Sun Performance Tuning Rules

SMCC Product Marketing Engineering

Excerpted from the book
Sun Performance And Tuning: SPARC and Solaris
by Adrian Cockcroft
Published by SunSoft Press / PTR Prentice Hall
ISBN 0-13-149642-3

Sun Microsystems Computer Corporation
2550 Garcia Avenue
Mountain View, CA 94043
U.S.A.

Revision A, March 1995

Rules and Tunables Quick Reference Tables

A

This appendix summarizes some of the advice given in the book into rules and lists the kernel variables that are tunable. The data is tabulated with references to the book for more information and explanation.

If you have one of the many performance management tools that lets you set thresholds or rules, then you can use these tables to set up your tool.

Notation Used in Tables

Rules

To refer to a measured result, I use the name of the command with any options, then a "." followed by the variable name identification. Spaces are removed whenever the resulting name is unambiguous. For example, to indicate disk service time using `iostat -x` with a 30-second interval, I would specify `iostat-x30.svc_t`.

Variables are combined using the usual C language operators with "&&" for logical AND, "||" for logical OR, and "==" to test for equality. Temporary variables and multistage rules may be used. A range test expression, for example, 0 <= X < 100 is used for brevity.

Levels

The level entry in the table indicates how serious the situation is. The color coding used is white for low usage, blue for underutilization or imbalance of an expensive resource, green for target utilization levels or no problem, amber for a warning level, red for a critical level that needs to be fixed for good performance, and black for problems that can prevent your application or system from running at all.

Actions

The action to be taken when the rule is true is explained below the tables in a numbered paragraph. The action field contains a terse reminder of the problem and the number of the relevant statement where necessary.

Disk I/O Rules

Table A-1 Disk I/O Performance Rules for SunOS 4

Rule for Each Disk Drive	Level	Action
(`iostat-D30.util` < 5%) && (other disks white or green)	White	No Problem
(`iostat-D30.util` < 5%) && (other disks amber or red)	Blue	1. Idle Disk
5% <= `iostat-D30.util` < 35%	Green	No Problem
35% <= `iostat-D30.util` < 65%	Amber	2. Busy Disk
65% <= `iostat-D30.util`	Red	2. Busy Disk

Table A-2 Disk I/O Performance Rules for Solaris 2

Rule for Each Disk Drive	Level	Action
(`iostat-x30.b` < 5%) && (other disks white or green)	White	No Problem
(`iostat-x30.b` < 5%) && (other disks amber or red)	Blue	1. Idle Disk
(5% <= `iostat-x30.b`) && (`iostat-x30.svc_t` < 30 ms)	Green	No Problem
(20% <= `iostat-x30.b`) && (30 ms <= `iostat-x30.svc_t` < 50 ms)	Amber	2. Busy Disk
(20% <= `iostat-x30.b`) && (50 ms <= `iostat-x30.svc_t`)	Red	2. Busy Disk
(20% <= `iostat-x30.b`) && (50 ms <= `iostat-x30.svc_t`) && (`iostat-x30.disk` = = “fd0” || `iostat-x30.disk` = = “sd6”)	Amber	3. Floppy/CD
0% = = `iostat-x30.w`	Green	No Problem
0% < `iostat-x30.w` < 5%	Amber	4. SCSI Busy
5% <= `iostat-x30.w`	Red	4. SCSI Busy

1. **Idle Disk**
   An idle disk is a waste of I/O throughput when other disks are overloaded. Rebalance the load or stripe this disk together with a busy one.

2. **Busy Disk**
   A busy or slow disk reduces system throughput and increases user response times. Rebalance the load or stripe this disk together with an idle one. If there are no idle disks then reduce the I/O load somehow or configure more disks. See “Load Monitoring and Balancing” on page 75.

3. **Floppy/CD**
   To prevent the rules from being applied to much slower devices, check for the disk type of `fd0`, indicating a floppy. A CD appears as a normal but very slow disk that has never been written to, and there is no simple way to exclude it without accidentally excluding a very overloaded, read-only hard disk. The CD is often configured as target number 6 on the first SCSI bus, so specifying `sd6` is a crude

workaround for this problem. If the floppy or CD are active for long periods, then something somewhere is running slowly, and the data should be moved to a faster disk.

4. **SCSI Busy**
   If disk commands have to wait before they can be sent to the disk, then the waiting may be due to an overloaded SCSI bus. There are no direct measures of SCSI bus utilization levels.

Network Rules

Table A-3 Network Performance Rules Based on Ethernet Collisions

Rule or Each Network Interface	Level	Action
(0 < `netstat-i30.output.packets` < 10) && (100 * `netstat-i30.output.colls / netstat-i30.output.packets` < 0.5%) `&& (other nets white or green)`	White	No Problem
(`0 < netstat-i30.output.packets` < 10) && (100 `* netstat-i30.output.colls / netstat-i30.output.packets` < 0.5%) `&& (other nets amber or red)`	Blue	1. Inactive Net
(10 <= `netstat-i30.output.packets`) && (0.5% <= 100 * `netstat-i30.output.colls / netstat-i30.output.packets` < 2.0%)	Green	No Problem
(10 <= `netstat-i30.output.packets`) && (2.0% <= 100 * `netstat-i30.output.colls / netstat-i30.output.packets` < 5.0%)	Amber	2. Busy Net
(10 <= `netstat-i30.output.packets`) && (5.0% <= 100 * `netstat-i30.output.colls / netstat-i30.output.packets`)	Red	2. Busy Net
network type is not `ie`, `le`, `ne`, or `qe`; it is `bf` or `nf`.	Green	3. Not Ether

1. **Inactive net**
   An inactive network is a waste of throughput when other networks are overloaded. Rebalance the load so that all networks are used more evenly.

2. **Busy Net**
   A network with too many collisions reduces throughput and increases response time for users. Move some of the load to inactive networks if there are any. Add more Ethernets or upgrade to a faster interface type like FDDI, 100-Mbit Ethernet, or ATM.

3. **Not Ether**
   If the last letter of the interface name is not `e` then this is not an Ethernet, so the collision-based network performance rule should not be used.

NFS Client Rules

Table A-4 NFS Client Performance Rules

Rule for Each NFS Client	Level	Action
0 = = `nfsstat-rc.calls`	White	No Client NFS
`nfsstat-rc.timeout` < 0.05 * `nfsstat-rc.calls`	Green	No Problem
0.05 * `nfsstat-rc.calls` <= `nfsstat-rc.timeout &&` `nfsstat-rc.badxid` = = 0	Red	1. Bad Net
0.05 * `nfsstat-rc.calls` <= (`nfsstat-rc.timeout` = = `nfsstat-rc.badxid`)	Red	2. Slow NFS

1. **Bad Net**
   Packets are not making it to or from the NFS server. Fix the network hardware or reduce NFS packet sizes.

2. **Slow NFS**
   The NFS server is so slow that it is processing duplicate requests from the clients that were sent after time-outs. Increase client time-out.

Memory Rules

Table A-5 Virtual Memory Rules for SunOS 4

Virtual Memory Rule	Level	Action
100000k <= `pstat-s.available`	White	1. Swap Waste
10000k <= `pstat-s.available` < 100000k	Green	No Problem
4000k <= `pstat-s.available` < 10000k	Amber	2. Swap Low
1000k <= `pstat-s.available` < 4000k	Red	2. Swap Low
`pstat-s.available` < 1000k	Black	3. No Swap

Table A-6 Virtual Memory Rules For Solaris 2

Virtual Memory Rule	Level	Action
100000k <= `vmstat30.swap`	White	1. Swap Waste
10000k <= `vmstat30.swap` < 100000k	Green	No Problem
4000k <= `vmstat30.swap` < 10000k	Amber	2. Swap Low
1000k <= `vmstat30.swap` < 4000k	Red	2. Swap Low
`vmstat30.swap` < 1000k	Black	3. No Swap

1. **Swap Waste**
   You have a lot of unused swap space.
2. **Swap Low**
   There is not much swap space left, so the system may soon run out of virtual memory. Reduce the number and size of the programs running on the system or add more swap space before it runs out completely. See “Swap Space” on page 4.
3. **No Swap**
   The system has effectively run out of swap space. Programs will fail or hang. Add more swap space or shut down applications immediately.

Table A-7 Physical Memory Rules for SunOS 4 and Solaris 2

Physical Memory Rule	Level	Action
`vmstat30.sr` = = 0	White	1. RAM Waste
0 < `vmstat30.sr` < 200	Green	No Problem
200 <= `vmstat30.sr` < 300	Amber	2. RAM Low
300 <= `vmstat30.sr`	Red	2. RAM Low

1. **RAM Waste**
   You have more RAM than you need. The system does not even need to reclaim inactive pages.
2. **RAM Low**
   The system is scanning through memory looking for pages to free at a high rate. This indicates that, as well as inactive pages, active pages may be stolen from processes.

Table A-8 Kernel Memory Rules for Solaris 2

<table>
<tr><th>Kernel Memory Rule</th><th>Level</th><th>Action</th></tr>
<tr><td>((0 < sar-k1.sml_mem.fail) || (0 < sar-k1.lg_mem_fail) || (0 < sar-k1.ovsz.alloc.fail)) && (64 < sar-r1.freemem)</td><td>Black</td><td>1. No Kmap</td></tr>
<tr><td>((0 < sar-k1.sml_mem.fail) || (0 < sar-k1.lg_mem_fail) || (0 < sar-k1.ovsz.alloc.fail)) && (sar-r1.freemem <= 64)</td><td>Black</td><td>2. No Kmem</td></tr>
</table>

1. **No Kmap**
   There are available memory pages, but the kernel cannot allocate more memory. The *kernelmap* or kernel address space has been used up; see "Kernel Memory Allocation" on page 216. Several parts of the system may fail; a reboot is advised to restore the system to a known stable state. Reducing *maxusers* and limiting the buffer cache size may help. See "Basic Sizing with Maxusers" on page 186 and "Buffer Cache" on page 193.
2. **No Kmem**
   The system is short of memory; when the kernel tried to allocate pages for its own use there was no free RAM. This is likely to occur only with Solaris 2.4, which has a smaller free list than earlier releases do. On a high-end MP server that has a lot of minor faults and page-ins this problem can be avoided by setting *lotsfree* higher to increase the free list size. See the description of "lotsfree and deficit" on page 209.

CPU Rules

The number of CPUs in use on this system must be known. It is referred to as *ncpus* in the rules. For SunOS 4, the `/usr/kvm/mpstat` command shows you how many CPUs are on SPARCserver 600MP systems. For Solaris 2, the `psrinfo` command or the `sysinfo` library routine can be used. The run queue length is divided by the number of CPUs because every CPU takes a job off the run queue in each time slice.

Table A-9 CPU Rule for SunOS 4 And Solaris 2

CPU Rule	Level	Action
0 == `vmstat30.r`	White	1. CPU Idle
0 < (`vmstat30.r` / ncpus) < 3.0	Green	No Problem
3.0 <= (`vmstat30.r` / ncpus) <= 5	Amber	2. CPU Busy
5.0 <=(`vmstat30.r` / ncpus)	Red	2. CPU Busy
`mpstat30.smtx` < 200	Green	No Problem
200 <= `mpstat30.smtx` < 400	Amber	3. Mutex Stall
400 <= `mpstat30.smtx`	Red	3. Mutex Stall

1. **CPU Idle**
   The CPU power of this system is underutilized. Fewer or less powerful CPUs could be used to do this job.
2. **CPU Busy**
   There is insufficient CPU power, and jobs are spending an increasing amount of time in the queue before being assigned to a CPU. This reduces throughput and increases interactive response times.
3. **Mutex Stall**
   If the number of stalls per CPU per second exceeds the limit, mutex contention is happening in the kernel, which wastes CPU time and degrades multiprocessor scaling.

Kernel Tables Rules

Table A-10 Kernel Table Rules for SunOS 4

Kernel Table Rule	Level	Action
`vmstat-s.name.hits` < 90%	Red	1. DNLC Miss
90% <= `vmstat-s.name.hits` < 95%	Amber	1. DNLC Miss
95% <= `vmstat-s.name.hit` <= 100%	Green	No Problem

Table A-11 Kernel Table Rules for Solaris 2

Kernel Table Rule	Level	Action
`dnlc.hitrate` = 100 * (`sar-a30.namei - sar-a30.iget`) / `sar-a30.namei`	All	Define
(`dnlc.hitrate` < 90%) && (50 < `sar-a30.namei`)	Amber[1]	1. DNLC Miss
90% <= `dnlc.hitrate` <= 100% || (`sar-a30.namei` <= 50)	Green	No Problem
0.00% == `sar-g1.ufs_ipf`	Green	No Problem
0.00% < `sar-g1.ufs_ipf` <= 100.00%	Amber	2. Inode Flush

1. This condition is given an Amber level because it needs more research to verify that it is the right threshold and rule.

1. **DNLC Miss**
   The directory name lookup cache is too small and a lot of inodes are being read. See "Directory Name Lookup Cache" on page 189.

2. **Inode Flush**
   The inode cache limit is set too low, and file data is being flushed as inodes are being reused. Try doubling *ufs_ninode*. See "The Inode Cache and File Data Caching" on page 191.

Tunable Kernel Parameters

The kernel chapter of this book explains how the main kernel algorithms work. The kernel tunable values listed in this section include *the main tunables that are worth worrying about*. There are a huge number of global values defined in the kernel; if you hear of a tweak that is not listed in this book, think twice before using it. The algorithms, default values, and existence of many of these variables vary from one release to the next. Do not assume that an undocumented tweak that works well for one kernel will apply to other releases, other kernel architectures of the same release, or even a different patch level.

Table A-12 Primary Configuration Variables in Solaris 2.3 and Solaris 2.4

Name	Default	Min	Max	Reference
maxusers	MB available RAM (*physmem*)	8	2048	"Autoconfiguration of maxusers in Solaris 2.3 and Solaris 2.4" on page 188
pt_cnt	48	48	3000	"Changing maxusers and Pseudo-ttys in Solaris 2" on page 187

Table A-13 File Name and Attribute Cache Sizes for SunOS 4 and Solaris 2

Name	Default	Min	Max	Reference
ncsize	(*maxusers* * 17) + 90	226	34906	"Directory Name Lookup Cache" on page 189
ufs_ninode	(*maxusers* * 17) + 90	226	34906	"The Inode Cache and File Data Caching" on page 191

Table A-14 Hardware-specific Configuration Tunables

Name	Default	Min	Max	Reference
use_mxcc_prefetch	0 (sun4d) 1 (sun4m)	0	1	"The SuperSPARC with SuperCache Two-level Cache Architecture" on page 159.
npts	dynamic, based on memory	default, do not set lower	2 * default	"The SPARC Reference MMU Table Walk Operation" on page 168

Shared memory parameters are usually set based on the needs of specific application programs.

Table A-15 Shared Memory and Semaphore Tunables in Solaris 2

Name	Default	Min	Max	Reference
shmsys:shminfo_shmmax	1048576	1048576	Available RAM	Maximum shm segment size in bytes
shmsys:shminfo_shmmin	1	1		Minimum shm segment size in bytes
shmsys:shminfo_shmni	100	100		Number of shm identifiers to pre-allocate
shmsys:shminfo_shmseg	6	6		Maximum number of shm segments per process
semsys:seminfo_semmap	10	10		Number of entries in semaphore map
semsys:seminfo_semmni	10	10	65535	Number of semaphore identifiers
semsys:seminfo_semmns	60			Number of semaphores in system
semsys:seminfo_semmnu	30			Number of undo structures in system
semsys:seminfo_semmsl	25			Maximum number of semaphores per ID
semsys:seminfo_semopm	10			Maximum number of operations per semop call
semsys:seminfo_semume	10			Maximum number of undo entries per process
semsys:seminfo_semusz	96	-	-	Size in bytes of undo structure, derived from semume
semsys:seminfo_semvmx	32767			Semaphore maximum value
semsys:seminfo_semaem	16384			Adjust on exit maximum value

Sun Microsystems
Computer Corporation
A Sun Microsystems, Inc. Business

For U.S. Sales Office locations, call: 800 821-4643
In California: 800 821-4642

Australia: (02) 413 2666
Belgium: +32 2 759 38 11
Canada: 416 477-6745
Finland: +358-0-502 27 00
France: (1) 30 67 50 00
Germany: (0) 89-46 00 8-0

Hong Kong: 852 802 4188
Italy: 039 60551
Japan: (03) 3221-7021
Korea: 822-563-8700
Latin America: 415 688-9464
The Netherlands: 033 501234

New Zealand: (04) 499 2344
Nordic Countries: +46 (0) 8 623 90 00
PRC: 861-831-5568
Singapore: 224 3388
Spain: (91) 5551648
Switzerland: (01) 825 71 11

Taiwan: 2-514-0567
UK: 0276 20444
Elsewhere in the world, call
Corporate Headquarters:
415 960-1300
Intercontinental Sales:
415 688-9000

Printed in USA

March 1995