SGI)

Processor types

Uni-Processor

Symmetric Multi Processor

Multi-Core

Symmetric Multi-Thread(Hyper threaded)

Combinations

Processor types & locations

```
[root@intel-s3e36-01 node1]# cat /proc/cpuinfo
processor          : 0  <logical cpu #>

physical id      : 0  <socket #>

siblings         : 16 <logical cpus per socket>

core id          : 0  <core # in socket>

cpu cores        : 8  <physical cores per socket>

# cat /sys/devices/system/node/node*/cpulist
node0: 0-3
node1: 4-7
```

NUMA & Multi Core Support

Cpusets (2.6.12)

Enable CPU & Memory assignment to sets of tasks

Allow dynamic job placement on large systems

Numa-aware slab allocator (2.6.14)

Optimized locality & management of slab creation

Swap migration. (2.6.16)

Swap migration relocates physical pages between nodes in a NUMA system while the process is running – improves performance

Huge page support for NUMA (2.6.16)

Netfilter ip_tables: NUMA-aware allocation (2.6.16)

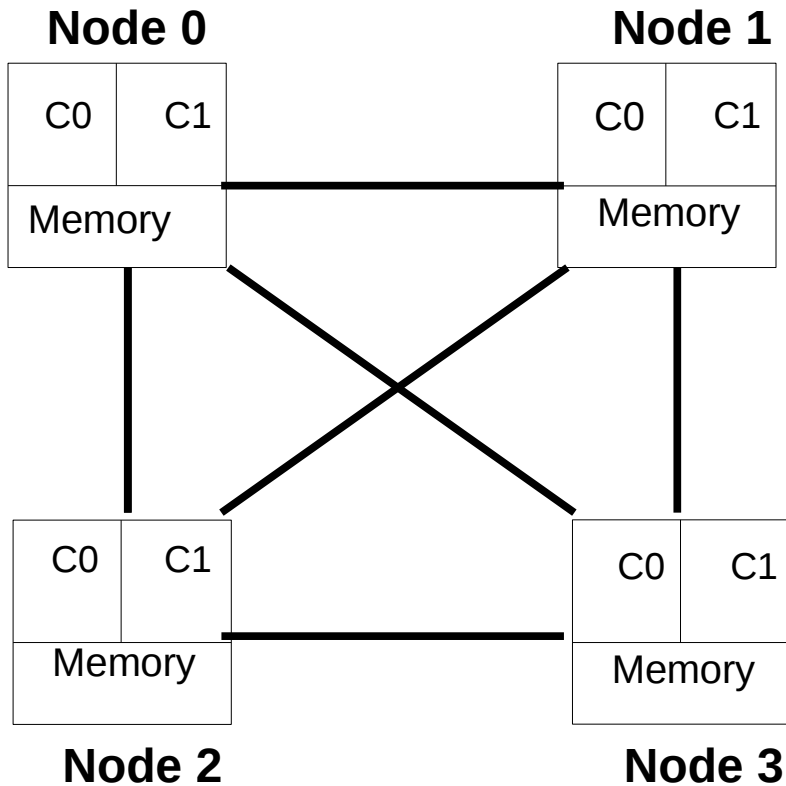
Multi-core

Scheduler improvements for shared-cache multi-core systems (2.6.17)

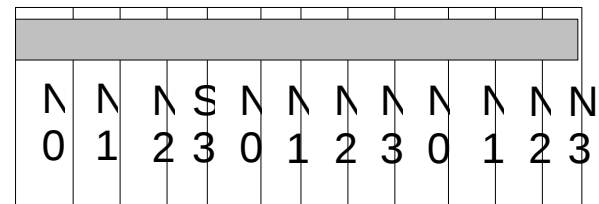
Scheduler power saving policy

Power consumption improvements through optimized task spreading

Typical NUMA System Layout

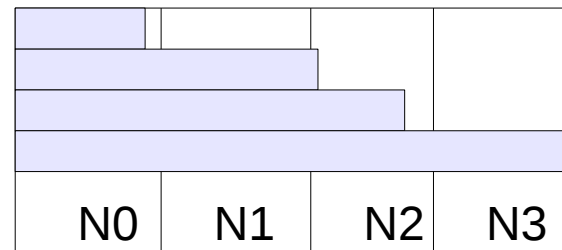


Process memory on N1C0



interleaved (Non-NUMA)

Process memory on N1C0



Non-Interleaved (NUMA)

NUMA Support

RHEL4 NUMA Support

- NUMA aware memory allocation policy

- NUMA aware memory reclamation

- Multi-core support

RHEL5 NUMA Support

- RHEL4 NUMA support (taskset, numactl)

- NUMA aware scheduling

- NUMA-aware slab allocator

- NUMA-aware hugepages

Memory Management

Physical Memory(RAM) Management

Virtual Address Space Maps

Kernel Wired Memory

Reclaimable User Memory

Page Reclaim Dynamics

Physical Memory(RAM) Management

Physical Memory Layout

NUMA versus Non-NUMA(UMA)

NUMA Nodes

Zones

mem_map array

Page lists

Free

Active

Inactive

Physical Memory Supported/Tested

RHEL4

x86 – 4GB, 16GB, 64GB

x86_64 – 512GB

ia64 – 1TB

RHEL5

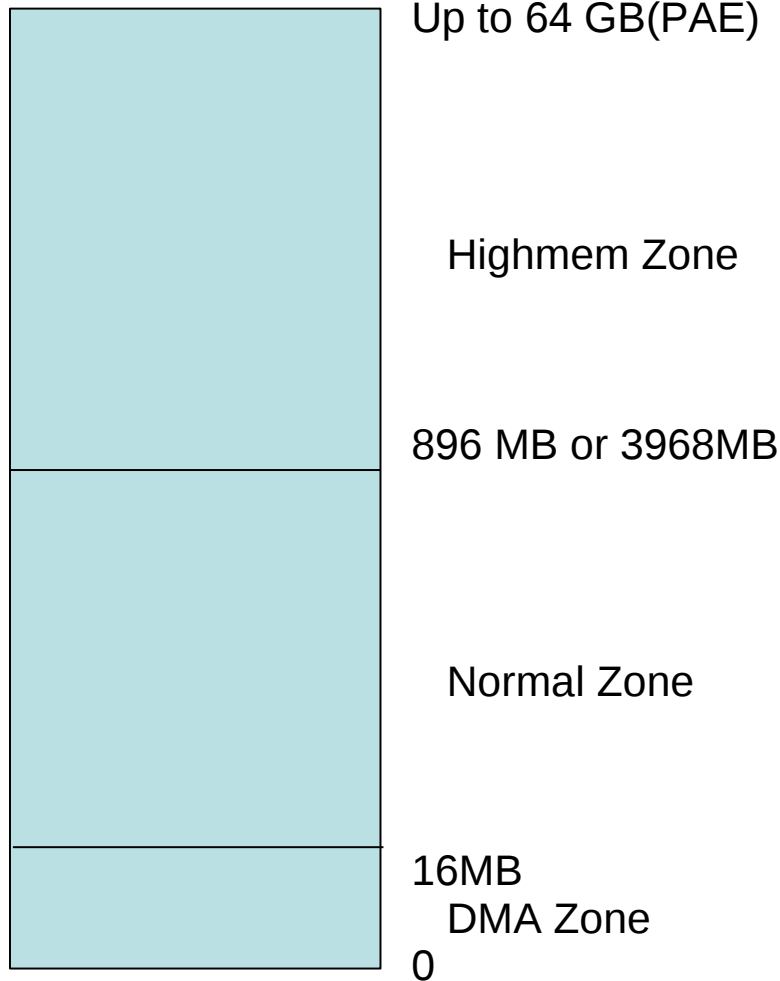
x86 – 4GB, 16GB

x86_64 – 512GB/1TB

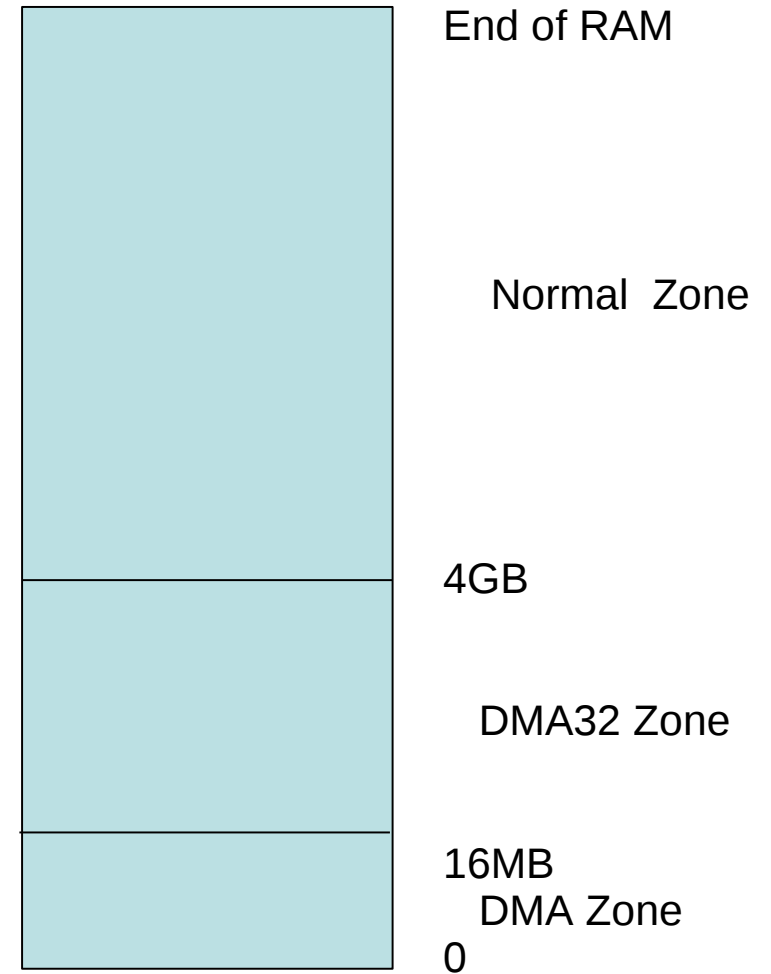
ia64 - 2TB

Memory Zones

32-bit



64-bit



Memory Zone Utilization_(x86)

DMA	Normal	(Highmem x86)
-----	--------	---------------

24bit I/O

Kernel Static
Kernel Dynamic
slabcache
bounce buffers
driver allocations
User Overflow

User
Anonymous
Pagecache
Pagetables

Memory Zone Utilization(x86_64)

DMA	DMA32	Normal
-----	-------	--------

24bit I/O

**32bit I/O
Normal overflow**

**Kernel Static
Kernel Dynamic
slabcache
bounce buffers
driver allocations**

**User
Anonymous
Pagecache
Pagetables**

Per-Zone Resources

RAM

mem_map

Page lists: free, active and inactive

Page allocation and reclamation

Page reclamation watermarks

mem_map

Kernel maintains a “page” struct for each 4KB(16KB on IA64 and 64KB for PPC64/RHEL5) page of RAM

mem_map is the global array of page structs

Page struct size(x86, x86_64):

32-bit = 32bytes

64-bit = 56bytes

Per-zone page lists

Active List - most recently referenced

Anonymous-stack, heap, bss

Pagecache-filesystem data/meta-data

Inactive List - least recently referenced

Dirty-modified

writeback in progress

Clean-ready to free

Free

Coalesced buddy allocator

Per zone Free list/buddy allocator lists

Kernel maintains per-zone free list

Buddy allocator coalesces free pages into larger physically contiguous pieces

DMA

1*4kB 4*8kB 6*16kB 4*32kB 3*64kB 1*128kB 1*256kB 1*512kB 0*1024kB 1*2048kB 2*4096kB = 11588kB)

Normal

217*4kB 207*8kB 1*16kB 1*32kB 0*64kB 1*128kB 1*256kB 1*512kB 0*1024kB 0*2048kB 0*4096kB = 3468kB)

HighMem

847*4kB 409*8kB 17*16kB 1*32kB 1*64kB 1*128kB 1*256kB 1*512kB 0*1024kB 0*2048kB 0*4096kB = 7924kB)

Memory allocation failures

Freelist exhaustion.

Freelist fragmentation.

Per NUMA-Node Resources

Memory zones(DMA & Normal zones)

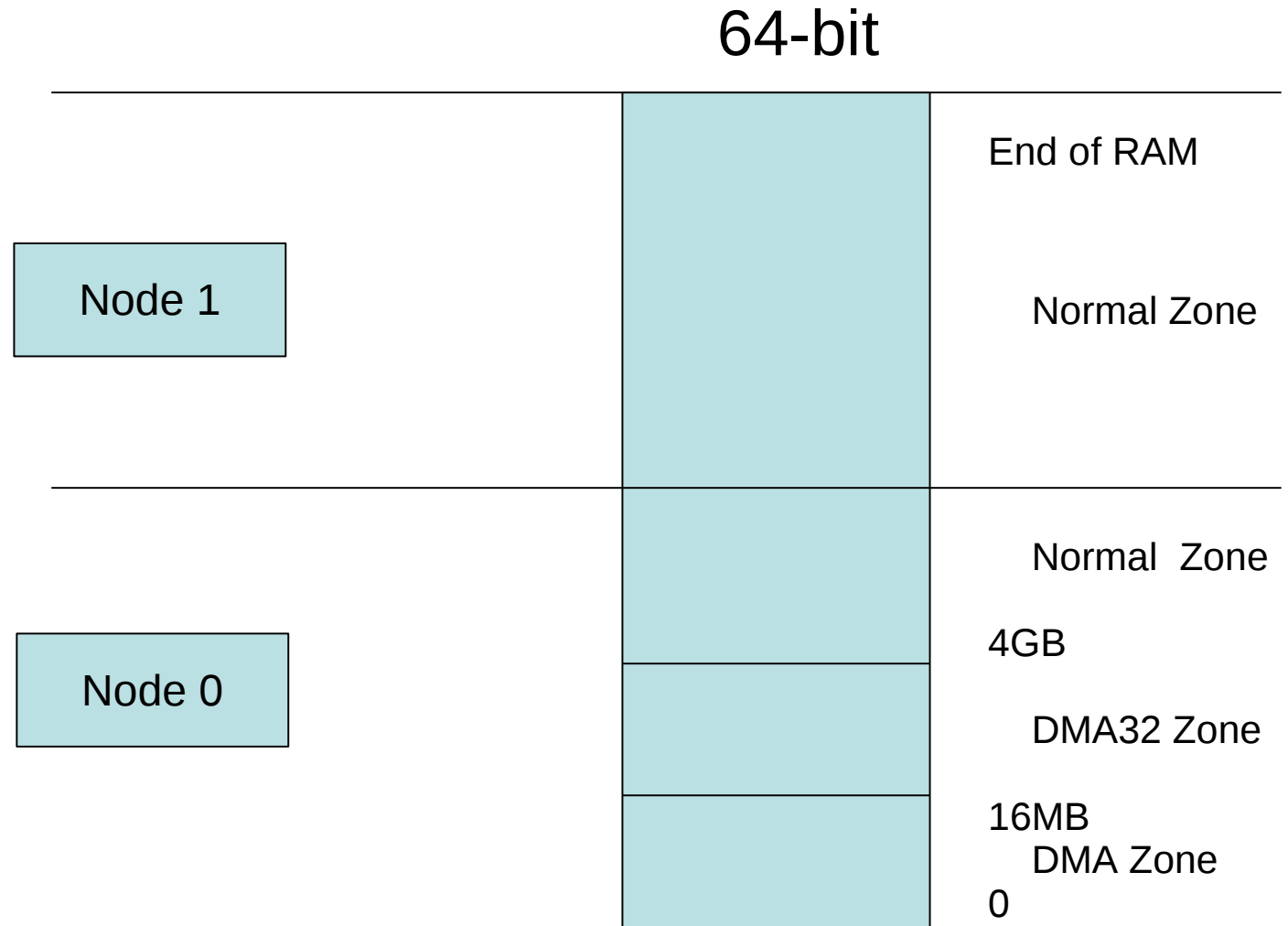
CPUs

IO/DMA capacity

Interrupt processing

Page reclamation kernel thread(kswapd#)

NUMA Nodes and Zones



Virtual Address Space Maps

32-bit

3G/1G address space

4G/4G address space(RHEL4 only)

64-bit

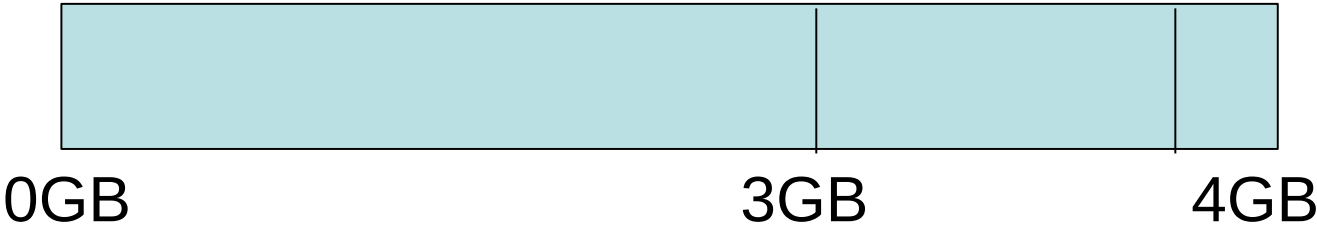
X86_64

IA64

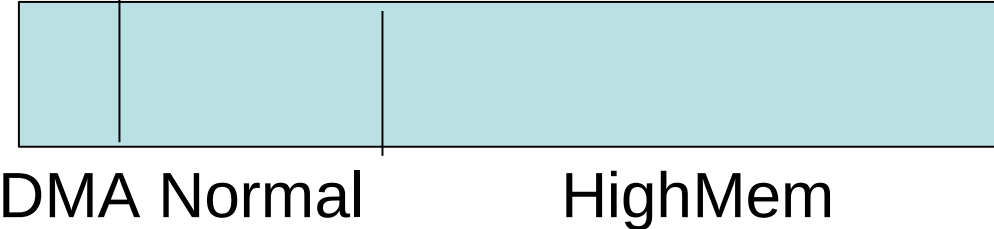
Linux 32-bit Address Spaces(SMP)

Virtual

3G/1G Kernel(SMP)



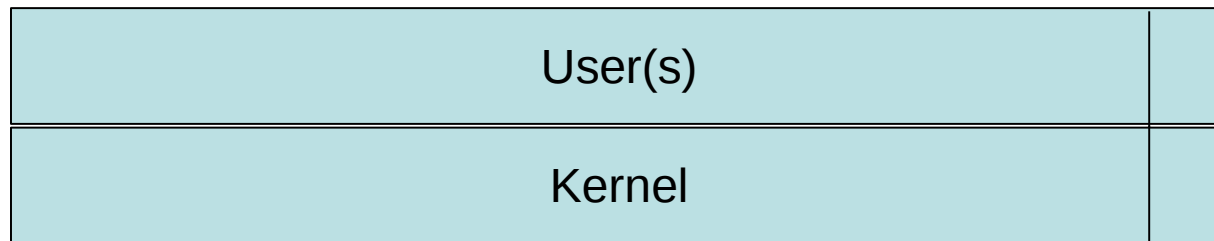
RAM



RHEL4 32-bit Address Space(Hugemem)

Virtual

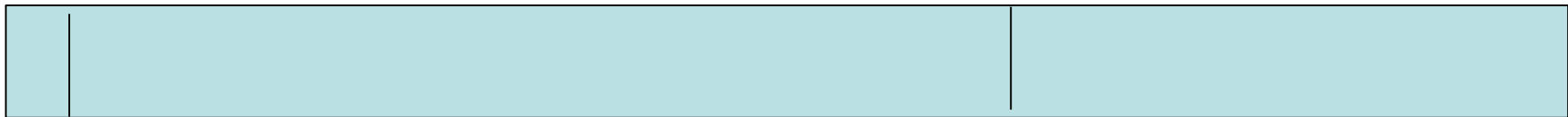
4G/4G Kernel(Hugemem)



0 GB

3968MB

RAM



DMA

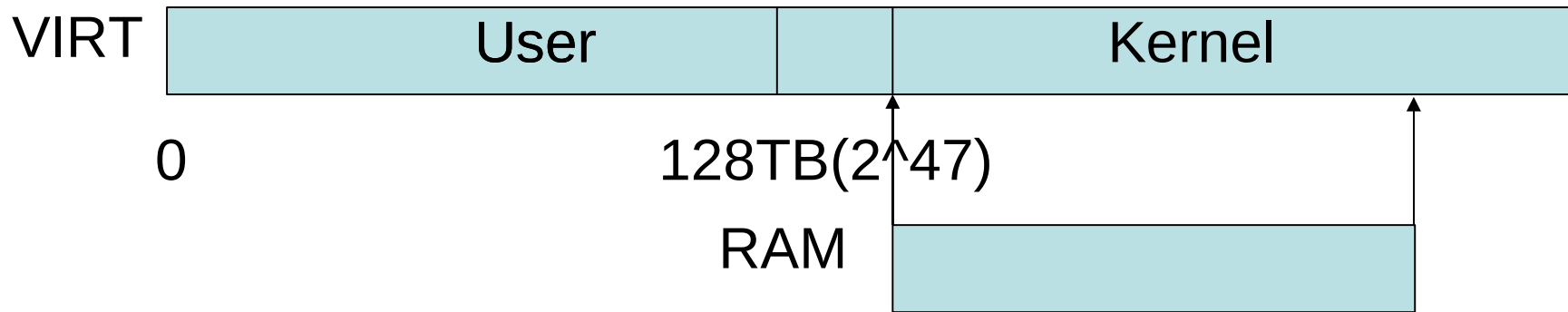
Normal

3968MB

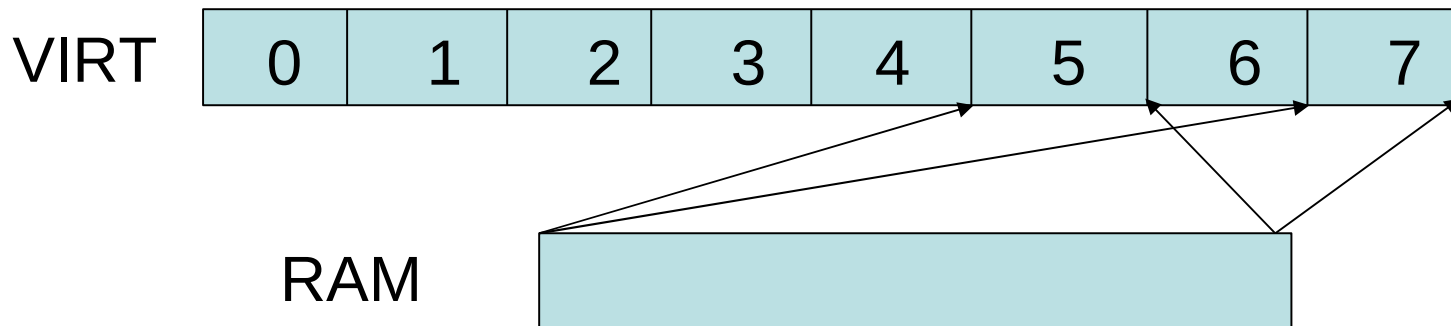
HighMem

Linux 64-bit Address Space

x86_64



IA64



Memory Pressure

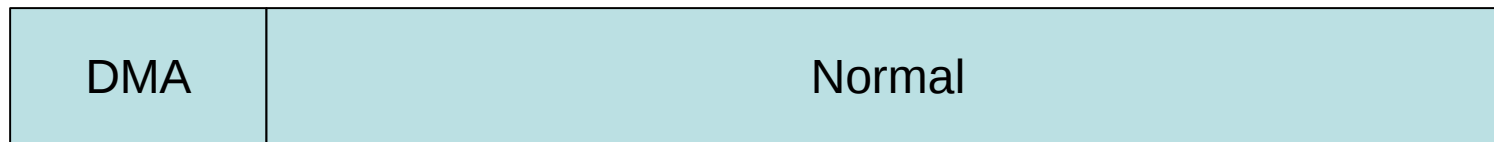
32- bit



Kernel Allocations

User Allocations

64- bit



Kernel and User Allocations

Kernel Memory Pressure

Static – Boot-time(DMA and Normal zones)

Kernel text, data, BSS

Bootmem allocator, tables and hashes(mem_map)

Dynamic

Slabcache(Normal zone)

Kernel data structs

Inode cache, dentry cache and buffer header dynamics

Pagetables(Highmem/Normal zone)

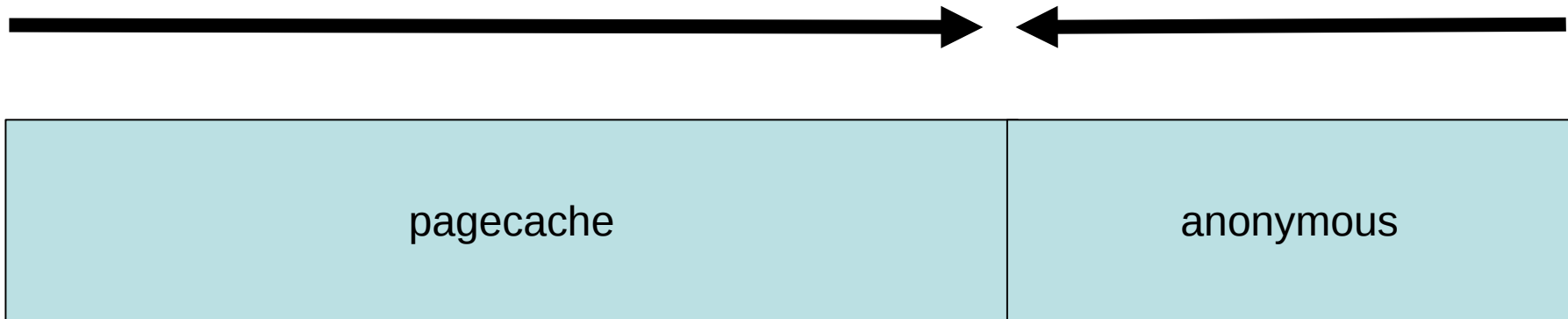
HughTLBfs(Highmem/Normal zone)

User Memory Pressure

Anonymous/pagecache split

Pagecache Allocations

Page Faults



PageCache/Anonymous memory split

Pagecache memory is global and grows when filesystem data is accessed until memory is exhausted.

Pagecache is freed:

- Underlying files are deleted.

- Unmount of the filesystem.

- Kswapd reclaims pagecache pages when memory is exhausted.

- `/proc/sys/vm/drop_caches`

Anonymous memory is private and grows on user demand

- Allocation followed by pagefault.

- Swapin.

Anonymous memory is freed:

- Process unmaps anonymous region or exits.

- Kswapd reclaims anonymous pages (swapout) when memory is exhausted

PageCache/Anonymous memory split

Balance between pagecache and anonymous memory.

Dynamic.

Controlled via:

`/proc/sys/vm/pagecache.`

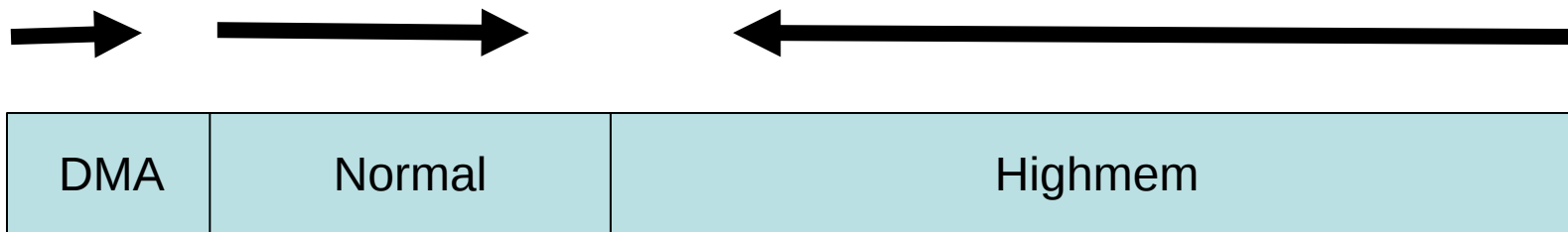
`/proc/sys/vm/swappiness.`

Swap files/partitions.

32-bit Memory Reclamation

Kernel Allocations

User Allocations



**Kernel Reclamation
(kswapd)**
slapcache reaping
inode cache pruning
bufferhead freeing
dentry cache pruning

**User Reclamation
(kswapd/pdflush)**
page aging
pagecache shrinking
swapping

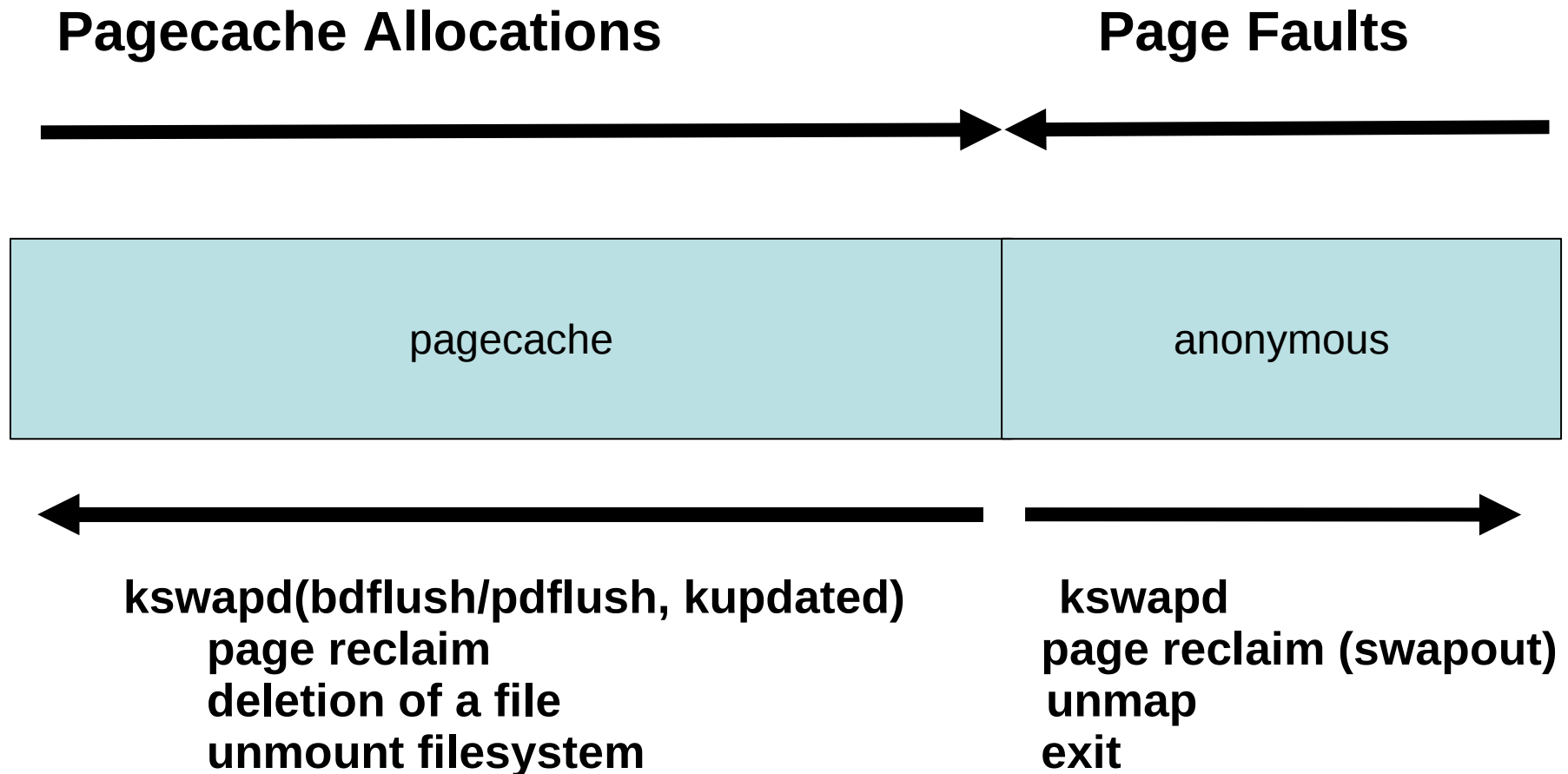
64-bit Memory Reclamation



Kernel and User Allocations

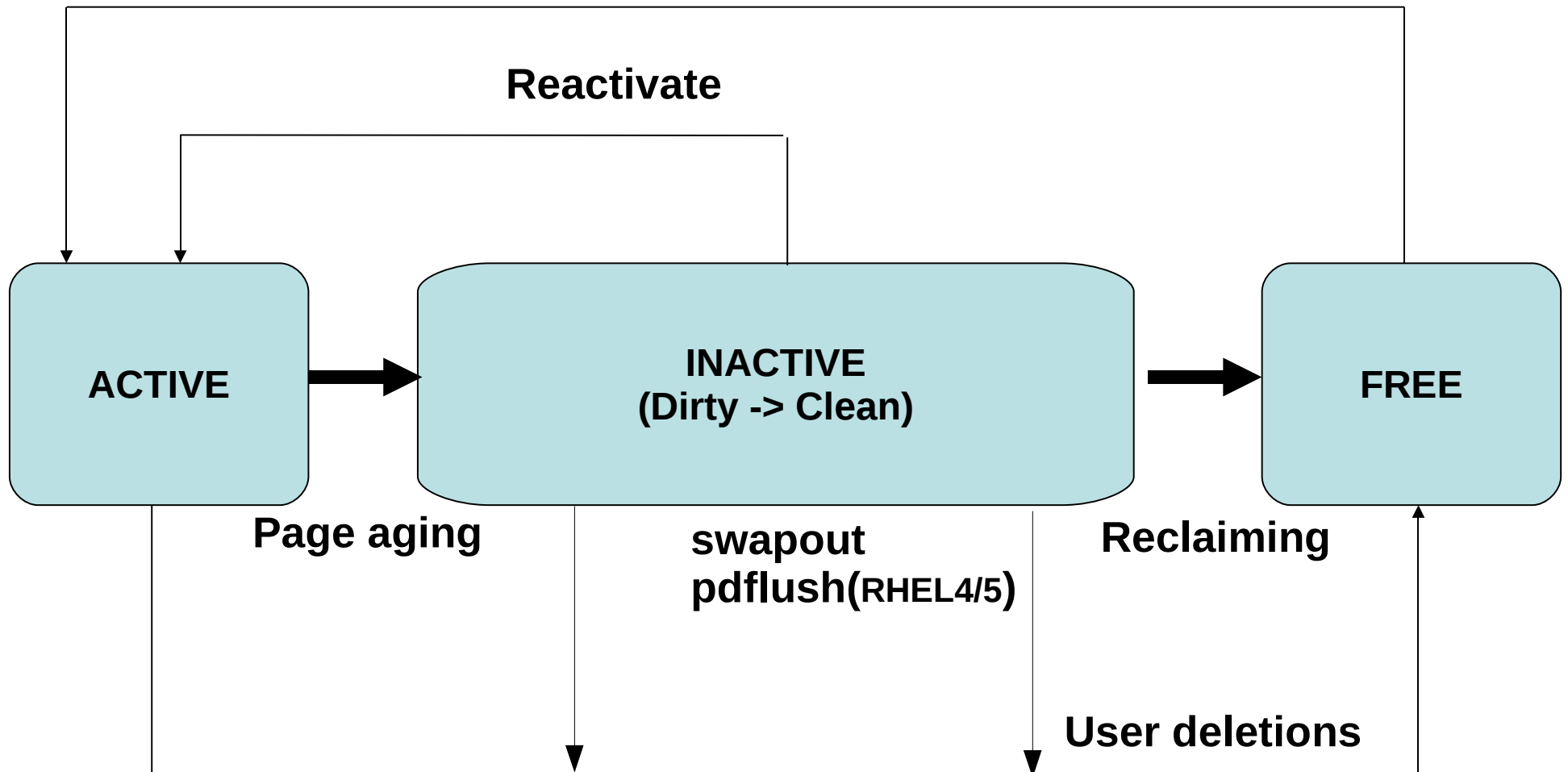


Anonymous/pagecache reclaiming



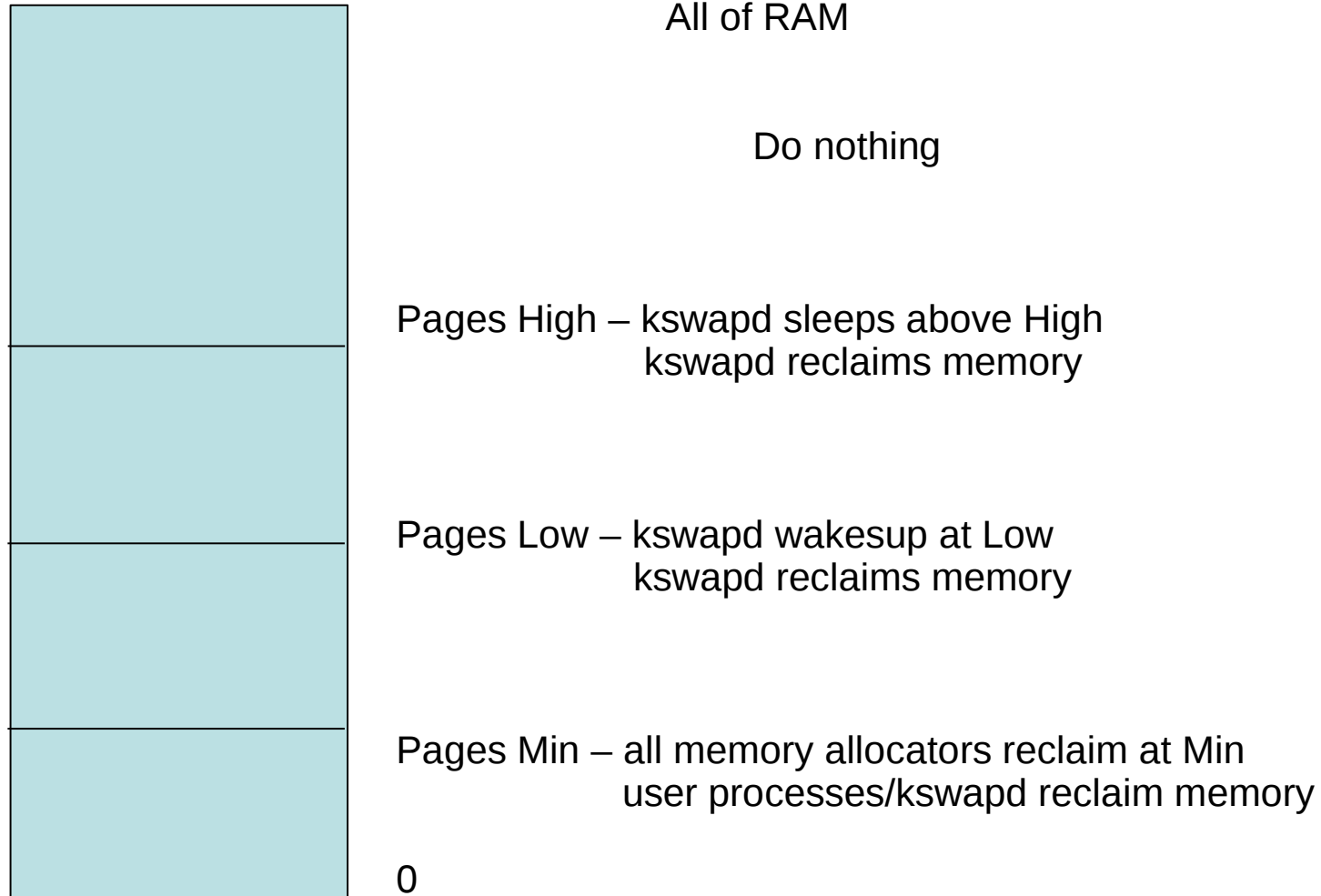
Per Node/Zone Paging Dynamics

User Allocations

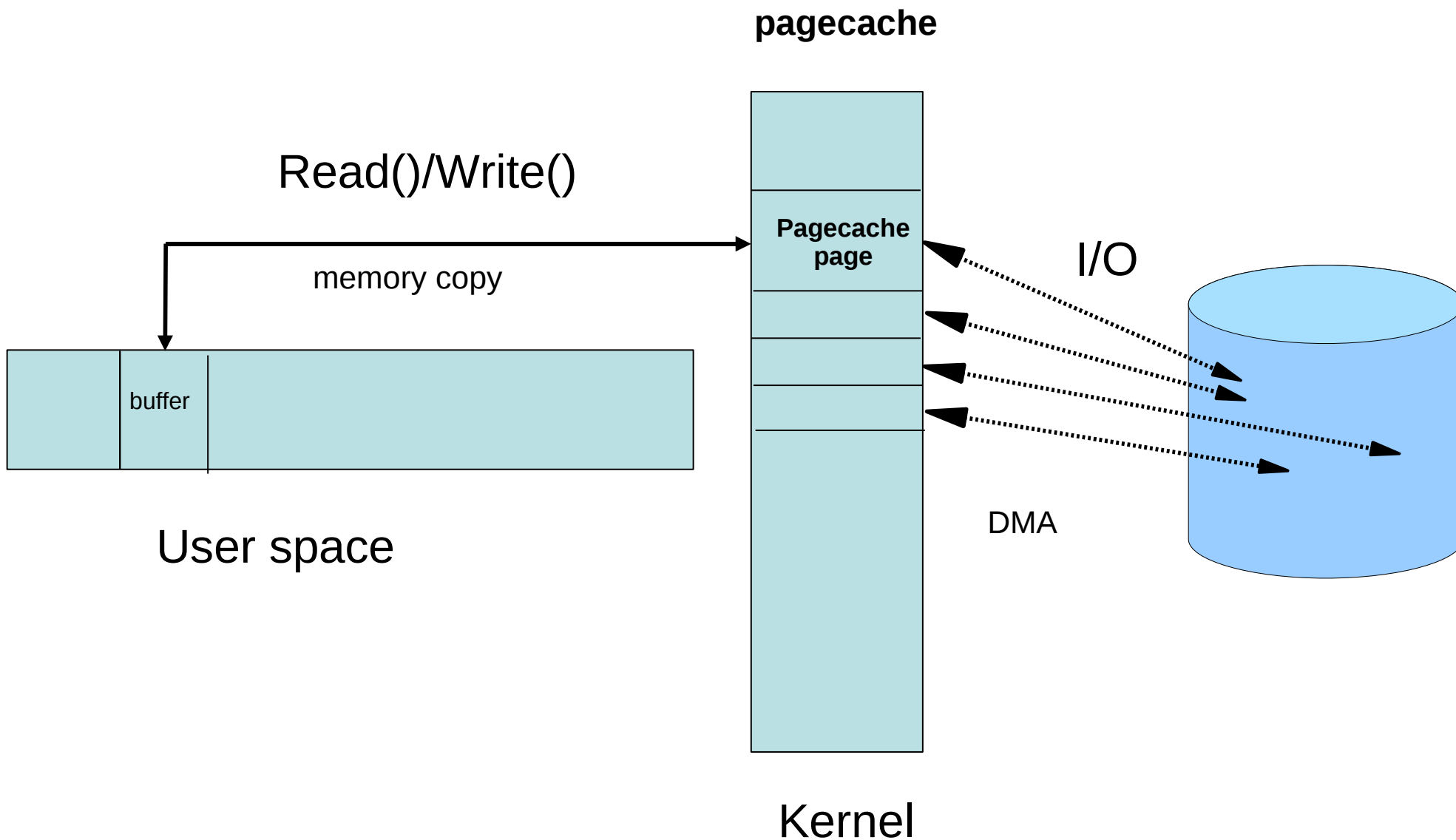


Memory reclaim Watermarks

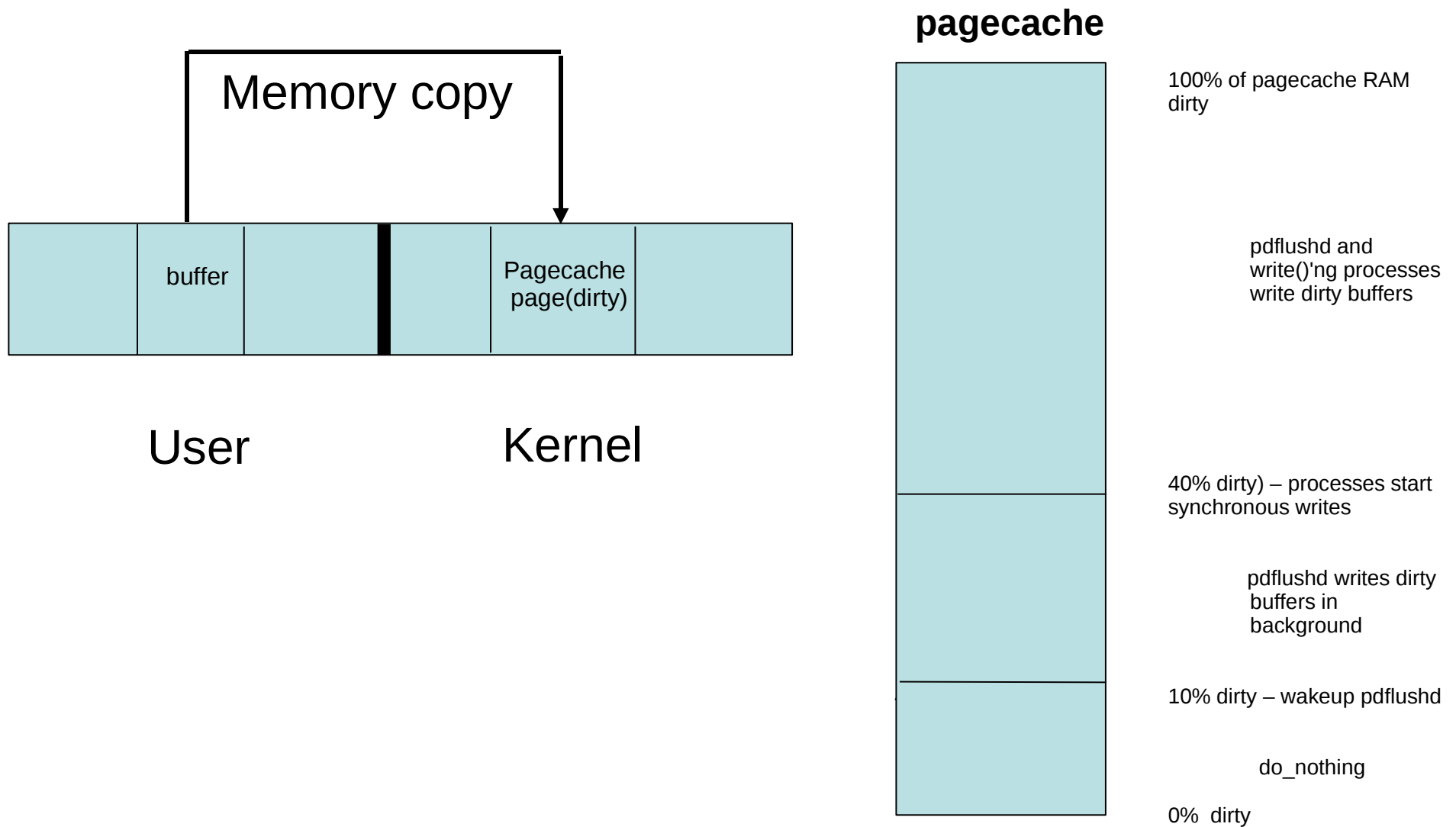
Free List



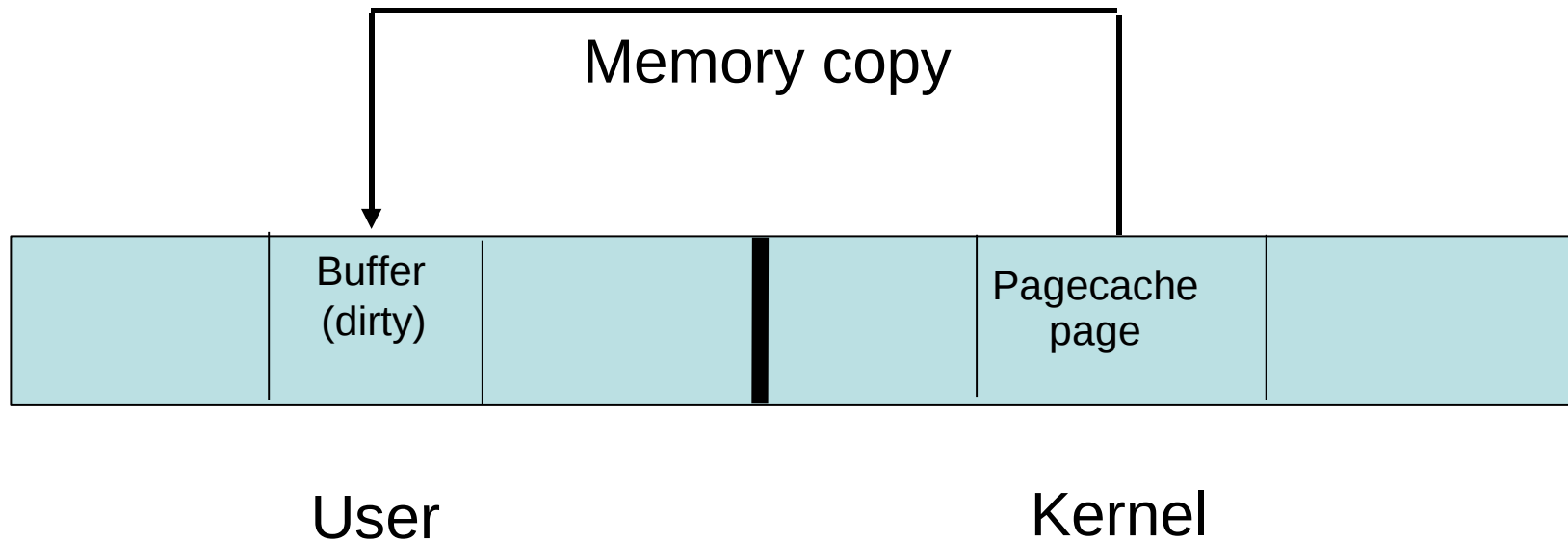
File System & Disk IO



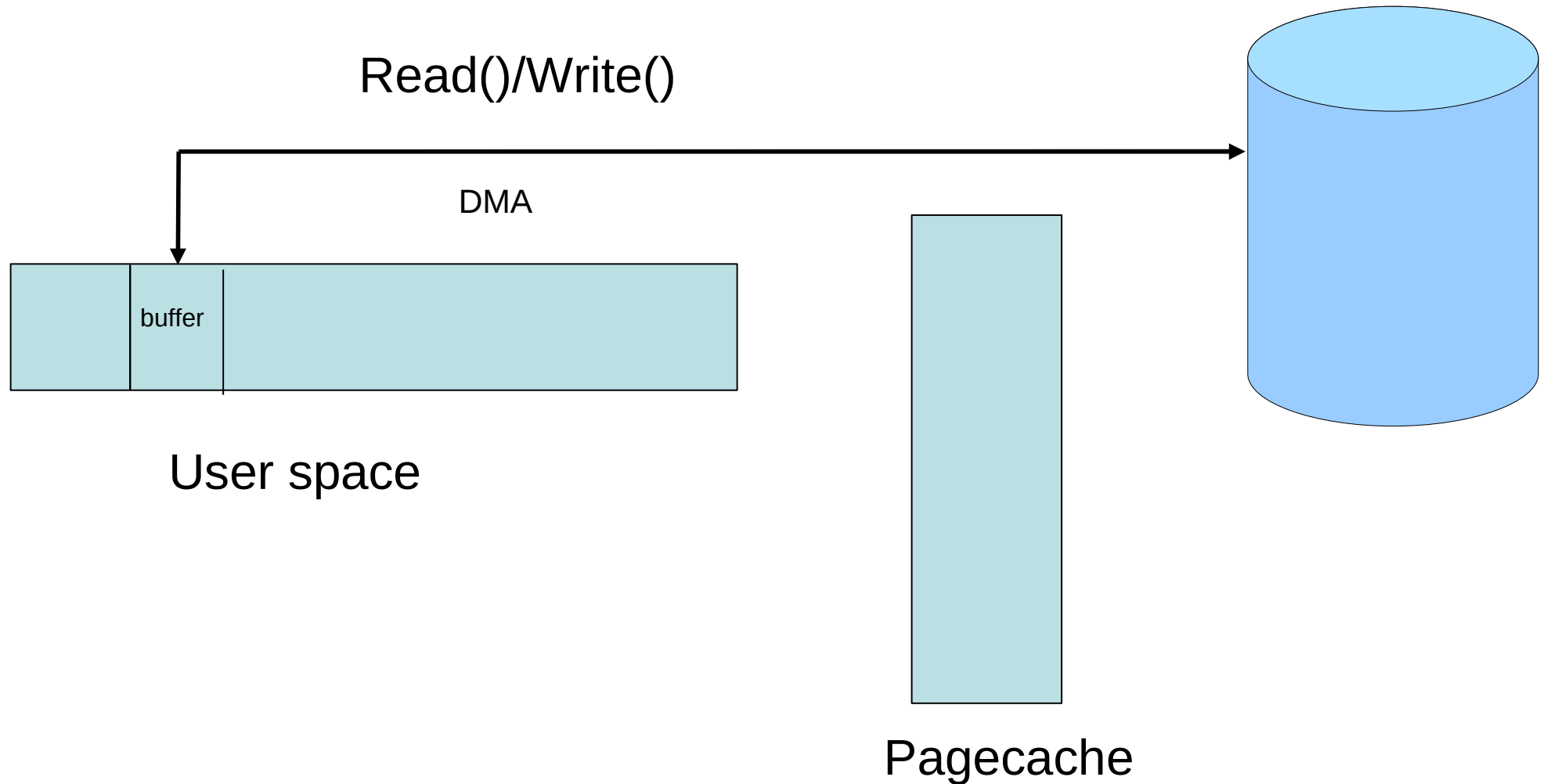
Buffered file system write



Buffered file system read



DirectIO file system read()/write()



Section 2 - Analyzing System Performance

Performance Monitoring Tools

What to run under certain loads

Analyzing System Performance

What to look for

Performance Monitoring Tools

- Standard Unix OS tools
 - Monitoring - cpu, memory, process, disk
 - oprofile
- Kernel Tools
 - /proc, info (cpu, mem, slab), dmesg, AltSysrq
- Networking
 -
- Profiling
 - nmi_watchdog=1, profile=2
 - Tracing strace, ltrace
 - dprobe, kprobe
- 3rd party profiling/ capacity monitoring
 - Perfmon, Caliper, vtune
 - SARcheck, KDE, BEA Patrol, HP Openview

Red Hat Top Tools

CPU Tools

- 1 – top
- 2 – vmstat
- 3 – ps aux
- 4 – mpstat -P all
- 5 – sar -u
- 6 – iostat
- 7 – oprofile
- 8 – gnome-system-monitor
- 9 – KDE-monitor
- 10 – /proc

Memory Tools

- 1 – top
- 2 – vmstat -s
- 3 – ps aux
- 4 – ipcs
- 5 – sar -r -B -W
- 6 – free
- 7 – oprofile
- 8 – gnome-system-monitor
- 9 – KDE-monitor
- 10 – /proc

Process Tools

- 1 – top
- 2 – ps -o pmem
- 3 – gprof
- 4 – strace,ltrace
- 5 – sar

Disk Tools

- 1 – iostat -x
- 2 – vmstat - D
- 3 – sar -DEV #
- 4 – nfsstat
- 5 – NEED MORE!

Monitoring Tools

mpstat – reveals per cpu stats, Hard/Soft Interrupt usage

vmstat – vm page info, context switch, total ints/s, cpu

netstat – per nic status, errors, statistics at driver level

lspci – list the devices on pci, indepth driver flags

oprofile – system level profiling, kernel/driver code

modinfo – list information about drivers, version, options

sar – collect, report, save system activity information

Many others available- iptraf, wireshark, etc

Sample use for some of these embedded in talk

top - press h – help,1-show cpus, m-memory, t-threads, > - column sort

top - 09:01:04 up 8 days, 15:22, 2 users, load average: 1.71, 0.39, 0.12

Tasks: 114 total, 1 running, 113 sleeping, 0 stopped, 0 zombie

Cpu0 : 5.3% us, 2.3% sy, 0.0% ni, 0.0% id, 92.0% wa, 0.0% hi, 0.3% si

Cpu1 : 0.3% us, 0.3% sy, 0.0% ni, 89.7% id, 9.7% wa, 0.0% hi, 0.0% si

Mem: 2053860k total, 2036840k used, 17020k free, 99556k buffers

Swap: 2031608k total, 160k used, 2031448k free, 417720k cached

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
27830	oracle	16	0	1315m	1.2g	1.2g	D	1.3	60.9	0:00.09	oracle
27802	oracle	16	0	1315m	1.2g	1.2g	D	1.0	61.0	0:00.10	oracle
27811	oracle	16	0	1315m	1.2g	1.2g	D	1.0	60.8	0:00.08	oracle
27827	oracle	16	0	1315m	1.2g	1.2g	D	1.0	61.0	0:00.11	oracle
27805	oracle	17	0	1315m	1.2g	1.2g	D	0.7	61.0	0:00.10	oracle
27828	oracle	15	0	27584	6648	4620	S	0.3	0.3	0:00.17	tpcc.exe
1	root	16	0	4744	580	480	S	0.0	0.0	0:00.50	init
2	root	RT	0	0	0	0	S	0.0	0.0	0:00.11	migration/0
3	root	34	19	0	0	0	S	0.0	0.0	0:00.00	ksoftirqd/0

vmstat(paging vs swapping)

Vmstat 10

procs		memory				swap		io		system			cpu		
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	wa	id
2	0	0	5483524	200524	234576	0	0	54	63	152	513	0	3	0	96
0	2	0	1697840	200524	3931440	0	0	578	50482	1085	3994	1	22	14	63
3	0	0	7844	200524	5784109	0	0	59330	58946	3243	14430	7	32	18	42

Vmstat 10

procs		memory				swap		io		system			cpu		
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	wa	id
2	0	0	5483524	200524	234576	0	0	54	63	152	513	0	3	0	96
0	2	0	1662340	200524	234576	0	0	578	50482	1085	3994	1	22	14	63
3	0	235678	7384	200524	234576	18754	23745	193	58946	3243	14430	7	32	18	42

Vmstat - IOzone(8GB file with 6GB RAM)

#! deplete memory until pdflush turns on

procs		memory				swap		io		system			cpu		
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	wa	id
2	0	0	4483524	200524	234576	0	0	54	63	152	513	0	3	0	96
0	2	0	1697840	200524	2931440	0	0	578	50482	1085	3994	1	22	14	63
3	0	0	1537884	200524	3841092	0	0	193	58946	3243	14430	7	32	18	42
0	2	0	528120	200524	6228172	0	0	478	88810	1771	3392	1	32	22	46
0	1	0	46140	200524	6713736	0	0	179	110719	1447	1825	1	30	35	35
2	2	0	50972	200524	6705744	0	0	232	119698	1316	1971	0	25	31	44

#! now transition from write to reads

procs		memory				swap		io		system			cpu		
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	wa	id
1	4	0	51040	200524	6705544	0	0	2	133519	1265	839	0	26	56	18
1	1	0	35064	200524	6712724	0	0	40	118911	1367	2021	0	35	42	23
0	1	0	68264	234372	6647020	0	0	76744	54	2048	4032	0	7	20	73
0	1	0	34468	234372	6678016	0	0	77391	34	1620	2834	0	9	18	72
0	1	0	47320	234372	6690356	0	0	81050	77	1783	2916	0	7	20	73
1	0	0	38756	234372	6698344	0	0	76136	44	2027	3705	1	9	19	72
0	1	0	31472	234372	6706532	0	0	76725	33	1601	2807	0	8	19	73

iostat -x of same IOzone EXT3 file system

Iostat metrics

rates per sec

sizes and response time

r|w rqm/s – request merged/s

averq-sz – average request sz

r|w sec/s – 512 byte sectors/s

avequ-sz – average queue sz

r|w KB/s – Kilobyte/s

await – average wait time ms

r|w /s – operations/s

svctm – ave service time m

Linux 2.4.21-27.0.2.ELsmp (node1)

avg-cpu:	%user	%nice	%sys	%iowait	%idle
	0.40	0.00	2.63	0.91	96.06

Device:	rrqm/s	wrqm/s	r/s	w/s	rsec/s	wsec/s	rkB/s	wkB/s	avgrq-sz	avgqu-sz	await	svctm	%util
sdi	16164.60	0.00	523.40	0.00	133504.00	0.00	66752.00	0.00	255.07	1.00	1.91	1.88	98.40
sdi	17110.10	0.00	553.90	0.00	141312.00	0.00	70656.00	0.00	255.12	0.99	1.80	1.78	98.40
sdi	16153.50	0.00	522.50	0.00	133408.00	0.00	66704.00	0.00	255.33	0.98	1.88	1.86	97.00
sdi	17561.90	0.00	568.10	0.00	145040.00	0.00	72520.00	0.00	255.31	1.01	1.78	1.76	100.00

SAR

```
[root@localhost redhat]# sar -u 3 3
```

```
Linux 2.4.21-20.EL (localhost.localdomain) 05/16/2005
```

10:32:28 PM	CPU	%user	%nice	%system	%idle
10:32:31 PM	all	0.00	0.00	0.00	100.00
10:32:34 PM	all	1.33	0.00	0.33	98.33
10:32:37 PM	all	1.34	0.00	0.00	98.66
Average:	all	0.89	0.00	0.11	99.00

```
[root] sar -n DEV
```

```
Linux 2.4.21-20.EL (localhost.localdomain) 03/16/2005
```

01:10:01 PM	IFACE	rxpck/s	txpck/s	rxbyt/s	txbyt/s	rxcmp/s	txcmp/s
01:20:00 PM	lo	3.49	3.49	306.16	306.16	0.00	0.00
01:20:00 PM	eth0	3.89	3.53	2395.34	484.70	0.00	0.00
01:20:00 PM	eth1	0.00	0.00	0.00	0.00	0.00	0.00

Networking tools

Tuning tools

- ethtool – View and change Ethernet card settings
- sysctl – View and set */proc/sys* settings
- ifconfig – View and set ethX variables
- setpci – View and set pci bus params for device
- netperf – Can run a bunch of different network tests
- /proc – OS info, place for changing device tunables

ethtool

Works mostly at the HW level

ethtool -S – provides HW level stats

Counters since boot time, create scripts to calculate diffs

ethtool -c - Interrupt coalescing

ethtool -g - provides ring buffer information

ethtool -k - provides hw assist information

ethtool -i - provides the driver information

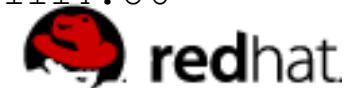
CPU Utilization – Raw vs. Tuned IRQ, NAPI

Not Tuned

CPU	%user	%nice	%system	%iowait	%irq	%soft	%idle	intr/s
all	0.23	0.00	8.01	0.02	0.00	10.78	80.96	21034.49
0	0.00	0.00	0.00	0.01	0.00	52.16	47.83	20158.58
1	0.00	0.00	0.00	0.02	0.00	0.00	100.00	125.14
2	0.00	0.00	0.00	0.08	0.00	0.00	99.93	125.14
3	0.00	0.00	0.00	0.03	0.00	0.00	99.99	125.13
4	1.79	0.00	64.11	0.00	0.00	34.11	0.01	125.14
5	0.01	0.00	0.00	0.02	0.00	0.00	99.99	125.14
6	0.00	0.00	0.00	0.00	0.00	0.00	100.01	125.14
7	0.00	0.00	0.00	0.02	0.00	0.00	99.99	125.14

With Tuning

CPU	%user	%nice	%system	%iowait	%irq	%soft	%idle	intr/s
all	0.26	0.00	10.44	0.00	0.00	12.50	76.79	1118.61
0	0.00	0.00	0.00	0.00	0.00	0.00	100.00	1.12
1	0.01	0.00	0.00	0.00	0.00	0.00	99.99	0.00
2	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00
3	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00
4	2.08	0.00	83.54	0.00	0.00	0.00	14.38	0.00
5	0.00	0.00	0.01	0.00	0.00	100.00	0.00	1.95
6	0.00	0.00	0.00	0.00	0.00	0.02	99.98	0.68
7	0.00	0.00	0.00	0.00	0.03	0.00	99.98	1114.86



free/numastat – memory allocation

```
[root@localhost redhat]# free -l
```

	total	used	free	shared	buffers
cached					
Mem:	511368	342336	169032	0	29712
167408					
Low:	511368	342336	169032	0	0
0					
High:	0	0	0	0	0
0					
-/+ buffers/cache:		145216	366152		
Swap:	1043240	0	1043240		

numastat (on 2-cpu x86_64 based system)

	node1	node0
numa_hit	9803332	10905630
numa_miss	2049018	1609361
numa_foreign	1609361	2049018
interleave_hit	58689	54749
local_node	9770927	10880901
other_node	2081423	1634090

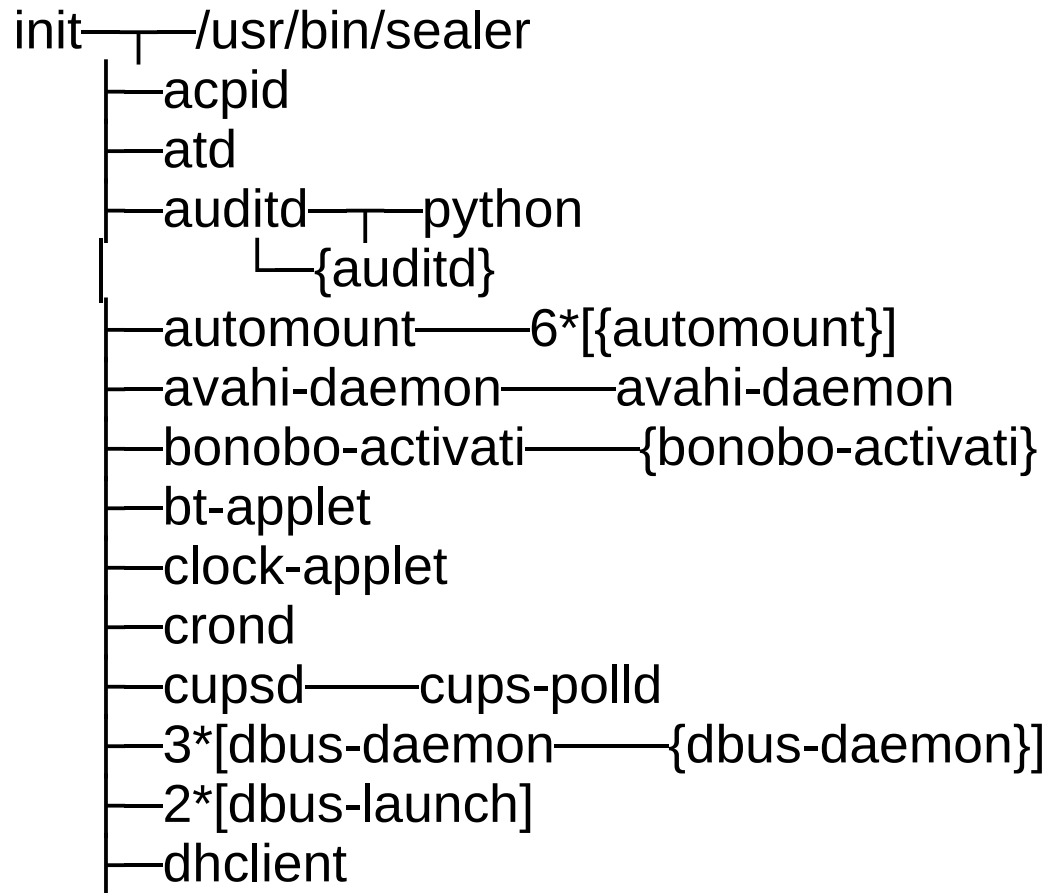
ps

```
[root@localhost root]# ps aux
```

```
[root@localhost root]# ps -aux | more
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.1	0.1	1528	516	?	S	23:18	0:04	init
root	2	0.0	0.0	0	0	?	SW	23:18	0:00	[keventd]
root	3	0.0	0.0	0	0	?	SW	23:18	0:00	[kapmd]
root	4	0.0	0.0	0	0	?	SWN	23:18	0:00	[ksoftirqd/0]
root	7	0.0	0.0	0	0	?	SW	23:18	0:00	[bdflush]
root	5	0.0	0.0	0	0	?	SW	23:18	0:00	[kswapd]
root	6	0.0	0.0	0	0	?	SW	23:18	0:00	[kscand]

pstree



The /proc filesystem

/proc

meminfo

slabinfo

cpuinfo

pid<#>/maps

vmstat(RHEL4 & RHEL5)

zoneinfo(RHEL5)

sysrq-trigger

/proc/meminfo

```
RHEL4> cat /proc/meminfo
MemTotal: 32749568 kB
MemFree: 31313344 kB
Buffers: 29992 kB
Cached: 1250584 kB
SwapCached: 0 kB
Active: 235284 kB
Inactive: 1124168 kB
HighTotal: 0 kB
HighFree: 0 kB
LowTotal: 32749568 kB
LowFree: 31313344 kB
SwapTotal: 4095992 kB
SwapFree: 4095992 kB
Dirty: 0 kB
Writeback: 0 kB
Mapped: 1124080 kB
Slab: 38460 kB
CommitLimit: 20470776 kB
Committed_AS: 1158556 kB
PageTables: 5096 kB
VmallocTotal: 536870911 kB
VmallocUsed: 2984 kB
VmallocChunk: 536867627 kB
HugePages_Total: 0
HugePages_Free: 0
Hugepagesize: 2048 kB
```

```
RHEL5> cat /proc/meminfo
MemTotal: 1025220 kB
MemFree: 11048 kB
Buffers: 141944 kB
Cached: 342664 kB
SwapCached: 4 kB
Active: 715304 kB
Inactive: 164780 kB
HighTotal: 0 kB
HighFree: 0 kB
LowTotal: 1025220 kB
LowFree: 11048 kB
SwapTotal: 2031608 kB
SwapFree: 2031472 kB
Dirty: 84 kB
Writeback: 0 kB
AnonPages: 395572 kB
Mapped: 82860 kB
Slab: 92296 kB
PageTables: 23884 kB
NFS_Unstable: 0 kB
Bounce: 0 kB
CommitLimit: 2544216 kB
Committed_AS: 804656 kB
VmallocTotal: 34359738367 kB
VmallocUsed: 263472 kB
VmallocChunk: 34359474711 kB
HugePages_Total: 0
HugePages_Free: 0
HugePages_Rsvd: 0
Hugepagesize: 2048 kB
```

/proc/slabinfo

slabinfo - version: 2.1

```
# name          <active_objs> <num_objs> <objsize> <objperslab> <pagesperslab> : tunables <limit>
<batchcount> <sharedfactor>: slabdata <active_slabs> <num_slabs> <sharedavail>
nfsd4_delegations      0      0    656     6     1 : tunables   54    27     8 : slabdata    0     0     0
nfsd4_stateids         0      0    128    30     1 : tunables  120    60     8 : slabdata    0     0     0
nfsd4_files            0      0     72    53     1 : tunables  120    60     8 : slabdata    0     0     0
nfsd4_stateowners     0      0    424     9     1 : tunables   54    27     8 : slabdata    0     0     0
nfs_direct_cache      0      0    128    30     1 : tunables  120    60     8 : slabdata    0     0     0
nfs_write_data        36     36    832     9     2 : tunables   54    27     8 : slabdata     4     4     0
nfs_read_data         32     35    768     5     1 : tunables   54    27     8 : slabdata     7     7     0
nfs_inode_cache       1383   1389   1040     3     1 : tunables   24    12     8 : slabdata   463   463     0
nfs_page              0      0    128    30     1 : tunables  120    60     8 : slabdata    0     0     0
fscache_cookie_jar    3      53     72    53     1 : tunables  120    60     8 : slabdata     1     1     0
ip_contrack_expect    0      0    136    28     1 : tunables  120    60     8 : slabdata    0     0     0
ip_contrack           75     130    304    13     1 : tunables   54    27     8 : slabdata    10    10     0
bridge_fdb_cache      0      0     64    59     1 : tunables  120    60     8 : slabdata    0     0     0
rpc_buffers           8      8   2048     2     1 : tunables   24    12     8 : slabdata     4     4     0
rpc_tasks             30     30    384    10     1 : tunables   54    27     8 : slabdata     3     3     0
```

/proc/cpuinfo

```
[lwoodman]$ cat /proc/cpuinfo
processor      : 0
vendor_id    : GenuineIntel
cpu family   : 6
model        : 15
model name   : Intel(R) Xeon(R) CPU           3060  @ 2.40GHz
stepping     : 6
cpu MHz      : 2394.070
cache size   : 4096 KB
physical id  : 0
siblings     : 2
core id      : 0
cpu cores    : 2
fpu          : yes
fpu_exception : yes
cpuid level  : 10
wp           : yes
flags        : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush
dts acpi mmx fxsr sse sse2 ss ht tm syscall nx lm constant_tsc pni monitor ds_cpl vmx est tm2 cx16
xtpr lahf_lm
bogomips     : 4791.41
clflush size : 64
cache_alignment : 64
address sizes : 36 bits physical, 48 bits virtual
power management:
```

32-bit /proc/<pid>/maps

```
[root@dhcp83-36 proc]# cat 5808/maps
```

```
0022e000-0023b000 r-xp 00000000 03:03 4137068 /lib/tls/libpthread-0.60.so
0023b000-0023c000 rw-p 0000c000 03:03 4137068 /lib/tls/libpthread-0.60.so
0023c000-0023e000 rw-p 00000000 00:00 0
0037f000-00391000 r-xp 00000000 03:03 523285 /lib/libnsl-2.3.2.so
00391000-00392000 rw-p 00011000 03:03 523285 /lib/libnsl-2.3.2.so
00392000-00394000 rw-p 00000000 00:00 0
00c45000-00c5a000 r-xp 00000000 03:03 523268 /lib/ld-2.3.2.so
00c5a000-00c5b000 rw-p 00015000 03:03 523268 /lib/ld-2.3.2.so
00e5c000-00f8e000 r-xp 00000000 03:03 4137064 /lib/tls/libc-2.3.2.so
00f8e000-00f91000 rw-p 00131000 03:03 4137064 /lib/tls/libc-2.3.2.so
00f91000-00f94000 rw-p 00000000 00:00 0
08048000-0804f000 r-xp 00000000 03:03 1046791 /sbin/ypbind
0804f000-08050000 rw-p 00007000 03:03 1046791 /sbin/ypbind
09794000-097b5000 rw-p 00000000 00:00 0
b5fdd000-b5fde000 ---p 00000000 00:00 0
```

Red Hat Summit 2009 | John Shakshober / Larry Woodman

RED HAT :: CHICAGO :: 2009

SUMMIT

64-bit /proc/<pid>/maps

```
# cat /proc/2345/maps
00400000-0100b000 r-xp 00000000 fd:00 1933328 /usr/sybase/ASE-12_5/bin/dataserver.esd3
0110b000-01433000 rw-p 00c0b000 fd:00 1933328 /usr/sybase/ASE-12_5/bin/dataserver.esd3
01433000-014eb000 rwxp 01433000 00:00 0
40000000-40001000 ---p 40000000 00:00 0
40001000-40a01000 rwxp 40001000 00:00 0
2a95f73000-2a96073000 ---p 0012b000 fd:00 819273 /lib64/tls/libc-2.3.4.so
2a96073000-2a96075000 r--p 0012b000 fd:00 819273 /lib64/tls/libc-2.3.4.so
2a96075000-2a96078000 rw-p 0012d000 fd:00 819273 /lib64/tls/libc-2.3.4.so
2a96078000-2a9607e000 rw-p 2a96078000 00:00 0
2a9607e000-2a98c3e000 rw-s 00000000 00:06 360450 /SYSV0100401e (deleted)
2a98c3e000-2a98c47000 rw-p 2a98c3e000 00:00 0
2a98c47000-2a98c51000 r-xp 00000000 fd:00 819227 /lib64/libnss_files-2.3.4.so
2a98c51000-2a98d51000 ---p 0000a000 fd:00 819227 /lib64/libnss_files-2.3.4.so
2a98d51000-2a98d53000 rw-p 0000a000 fd:00 819227 /lib64/libnss_files-2.3.4.so
2a98d53000-2a98d57000 r-xp 00000000 fd:00 819225 /lib64/libnss_dns-2.3.4.so
2a98d57000-2a98e56000 ---p 00004000 fd:00 819225 /lib64/libnss_dns-2.3.4.so
2a98e56000-2a98e58000 rw-p 00003000 fd:00 819225 /lib64/libnss_dns-2.3.4.so
2a98e58000-2a98e69000 r-xp 00000000 fd:00 819237 /lib64/libresolv-2.3.4.so
2a98e69000-2a98f69000 ---p 00011000 fd:00 819237 /lib64/libresolv-2.3.4.so
2a98f69000-2a98f6b000 rw-p 00011000 fd:00 819237 /lib64/libresolv-2.3.4.so
2a98f6b000-2a98f6d000 rw-p 2a98f6b000 00:00 0
35c7e00000-35c7e08000 r-xp 00000000 fd:00 819469 /lib64/libpam.so.0.77
35c7e08000-35c7f08000 ---p 00008000 fd:00 819469 /lib64/libpam.so.0.77
35c7f08000-35c7f09000 rw-p 00008000 fd:00 819469 /lib64/libpam.so.0.77
35c8000000-35c8011000 r-xp 00000000 fd:00 819468 /lib64/libaudit.so.0.0.0
35c8011000-35c8110000 ---p 00011000 fd:00 819468 /lib64/libaudit.so.0.0.0
35c8110000-35c8118000 rw-p 00010000 fd:00 819468 /lib64/libaudit.so.0.0.0
35c9000000-35c900b000 r-xp 00000000 fd:00 819457 /lib64/libgcc_s-3.4.4-20050721.so.1
35c900b000-35c910a000 ---p 0000b000 fd:00 819457 /lib64/libgcc_s-3.4.4-20050721.so.1
35c910a000-35c910b000 rw-p 0000a000 fd:00 819457 /lib64/libgcc_s-3.4.4-20050721.so.1
7fbfff1000-7fc0000000 rwxp 7fbfff1000 00:00 0
ffffffff600000-fffffffffe000000 ---p 00000000 00:00 0
```

/proc/vmstat

```
cat /proc/vmstat
nr_anon_pages 98893
nr_mapped 20715
nr_file_pages 120855
nr_slab 23060
nr_page_table_pages
5971
nr_dirty 21
nr_writeback 0
nr_unstable 0
nr_bounce 0
numa_hit 996729666
numa_miss 0
numa_foreign 0
numa_interleave 87657
numa_local 996729666
numa_other 0
pgpgin 2577307
pgpgout 106131928
pswpin 0
pswpout 34
pgalloc_dma 198908
pgalloc_dma32
997707549
pgalloc_normal 0
pgalloc_high 0
pgfree 997909734
pgactivate 1313196
pgdeactivate 470908
pgfault 2971972147
pgmajfault 8047.
```

```
CONTINUED...
pgrefill_dma 18338
pgrefill_dma32 1353451
pgrefill_normal 0
pgrefill_high 0
pgsteal_dma 0
pgsteal_dma32 0
pgsteal_normal 0
pgsteal_high 0
pgscan_kswapd_dma 7235
pgscan_kswapd_dma32 417984
pgscan_kswapd_normal 0
pgscan_kswapd_high 0
pgscan_direct_dma 12
pgscan_direct_dma32 1984
pgscan_direct_normal 0
pgscan_direct_high 0
pginodesteal 166
slabs_scanned 1072512
kswapd_steal 410973
kswapd_inodesteal 61305
pageoutrun 7752
allocstall 29
pgrotated 73
```

Alt Sysrq M

```
Free pages:      15809760kB (0kB HighMem)
Active:51550 inactive:54515 dirty:44 writeback:0 unstable:0 free:3952440 slab:8727 mapped-
file:5064 mapped-anon:20127 pagetables:1627
Node 0 DMA free:10864kB min:8kB low:8kB high:12kB active:0kB inactive:0kB present:10460kB
pages_scanned:0 all_unreclaimable? no
Node 0 DMA32 free:2643124kB min:2760kB low:3448kB high:4140kB active:0kB inactive:0kB
present:2808992kB pages_scanned:0 all_unreclaimable? no
Node 0 Normal free:13155772kB min:13480kB low:16848kB high:20220kB active:206200kB
inactive:218060kB present:13703680kB pages_scanned:0 all_unreclaimable? no
Node 0 HighMem free:0kB min:128kB low:128kB high:128kB active:0kB inactive:0kB present:0kB
pages_scanned:0 all_unreclaimable? no
Node 0 DMA: 4*4kB 2*8kB 3*16kB 1*32kB 2*64kB 1*128kB 1*256kB 0*512kB 2*1024kB 0*2048kB
2*4096kB = 10864kB
Node 0 DMA32: 1*4kB 0*8kB 1*16kB 1*32kB 0*64kB 1*128kB 0*256kB 2*512kB 2*1024kB 3*2048kB
643*4096kB = 2643124kB
Node 0 Normal: 453*4kB 161*8kB 44*16kB 15*32kB 4*64kB 4*128kB 0*256kB 1*512kB 0*1024kB
1*2048kB 3210*4096kB = 13155772kB
Node 0 HighMem: empty
85955 pagecache pages
Swap cache: add 0, delete 0, find 0/0, race 0+0
Free swap = 2031608kB
Total swap = 2031608kB
Free swap:      2031608kB
4521984 pages of RAM
446612 reserved pages
21971 pages shared
0 pages swap cached
```

Alt Sysrq M - NUMA

```
Free pages:      15630596kB (0kB HighMem)
Active:77517 inactive:67928 dirty:1000 writeback:0 unstable:0 free:3907649 slab:10391 mapped-file:8975
mapped-anon:38003 pagetables:4731
Node 0 DMA free:10864kB min:8kB low:8kB high:12kB active:0kB inactive:0kB present:10460kB pages_scanned:0
all_unreclaimable? no
lowmem_reserve[]: 0 2743 8045 8045
Node 0 DMA32 free:2643480kB min:2760kB low:3448kB high:4140kB active:0kB inactive:0kB present:2808992kB
pages_scanned:0 all_unreclaimable? no
Node 0 Normal free:4917364kB min:5340kB low:6672kB high:8008kB active:204836kB inactive:197340kB
present:5429760kB pages_scanned:0 all_unreclaimable? no
Node 0 HighMem free:0kB min:128kB low:128kB high:128kB active:0kB inactive:0kB present:0kB
pages_scanned:0 all_unreclaimable? no
Node 1 DMA free:0kB min:0kB low:0kB high:0kB active:0kB inactive:0kB present:0kB pages_scanned:0
all_unreclaimable? no
Node 1 DMA32 free:0kB min:0kB low:0kB high:0kB active:0kB inactive:0kB present:0kB pages_scanned:0
all_unreclaimable? no
Node 1 Normal free:8058888kB min:8140kB low:10172kB high:12208kB active:105232kB inactive:74372kB
present:8273920kB pages_scanned:0 all_unreclaimable? no
Node 1 HighMem free:0kB min:128kB low:128kB high:128kB active:0kB inactive:0kB present:0kB
pages_scanned:0 all_unreclaimable? no
Node 0 DMA: 6*4kB 5*8kB 3*16kB 2*32kB 3*64kB 2*128kB 0*256kB 0*512kB 2*1024kB 0*2048kB 2*4096kB = 10864kB
Node 0 DMA32: 2*4kB 2*8kB 0*16kB 2*32kB 1*64kB 1*128kB 1*256kB 2*512kB 2*1024kB 3*2048kB 643*4096kB =
2643480kB
Node 0 Normal: 91*4kB 47*8kB 27*16kB 5*32kB 5*64kB 0*128kB 0*256kB 1*512kB 2*1024kB 1*2048kB 1199*4096kB
= 4917364kB
Node 1 Normal: 78*4kB 48*8kB 477*16kB 326*32kB 261*64kB 105*128kB 55*256kB 33*512kB 20*1024kB 0*2048kB
1943*4096kB = 8058888kB
107476 pagecache pages
4521984 pages of RAM
```

Alt Sysrq T

```
gdmgreeter      S ffff810009036800      0 7511 7483      7489 (NOTLB)
ffff81044ae05b38 0000000000000082 0000000000000080 0000000000000000
0000000000000000 000000000000000a ffff810432ed97a0 ffff81010f387080
0000002a3a0d4398 00000000000003b57 ffff810432ed9988 0000000600000000
```

Call Trace:

```
[<ffffffff8006380f>] schedule_timeout+0x1e/0xad
[<ffffffff80049b33>] add_wait_queue+0x24/0x34
[<ffffffff8002db7e>] pipe_poll+0x2d/0x90
[<ffffffff8002f764>] do_sys_poll+0x277/0x360
[<ffffffff8001e99c>] __pollwait+0x0/0xe2
[<ffffffff8008be44>] default_wake_function+0x0/0xe
[<ffffffff8008be44>] default_wake_function+0x0/0xe
[<ffffffff8008be44>] default_wake_function+0x0/0xe
[<ffffffff80012f1a>] sock_def_readable+0x34/0x5f
[<ffffffff8004a81a>] unix_stream_sendmsg+0x281/0x346
[<ffffffff80037c3a>] do_sock_write+0xc6/0x102
[<ffffffff801277da>] avc_has_perm+0x43/0x55
[<ffffffff80276a6e>] unix_ioctl+0xc7/0xd0
[<ffffffff8021f48f>] sock_ioctl+0x1c1/0x1e5
[<ffffffff800420a7>] do_ioctl+0x21/0x6b
[<ffffffff800302a0>] vfs_ioctl+0x457/0x4b9
[<ffffffff800b6193>] audit_syscall_entry+0x180/0x1b3
[<ffffffff8004c4f6>] sys_poll+0x2d/0x34
[<ffffffff8005d28d>] tracesys+0xd5/0xe0
```

Alt Sysrq W and P

SysRq : Show CPUs

CPU2:

```
ffff81010f30bf48 0000000000000000 ffff81010f305e20 ffffffff801ae69e
0000000000000000 00000000000000200 ffffffff803ea2a0 ffffffff801ae6cd
ffffffff801ae69e ffffffff80022d85 ffffffff80197393 00000000000000ff
```

Call Trace:

```
<IRQ> [] showacpu+0x0/0x3b
[] showacpu+0x2f/0x3b
[] showacpu+0x0/0x3b
[] smp_call_function_interrupt+0x57/0x75
[] acpi_processor_idle+0x0/0x463
[] call_function_interrupt+0x66/0x6c
<EOI> [] acpi_safe_halt+0x25/0x36
[] acpi_processor_idle+0x187/0x463
[] acpi_processor_idle+0x2/0x463
[] acpi_processor_idle+0x0/0x463
[] acpi_processor_idle+0x0/0x463
[] cpu_idle+0x95/0xb8
[] start_secondary+0x45a/0x469
```

Profiling Tools: OProfile

Open source project –

<http://oprofile.sourceforge.net>

Upstream; Red Hat contributes

Originally modeled after DEC Continuous Profiling Infrastructure (DCPI)

System-wide profiler (both kernel and user code)

Sample-based profiler with SMP machine support

Performance monitoring hardware support

Relatively low overhead, typically <10%

Designed to run for long times

Included in base Red Hat Enterprise Linux product

Events to measure with Oprofile:

Initially time-based samples most useful:

PPro/PII/PIII/AMD: CPU_CLK_UNHALTED

P4: GLOBAL_POWER_EVENTS

IA64: CPU_CYCLES

TIMER_INT (fall-back profiling mechanism) default

Processor specific performance monitoring hardware can provide additional kinds of sampling

Many events to choose from

Branch mispredictions

Cache misses - TLB misses

Pipeline stalls/serializing instructions

oprofile – builtin to RHEL4 & 5 (smp)

opcontrol – on/off data

--start start collection

--stop stop collection

--dump output to disk

--event=:name:count

Example:

```
# opcontrol –start
```

```
# /bin/time test1 &
```

```
# sleep 60
```

```
# opcontrol –stop
```

```
# opcontrol dump
```

opreport – analyze profile

-r reverse order sort

-t [percentage] threshold to view

-f /path/filename

-d details

opannotate

-s /path/source

-a /path/assembly

oprofile – opcontrol and oprofile cpu_cycles

```
# CPU: Core 2, speed 2666.72 MHz (estimated)
Counted CPU_CLK_UNHALTED events (Clock cycles when not halted) with a unit mask of 0x00 (Unhalted core
cycles) count 100000
CPU_CLK_UNHALT...|
  samples|      %|
-----|
397435971 84.6702 vmlinux
 19703064  4.1976 zeus.web
 16914317  3.6034 e1000
 12208514  2.6009 ld-2.5.so
 11711746  2.4951 libc-2.5.so
  5164664  1.1003 sim.cgi
  2333427  0.4971 oprofiled
  1295161  0.2759 oprofile
  1099731  0.2343 zeus.cgi
   968623  0.2064 ext3
   270163  0.0576 jbd
```

Profiling Tools: SystemTap

Technology: Kprobes:

In current 2.6 kernels

Upstream 2.6.12, backported to RHEL4 kernel

Kernel instrumentation without recompile/reboot

Uses software int and trap handler for instrumentation

Debug information:

Provides map between executable and source code

Generated as part of RPM builds

Available at: <ftp://ftp.redhat.com>

Safety: Instrumentation scripting language:

No dynamic memory allocation or assembly/C code

Types and type conversions limited

Restrict access through pointers

Script compiler checks:

Infinite loops and recursion – Invalid variable access

Profiling Tools: SystemTap

Red Hat, Intel, IBM & Hitachi collaboration

Linux answer to Solaris Dtrace

Dynamic instrumentation

Tool to take a deep look into a running system:

Assists in identifying causes of performance problems

Simplifies building instrumentation

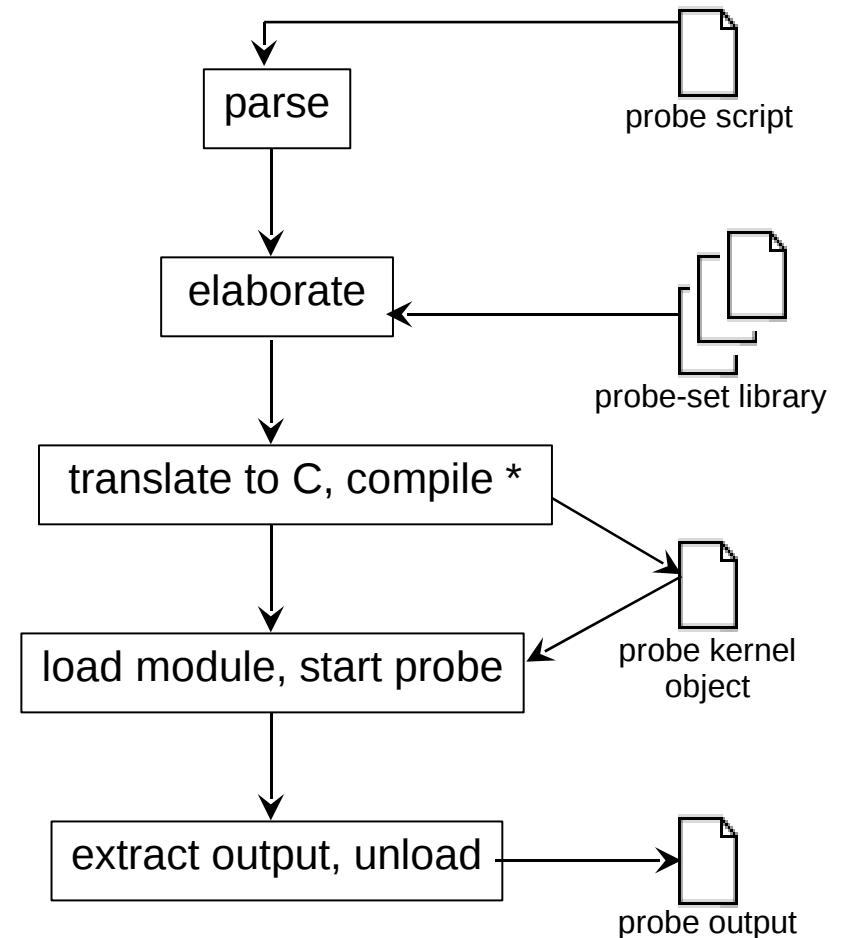
Current snapshots available from:

<http://sources.redhat.com/systemtap>

Source for presentations/papers

Kernel space tracing today, user space tracing under development

Technology preview status until 5.1



* Solaris Dtrace is interpretive

SystemTap: Kernel debugging

Several tracepoints were added to RHEL5 kernel

```
trace_mm_filemap_fault(area->vm_mm, address, page);
trace_mm_anon_userfree(mm, addr, page);
trace_mm_filemap_userunmap(mm, addr, page);
trace_mm_filemap_cow(mm, address, new_page);
trace_mm_anon_cow(mm, address, new_page);
trace_mm_anon_pgin(mm, address, page);
trace_mm_anon_fault(mm, address, page);
trace_mm_page_free(page);
trace_mm_page_allocation(page, zone->free_pages);
trace_mm_pdflush_bgwriteout(_min_pages);
trace_mm_pdflush_kupdate(nr_to_write);
trace_mm_anon_unmap(page, ret == SWAP_SUCCESS);
trace_mm_filemap_unmap(page, ret == SWAP_SUCCESS);
trace_mm_pagereclaim_pgout(page, PageAnon(page));
trace_mm_pagereclaim_free(page, PageAnon(page));
trace_mm_pagereclaim_shrinkinactive_i2a(page);
trace_mm_pagereclaim_shrinkinactive_i2i(page);
trace_mm_pagereclaim_shrinkinactive(nr_reclaimed);
trace_mm_pagereclaim_shrinkactive_a2a(page);
trace_mm_pagereclaim_shrinkactive_a2i(page);
trace_mm_pagereclaim_shrinkactive(pgscanned);
trace_mm_pagereclaim_shrinkzone(nr_reclaimed);
trace_mm_directreclaim_reclaimall(priority);
trace_mm_kswapd_runs(sc.nr_reclaimed);
```

SystemTap: Kernel debugging

Several custom scripts enable/use tracepoints

(/usr/local/share/doc/systemtap/examples)

```
#!/usr/local/bin/stap
global traced_pid
function log_event:long ()
{
    return (!traced_pid ||traced_pid == (task_pid(task_current())))
}
probe kernel.trace("mm_pagereclaim_shrinkinactive") {
    if (!log_event()) next
    reclaims[pid()]++
    command[pid()]=execname()
}
//MM kernel tracepoints prolog and epilog routines
probe begin {
    printf("Starting mm tracepoints\n");
    traced_pid = target();
    if (traced_pid) {
        printf("mode Specific Pid, traced pid: %d\n", traced_pid);
    } else {
        printf("mode - All Pids\n");
    }
    printf("\n");
}
probe end {
    printf("Terminating mm tracepoints\n");
    printf("Command      Pid      Direct   Activate  Deactivate Reclaims  Freed\n");
    printf("-----      ---      -----   - - - - -  - - - - -  - - - - -  - - - - -\n");
    foreach (pid in reclaims-)
```

SystemTap: Kernel debugging

Command	Pid	Direct	Activate	Deactivate	Reclaims	Freed
-----	-----	-----	-----	-----	-----	-----
kswapd0	544	0	1503767	919437	15157	430730
kswapd1	545	0	1806788	824347	12117	341408
memory	25435	997	569757	308360	4621	115837
mixer_applet2	7687	6	4180	1013	33	981
Xorg	7491	5	1906	2839	20	382
gnome-terminal	7161	2	1038	695	12	320
gnome-terminal	7701	5	2614	2245	7	172
cupsd	7100	1	927	0	4	128

SystemTap: Kernel debugging

Command	Pid	Alloc	Free	A_fault	A_ufree	A_pgin	A_cow	A_unmap
-----	---	-----	----	-----	-----	-----	-----	-----
memory	25685	2842784	4064408	2834840	3989816	14	0	48185
kswapd1	545	3007	53257	0	0	0	0	49884
kswapd0	544	620	25241	0	0	0	0	17568
mixer_applet2	7687	302	2827	0	0	1	0	1241
sshd	25051	227	0	0	0	6	0	0
kjournald	863	207	283	0	0	0	0	2149
Xorg	7491	169	898	0	0	0	0	310
gnome-power-man	7653	152	0	0	0	18	0	0
avahi-daemon	7252	150	1280	0	0	48	0	160
irqbalance	6725	126	364	13	13	18	0	190
bash	25053	122	0	0	0	13	0	0
hald	7264	89	0	0	0	83	0	0
gconfd-2	7163	82	526	0	0	68	0	116

Red Hat MRG Tuna

Tuna (on perf20.lab.bos.redhat.com)

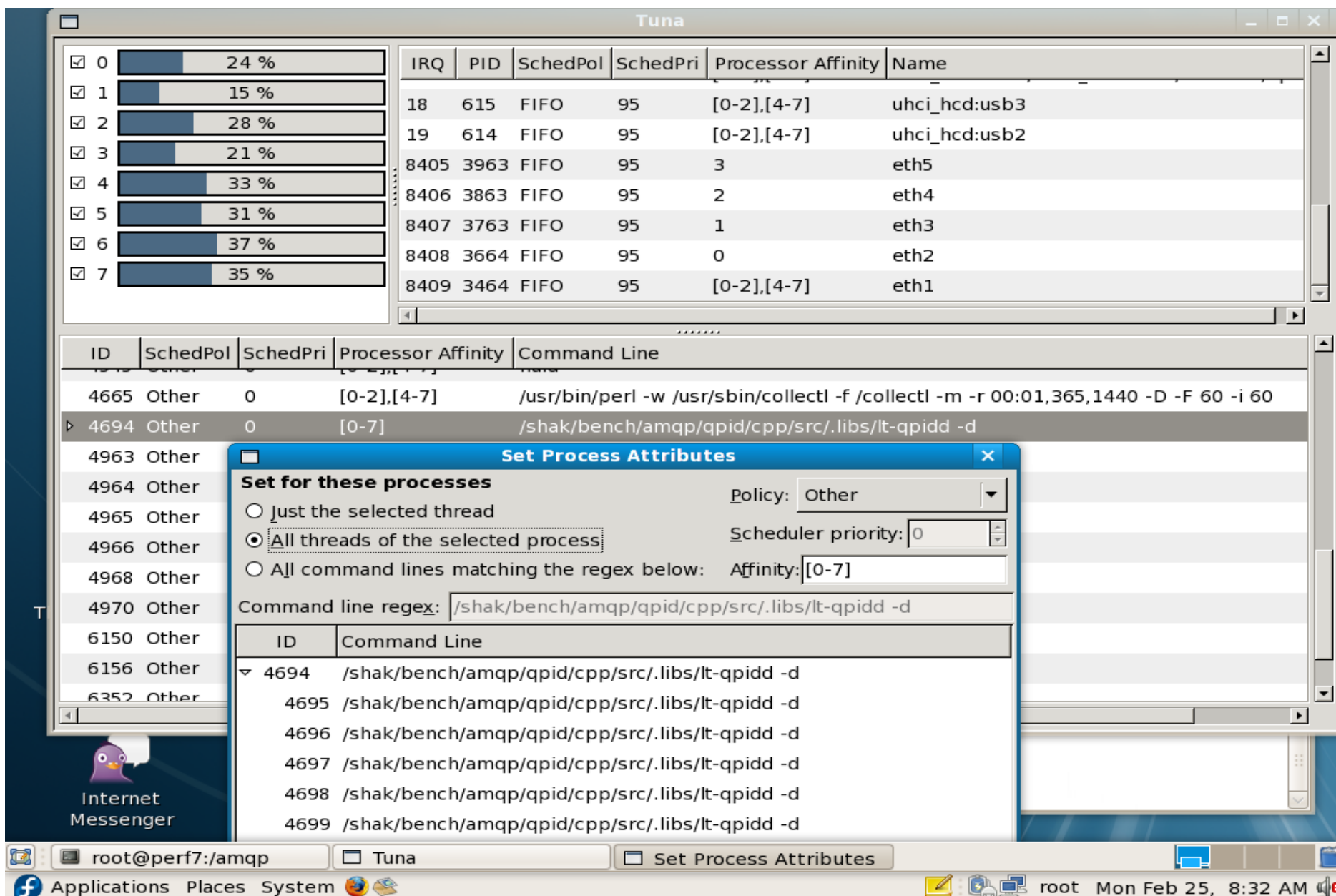
Socket 0			Socket 1		
Filter	CPU	Usage	Filter	CPU	Usage
<input checked="" type="checkbox"/>	0	0	<input checked="" type="checkbox"/>	3	21
<input checked="" type="checkbox"/>	1	0	<input checked="" type="checkbox"/>	4	0
<input checked="" type="checkbox"/>	2	0	<input checked="" type="checkbox"/>	5	0
<input checked="" type="checkbox"/>	12	0	<input checked="" type="checkbox"/>	15	0
<input checked="" type="checkbox"/>	13	0	<input checked="" type="checkbox"/>	16	0
<input checked="" type="checkbox"/>	14	0	<input checked="" type="checkbox"/>	17	0

Socket 2			Socket 3		
Filter	CPU	Usage	Filter	CPU	Usage
<input checked="" type="checkbox"/>	6	0	<input checked="" type="checkbox"/>	9	0
<input checked="" type="checkbox"/>	7	0	<input checked="" type="checkbox"/>	10	0
<input checked="" type="checkbox"/>	8	0	<input checked="" type="checkbox"/>	11	0
<input checked="" type="checkbox"/>	18	0	<input checked="" type="checkbox"/>	21	0
<input checked="" type="checkbox"/>	19	0	<input checked="" type="checkbox"/>	22	0
<input checked="" type="checkbox"/>	20	0	<input checked="" type="checkbox"/>	23	0

IRQ	PID	Policy	Priority	Affinity	Events	Users
17	1473	FIFO	50	1,13	51525	megasas
22	1321	FIFO	50	1,13	858	uhci_hcd:usb2,uhci_hcd:usb3,uhci_hcd:usb4,uhci_h
23	1270	FIFO	50	2,14	30	ehci_hcd:usb1
2229	6529	FIFO	50	0	46098	eth3(e1000)
2230	6320	FIFO	50	13	1624017	eth2(e1000)
2231	6148	FIFO	50	0-23	1	eth0:lsc
2232	6147	FIFO	50	13	56938	eth0:v15-Rx
2233	6146	FIFO	50	2	55448	eth0:v14-Rx
2234	6145	FIFO	50	12	55406	eth0:v13-Rx
2235	6144	FIFO	50	14	56700	eth0:v12-Rx
2236	6143	FIFO	50	1	56803	eth0:v11-Rx
2237	6142	FIFO	50	14	58014	eth0:v10-Rx
2238	6141	FIFO	50	1	57371	eth0:v9-Rx
2239	6140	FIFO	50	14	58816	eth0:v8-Rx
2240	6139	FIFO	50	0	60573	eth0:v7-Rx

PID	Policy	Priority	Affinity	VolCtxtSwitch	NonVolCtxtSwitch	Command Line
1	OTHER	0	0-23	20259	2744	init [3]
2	OTHER	0	0-23	530	1320	kthreadd
3	FIFO	99	0	702	0	migration/0
4	FIFO	99	0	2	0	posixcpumr/0
5	FIFO	50	0	2	0	sirq-high/0
6	FIFO	50	0	90298186	0	sirq-timer/0
7	FIFO	50	0	15	0	sirq-net-tx/0
8	FIFO	50	0	133467	0	sirq-net-rx/0
9	FIFO	50	0	1055	0	sirq-block/0
10	FIFO	50	0	567	0	sirq-tasklet/0

Applications Places System 1 GHz root Tue Nov 4, 2:44 PM



The screenshot shows the Tuna application interface. On the left, there are CPU usage bars for processors 0 through 7, with usage percentages ranging from 15% to 35%. The main window displays a table of processes with columns for IRQ, PID, SchedPol, SchedPri, Processor Affinity, and Name. Below this, a detailed view of processes is shown with columns for ID, SchedPol, SchedPri, Processor Affinity, and Command Line. A 'Set Process Attributes' dialog box is open, showing configuration for process 4694. The dialog has three radio buttons: 'Just the selected thread', 'All threads of the selected process' (which is selected), and 'All command lines matching the regex below'. The 'Policy' is set to 'Other', 'Scheduler priority' is 0, and 'Affinity' is [0-7]. The 'Command line regex' is /shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d. A list of matching processes is shown below the dialog.

IRQ	PID	SchedPol	SchedPri	Processor Affinity	Name
18	615	FIFO	95	[0-2],[4-7]	uhci_hcd:usb3
19	614	FIFO	95	[0-2],[4-7]	uhci_hcd:usb2
8405	3963	FIFO	95	3	eth5
8406	3863	FIFO	95	2	eth4
8407	3763	FIFO	95	1	eth3
8408	3664	FIFO	95	0	eth2
8409	3464	FIFO	95	[0-2],[4-7]	eth1

ID	SchedPol	SchedPri	Processor Affinity	Command Line
4665	Other	0	[0-2],[4-7]	/usr/bin/perl -w /usr/sbin/collectl -f /collectl -m -r 00:01,365,1440 -D -F 60 -i 60
4694	Other	0	[0-7]	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4963	Other			
4964	Other			
4965	Other			
4966	Other			
4968	Other			
4970	Other			
6150	Other			
6156	Other			
6352	Other			

Set Process Attributes

Set for these processes

Just the selected thread

All threads of the selected process

All command lines matching the regex below:

Policy: Other

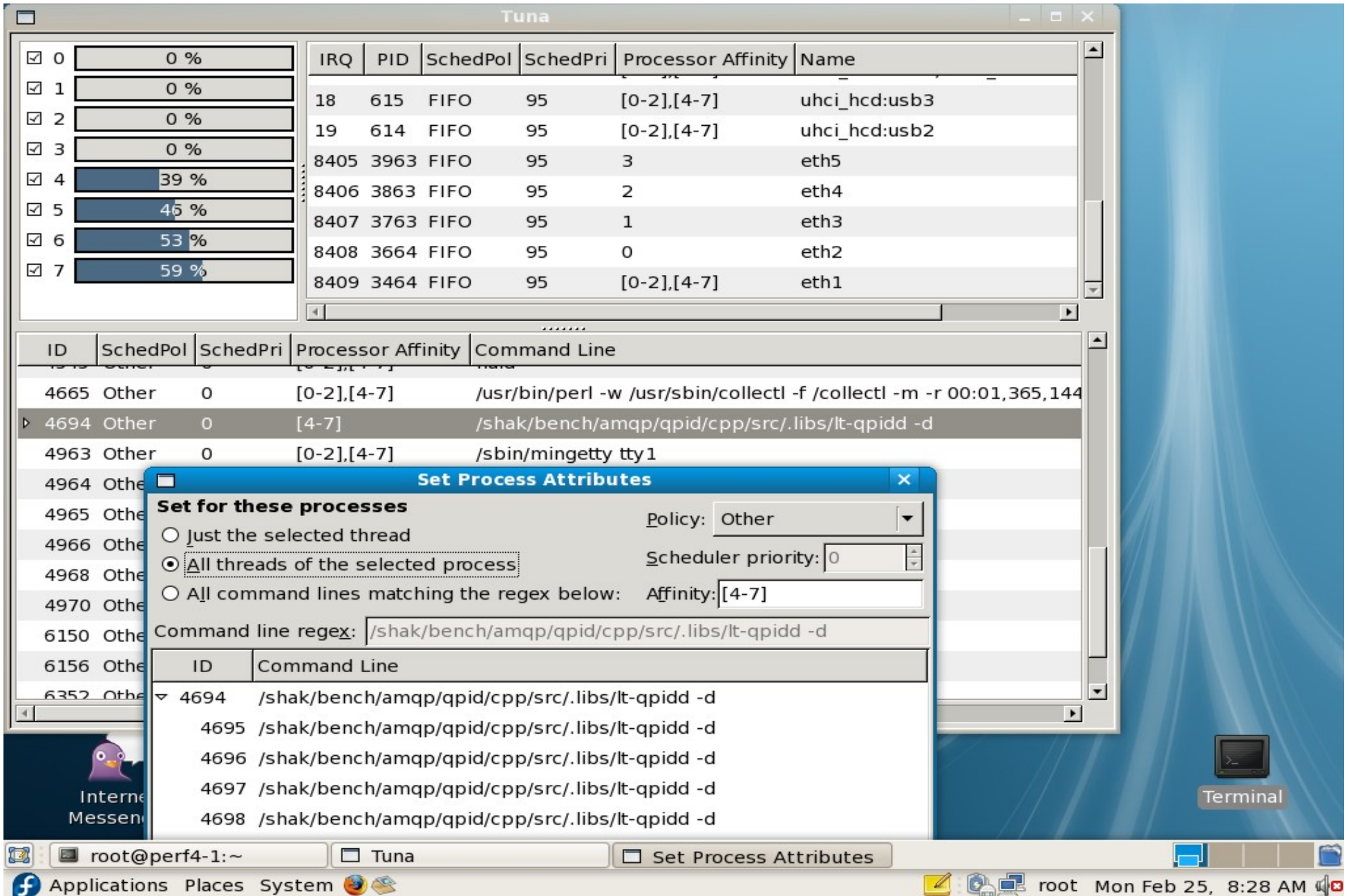
Scheduler priority: 0

Affinity: [0-7]

Command line regex: /shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d

ID	Command Line
4694	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4695	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4696	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4697	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4698	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4699	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d

Red Hat MRG Tuna



The screenshot shows the Tuna application window with a table of process scheduling parameters and a dialog box for setting process attributes.

IRQ	PID	SchedPol	SchedPri	Processor Affinity	Name
18	615	FIFO	95	[0-2],[4-7]	uhci_hcd:usb3
19	614	FIFO	95	[0-2],[4-7]	uhci_hcd:usb2
8405	3963	FIFO	95	3	eth5
8406	3863	FIFO	95	2	eth4
8407	3763	FIFO	95	1	eth3
8408	3664	FIFO	95	0	eth2
8409	3464	FIFO	95	[0-2],[4-7]	eth1

ID	SchedPol	SchedPri	Processor Affinity	Command Line
4665	Other	0	[0-2],[4-7]	/usr/bin/perl -w /usr/sbin/collectl -f /collectl -m -r 00:01,365,144
4694	Other	0	[4-7]	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4963	Other	0	[0-2],[4-7]	/sbin/mingetty tty1

Set Process Attributes

Set for these processes

- Just the selected thread
- All threads of the selected process
- All command lines matching the regex below:

Policy: Other

Scheduler priority: 0

Affinity: [4-7]

Command line regex: /shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d

ID	Command Line
4694	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4695	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4696	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4697	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d
4698	/shak/bench/amqp/qpid/cpp/src/.libs/lt-qpid -d

Red Hat MRG Tuna

TUNA command line

Usage: tuna [OPTIONS]

-h, --help	Give this help list
-g, --gui	Start the GUI
-c, --cpus=CPU-LIST	CPU-LIST affected by commands
-C, --affect_children	Operation will affect children threads
-f, --filter	Display filter the selected entities
-i, --isolate	Move all threads away from CPU-LIST
-l, --include	Allow all threads to run on CPU-LIST
-K, --no_kthreads	Operations will not affect kernel threads
-m, --move	move selected entities to CPU-LIST
-p, --priority=[POLICY]:RTPRIO	set thread scheduler POLICY and RTPRIO
-P, --show_threads	show thread list
-s, --save=FILENAME	save kthreads sched tunables to FILENAME
-S, --sockets=CPU-SOCKET-LIST	CPU-SOCKET-LIST affected by commands
-t, --threads=THREAD-LIST	THREAD-LIST affected by commands
-U, --no_uthreads	Operations will not affect user threads
-W, --what_is	Provides help about selected entities

Examples

tuna -c 0-3 -i (isolate cpu 0-3), tune -S 1 -i (isolate socket 1 = cpu 0-3 intelq)
tuna -t PID -C -p fifo:50 -S 1 -m -P (move PID# to socket 1, sched:fifo +50 prior

Section 3: Tuning RHEL

How to tune Linux

Capacity tuning

Fix problems by adding resources

Performance Tuning

Methodology

- 1) Document config
- 2) Baseline results
- 3) While results non-optimal
 - a) Monitor/Instrument system/workload
 - b) Apply tuning 1 change at a time
 - c) Analyze results, exit or loop
- 4) Document final config

Tuning - setting kernel parameters

/proc

```
[root@foobar fs]# cat /proc/sys/kernel/sysrq (see "0")
```

```
[root@foobar fs]# echo 1 > /proc/sys/kernel/sysrq
```

```
[root@foobar fs]# cat /proc/sys/kernel/sysrq (see "1")
```

Sysctl command

```
[root@foobar fs]# sysctl kernel.sysrq
```

```
kernel.sysrq = 0
```

```
[root@foobar fs]# sysctl -w kernel.sysrq=1
```

```
kernel.sysrq = 1
```

```
[root@foobar fs]# sysctl kernel.sysrq
```

```
kernel.sysrq = 1
```

Edit the /etc/sysctl.conf file

```
# Kernel sysctl configuration file for Red Hat Linux
```

```
# Controls the System Request debugging functionality of the kernel
```

```
kernel.sysrq = 1
```

Capacity Tuning

- Memory

- /proc/sys/vm/overcommit_memory
- /proc/sys/vm/overcommit_ratio
- /proc/sys/vm/max_map_count
- /proc/sys/vm/nr_hugepages

- Kernel

- /proc/sys/kernel/msgmax
- /proc/sys/kernel/msgmnb
- /proc/sys/kernel/msgmni
- /proc/sys/kernel/shmall
- /proc/sys/kernel/shmmax
- /proc/sys/kernel/shmmni
- /proc/sys/kernel/threads-max

- Filesystems

- /proc/sys/fs/aio_max_nr
- /proc/sys/fs/file_max

- OOM kills

OOM kills – lowmem consumption

```
Free pages:      9003696kB (8990400kB HighMem)
Active:323264 inactive:346882 dirty:327575 writeback:3686 unstable:0 free:2250924 slab:177094
mapped:15855 pagetables:987
DMA free:12640kB min:16kB low:32kB high:48kB active:0kB inactive:0kB present:16384kB
pages_scanned:149 all_unreclaimable? yes
protections[]: 0 0 0
Normal free:656kB min:928kB low:1856kB high:2784kB active:6976kB inactive:9976kB present:901120kB
pages_scanned:28281 all_unreclaimable? yes
protections[]: 0 0 0
HighMem free:8990400kB min:512kB low:1024kB high:1536kB active:1286080kB inactive:1377552kB
present:12451840kB pages_scanned:0 all_unreclaimable? no
protections[]: 0 0 0
DMA: 4*4kB 4*8kB 3*16kB 4*32kB 4*64kB 1*128kB 1*256kB 1*512kB 1*1024kB 1*2048kB 2*4096kB =
12640kB
Normal: 0*4kB 2*8kB 0*16kB 0*32kB 0*64kB 1*128kB 0*256kB 1*512kB 0*1024kB 0*2048kB 0*4096kB =
656kB
HighMem: 15994*4kB 17663*8kB 11584*16kB 8561*32kB 8193*64kB 1543*128kB 69*256kB 2101*512kB
1328*1024kB 765*2048kB 875*4096kB = 8990400kB
Swap cache: add 0, delete 0, find 0/0, race 0+0
Free swap:      8385912kB
3342336 pages of RAM
2916288 pages of HIGHMEM
224303 reserved pages
666061 pages shared
0 pages swap cached
Out of Memory: Killed process 22248 (httpd).
oom-killer: gfp_mask=0xd0
```

OOM kills – IO system stall

```
Free pages: 15096kB (1664kB HighMem) Active:34146 inactive:1995536 dirty:255
writeback:314829 unstable:0 free:3774 slab:39266 mapped:31803 pagetables:820
DMA free:12552kB min:16kB low:32kB high:48kB active:0kB inactive:0kB present:16384kB
pages_scanned:2023 all_unreclaimable? yes
protections[]: 0 0 0
Normal free:880kB min:928kB low:1856kB high:2784kB active:744kB inactive:660296kB
present:901120kB pages_scanned:726099 all_unreclaimable? yes
protections[]: 0 0 0
HighMem free:1664kB min:512kB low:1024kB high:1536kB active:135840kB inactive:7321848kB
present:7995388kB pages_scanned:0 all_unreclaimable? no
protections[]: 0 0 0
DMA: 2*4kB 4*8kB 2*16kB 4*32kB 3*64kB 1*128kB 1*256kB 1*512kB 1*1024kB 1*2048kB 2*4096kB =
12552kB
Normal: 0*4kB 18*8kB 14*16kB 0*32kB 0*64kB 0*128kB 0*256kB 1*512kB 0*1024kB 0*2048kB
0*4096kB = 880kB
HighMem: 6*4kB 9*8kB 66*16kB 0*32kB 0*64kB 0*128kB 0*256kB 1*512kB 0*1024kB 0*2048kB
0*4096kB = 1664kB
Swap cache: add 856, delete 599, find 341/403, race 0+0
0 bounce buffer pages
Free swap:      4193264kB
2228223 pages of RAM
1867481 pages of HIGHMEM
150341 reserved pages
343042 pages shared
257 pages swap cached
kernel: Out of Memory: Killed process 3450 (hpsmhd).
```

Eliminating OOMkills

- RHEL4

- `/proc/sys/vm/oom-kill` – oom kill enable/disable flag(default 1).

- RHEL5

- `/proc/<pid>/oom_adj` – per-process OOM adjustment(-17 to +15)
- Set to -17 to disable that process from being OOM killed
- Decrease to decrease OOM kill likelihood.
- Increase to increase OOM kill likelihood.
- `/proc/<pid>/oom_score` – current OOM kill priority.

General Performance Tuning Considerations

Over Committing RAM

Swap device location

Storage device and limits limits

Kernel selection

Performance Tuning

Kernel Selection

VM tuning

Processor related tuning

NUMA related tuning

Disk & IO tuning

Hugepages

KVM host and guests

RHEL4 kernel selection

- x86

- Standard kernel(no PAE, 3G/1G)
 - UP systems with ≤ 4 GB RAM
- SMP kernel(PAE, 3G/1G)
 - SMP systems with $< \sim 16$ GB RAM
- Highmem/Lowmem ratio $\leq 16:1$
 - Hugemem kernel(PAE, 4G/4G)
 - SMP systems $> \sim 16$ GB RAM

- X86_64

- Standard kernel for UP systems
- SMP kernel for systems with up to 8 CPUs
- LargeSMP kernel for systems up to 512 CPUs

RHEL5 kernel selection

- x86

- Standard kernel(no PAE, 3G/1G)
- UP and SMP systems with \leq 4GB RAM
- PAE kernel(PAE, 3G/1G)
- UP and SMP systems with $>$ 4GB RAM

- X86_64

- Standard kernel for all systems

- IA64

- Standard kernel for all systems

VM: swappiness

Controls how aggressively the system reclaims “mapped” memory:

Anonymous memory - swapping

Mapped file pages – writing if dirty and freeing

System V shared memory - swapping

Decreasing: more aggressive reclaiming of unmapped pagecache memory

Increasing: more aggressive swapping of mapped memory

/proc/sys/vm/swappiness

Sybase server with /proc/sys/vm/swappiness set to 60(default)

procs		-----memory-----				---swap--		-----io-----		--system--		----cpu----			
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa
5	1	643644	26788	3544	32341788	880	120	4044	7496	1302	20846	25	34	25	16

Sybase server with /proc/sys/vm/swappiness set to 10

procs		-----memory-----				---swap--		-----io-----		--system--		----cpu----			
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa
8	3	0	24228	6724	32280696	0	0	23888	63776	1286	20020	24	38	13	26

`/proc/sys/vm/min_free_kbytes`

Directly controls the page reclaim watermarks in KB

```
# echo 1024 > /proc/sys/vm/min_free_kbytes
```

```
-----  
Node 0 DMA free:4420kB min:8kB low:8kB high:12kB
```

```
Node 0 DMA32 free:14456kB min:1012kB low:1264kB high:1516kB  
-----
```

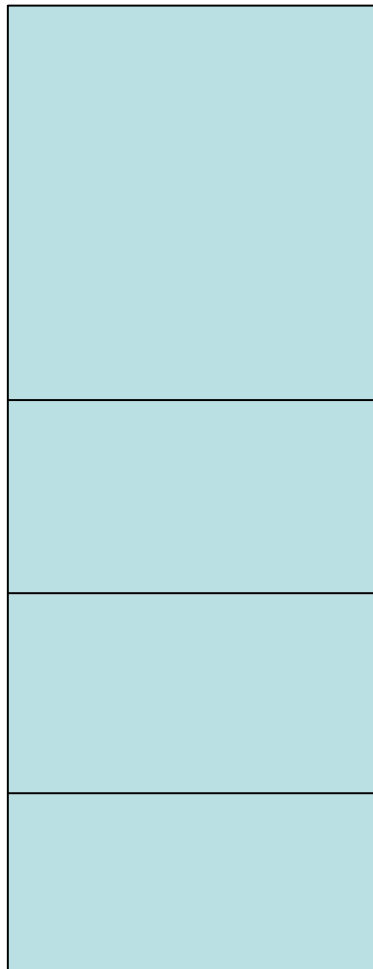
```
echo 2048 > /proc/sys/vm/min_free_kbytes
```

```
-----  
Node 0 DMA free:4420kB min:20kB low:24kB high:28kB
```

```
Node 0 DMA32 free:14456kB min:2024kB low:2528kB high:3036kB  
-----
```

Memory reclaim Watermarks - min_free_kbytes

Free List



All of RAM

Do nothing

Pages High – kswapd sleeps above High

kswapd reclaims memory

Pages Low – kswapd wakes up at Low

kswapd reclaims memory

Pages Min – all memory allocators reclaim at Min

user processes/kswapd reclaim memory

0

`/proc/sys/vm/dirty_ratio`

Absolute limit to percentage of dirty pagecache memory

Default is 40%

Lower means less dirty pagecache and smaller IO streams

Higher means more dirty pagecache and larger IO streams

`/proc/sys/vm/dirty_background_ratio`

- Controls when dirty pagecache memory starts getting written.
- Default is 10%
- Lower
 - pdflush starts earlier
 - less dirty pagecache and smaller IO streams
- Higher
 - pdflush starts later
 - more dirty pagecache and larger IO streams

dirty_ratio and dirty_background_ratio

pagecache



100% of pagecache RAM dirty

pdflushd and write()'ng processes write dirty buffers

dirty_ratio(40% of RAM dirty) – processes start synchronous writes

pdflushd writes dirty buffers in background

dirty_background_ratio(10% of RAM dirty) – wakeup pdflushd

do_nothing

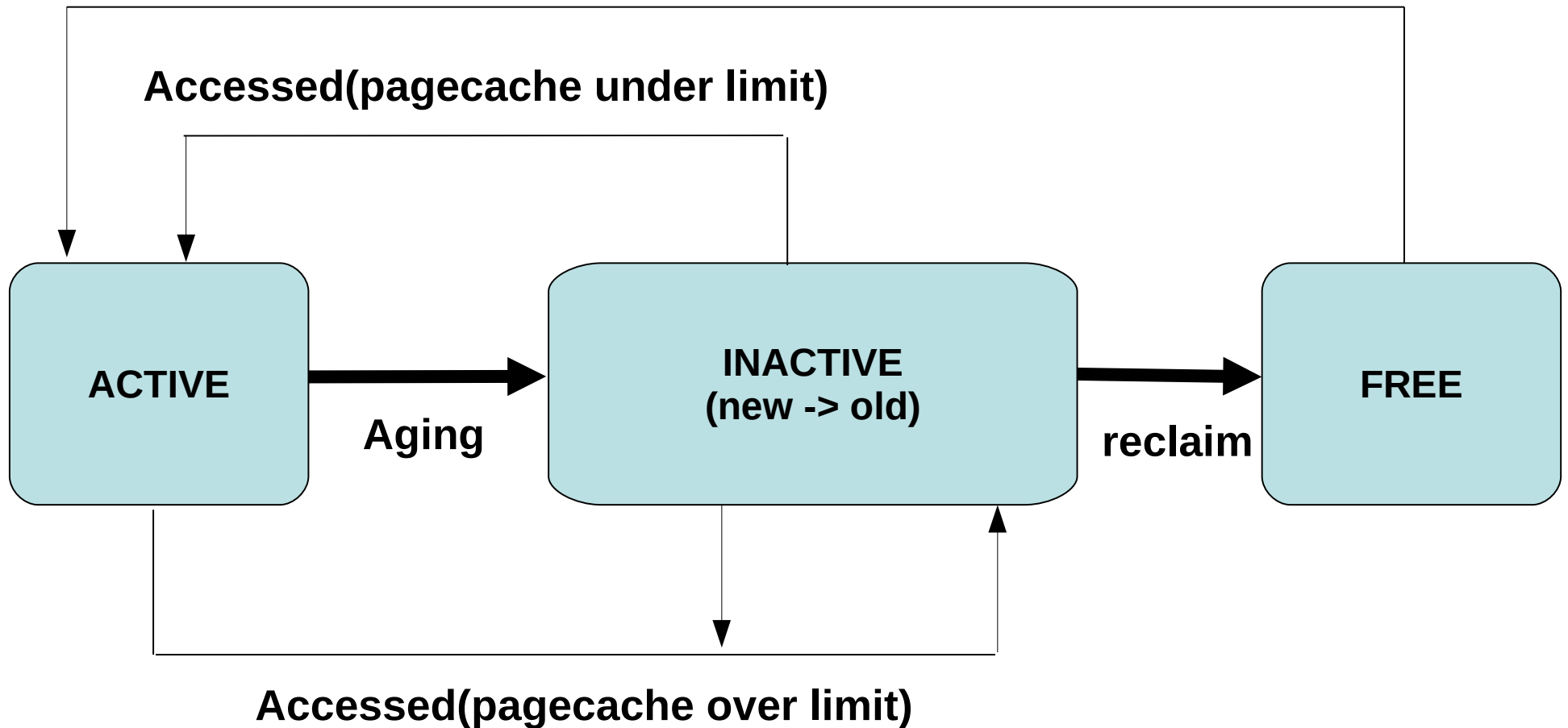
0% of pagecache RAM dirty

`/proc/sys/vm/pagecache`

- Controls when pagecache memory is deactivated.
- Default is 100%
- Lower
 - Prevents swapping out anonymous memory
- Higher
 - Favors pagecache pages
 - Disabled at 100%

Pagecache Tuning

Filesystem/pagecache Allocation



(Hint)flushing the pagecache

```
echo 1 > /proc/sys/vm/drop_caches
```

```
procs -----memory----- ---swap-- -----io----- --system-- -----cpu-----
r  b   swpd   free   buff  cache   si   so    bi    bo    in    cs  us  sy  id  wa
0  0     224   57184 107808 3350196   0    0     0    56  1136   212  0  0  83  17
0  0     224   57184 107808 3350196   0    0     0     0  1039   198  0  0 100  0
0  0     224   57184 107808 3350196   0    0     0     0  1021   188  0  0 100  0
0  0     224   57184 107808 3350196   0    0     0     0  1035   204  0  0 100  0
0  0     224   57248 107808 3350196   0    0     0     0  1008   164  0  0 100  0
3  0     224 2128160    176 1438636   0    0     0     0  1030   197  0  0  15  85  0
0  0     224 3610656    204  34408   0    0    28    36  1027   177  0  0  32  67  2
0  0     224 3610656    204  34408   0    0     0     0  1026   180  0  0 100  0
0  0     224 3610720    212  34400   0    0     8     0  1010   183  0  0  99  1
```

(Hint)flushing the slabcache

```
echo 2 > /proc/sys/vm/drop_caches
```

```
[tmp]# cat /proc/meminfo  
MemTotal:    3907444 kB  
MemFree:     3104576 kB
```

```
Slab:        415420 kB
```

```
Hugepagesize: 2048 kB
```

```
tmp]# cat /proc/meminfo  
MemTotal:    3907444 kB  
MemFree:     3301788 kB
```

```
Slab:        218208 kB
```

```
Hugepagesize: 2048 kB
```

RHEL5 CPU speed and performance:

Enabled = governor set to “ondemand”

Looks at cpu usage to regulate power

Within 3-5% of performance for cpu loads

IO loads can keep cpu stepped down -15-30%

Supported in RHEL5 virtualization

To turn off – else may leave cpu’s in reduced step

If its not using performance, then:

```
# echo performance > /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
```

Then check to see if it stuck:

```
# cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
```

Check /proc/cpuinfo to make sure your seeing the expected CPU freq.

Proceed to “normal” service disable

Service cpuspeed stop

Chkconfig cpuspeed off

CPU Scheduler

Recognizes differences between logical and physical processors

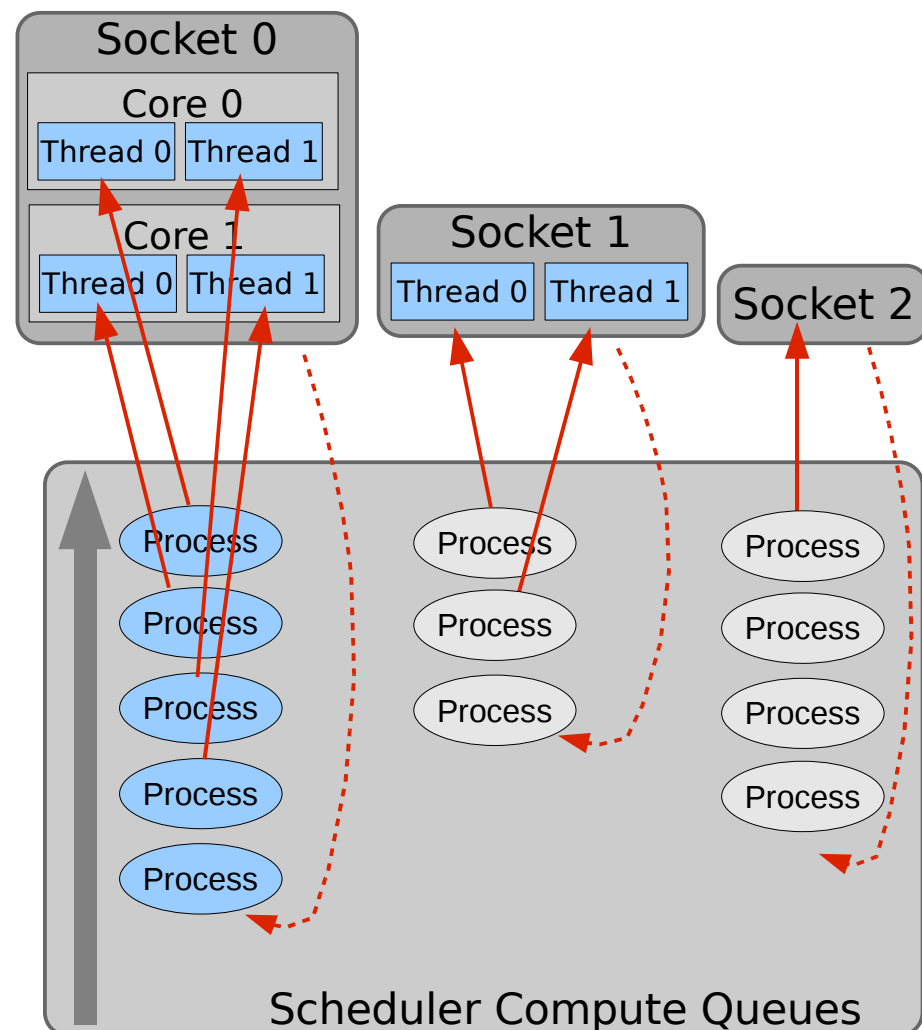
I.E. Multi-core, hyperthreaded & chips/sockets

Optimizes process scheduling to take advantage of shared on-chip cache, and NUMA memory nodes

Implements multilevel run queues for sockets and cores (as opposed to one run queue per processor or per system)

Strong CPU affinity avoids task bouncing

Requires system BIOS to report CPU topology correctly



NUMA related Tuning

Numastat

Numactl

Hugetlbfs

`/sys/devices/system/node`

RHEL4&5 NUMAstat and NUMActl

EXAMPLES

`numactl --interleave=all bigdatabase arguments` Run big database with its memory interleaved on all CPUs.

`numactl --cpubind=0--membind=0,1 process` Run process on node 0 with memory allocated on node 0 and 1.

`numactl --preferred=1` `numactl --show` Set preferred node 1 and show the resulting state.

`numactl --interleave=all --shmkeyfile /tmp/shmkey` Interleave all of the sysv shared memory region specified by /tmp/shmkey over all nodes.

`numactl --offset=1G --length=1G --membind=1 --file /dev/shm/A --touch` Bind the second gigabyte in the tmpfs file /dev/shm/A to node 1.

`numactl --localalloc /dev/shm/file` Reset the policy for the shared memory file file to the default localalloc policy.

RHEL4&5 NUMAstat and NUMActl

NUMAstat to display system NUMA characteristics on a numasystem

```
[root@perf5 ~]# numastat
```

	node3	node2	node1	node0
numa_hit	72684	82215	157244	325444
numa_miss	0	0	0	0
numa_foreign	0	0	0	0
interleave_hit	2668	2431	2763	2699
local_node	67306	77456	152115	324733
other_node	5378	4759	5129	711

NUMActl to control process and memory”

TIP numactl [--interleave nodes] [--preferred node] [--mbind nodes]
[--cpubind nodes] [--localalloc] command {arguments ...}

App < memory single NUMA zone

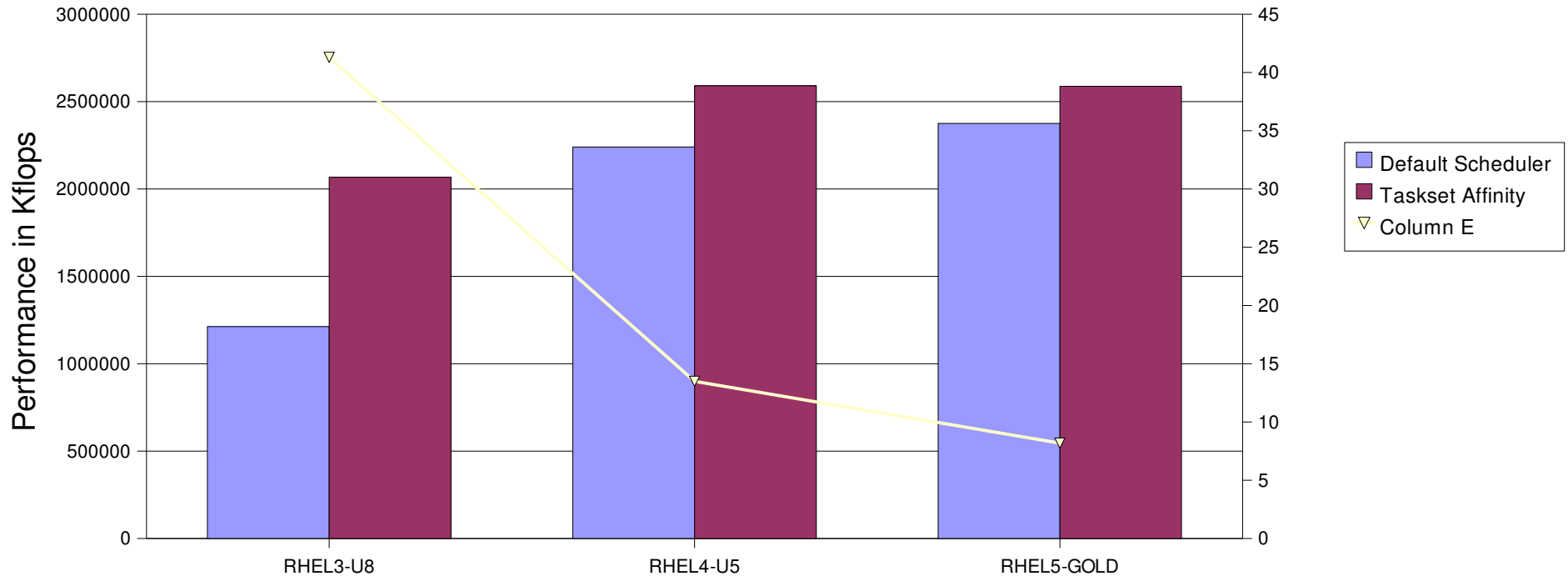
Numactl use -cpubind cpus within same socket

App > memory of a single NUMA zone

Numactl -interleave XY and -cpubind XY

Linux NUMA Evolution (NEWer)

RHEL3, 4 and 5 Linpack Multi-stream
AMD64, 8cpu - dualcore (1/2 cpus loaded)



Limitations :

Numa “spill” to different numa boundaries

Process migrations – no way back

Lack of page replication – text, read mostly

HugeTLBFS

The Translation Lookaside Buffer (TLB) is a small CPU cache of recently used virtual to physical address mappings

TLB misses are extremely expensive on today's very fast, pipelined CPUs

Large memory applications can incur high TLB miss rates

HugeTLBs permit memory to be managed in very large segments

Example: x86_64

Standard page: 4KB

Huge page: 2MB

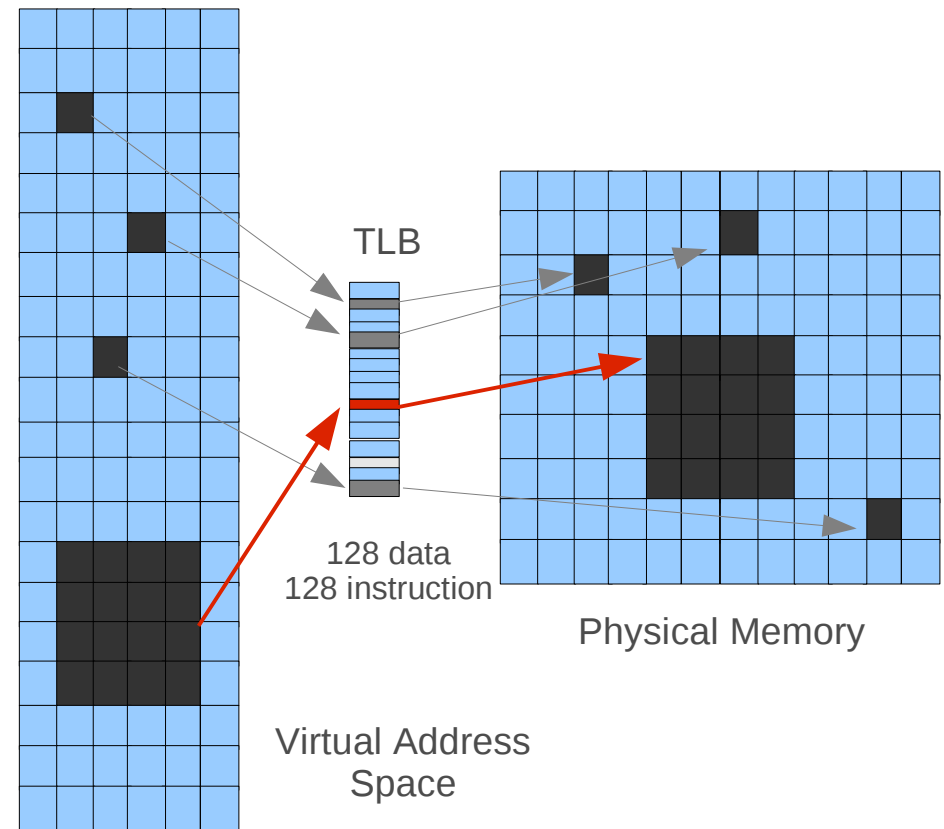
512:1 difference

File system mapping interface

Ideal for databases

Example: 128 entry TLB can fully map 256MB

* RHEL6 – 1GB hugepage support



Hugepages - before

\$vmstat

```
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 15623656 31044 401120    0    0   187   14   163   75  1  0  97  2  0
```

\$cat /proc/meminfo

```
MemTotal:      16301368 kB
MemFree:       15623604 kB
...
HugePages_Total:      0
HugePages_Free:      0
HugePages_Rsvd:      0
Hugepagesize:      2048 kB
```

Hugepages reserving

```
$echo 2000 > /proc/sys/vm/nr_hugepages
```

```
$vmstat
```

```
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 11526632 31168 401780    0    0   129   10  156   63  1  0 98  1
0
```

```
$cat /proc/meminfo
```

```
MemTotal:      16301368 kB
```

```
MemFree:       11526520 kB
```

```
...
```

```
HugePages_Total: 2000
```

```
HugePages_Free: 2000
```

```
HugePages_Rsvd: 0
```

```
Hugepagesize: 2048 kB
```

Hugepages - using

```
$mount -t hugetlbfs hugetlbfs /huge
$cp 1GB-file /huge/junk
```

```
$vmstat
```

```
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 10526632  31168 1401780    0    0    129    10   156   63   1   0  98   1
0
```

```
$cat /proc/meminfo
```

```
LowTotal:      16301368 kB
LowFree:       11524756 kB
...
HugePages_Total: 2000
HugePages_Free: 1488
HugePages_Rsvd: 0
Hugepagesize: 2048 kB
```

Hugepages - releasing

```
$rm /huge/junk
```

```
$cat /proc/meminfo
```

```
MemTotal:      16301368 kB
```

```
MemFree:       11524776 kB
```

```
...
```

```
HugePages_Total: 2000
```

```
HugePages_Free: 2000
```

```
HugePages_Rsvd: 0
```

```
Hugepagesize: 2048 kB
```

```
$echo 0 > /proc/sys/vm/nr_hugepages
```

```
$vmstat
```

```
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 15620488  31512 401944    0    0    71    6    6  149  59  1  0 98  1  0
```

```
$cat /proc/meminfo
```

```
MemTotal:      16301368 kB
```

```
MemFree:       15620500 kB
```

```
...
```

```
HugePages_Total: 0
```

```
HugePages_Free: 0
```

```
HugePages_Rsvd: 0
```

```
Hugepagesize: 2048 kB
```

NUMA Hugepages - reserving

```
[root@dhcp-100-19-50 ~]# cat /sys/devices/system/node/*/meminfo | grep Huge
```

```
Node 0 HugePages_Total: 0
```

```
Node 0 HugePages_Free: 0
```

```
Node 1 HugePages_Total: 0
```

```
Node 1 HugePages_Free: 0
```

```
[root@dhcp-100-19-50 ~]# echo 6000 > /proc/sys/vm/nr_hugepages
```

```
[root@dhcp-100-19-50 ~]# cat /sys/devices/system/node/*/meminfo | grep Huge
```

```
Node 0 HugePages_Total: 2980
```

```
Node 0 HugePages_Free: 2980
```

```
Node 1 HugePages_Total: 3020
```

```
Node 1 HugePages_Free: 3020
```

NUMA Hugepages - using

```
[root@dhcp-100-19-50 ~]# mount -t hugetlbfs hugetlbfs /huge
```

```
[root@dhcp-100-19-50 ~]# /usr/tmp/mmapwrite /huge/junk 32 &
```

```
[1] 18804
```

```
[root@dhcp-100-19-50 ~]# Writing 1048576 pages of random junk to file /huge/junk
```

```
wrote 4294967296 bytes to file /huge/junk
```

```
[root@dhcp-100-19-50 ~]# cat /sys/devices/system/node/*/meminfo | grep Huge
```

```
Node 0 HugePages_Total: 2980
```

```
Node 0 HugePages_Free: 2980
```

```
Node 1 HugePages_Total: 3020
```

```
Node 1 HugePages_Free: 972
```

NUMA Hugepages - using_(overcommit)

```
[root@dhcp-100-19-50 ~]# /usr/tmp/mmapwrite /huge/junk 33 &
```

```
[1] 18815
```

```
[root@dhcp-100-19-50 ~]# Writing 2097152 pages of random junk to file /huge/junk
```

```
wrote 8589934592 bytes to file /huge/junk
```

```
[root@dhcp-100-19-50 ~]# cat /sys/devices/system/node/*/meminfo | grep Huge
```

```
Node 0 HugePages_Total: 2980
```

```
Node 0 HugePages_Free: 1904
```

```
Node 1 HugePages_Total: 3020
```

```
Node 1 HugePages_Free: 0
```

NUMA Hugepages - reducing

```
[root@dhcp-100-19-50 ~]# cat /sys/devices/system/node/*/meminfo | grep Huge
```

```
Node 0 HugePages_Total: 2980
```

```
Node 0 HugePages_Free: 2980
```

```
Node 1 HugePages_Total: 3020
```

```
Node 1 HugePages_Free: 3020
```

```
[root@dhcp-100-19-50 ~]# echo 3000 > /proc/sys/vm/nr_hugepages
```

```
[root@dhcp-100-19-50 ~]# cat /sys/devices/system/node/*/meminfo | grep Huge
```

```
Node 0 HugePages_Total: 0
```

```
Node 0 HugePages_Free: 0
```

```
Node 1 HugePages_Total: 3000
```

```
Node 1 HugePages_Free: 3000
```

NUMA Hugepages - freeing/reserving

```
[root@dhcp-100-19-50 ~]# echo 6000 > /proc/sys/vm/nr_hugepages
```

```
[root@dhcp-100-19-50 ~]# cat /sys/devices/system/node/*/meminfo | grep Huge
```

```
Node 0 HugePages_Total: 2982
```

```
Node 0 HugePages_Free: 2982
```

```
Node 1 HugePages_Total: 3018
```

```
Node 1 HugePages_Free: 3018
```

```
[root@dhcp-100-19-50 ~]# echo 0 > /proc/sys/vm/nr_hugepages
```

```
[root@dhcp-100-19-50 ~]# echo 3000 > /proc/sys/vm/nr_hugepages
```

```
[root@dhcp-100-19-50 ~]# cat /sys/devices/system/node/*/meminfo | grep Huge
```

```
Node 0 HugePages_Total: 1500
```

```
Node 0 HugePages_Free: 1500
```

```
Node 1 HugePages_Total: 1500
```

```
Node 1 HugePages_Free: 1500
```

JVM Tuning

Eliminate swapping

Lower swappiness to 10%(or lower if necessary).

Promote pagecache reclaiming

Lower dirty_background_ratio to 10%

Lower dirty_ratio if necessary

Promote inode cache reclaiming

Lower vfs_cache_pressure

Section 4: Tuning Examples

JVM Tuning

SPECjbb

File systems

EXT3, GFS, NFS, EXT4, XFS

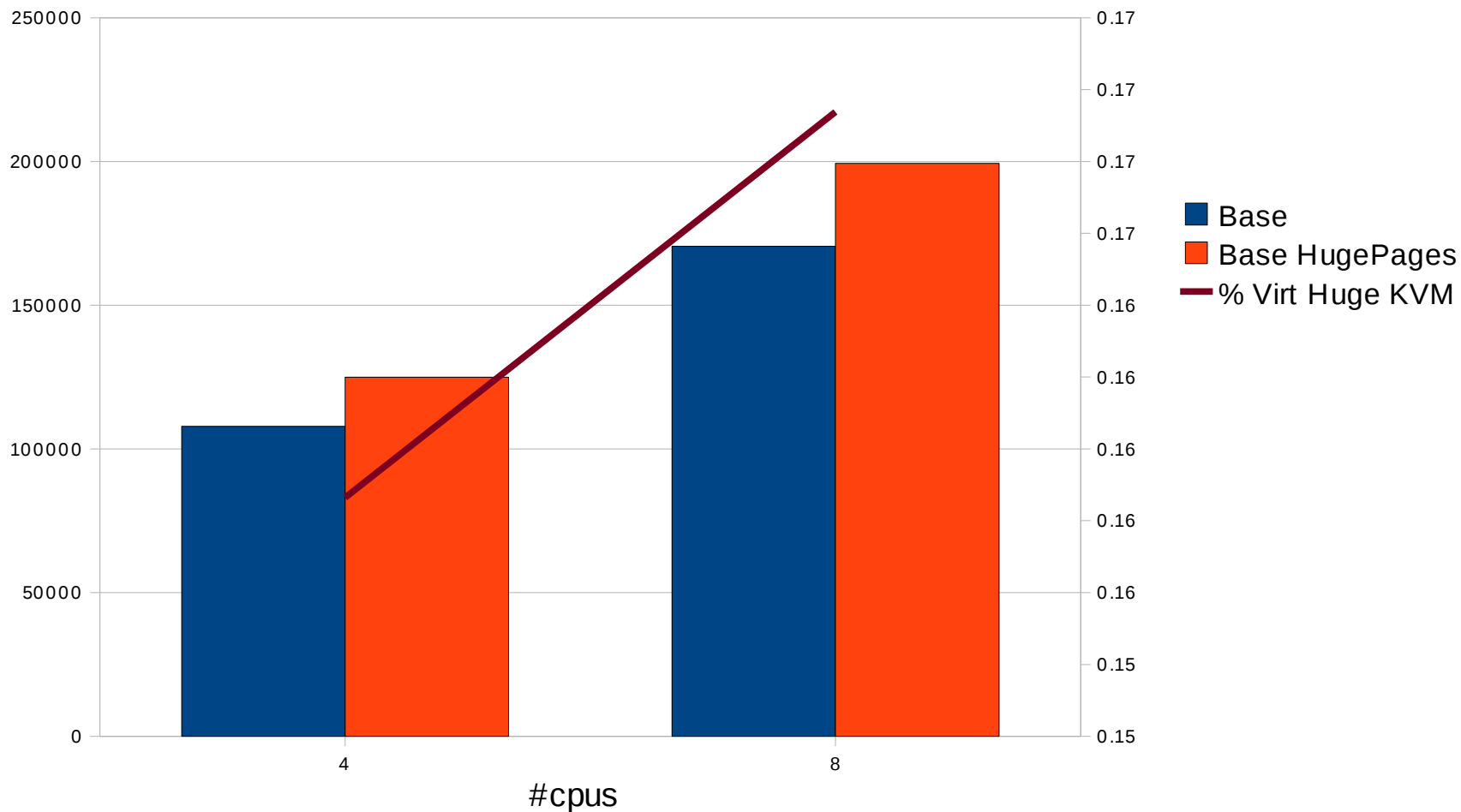
Database Performance

Scaling, Hugepages, AIO/DIO, Elevators, SELinux

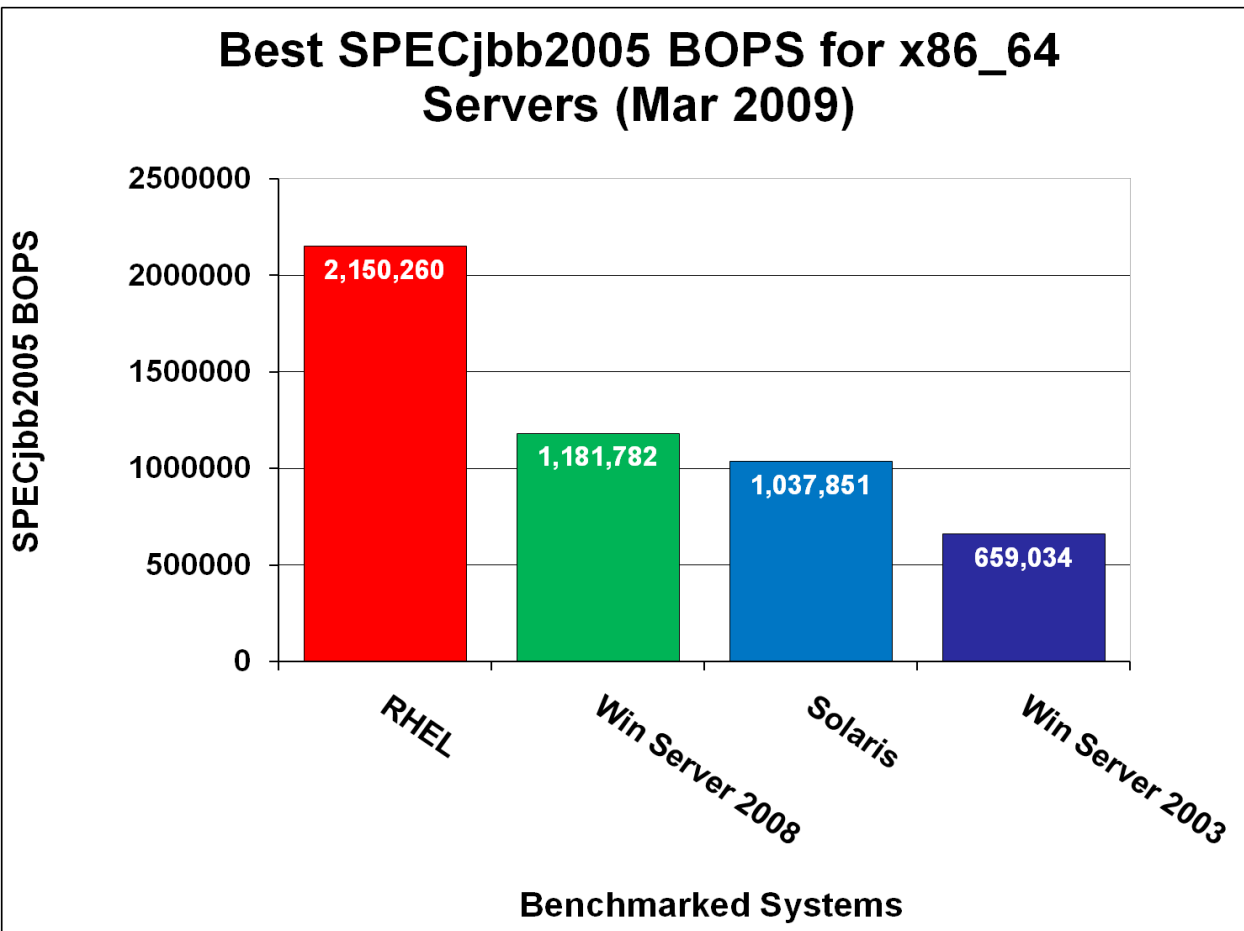
Network Tuning

RHEL5.4 KVM Java OpenJDK Performance

Intel Nahalem 2.4 Ghz, 24 Gb mem



Java Performance – SPECjbb[®]2005 Benchmark on RHEL (contd.)



- In February 2009 RHEL running on a 96-core Xeon based server from NEC achieved the best score achieved on a x86_64 server (= 2,150,260 BOPS).
- **Use of NUMActl and HugePages critical to result**
- **See Reference Architecture talk on Benchmark Papers/ Results**

General Performance Tuning Guidelines

Use hugepages whenever possible.

Minimize swapping.

Maximize pagecache reclaiming

Place swap partition(s) on quite device(s).

Direct IO if possible.

Beware of turning NUMA off.

Understanding IOzone Results

GeoMean per category are statistically meaningful.

Understand HW setup

Disk, RAID, HBA, PCI

Layout file systems

LVM or MD devices

Partions w/ fdisk

Baseline raw IO DD/DT

EXT3 perf w/ IOzone

In-cache – file sizes which fit goal -> 90% memory BW.

Out-of-cache – file sizes more tan 2x memory size

O_DIRECT – 95% of raw

Global File System – GFS goal --> 90-95% of local EXT3

Use raw command

```
fdisk /dev/sdX
```

```
raw /dev/raw/rawX /dev/sdX1
```

```
dd if=/dev/raw/rawX bs=64k
```

Mount file system

```
mkfs -t ext3 /dev/sdX1
```

```
Mount -t ext3 /dev/sdX1 /perf1
```

IOzone commands

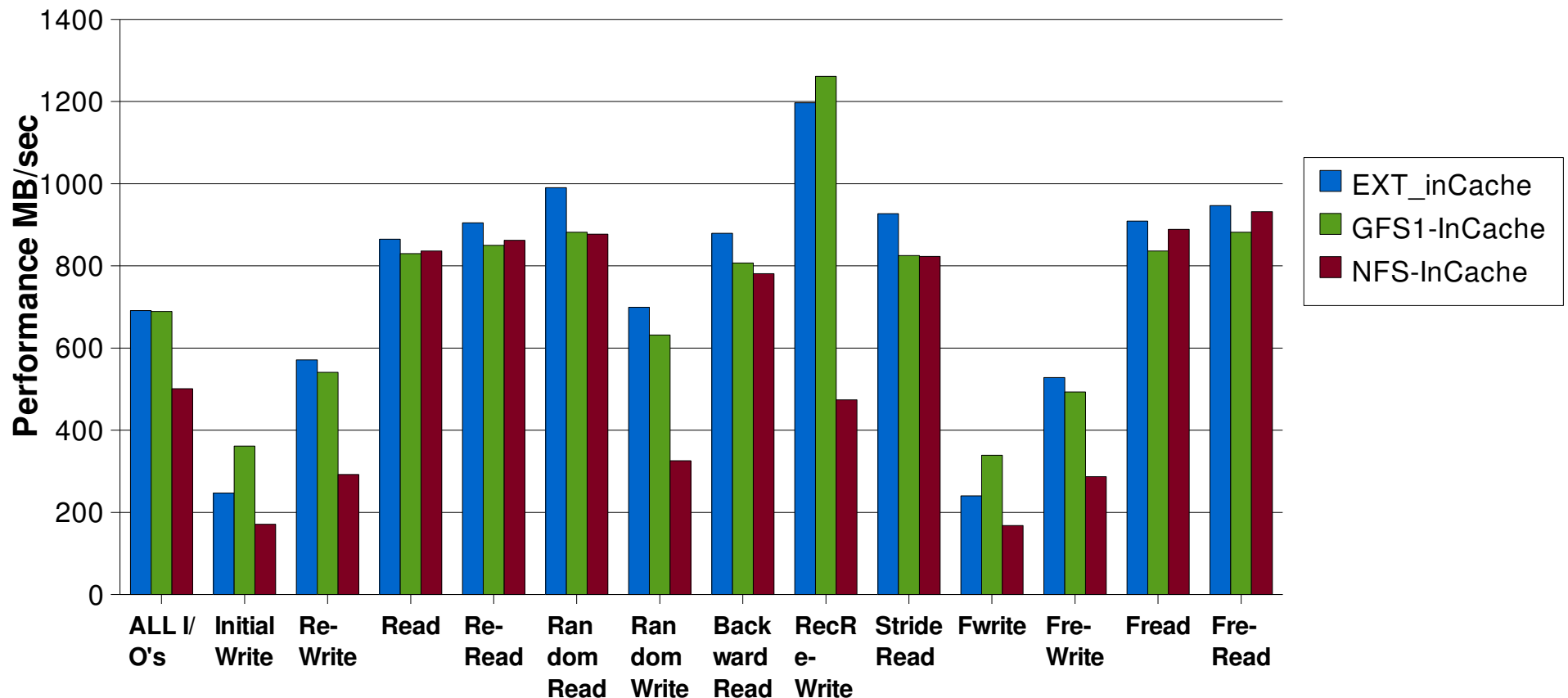
```
lozone -a -f /perf1/t1 (incache)
```

```
lozone -a -l -f /perf1/t1 (w/ dio)
```

```
lozone -s 2xmem -f /perf1/t1 (big)
```

EXT3, GFS, NFS Iozone in cache

RHEL5 In-Cache IOzone EXT3, GFS1, GFS2
(Geom 1M-4GB, 1k-1m)



Using IOzone w/ o_direct – mimic database

Problem :

Filesystems use memory for file cache

Databases use memory for database cache

Users want filesystem for management outside database access (copy, backup etc)

You DON'T want BOTH to cache.

Solution :

Filesystems that support Direct IO

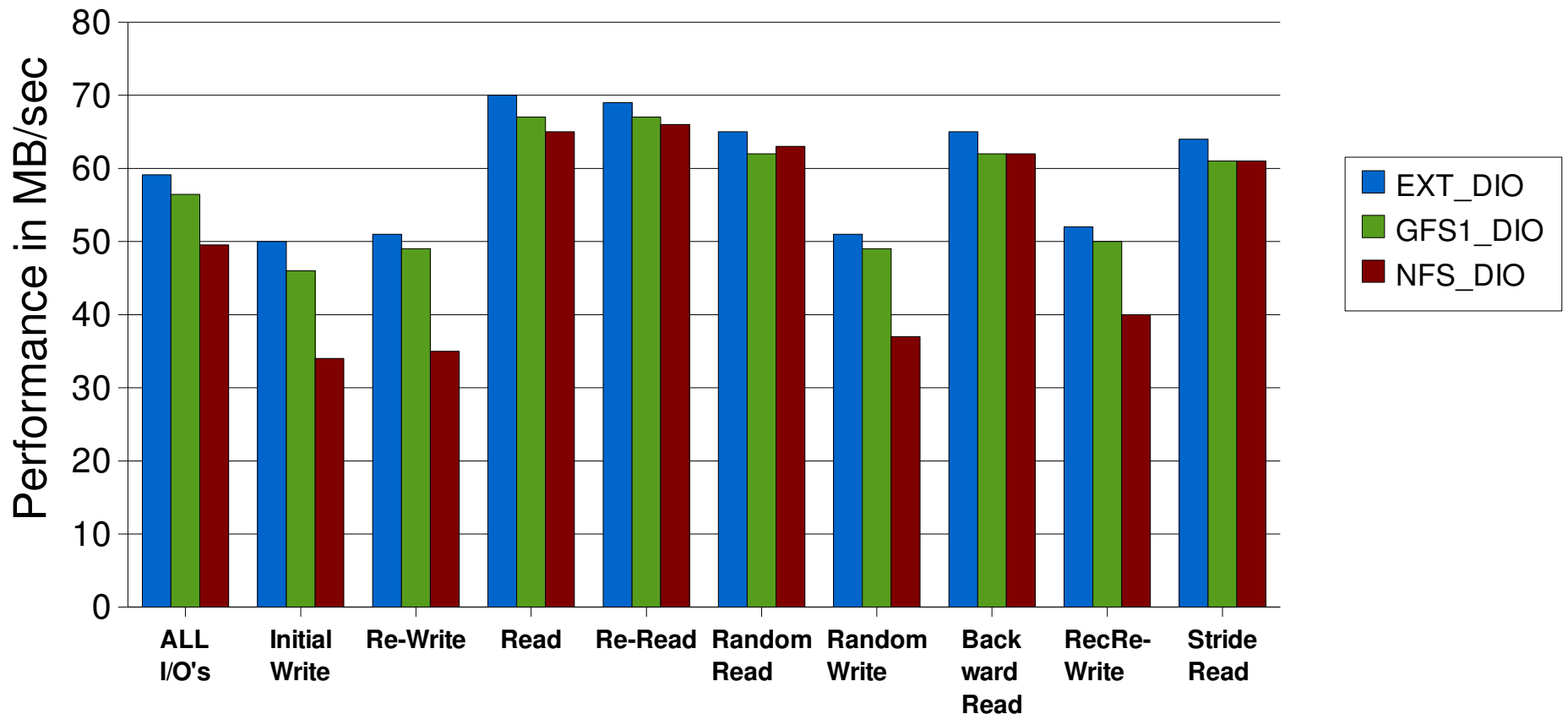
Open files with o_direct option

Databases which support Direct IO (ORACLE)

NO DOUBLE CACHING!

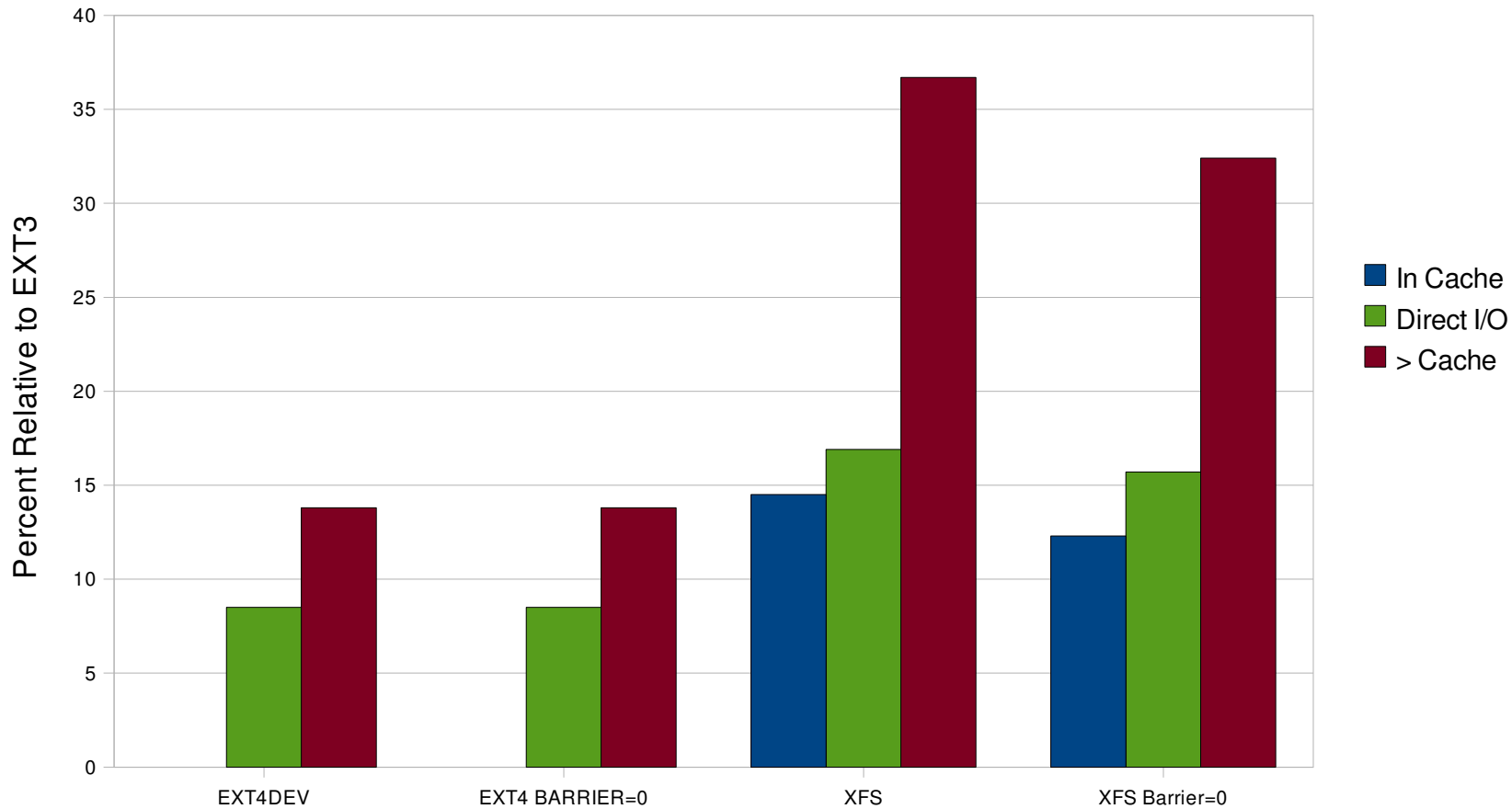
EXT3, GFS, NFS I/Ozone w/ DirectIO

RHEL5 Direct_IO IOzone EXT3, GFS, NFS
(Geom 1M-4GB, 1k-1m)



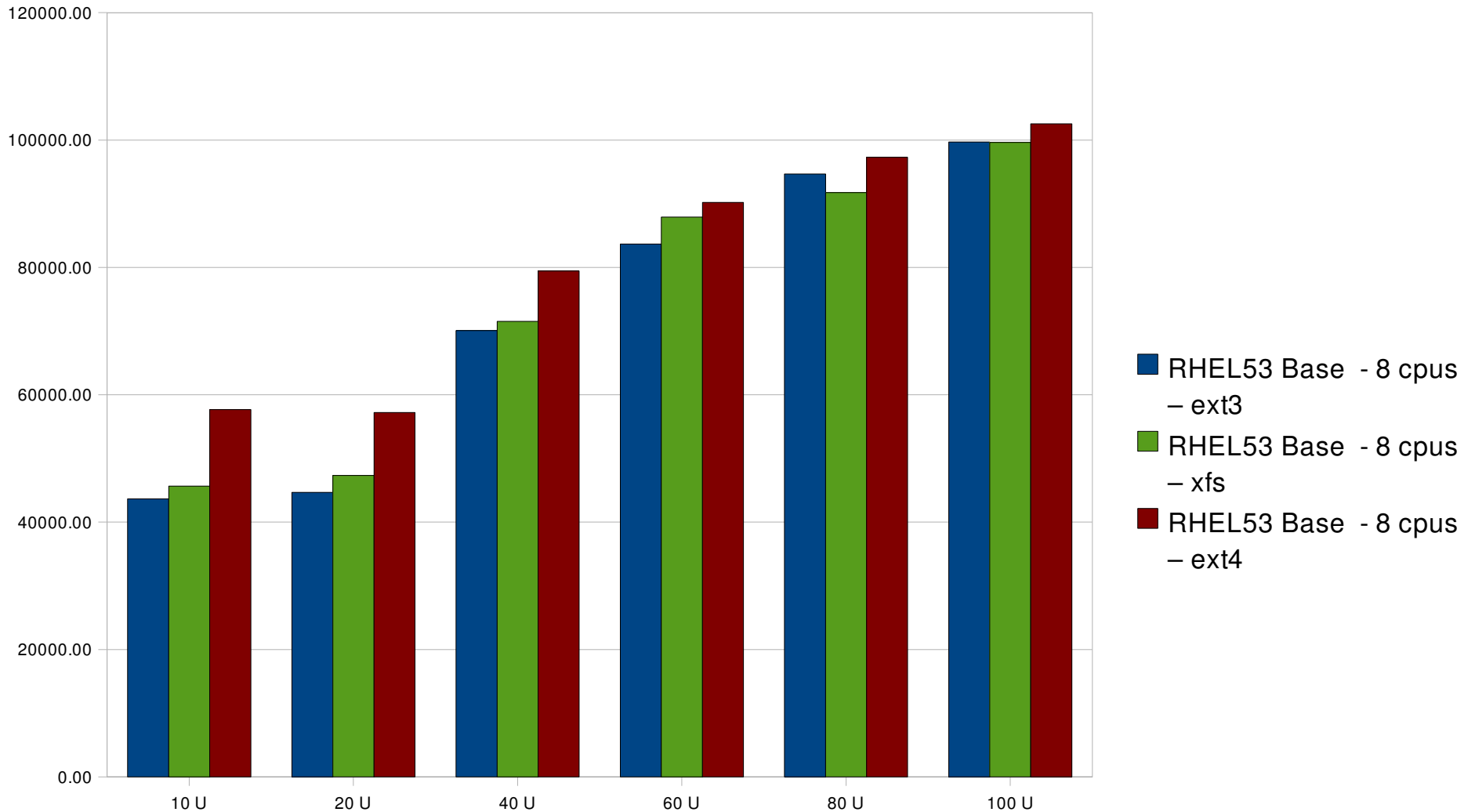
RHEL5.3 IOzone EXT3, EXT4, XFS eval

RHEL53 (120), IOzone Performance
Geo Mean 1k points, Intel 8cpu, 16GB, FC



RHEL5 Oracle 10.2 Performance Filesystems

Intel 8-cpu, 16GB, 2 FC MPIO, AIO/DIO



Large App and Database Performance

Scaling 1-24 core single servers

Huge Pages

2MB huge pages

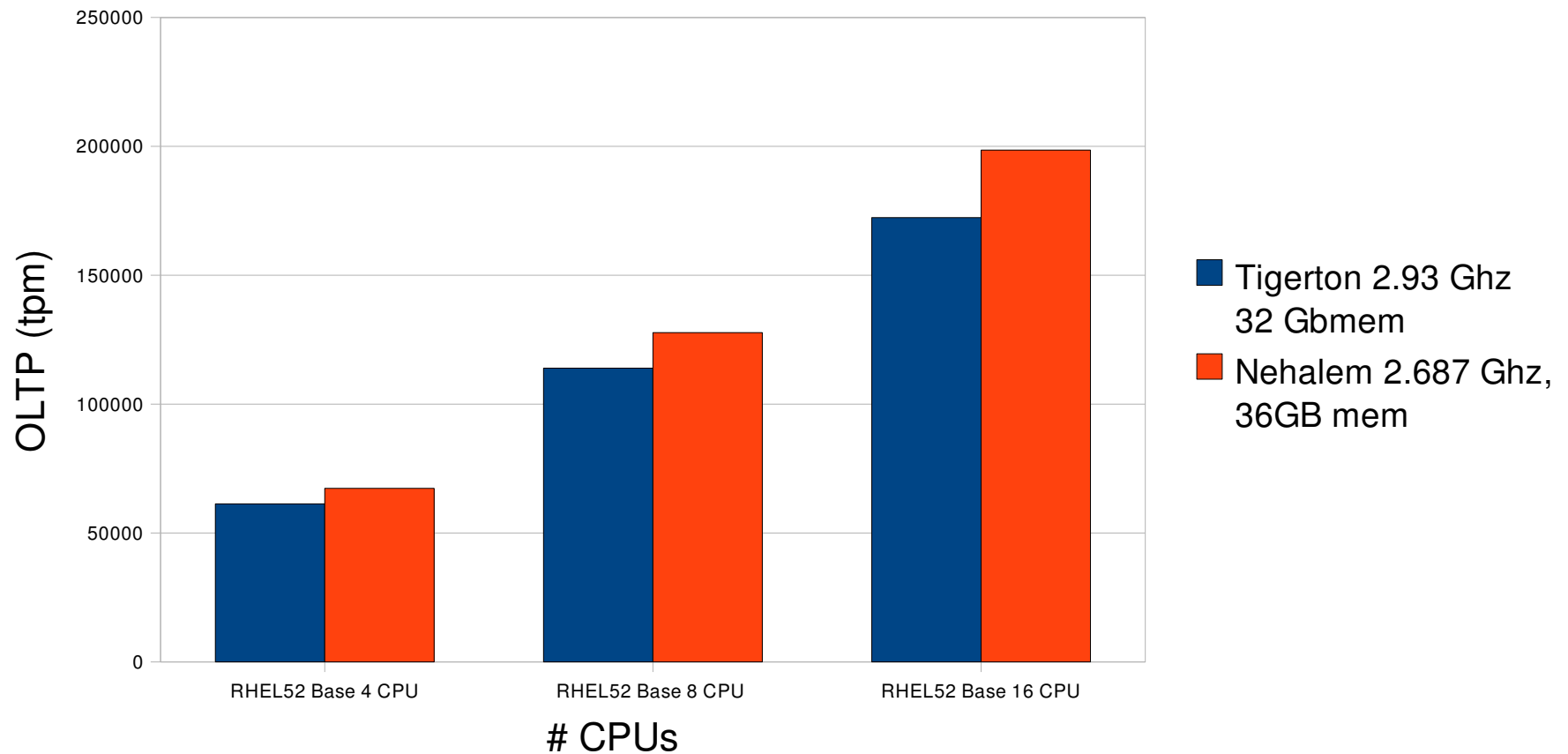
Set value in `/etc/sysctl.conf` (`vm.nr_hugepages`)

NUMA

Localized memory access for certain workloads improves performance

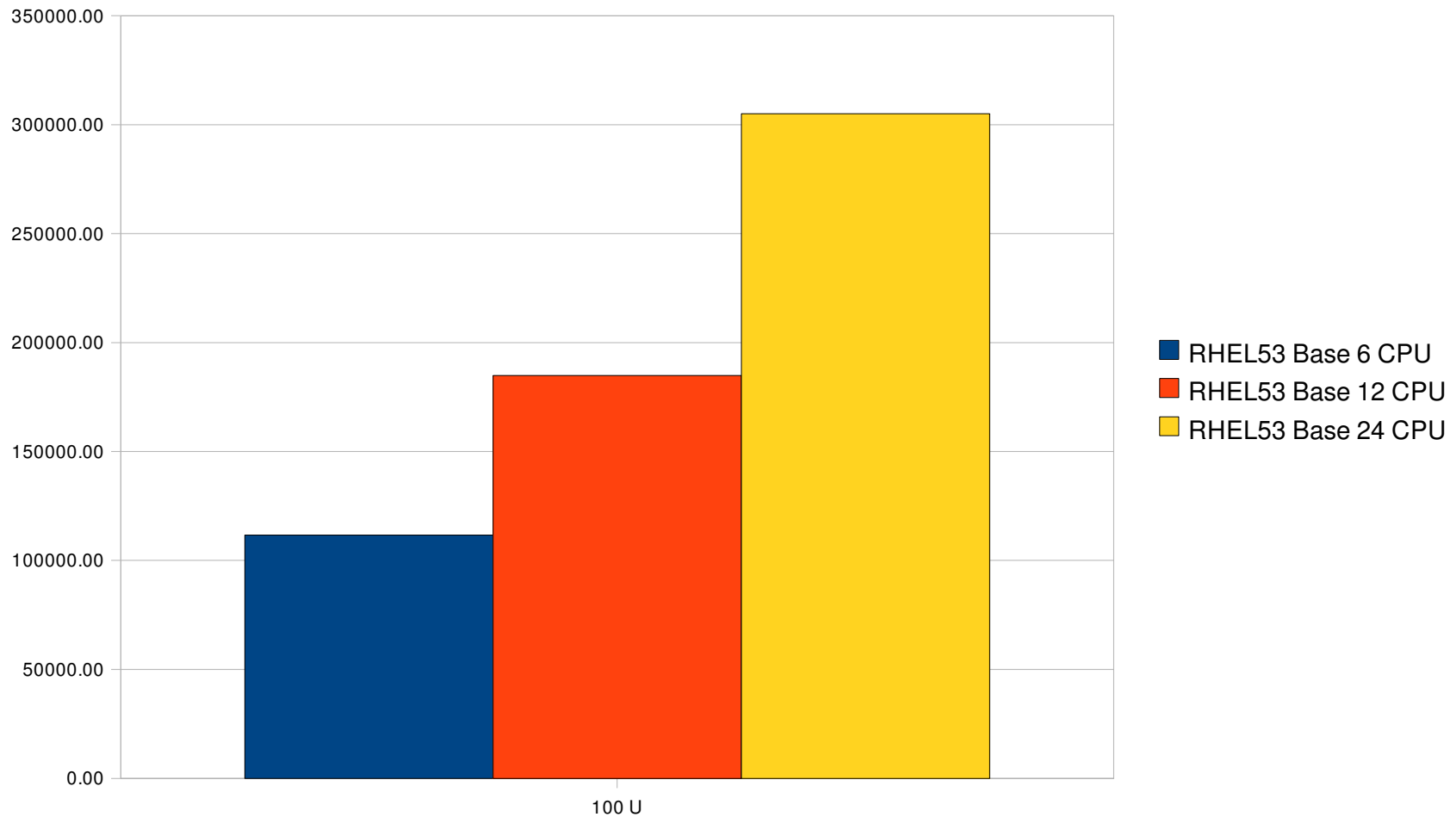
Oracle OLTP Performance Scaling at 16-cpu

RHEL5.4 Oracle OLTP Performance



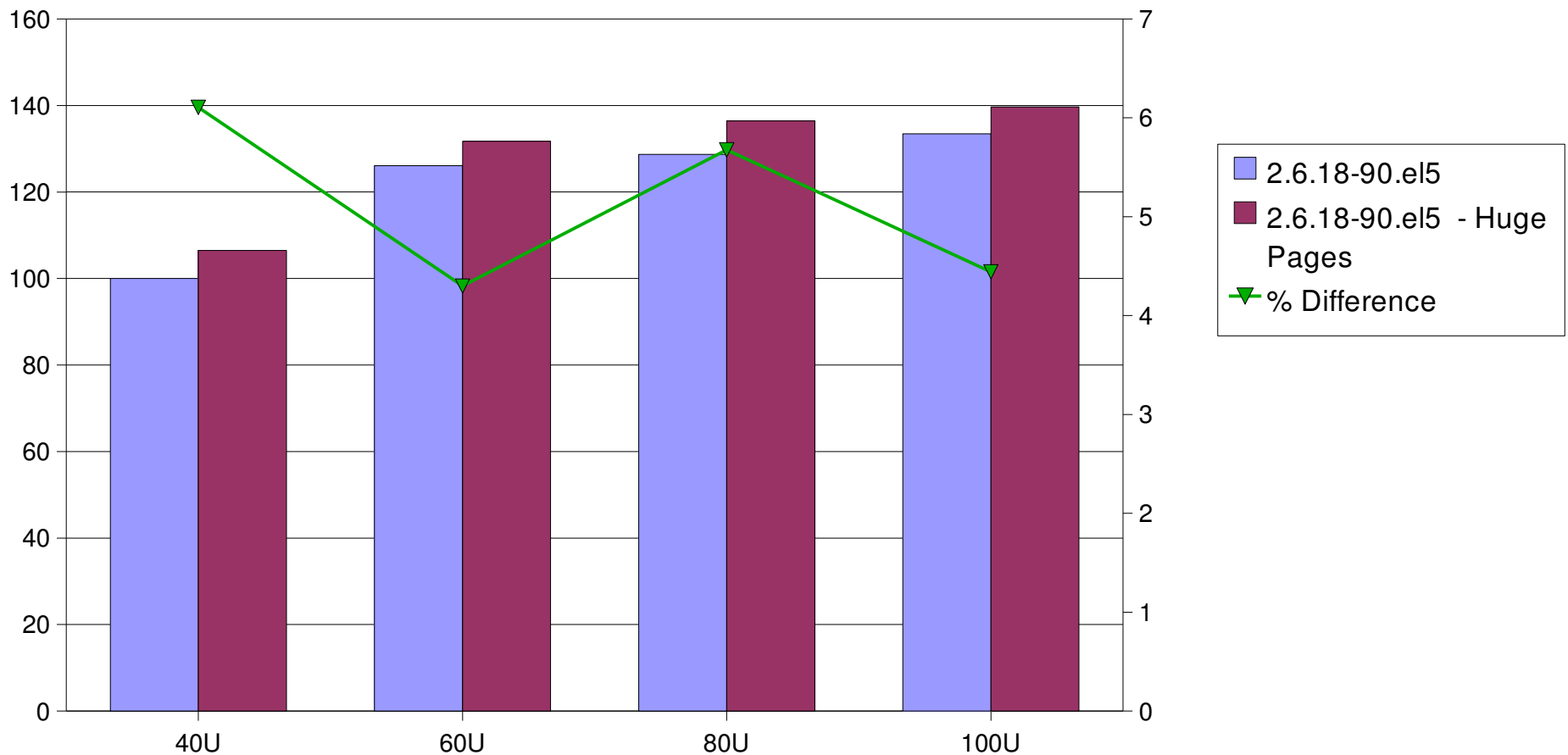
Oracle OLTP Performance Scaling at 24-cpu

Oracle Scaling on Intel Dunnington



RHEL5.2 -Oracle 10.2 Huge pages

Relative performance



Asynchronous I/O to File Systems

Allows application to continue processing while I/O is in progress

Eliminates Synchronous I/O stall

Critical for I/O intensive server applications

Red Hat Enterprise Linux – since 2002

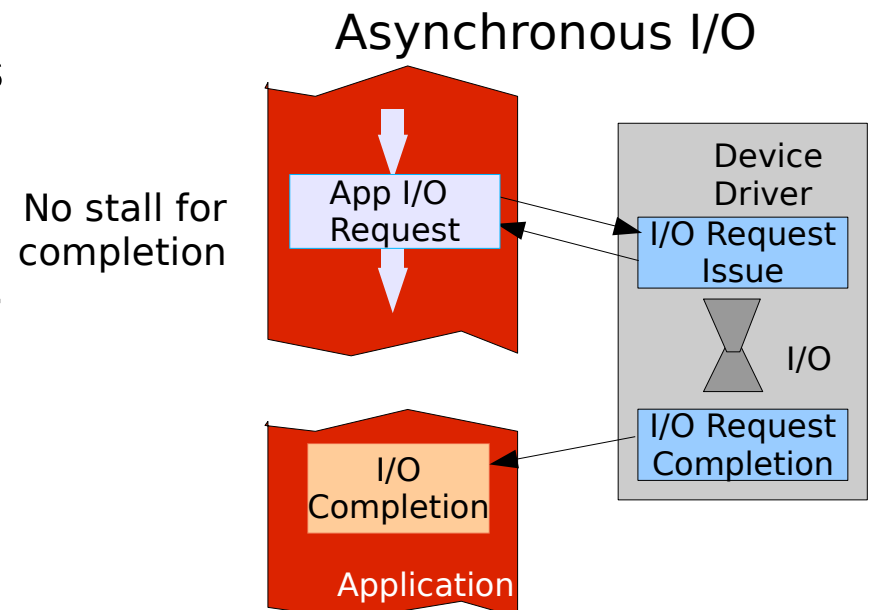
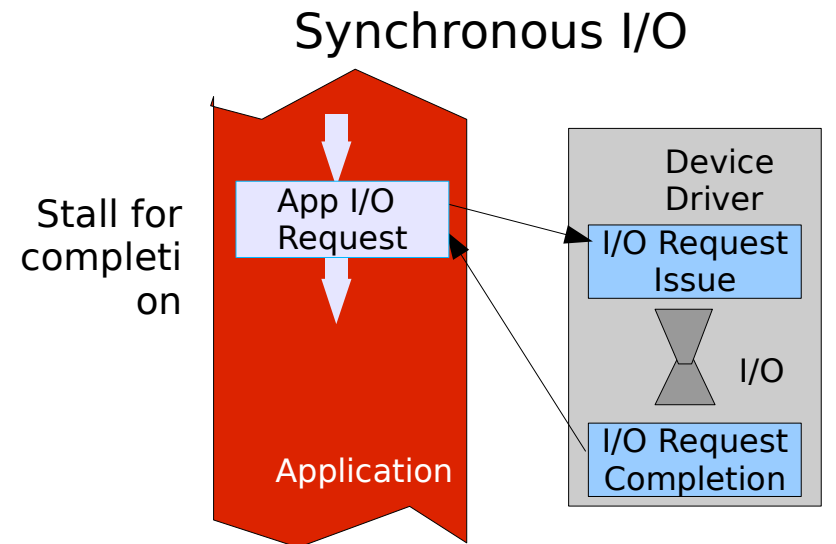
Support for RAW devices only

With Red Hat Enterprise Linux 4, significant improvement:

Support for Ext3, NFS, GFS file system access

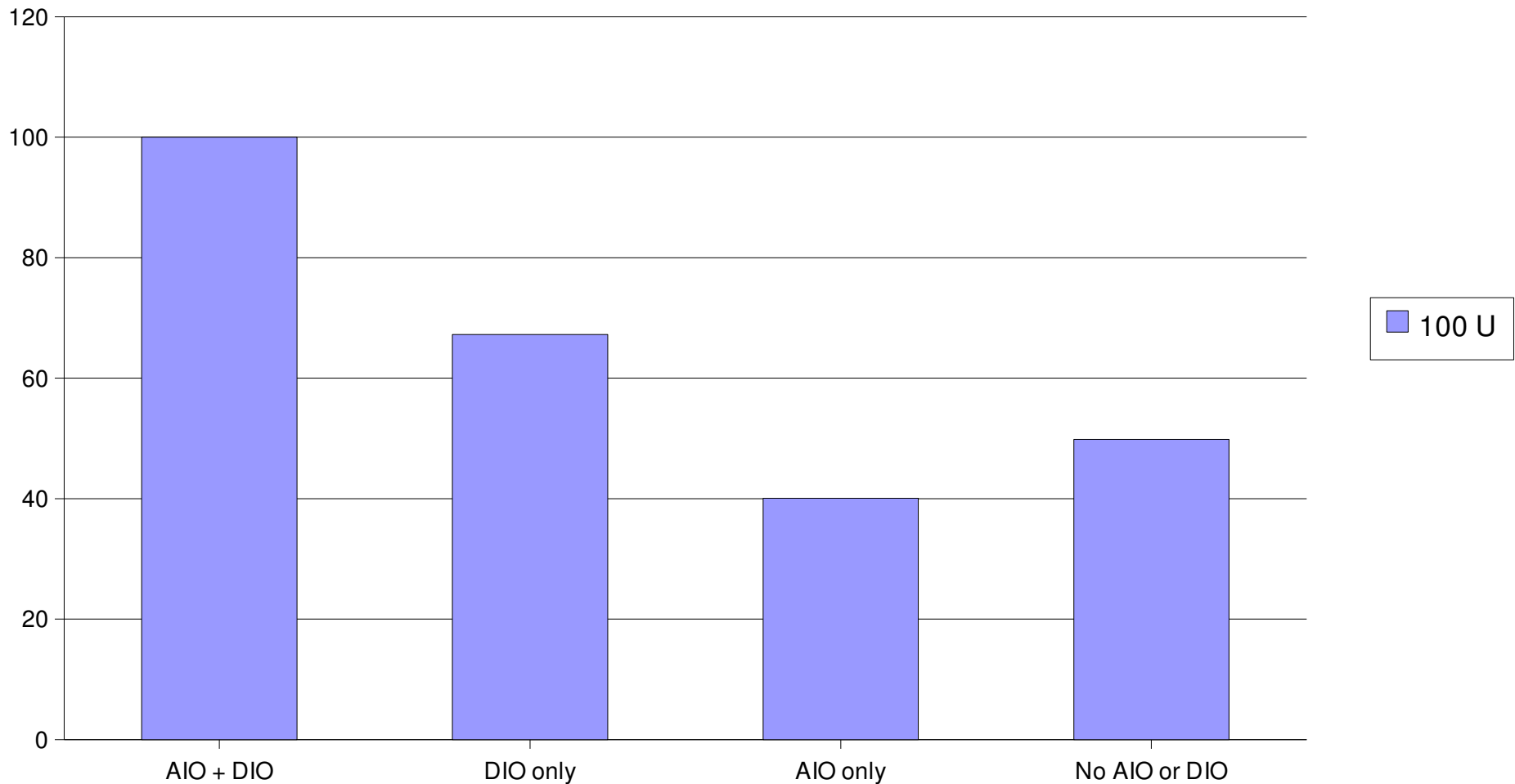
Supports Direct I/O (e.g. Database applications)

Makes benchmark results more appropriate for real-world comparisons



RHEL5.1 Oracle 10.2 - I/O options

RHEL5.1 with Oracle 10.2 - I/O Options



Disk IO tuning - RHEL4/5

RHEL4/5 – 4 tunable I/O Schedulers

CFQ – elevator=cfq. Completely Fair Queuing default, balanced, fair for multiple luns, adaptors, smp servers

NOOP – elevator=noop. No-operation in kernel, simple, low cpu overhead, leave opt to ramdisk, raid cntrl etc.

Deadline – elevator=deadline. Optimize for run-time-like behavior, low latency per IO, balance issues with large IO luns/controllers (NOTE: current best for FC5)

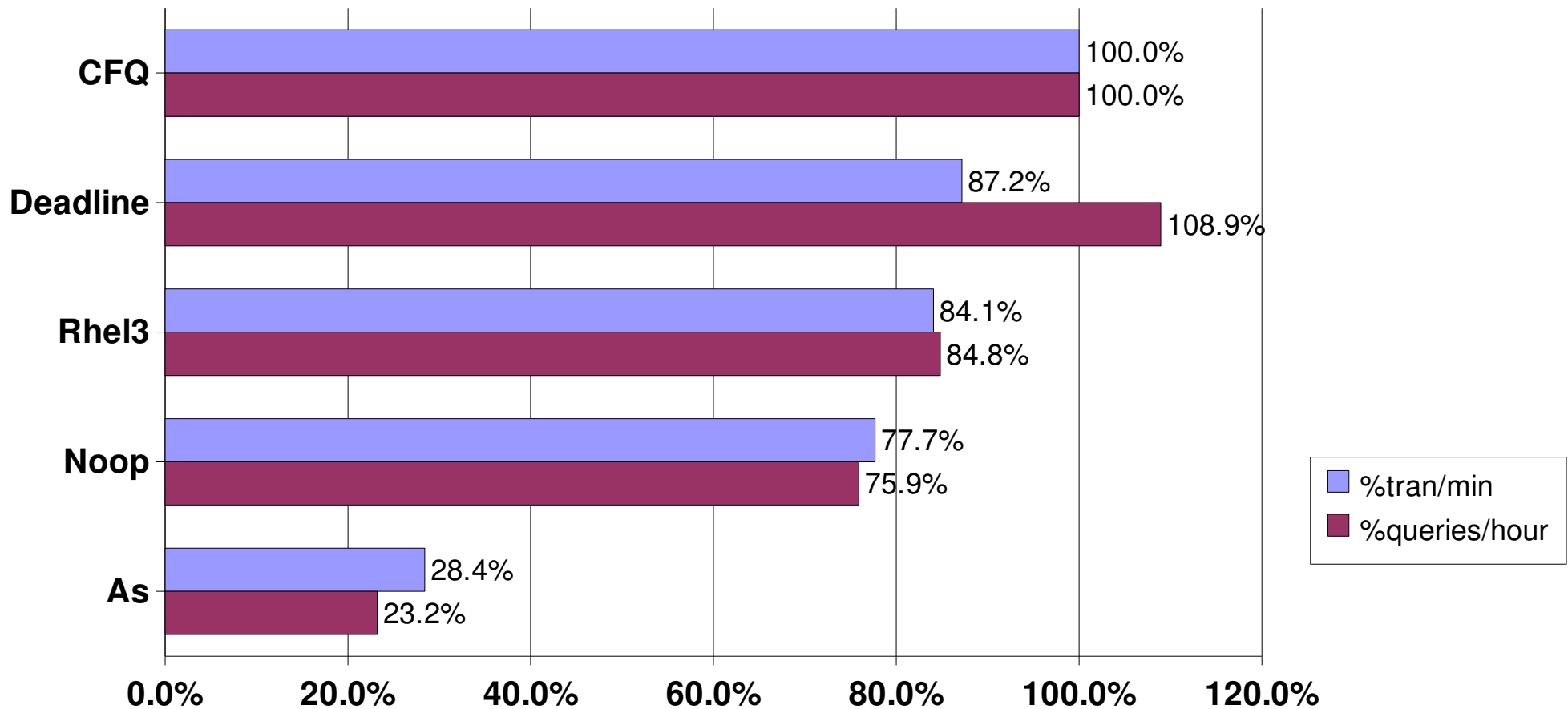
Anticipatory – elevator=as. Inserts delays to help stack aggregate IO, best on system w/ limited physical IO – SATA

RHEL4 - Set at boot time on command line

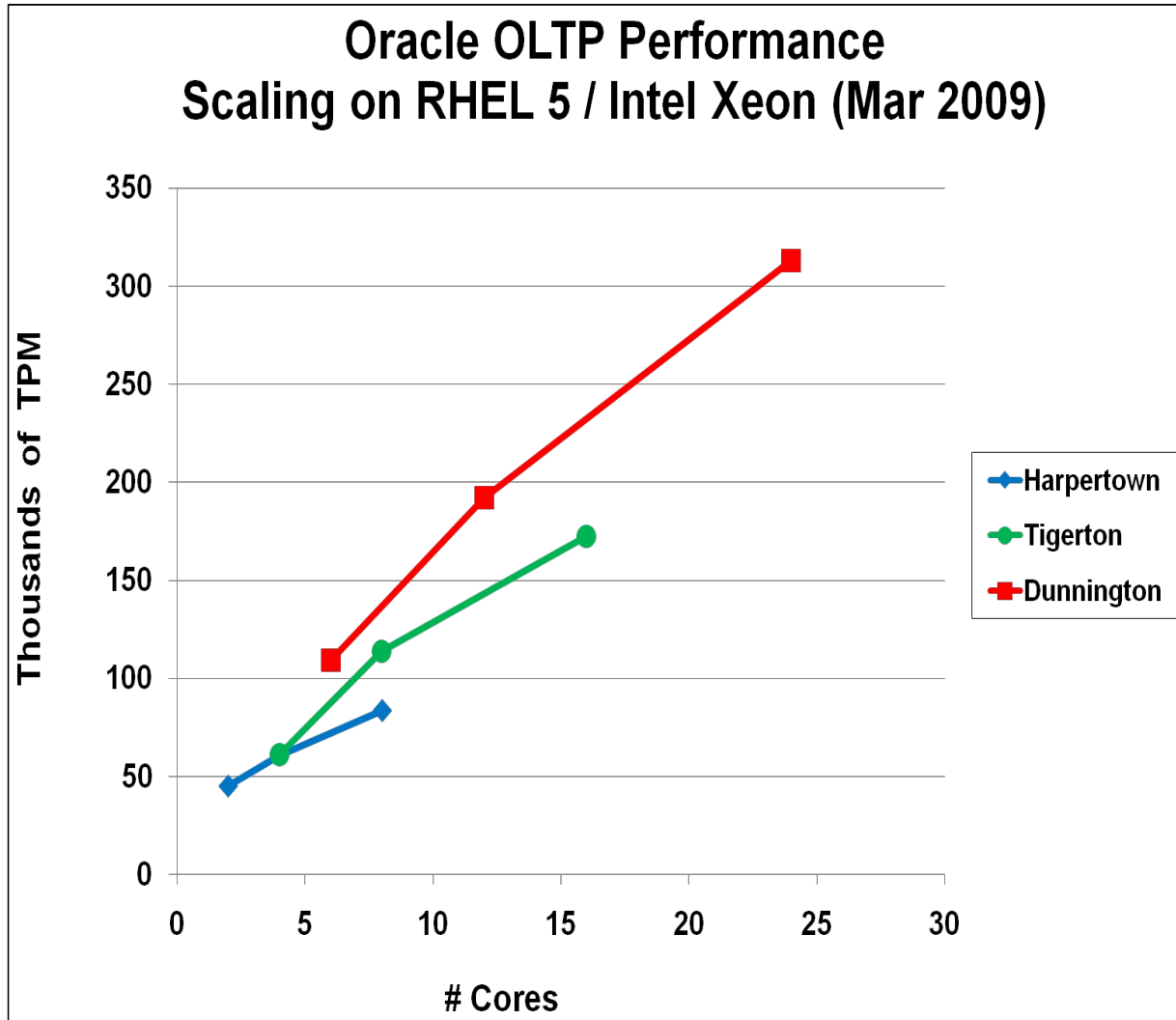
RHEL5 – Change on the fly

echo deadline > /sys/block/<sdX>/queue/scheduler

RHEL5 IO schedules vs RHEL3 for Database Oracle 10G oltp/dss (relative performance)

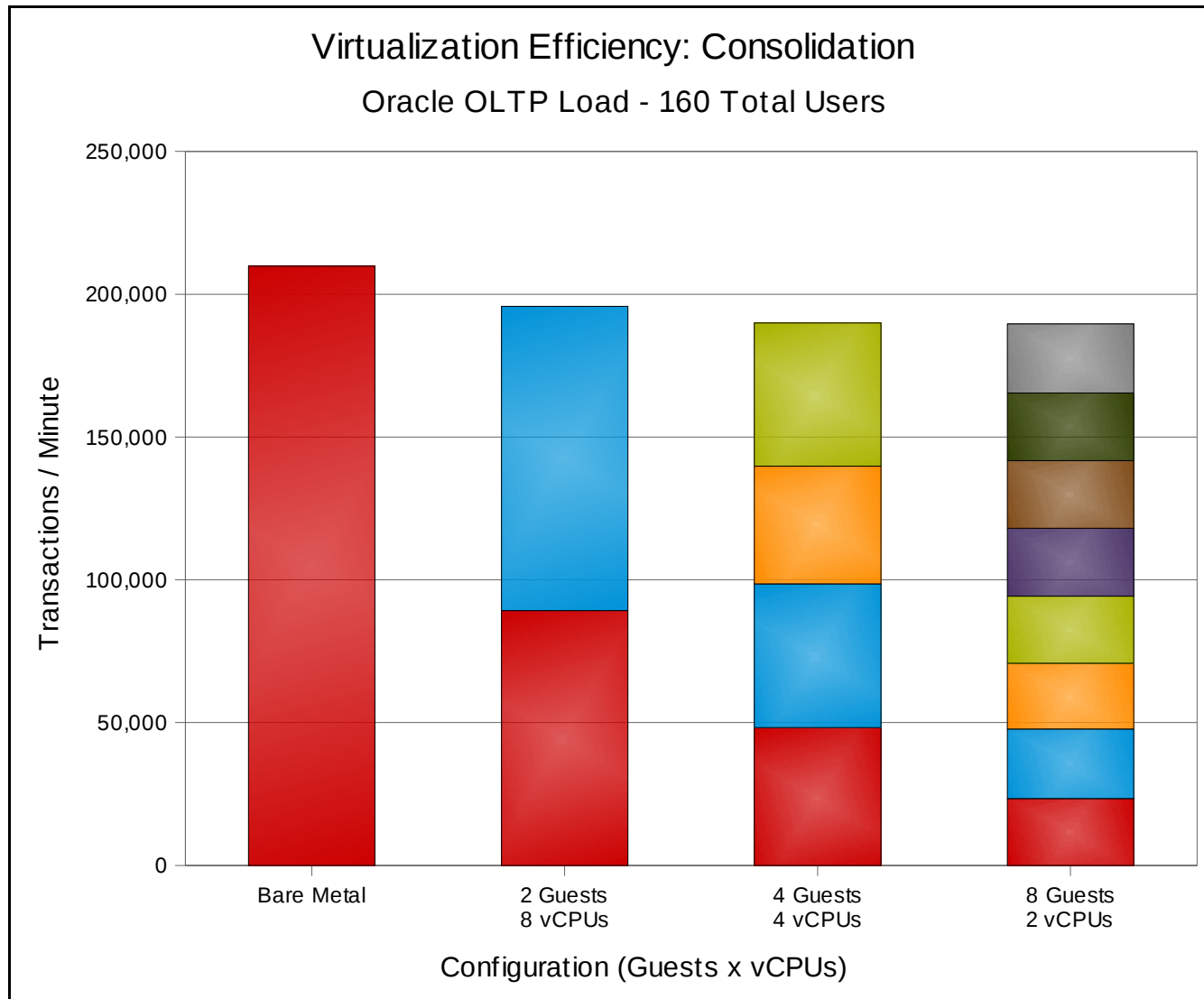


Oracle 10g Performance Scaling on RHEL5



- Oracle OLTP performance on RHEL scales well to 24 cores.
- See Reference Architecture talk on Benchmark Papers/Results
- **Testing on larger servers with the most recent x86_64 technology is anticipated in the coming year.**

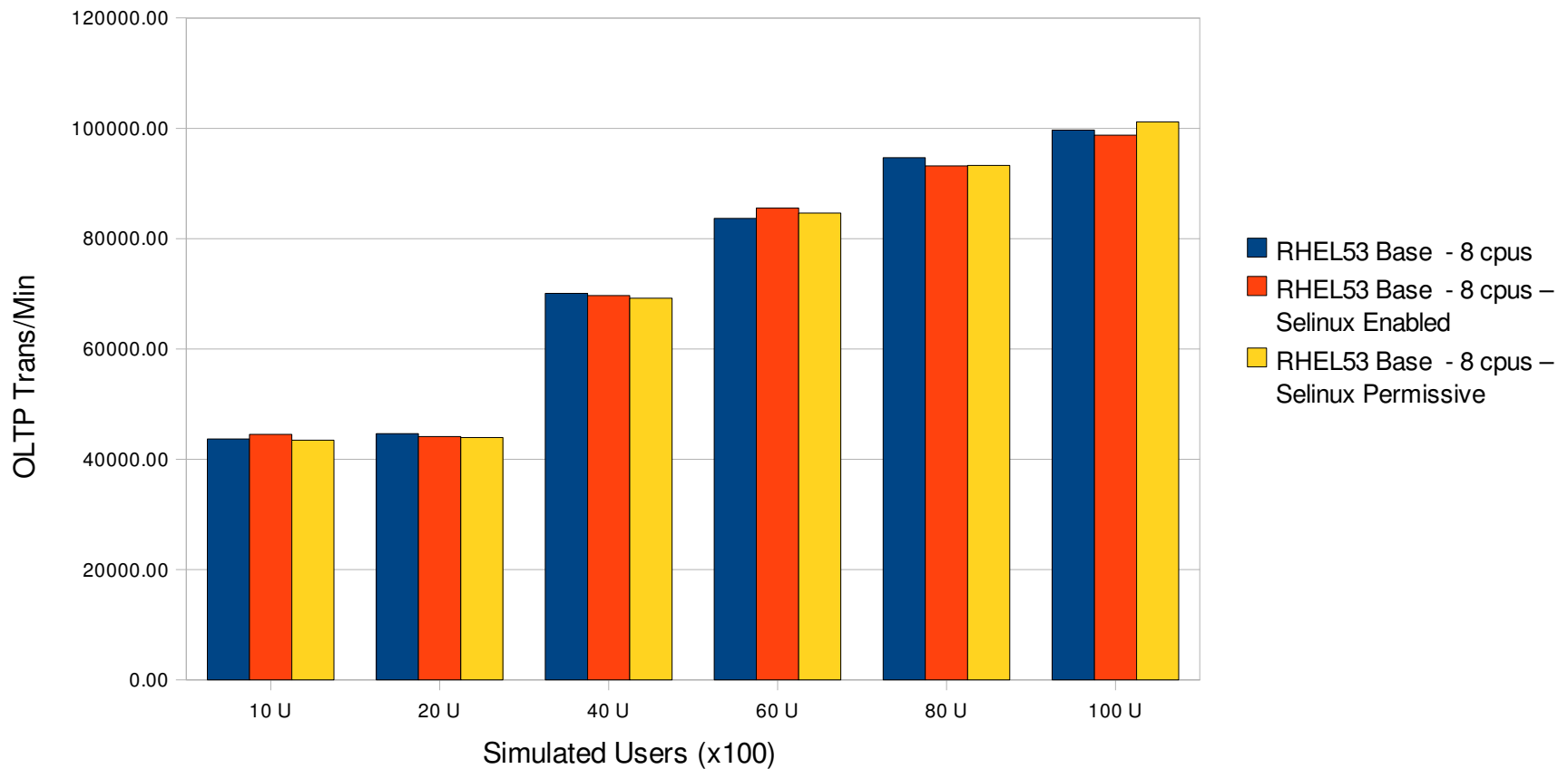
Oracle 10g: RHEL5.4 KVM Virtualization Efficiency



SELinux w/ Oracle OLTP Performance (steady state performance <5%)

RHEL5.2 Oracle 10GR4 w/ SELinux

Intel 8-cpu, 16GB, 2 FC, OLTP Perf



Benchmark Tuning

Use Hugepages.

Dont overcommit memory

If memory must be over committed

Eliminate all swapping.

Maximize pagecache reclaiming

Place swap partition(s) on separate device(s).

Use Direct IO

Dont turn NUMA off.

Network Performance Tuning Outline

IRQ Affinity / Processor Affinity - No magic formula

experiment to get the best results

Interrupt coalescing

Chip architectures play a big role

Try to match TX and RX on same socket / data caches

sysctl.conf

Increase/decrease memory parameter for **network**

Driver Setting

NAPI – if driver supports

HW ring buffers

TSO, UFO, GSO

sysctl

sysctl is a mechanism to view and control the entries under the `/proc/sys` tree

`sysctl -a` - lists all variables

`sysctl -q` - queries a variable

`sysctl -w` - writes a variable

When setting values, spaces are not allowed

```
sysctl -w net.ipv4.conf.lo.arp_filter=0
```

Look at documentation in `/usr/src`

`/usr/src/redhat/BUILD/kernel-2.6.18/linux-2.6.18.x86_64/Documentation/networking`

By default, Linux networking not tuned for max performance, more for reliability

Buffers are especially not tuned for local 10GbE traffic

Remember that Linux “autotunes” buffers for connections

Some Important settings for sysctl

net.ipv4.tcp_window_scaling - toggles window scaling

Misc TCP protocol

net.ipv4.tcp_timestamps - toggles TCP timestamp support

net.ipv4.tcp_sack - toggles SACK (Selective ACK) support

TCP Memory Allocations - min/pressure/max

net.ipv4.tcp_rmem - TCP read buffer - in bytes

overridden by core.rmem_max

net.ipv4.tcp_wmem - TCP write buffer - in bytes

overridden by core/wmem_max

net.ipv4.tcp_mem - TCP buffer space

measured in pages, not bytes !

Some Important settings for sysctl

CORE memory settings

net.core.rmem_max - max size of rx socket buffer

net.core.wmem_max -max size of tx socket buffer

net.core.rmem_default - default rx size of socket buffer

net.core.wmem_default - default tx size of socket buffer

net.core.optmem_max - maximum amount of option memory buffers

net.core.netdev_max_backlog how many unprocessed rx packets
before kernel starts to drop them

These settings also impact UDP

Control your network route :

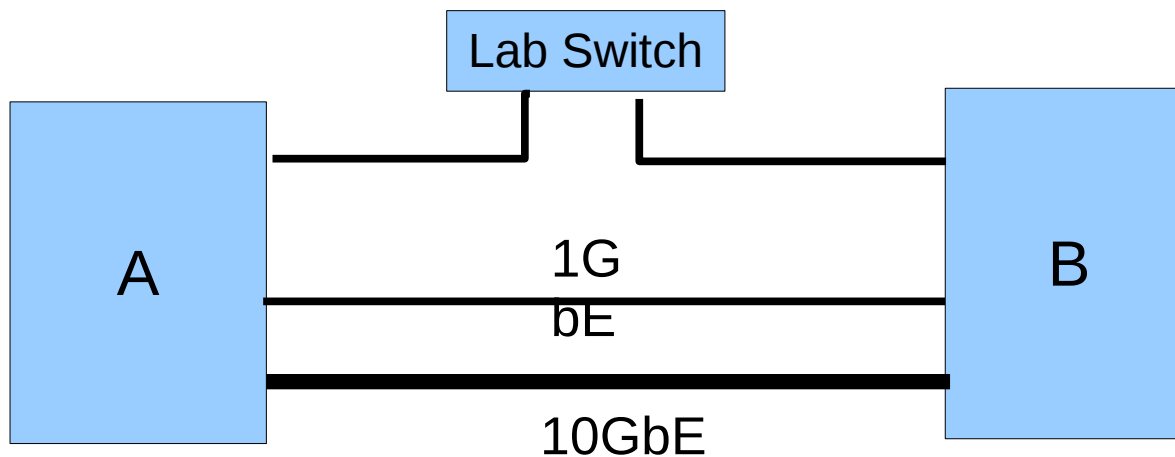
Check arp_filter settings with sysctl

```
sysctl -a | grep arp_filter
```

A setting of 0 says uses any path

If more than one path between machines, set arp_filter=1

Look for increasing interrupt counts in /proc/interrupt or increasing counters via ifconfig or netstat



netperf

<http://netperf.org>

Feature Rich

Read documentation

Default test is TCP_STREAM – uses *send()* call

TCP_SENDFILE – uses *sendfile()* call – much less copying

TCP_RR – Request / Response tests

UDP_STREAM

Many others

General Network/Disk I/O Tuning Guidelines

To maximize network throughput;

Disable irqbalance

```
service irqbalance stop  
chkconfig irqbalance off
```

Disable cpuspeed

default gov=ondemand, set governor to performance

Use affinity to maximize multi-core shared cache environments

Process affinity

Use taskset or MRG's "Tuna"

Interrupt affinity

```
grep eth2 /proc/interrupts  
echo 80 > /proc/irq/177/smp_affinity
```

Linux Performance Tuning References

Alikins,  System Tuning Info for Linux Servers,
http://people.redhat.com/alikins/system_tuning.html

Axboe, J.,  Deadline IO Scheduler Tunables, SuSE, EDF R&D, 2003.

Braswell, B, Ciliendo, E,  Tuning Red Hat Enterprise Linux on IBM eServer xSeries Servers, <http://www.ibm.com/redbooks>

Corbet, J.,  The Continuing Development of IO Scheduling ,
<http://lwn.net/Articles/21274>.

Ezolt, P, Optimizing Linux Performance, www.hp.com/hpbooks, Mar 2005.

Heger, D, Pratt, S,  Workload Dependent Performance Evaluation of the Linux 2.6 IO Schedulers , Linux Symposium, Ottawa, Canada, July 2004.

Red Hat Enterprise Linux “Performance Tuning Guide”

http://people.redhat.com/dshaks/rhel3_perf_tuning.pdf

Network, NFS Performance covered in separate talks

<http://nfs.sourceforge.net/nfs-howto/performance.html>

Questions?

