

# AIX Performance Tuning

Jaqui Lynch  
Mainline Information Systems  
jaqui.lynch@mainline.com  
Webcast June 03 2004



1

## Agenda

- CPU
- Network
- I/O
- Memory
  
- All recommendations are starting points and need to be tested before putting into production on your systems



2

## CPU Problems

- Use vmstat, ps, nmon
- Is it system, user or wait time
- How are the run queue and block queue?
- To fix:
  - Profiling
  - Optimize compiles
  - Change priorities
  - Tweak the scheduler (schedo – was schedtune)
  - CPU upgrade



## CPU Time

- Real
  - Wallclock time
- User State
  - Time running a users program
  - Includes library calls
  - Affected by optimization and inefficient code
- System
  - Time spent in system state on behalf of user
  - Kernel calls and all I/O routines
  - Affected by blocking I/O transfers
- I/O and Network
  - Time spent moving data and servicing I/O requests



# vmstat (AIX)

**IGNORE FIRST LINE - average since boot**  
Run vmstat over an interval (i.e. vmstat 2 30)

Kthr		memory		Page				Faults				CPU				
r	b	avm	fre	re	pi	po	fr	sr	cy	in	sy	cs	us	sy	id	wa
2	3	253292	5	0	1	262	1783	2949	0	941	13281	1877	12	29	0	60
2	3	255503	0	0	0	260	1110	1879	0	911	14725	1049	22	25	0	53
r	b	avm	fre	re	pi	po	fr	sr	cy	in	sy	cs	us	sy	id	wa
1	2	273017	487	0	0	0	0	0	0	507	12627	422	12	20	65	3
1	5	309809	465	0	2	580	1001	2033	0	1174	83870	1203	12	32	1	55
1	3	310054	321	0	0	76	55	83	0	665	98714	619	20	30	1	48

fre should be between minfree and maxfree

fr:sr ratio 1783:2949 means that for every 1783 pages freed 2949 pages had to be examined.

To get a 60 second average try: vmstat 60 2



5

# Other Tools

- nmon
  - [http://www-106.ibm.com/developerworks/eserver/articles/analyze\\_aix/](http://www-106.ibm.com/developerworks/eserver/articles/analyze_aix/)
- nmon analyzer
  - [http://www-106.ibm.com/developerworks/eserver/articles/nmon\\_analyser/](http://www-106.ibm.com/developerworks/eserver/articles/nmon_analyser/)
- sar
  - sar -A -o filename 2 30 >/dev/null
- ps
  - ps gv | head -n 1 >filename
  - ps gv | egrep -v "RSS" | sort +6b -7 -n -r >>filename
  - ps -ef
  - ps aux



6

## Network

- Buffers
  - Mbufs
    - Kernel buffer using pinned memory
    - thewall is max memory for mbufs
  - TCP and UDP receive and send buffers
  - Ethernet adapter attributes
  - no and nfso commands
  - nfsstat
  - rfc1323 and nfs\_rfc1323



## netstat

- netstat -i
  - Shows input and output packets and errors for each adapter
  - Also shows collisions
- netstat -ss
  - Shows summary info such as udp packets dropped due to no socket
- netstat -m
  - Memory information
- netstat -v
  - Statistical information on all adapters



## Network tuneables

- Using no
  - rfc1323 = 1
  - sb\_max=1310720
  - tcp\_sendspace=262144
  - tcp\_recvspace=262144
  - udp\_sendspace=65536
  - udp\_recvspace=655360
  - thewall=131072
- Using nfsd
  - Nfs\_rfc1323=1
  - nfs\_socketsize=60000
- Do a web search on “nagle effect”



## nfsstat

- Client and Server NFS Info
- nfsstat -cn or -r or -s
  - Retransmissions due to errors
    - Retrans>5% is bad
  - Badcalls
  - Timeouts
  - Waits
  - Reads



## Memory and I/O problems

- iostat
  - Look for overloaded disks and adapters
- vmstat
- vmo and ioo (replace vmtune)
- sar
- fileplace and filemon
- Asynchronous I/O
- Paging
- svmon
  - svmon -G >filename
- Nmon
- Check error logs



## iostat (AIX)

Run iostat ver an interval (i.e. iostat 2 30)

tty:	tin	tout	avg cpu:	%user	%sys	%idle	%iowait
(s1)	17.5	2.9		0.0	75.4	2.9	23.4
(s2)	0	0		9	8	24	59

Disks:	%tm_act	Kbps	tps	Kb_read	Kb_wrtn
hdisk1	3.5	32	3.5	0	64
hdisk0	59	27.6	59.5	44	508
hdisk2	12	64	16	0	128
hdisk3	3.5	32	3.5	0	64

%user + %sys <= 80%

Similar info from ps au or sar -u

%iowait – time waiting on disk I/O – includes local and nfs

Historical disk I/O collection is off by default on AIX 5.1

%tmact time physical disk was active - >40% means processes probably waiting

Adapter throughput – add up the kbps per drive on each adapter

Look for hot disks



## I/O Tuneables 1/2

- **minperm**
  - default is 20%
- **maxperm**
  - default is 80%
- **numperm**
  - This is what percent of real memory is currently being used for caching files - if it is high reduce maxperm to 30 to 50%
- **strict\_maxperm**
  - Used to avoid double caching – be extra careful!!!!
- Reducing maxperm stops file caching affecting programs that are running
- **maxrandwrt** is random write behind
  - default is 0 – try 32
- **numclust** is sequential write behind



## I/O Tuneables 2/2

- **minpgahead** and **maxpgahead**
  - Default min = 2 max = 8
  - Maxfree – minfree >= maxpgahead
- **lvm\_bufcnt**
  - Buffers for raw I/O. Default is 9
  - Increase if doing large raw I/Os (no jfs)
- **numfsbufs**
  - Helps write performance for large write sizes
- **hd\_pbuf\_cnt**
  - Pinned buffers to hold JFS I/O requests
  - Increase if large sequential I/Os to stop I/Os bottlenecking at the LVM
  - One pbuf is used per sequential I/O request regardless of the number of pages
- **sync\_release\_ilock**



## rc.tune 1/2

```
#
# rc.tune is called by inittab to tune the network and system parameters
#
# To make no changes permanent use the no -p option
/usr/sbin/no -rp -o rfc1323=1
/usr/sbin/no -rp -o sb_max=1310720
/usr/sbin/no -rp -o tcp_sendspace=262144
/usr/sbin/no -rp -o tcp_recvspace=262144
/usr/sbin/no -rp -o udp_sendspace=65536
/usr/sbin/no -rp -o udp_recvspace=655360
/usr/sbin/no -rp -o thewall=131072
#
/usr/sbin/nfso -p -o nfs_rfc1323=1
```



## rc.tune 2/2

```
#
# If this works well for you try setting maxperm to 30 instead of 50
# /usr/samples/kernel/vmtune -p 10 -P 50 -W 32
# vmtune is being phased out and replaced by ioo and vmo
#
/usr/sbin/vmo -p -o minperm%=10
/usr/sbin/vmo -p -o maxclient%=50
/usr/sbin/vmo -p -o maxperm%=50
/usr/sbin/ioo -p -o maxrandwrt=32
/usr/sbin/ioo -p -o sync_release_ilock=1
#
# If this works well for you try setting maxperm to 30 instead of 50
```



## ioo Output

```
minpgahead = 2
maxpgahead = 8
pd_npages = 65536
maxrandwrt = 0
numclust = 1
numfsbufs = 186
sync_release_ilock = 0
lvm_bufcnt = 9
j2_minPageReadAhead = 2
j2_maxPageReadAhead = 8
j2_nBufferPerPagerDevice = 512
j2_nPagesPerWriteBehindCluster = 32
j2_maxRandomWrite = 0
j2_nRandomCluster = 0
hd_pvs_opn = 1
hd_pbuf_cnt = 320
```

Info on ioo is at:

[http://publib16.boulder.ibm.com/pseries/en\\_US/cmds/aixcmds3/ioo.htm](http://publib16.boulder.ibm.com/pseries/en_US/cmds/aixcmds3/ioo.htm)



17



## vmo Output

```
memory_frames = 65536
maxfree = 128
minfree = 120
minperm% = 20
minperm = 11813
maxperm% = 80
maxperm = 47255
strict_maxperm = 0
maxpin% = 80
maxpin = 52429
maxclient% = 80
```

```
lrubucket = 131072
defps = 1
nokilluid = 0
numpsblks = 131072
npskill = 1024
npswarn = 4096
v_pinshm = 0
pta_balance_threshold = 50
pagecoloring = 0
framesets = 0
mempools = 0
lgpg_size = 0
lgpg_regions = 0
num_spec_dataseg = n/a
spec_dataseg_int = n/a
memory_affinity = n/a
```

Info on ioo and vmo can be found at:

[http://www-106.ibm.com/developerworks/eserver/articles/Keung\\_AIXPerf.html](http://www-106.ibm.com/developerworks/eserver/articles/Keung_AIXPerf.html)



18



## vmtune 1/2

vmtune: current values:

-p	-P	-r	-R	-f	-F	-N	-W	
minperm	maxperm	minpgahead	maxpgahead	minfree	maxfree	pd_npages	maxrandwrt	
1206105	6030528	2	64	360	513	65536	0	
-M	-w	-k	-c	-b	-B	-u	-l	-d
maxpin	npswarn	npskill	numclust	numfsbufs	hd_pbuf_cnt	lvm_bufcnt	lrubucket	defps
10066323	393216	98304	1	512	7168	9	131072	1
-s	-n	-S	-L	-g	-h			
sync_release_ilock	nokilluid	v_pinshm	lgpg_regions	lgpg_size	strict_maxperm			
0	0	0	0	0	0			



19

## vmtune 2/2

-t	-j	-J	-z
maxclient	j2_nPagesPer	j2_maxRandomWrite	j2_nRandomCluster
2412210	32	0	0
-Z	-q	-Q	-y
j2_nBufferPer	j2_minPageReadAhead	j2_maxPageReadAhead	memory_affinity
512	2	8	0
-V	-i	-e	-E
num_spec_dataseg	spec_dataseg_int	jfs_cread_enabled	jfs_use_read_lock
0	512	0	0

PTA balance threshold percentage = 50.0%

number of valid memory pages = 12582903	<b>maxperm=50.0%</b> of real memory
maximum pinable=80.0% of real memory	<b>minperm=10.0%</b> of real memory
number of file memory pages = 6030204	<b>numperm=76.0%</b> of real memory
number of compressed memory pages = 0	compressed=0.0% of real memory
number of client memory pages = 163	numclient=0.0% of real memory
# of remote pgs sched-pageout = 0	maxclient=20.0% of real memory



20

## JFS Logs

- Max size = 256mb
- Default is 1 x PP
- All filesystems in a VG use the samelog by default
- If sync I/O or many files created and deleted then this file will be hot
- `mklv -t jfslog -y JFSLOGdb2 testvg 2 hdisk4`
- `logform /dev/JFSLOGdb2`
- `chfs -a log=/dev/JFSLOGdb2 /filesystem`
- Umount and mount filesystem



## Isps -a (similar to pstat)

- Ensure all page datasets the same size although hd6 can be bigger - ensure more page space than memory
- Only includes pages allocated (default)
- Use `Isps -s` to get all pages (includes reserved via early allocation (PSALLOC=early))
- Use multiple page datasets on multiple disks



# Isps output

```
Isps -a
Page Space Physical Volume Volume Group Size %Used Active Auto Type
paging01   hdisk3         pagvg01   2072MB 1  yes  yes  lv
paging02   hdisk4         vgpaging01 504MB 1  yes  yes  lv
paging03   hdisk5         vgpaging02 168MB 1  yes  yes  lv
paging01   hdisk6         vgpagine03 168MB 1  yes  yes  lv
paging00   hdisk2         vgpaging04 168MB 1  yes  yes  lv
hd6        hdisk0         rootvg    512MB 1  yes  yes  lv
```

```
Isps -s
Total Paging Space  Percent Used
3592MB              1%
```

Bad Layout above  
Should be balanced  
Make hd6 the biggest by one lp or the same size as the others



# Intra and Inter Policies

- Intra Policy
  - How data is laid out on the disk
  - Outer edge, outer middle, center, inner middle, inner edge
  - If  $\leq 4$ gb disk then center is fastest seek
  - If  $> 4$ gb then outer edge
- Inter Policy
  - How data is laid out between/across disks
  - Minimum as few disks as possible
  - Maximum as many disks as possible



# I/O Pacing

- Set high value to multiple of  $(4*n)+1$
- Limits the number of outstanding I/Os against an individual file
- minpout – minimum
- maxpout – maximum
- If process reaches maxpout then it is suspended from creating I/O until outstanding requests reach minpout



# Other I/O Notes

- Mirroring of disks
  - Islv to check # copies
  - Mirror write consistency
- Mapping of backend disks
  - Don't software mirror if already RAIDed
  - RIO and adapter Limits
- Logical volume scheduling policy
  - Parallel
  - Parallel/sequential
  - Parallel/roundrobin
  - Sequential
- Async I/O
  - Oracle loves async I/O (minserver & maxserver)
  - DB/2 uses its own I/O cleaners if you turn them on



## Other tools

- filemon
  - filemon -v -o filename -O all
  - sleep 30
  - trcstop
- pstat to check async I/O
  - pstat -a | grep aio | wc -l
- perfpmr to build performance info for IBM
  - /usr/bin/perfpmr.sh 300



## Striping

- Spread data across drives
- Improves r/w performance of large sequential files
- Can mirror stripe sets
- Stripe width = # of stripes
- Stripe size
  - Set to 64kb for best sequential I/O throughput
  - Any  $N^2$  from 4kb to 128kb



## General Recommendations

- Different hot LVs on separate physical volumes
- Stripe hot LVs across disks
- Mirror read intensive data
- Ensure LVs are contiguous
  - Use lslv and look at in-band % and distrib
  - reorgvg if needed to reorg LVs
- writeverify=no
- minpgahead=2, maxpgahead=16 for 64kb stripe size
- Increase maxfree if you adjust maxpgahead
- Tweak minperm, maxperm and maxrandwrt
- Tweak lvm\_bufcnt if doing a lot of large raw I/Os



## Summary



# Adapter Throughput SCSI

Courtesy of <http://www.scsita.org/terms/scsiterms.html>

	100% mby/s	70% mby/s	Bits Bus	Max Devs Width
• SCSI-1	5	3.5	8	8
• Fast SCSI	10	7	8	8
• FW SCSI	20	14	16	16
• Ultra SCSI	20	14	8	8
• Wide Ultra SCSI	40	28	16	8
• Ultra2 SCSI	40	28	8	8
• Wide Ultra2 SCSI	80	56	16	16
• Ultra3 SCSI	160	112	16	16
• Ultra320 SCSI	320	224	16	16
• Ultra640 SCSI	640	448	16	16

- Watch for saturated adapters



31

# Adapter Throughput Fibre

	100% mbit/s	70% mbit/s
• 133		93
• 266		186
• 530		371
• 1 gbit		717
• 2 gbit		1434

- SSA comes in 80 and 160 mb/sec



32

## RAID Levels

- Raid-0
  - Disks combined into single volume stripeset
  - Data striped across the disks
- Raid-1
  - Every disk mirrored to another
  - Full redundancy of data but needs extra disks
  - At least 2 I/Os per random write
- Raid-0+1
  - Striped mirroring
  - Combines redundancy and performance



## RAID Levels

- RAID-5
  - Data striped across a set of disks
  - 1 more disk used for parity bits
  - Parity may be striped across the disks also
  - At least 4 I/Os per random write (read/write to data and read/write to parity)
  - Uses hot spare technology

