

Re: Solaris real-time thread capabilities

Source: <http://unix.derkeiler.com/Newsgroups/comp.unix.solaris/2005-09/0370.html>

From: Greg Menke (gregm-xyzpdq_at_toadmail.com)

Date: 09/07/05

Date: 06 Sep 2005 20:24:15 -0400

capnwhit@yahoo.com writes:

> *Greg Menke wrote:*

> > *Have you run empty loops using those various techniques to see*

> > *if you can achieve the timing you're looking for?*

> >

> *Hello Greg, All of my tests so far have been using empty loops! I can't*

> *get the timing right even on an empty loop... that is why I didn't*

> *bother explaining the size constraints for our jitter buffers, our*

> *packages, or the rest of our app.*

>

> > *there wasn't even the results of prstat showing how busy*

> > *the cpu was while executing your job.*

> >

> *The reason I didn't post this is because it was a non-issue (1 or*

> *2%)... like I said the test loop is empty, just calling nanosleep to*

> *try and achieve the timing... When using the spin loop technique, of*

> *course, the CPU usage went up, but I only tried said technique when*

> *using a processor set, and only when my test program was the only LWP*

> *assigned to the target processor. Therefore the system still had one*

> *entire processor in which to schedule other LWPs.*

Ahh thats the kind of info that we need. I'll sniff around as well.

Gregm