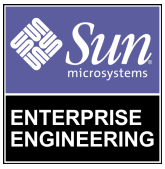
Using Solaris Resource Manager™ Software with Solaris™ PC NetLink Software - Part 2

By Don DeVitt - Enterprise Engineering

Sun BluePrints™ OnLine - June 2000



http://www.sun.com/blueprints

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303 USA
650 960-1300 fax 650 969-9131

Part No.: 806-5683-10
Revision 01, June 2000

Please
Recycle

Using Solaris Resource Manager™ with Solaris™ PC NetLink Software - Part 2

Part 1 of this article was published in April Edition of Sun BluePrints™ OnLine, and described how to incorporate Solaris™ PC NetLink software within the Solaris Resource Manager™ software. By accomplishing this task, the resources used by Solaris PC NetLink software can be managed. This Part 2 focuses primarily on managing CPU resources for Solaris PC NetLink software. There are two purposes for which these resources need to be managed. First, a minimum amount of resources need to be allocated to maintain Solaris PC NetLink service to the user community. Secondly, Solaris PC NetLink resources need to be managed to assure that the resources consumed do not starve other critical services such as database, email or web services that are running on the same server.

Keep in mind, that in order to effectively manage resources on a server, you need to perform controlled experiments. For example, without the following experiment, you would only be guessing on how many resources to allocate for Solaris PC NetLink operation. Trying to determine what Solaris Resource Manager shares to use by experimenting on a production server is not something you should consider doing.

This article shows the results of a lab exercise that will help guide you in determining how many resources, in the form of Solaris Resource Manager “shares”, that Solaris PC NetLink software could need for a particular server environment.

Solaris Resource Management Software and Solaris PC NetLink Throughput

To determine what reasonable levels of CPU shares should be allocated to the Solaris PC NetLink software, an experiment is setup to benchmark the software throughput as it is given fewer and fewer Solaris Resource Manager shares. In this experiment the following server configuration is used:

- Sun Enterprise™ 450 (4x300 MHz Processors), 4 Gbytes Memory
- `/tmp` is shared via the Solaris PC NetLink software and is used as the file system for the NetBench benchmark. This eliminates the possibility that a disk subsystem could be the bottleneck for the benchmark because `/tmp` is implemented in free memory by the Solaris™ Operating Environment.
- One 100 Mbit Ethernet connection is used for this experiment. This is not enough network bandwidth to allow the CPUs to saturate, but is adequate to derive the information needed for this experiment.
- Thirty 300 MHz PCs are used to run the benchmark, all PCs are attached to a network switch using full duplex 100 Mbit connections to a network switch.
- Four 100 percent CPU Duty-cycle processes (for a four-processor system) are used to consume 100 percent of the CPU resource if allowed to do so by the Solaris Resource Manager software.
- The NetBench benchmark is configured to test all 30 PCs with no wait time to allow for maximum potential load.

To measure performance differences caused by using Solaris Resource Manager, you must have a competing application that will attempt to consume CPU time. For this experiment, the following command is used to consume CPU resources.

```
# srmuser acctsrvr dd if=/dev/zero of=/dev/null &
```

This launches a Solaris Operating Environment `dd` command under the Solaris Resource Manager lnode group `acctsrvr` (see figure 1). This `dd` command moves an infinite stream of zeros (0x00) from (`/dev/zero`) to the bit bucket (`/dev/null`), as fast as possible. After executing this `dd` command once on an otherwise idle system, you will see one CPU show 0 percent idle time when viewing the `mpstat` command. To consume 100 percent of the CPU time for all processors, you will need to run the above `dd` command for each of the processors on the system. For a four processor system you will need to execute the command four times. The Solaris Operating Environment will do the job of distributing the processes across all

processors, and the `mpstat` command will report something similar to the following screen text. Note, the idle time and the IO wait (`wt`) time are 0 in the last two columns on the right.

```
# mpstat
CPU minf mjf xcal  intr ithr  csw icsw migr smtx  srw syscl  usr sys  wt idl
  0    0   0  171   654    0 1401  650    8  867    0 61700   35  65   0   0
  1    0   0   63   402    4  510  253    6 4769    0 68027   32  68   0   0
  2    0   0    0  2722 2675  190   91    5  579    0 56246   25  75   0   0
  3    0   0 1038   488    0 1158  482   11 2338    0 50320   31  69   0   0
```

Executing the above "`dd`" command for each of the four processors, forces the Solaris PC NetLink software to compete for needed CPU resources. It is now necessary to create a Solaris Resource Manager lnode tree from which CPU resource shares can be allocated between the competing resources and the Solaris PC NetLink processes. For this test, an lnode tree is created where both the Solaris PC NetLink processes and the other CPU consuming processes are part of a group called `srvgroup`. Branching off from `srvgroup`, are the two competing lnodes `lmxadmin` and `acctsrvr`. In this example, user and system processes remain inactive and will not compete for resources. The lnode tree looks like the following.

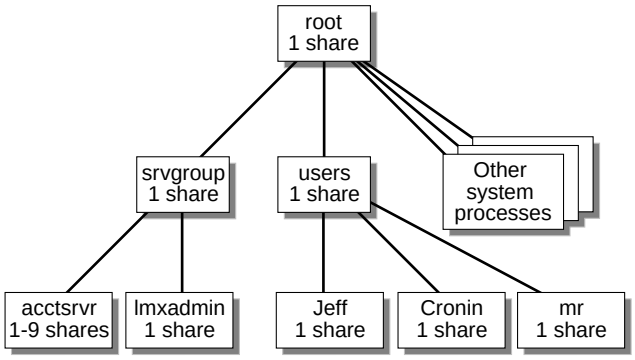


FIGURE 1 lnode Tree Used by Solaris Resource Manager to Include Solaris PC NetLink Test

Another way to view the lnode tree is to use the Solaris Resource Manager `limreport` command. The lnode tree can be reconstructed from the output of this command. The following is the output of the `limreport` command for this test environment.

```
# limreport flag.real "%s \t %s \t %i \n" lname sgroupname
cpu.shares
root            1s
daemon   root    1s
bin      root    1s
sys      root    1s
adm      root    1s
lp       root    1s
smtp             1s
uucp     root    1s
nuucp    root    1s
listen   root    1s
lanman   root    1s
lmxadmin         servgrp         1s
srmidle          root    0s
srmlost          root    1s
srmother         root    1s
jeff     users   1s
cronin   users   1s
mr       users   1s
servgrp          root    10s
acctsrvr         servgrp         8s
users    root    1s
```

FIGURE 2 `limreport` Command that Describes Solaris Resource Manager lnode Tree

The `flag.real` parameter selects only those UIDs that have mapped lnodes.

Each of the lnodes require an /etc/passwd entry even though the processes are running under the root or some other account. The following are the /etc/passwd entries for the lnodes used in this test:

```
. . . standard system /etc/passwd entries not shown . . .

lanman:x:100:13:SunLink Server account:/opt/lanman:/bin/false
lmxadmin:x:92780:13:SunLink Server Administrator:/var/opt/
lanman/lmxadmin:/bin/sh
lmxguest:x:92781:13:SunLink Server GUEST Login:/home/lmxguest:/
bin/false
lmworld:x:92782:13:SunLink Server World Login:/home/lmworld:/
bin/false
srmidle:x:41:1:SRM idle user:/:/bin/false
srmlost:x:42:1:SRM lost user:/:/bin/false
srmother:x:43:1:SRM other user:/:/bin/false
jeff:x:50:10::/export/home/logullo:/bin/ksh
cronin:x:51:10::/export/home/kevin:/bin/ksh
mr:x:53:10::/export/home/mr:/bin/ksh
servgrp:x:44:10::/:/bin/sh
acctsrvr:x:45:10::/:/bin/sh
users:x:46:10::/:/bin/sh
```

FIGURE 3 /etc/passwd Files Showing Entries Used by Solaris Resource Manager to Define lnodes

The specific change (discussed in Part 1 of this article) made to the system to incorporate the Solaris PC NetLink processes (processes starting with lmx) was a simple edit made to the /etc/init.d/ms_srv file that starts the Solaris PC NetLink environment at boot time. To edit the file, search for the entry LMXCRTL.

The differences in the file before and after the edit are following:

```
Before the Edit
    cd `dirname $LMXCTRL_PATH` && $LMXCTRL_NAME < $DEV_NULL_PATH

After the Edit

   cd `dirname $LMXCTRL_PATH` && /usr/srm/bin/srmuser lmxadmin \
                    $LMXCTRL_NAME < $DEV_NULL_PATH
```

This change to the file causes the system to launch the lmx.ctrl process from within the Solaris Resource Manager framework under the lmxadmin lnode. Keep in mind that the processes will run as root but will be in the Solaris Resource

Manager hierarchy as `lmxadmin`. All processes that are spawned by `lmx.ctrl` will also fall within the Solaris Resource Manager software hierarchy as `lmxadmin`, which allows them all to be controlled as one entity.

The next step is to assign shares to the two competing processes. Initially, each lnode was given one share using the `limadm` command.

```
# limadm set cpu.shares=1 acctsrvr
# limadm set cpu.shares=1 lmxadmin
```

Both `acctsvr` and `lmxadmin` have been assigned one share. In addition, they both fall under the `servgrp` lnode in the Solaris Resource Manager hierarchy; therefore, they will initially get 1/2 of whatever shares the `servgrp` lnode is allocated. Currently, this system is not performing any significant functions other than the two that need to be controlled for this test; therefore, you can start running a benchmark and see how the allocation of shares changes the throughput. The following table shows the measured results under the various settings of Solaris Resource Manager shares. This particular NetBench test forces all 30 PCs in the test to place a full load on the server. The first entry would normally provide a throughput much higher than than what is measured; however, the one 100 Mbit Ethernet connection, limits the bandwidth of the connection.

TABLE 1 Effect of Load on Throughput

NetBench Maximum Throughput	Setting of Solaris PC NetLink / acctsrvr(100% load)
10,346,435	1 / 1 No processes but Solaris PC NetLink running (Maximum possible, in this server configuration)
8,670,323	1 / 1 With 4 CPU `dd` command loads
6,712,454	1 / 2 With 4 CPU `dd` command loads
5,479,512	1 / 3 With 4 CPU `dd` command loads
5,006,338	1 / 4 With 4 CPU `dd` command loads
4,657,473	1 / 5 With 4 CPU dd command loads
4,409,744	1 / 6 With 4 CPU dd command loads
4,135,639	1 / 7 With 4 CPU dd command loads
3,693,225	1 / 8 With 4 CPU dd command loads
3,533,888	1 / 9 With 4 CPU dd command loads

As the competing processes are given more and more shares, the Solaris PC NetLink software receives a smaller and smaller percentage of the CPU resources. Thus, its maximum throughput performance decreases.

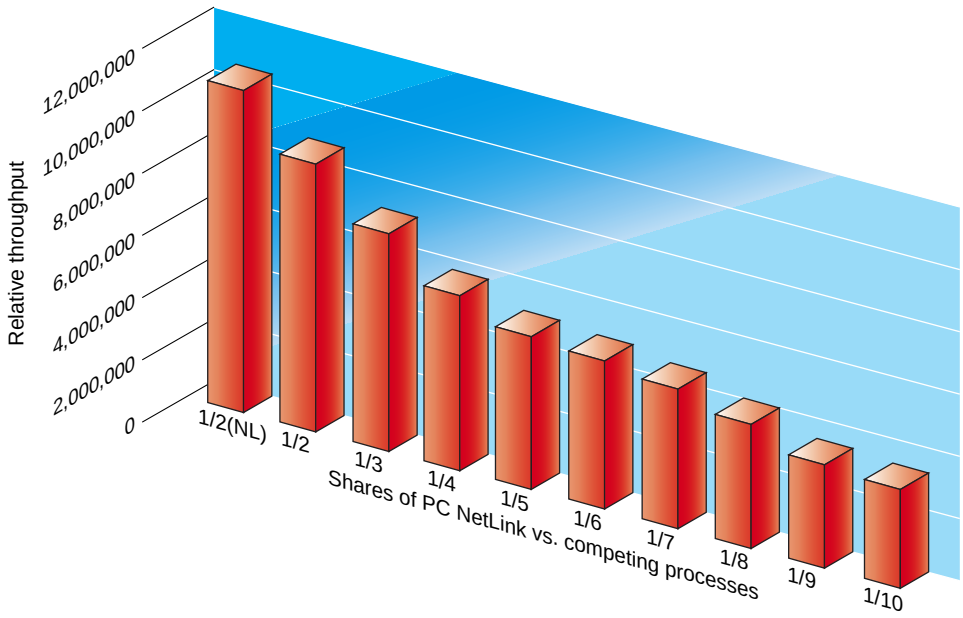


FIGURE 4 Relative Performance of the Solaris PC NetLink Software as Solaris Resource Manager Shares are Reallocated

The first entry of the chart (labeled 1/2(NL)) shows the performance of this system when no competing processes are running. Again, the system is using only one 100 Mbit full-duplex connection; therefore, the maximum throughput of the first value is gated by the network connection and is just over 10 Mbytes/sec. The CPU utilization is measured to be 60 percent for this data sample. When the four dd CPU loads are turned on to consume every remaining CPU cycle, and the Solaris PC NetLink processes are allocated one share (50 percent of the CPU resources), the throughput drops significantly, just as it should. As more shares are allocated to the CPU loading processes, with the dd command, the throughput continues to drop further and further allocating more CPU resources to the artificial load and less to the Solaris PC NetLink processes.

In a real-world setting, you could launch web servers, email servers, or any other server processes into the Solaris Resource Manager hierarchy with the Solaris PC NetLink software. By doing this, the other processes that Solaris PC NetLink software competes with for resources would never be given more than their allocated shares when they are all actively consuming resources.

Remember, you will not easily see Solaris Resource Manager software work unless all the processes it controls are actively seeking resources. If some processes are not using their shares and are in an idle state, the other processes with greater demands, will consume the resources that they otherwise could not use.

Note – Seeing Solaris Resource Manager work is sometimes difficult unless you perform tests like the one shown here, where you can guarantee CPU loads that will consume the full system.

In this test, neither the Solaris PC NetLink software, nor the PC clients that were running the NetBench benchmark, ever became so starved for resources that the benchmark timed-out or failed.

Continuing the test, from 1/10 to 1/20, 1/40, 1/60, 1/80, and 1/100 shares you may see some unexpected results. You would expect the benchmark throughput to continue to get lower until it starves the PC clients to the point where time outs would occur. You would also expect the curve to approach zero. However, for the results of this test, you see the benchmark settle at a line of ~1 Mbytes/sec instead. You may ask yourself what accounts for this additional throughput?

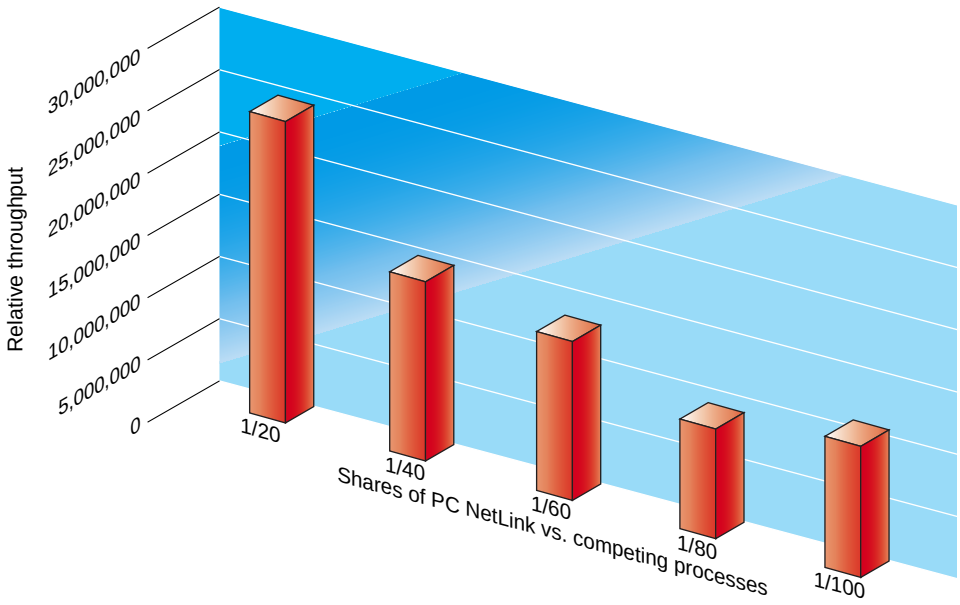


FIGURE 5 Extreme Control of the Solaris PC NetLink software via Solaris Resource Manager Down to 1 of 100 Shares

The additional throughput is due to SMB opportunistic locking that allows the Solaris PC clients to cache files locally. As the PC clients operate on files that they themselves are caching, the benchmark continues to show throughput as these locally cached files are accessed. What you need to keep in mind is that even when the Solaris PC NetLink software is allocated only one percent of a four-CPU server, with 99 percent of the CPU resources being consumed by other processes, the software will continue to function normally, although with degraded performance. In fact, none of the 30 PC clients failed to finish the benchmark under these extreme conditions.

With Solaris Resource Manager software, you can give the Solaris PC NetLink software as little as one of 100 shares. While performance may be slow, it will be reliable. In this test, the Solaris Operating Environment `/tmp` directory was used as the test file system that the Solaris PC NetLink software shared. This was done to make sure no disk waits could place the Solaris PC NetLink processes in an I/O wait state. As long as you distribute the file server load across as many subsystems as possible, you will see similar results.

Conclusion

Part 1 of this article listed and defined the most likely tools and techniques that can be used with Solaris PC NetLink software to participate in a resource management function. Part 1 also explained how system managers can implement a server that incorporates Solaris PC NetLink software within Solaris Resource Manager Software's scheduler.

In this article (Part 2) an experiment is performed to determine a useful range of shares that can be allocated to the Solaris PC NetLink software by the Solaris Resource Manager software.

References

Using Solaris™ Resource Manager with the Solaris™ PC NetLink Software Part 1 by Don DeVitt - Sun BluePrints™ OnLine Article May 2000 Edition available via `http://www.sun.com/blueprints/0500/srmpcnet1.pdf`

For additional, detailed information on Solaris PC NetLink software, refer to the Sun BluePrints book, *Solaris™ PC NetLink Software: Performance, Sizing, and Deployment*, (ISBN# 0-13-026686-8) which is scheduled for publication by Prentice-Hall in the May of 2000 and is expected to be available through www.sun.com/books, amazon.com, fatbrain.com, and Barnes & Noble bookstores.

Solaris Resource Manager 1.1 System Administration Guide search for the title on http:// docs.sun.com

Solaris Resource Manager by Richard McDougall - Sun BluePrints OnLine article available via
`http://www.sun.com/software/solutions/blueprints/0499/solaris1.pdf`

The Ziff Davis NetBench 6.0 benchmark is described and documented at the Ziff Davis Benchmark Operations website - `http://www.zdnet.com/zdbop/`

Author's Bio: Don De Vitt

Don is currently a Senior Staff Engineer and PC inter operability specialist within the Enterprise Engineering group, and is a member of the Solaris PC NetLink engineering team where he has focused on performance-related issues. Don has been on the development teams of almost every software and hardware PC interoperability product that Sun Microsystems has produced over the last 13 years.

Don DeVitt started his career as an electrical engineer and worked in the Automated Test industry (Teradyne Inc.), and PC operating system market (Digital Research from CP/M fame) before coming to Sun.