

Re: What can I check to fix system performance?

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.setup/2005-10/0561.html>

From: Enrique Perez-Terron (*enrio_at_online.no*)

Date: 10/19/05

Date: Wed, 19 Oct 2005 21:27:00 +0200

On Wed, 19 Oct 2005 19:47:42 +0200, tuxworks <ranjanr@gmail.com> wrote:

```
> vmstat output:
> *****
> procs -----memory-----swap-- -----io----- --system---
> -----cpu-----
> r b swpd free buff cache si so bi bo in cs us
> sy id wa
> 0 0 480 33924 71024 3577508 0 0 5 50 192 160 15
> 16 68 1
```

Are these memory data taken while the system was running that make?

If so, it seems you have plenty of memory available: 3.5 Gbytes.
Also notice the amount of swap usage: In: 0, out: 0.

(When the kernel does not need the memory for anything else, it keeps copies of files you have read of written lately, in a cache, in case you access them again. That makes it look like that there is never much memory free, but you should really consider "cache" as equivalent to "free" – except a resonable amount matching the amount of cache hits per 10 seconds or so.)

The next time you post data, please do not use Google to post. As you see above, Google breaks the lines, making it very hard to read this kind of tabular data. Since you obviously have access to Linux computers, get a newsreader. Myself I am using Opera, and, well, not bad, but not perfect, some frustrating bugs and limitations. I see people are talking about Pan, and I am considering giving it a try. I used Gnus (standard part of Emacs), quite good, but the docs are horrible: They try to be good, but unless you know the lingo, there is no hope. And no, you don't know the lingo.

As newbies go, you are unusually well behaved, posting a selection of data. Good! Unfortunately (and you are are aware of it,it seems, you hint at it) this problem is quite difficult and needs much more data and a bit of thinking and asking and replying.

comp.os.linux.setup: Re: What can I check to fix system performance?

Myself I am among the less expert in tuning issues, so I hope others will join in with suggestions, but here are my first reactions.

Is it possible that the increase in time is due to changes in the software package or its makefiles?

Builds can usually be made a lot faster by adding some "-j" options to make commands. This is true on uniprocessor systems too. I saw you said your system has 4 CPUs, then consider having at least 16 processes running, four per CPU. Quite likely, the optimum is closer to 10 processes per CPU, or 40 in all. (This is based on my own experience, but that was with a very different kind of systems about 10 years ago. If other readers on this list know differently and specifically for Linux systems or modern systems, please speak up.)

However, this has to be done judiciously. It is far from trivial to do. But I brought a build down from 30 hours to 13 that way. (Back then gmake did not have "-j", so I hacked a version of bash that allowed me to run 300 processes with "&", and the shell would only issue as many processes as I allowed in a special variable, and automatically issue another from the queue when any process terminated. I found 25 gave good results. I did not really experiment with it, I just was so happy to come down to 13 hours. But I believe the OS in question was more geared for batch processing than is customary today, that is why I would consider trying first with lower numbers per CPU.)

Consider the disk structure. See if you can arrange that the build reads one disk while it writes another. This measure and the -j option measure win time taken from the same pool: each measure can, with luck, give great returns, but when both are applied, the second one does not win nearly as much.

First of all learn more about what is going on when the system is loaded. Run "top". By default the list is sorted with the processes that use more CPU at the top. If the system becomes memory or disk constrained it is better to sort by other columns. Hit the ">" key. Then the list gets sorted by the next column, %mem (rather than %cpu). Play with ">" and "<". Do "man top" to learn about displaying other columns with other kinds of data.

Take some snapshots by copying and pasting into an editor. (Since you are new to Linux, do you know the Unix way of copying and pasting? There is no "Ctrl-C Ctrl-V", but just mark what you want to copy, and *middle*-click where you want to paste. You do have a mouse with three buttons? or you have configured the trick of using both buttons at once to emulate the middle button?) If you need more time between screen updates (default 1 second), find the commands to change that.

In the head part of the "top" screen, look at the "%wa" item in the Cpu(s) line. It measures the time the CPU is waiting, mostly this is

Re: What can I check to fix system performance?

comp.os.linux.setup: Re: What can I check to fix system performance?

disk wait. Notice that this is different from idle.

Tell us about the disks your system has. Is the system using any network mounted storage? If so, we will need data about the network performance too.

Be specific. If you are using software raid or logical volumes, use the tools to list the complete configuration and post it. Also post the output of "fdisk -l /dev/hda" and similarly for the other disks. You probably have most of your data on scsi disks or hardware raid units, use the appropriate tools to list the partitioning (if applicable) and setup. Post the file /etc/fstab, and tell us what mount points contain the source file tree. If the build tree is separate from the source tree, tell us what mount point contains it.

If the build writes files in the source tree, consider if you can modify the build process to have a separate build tree where all but the pristine source files are written. Then consider using a tmpfs to hold the build tree. This means the files are mostly not written to disk at all. The "make install" step should write the final results to a disk-backed file system, of course.

-Enrique