Subject: Re: [PATCH v2 5/5] expose per-taskgroup schedstats in cgroup
Posted by Glauber Costa on Wed, 18 Apr 2012 16:24:37 GMT
View Forum Message <> Reply to Message

>
> You define the idle time as the sum of task's sleeping time which i
> think it needs to
> discuss.

Where is it done ?

Idle time here is measured as the time between enqueue_sleeper() and
the group being put back in the rq.
But note it is enqueue sleeper for the group, not any tasks.

cfs will call this callback until it finds anything that is running
(task or not a task).

Maybe I made some mistake in the code - and in this case, please point
out - but that's the idea.

> IMHO, idle
> time can just
> be the true system value. Personally I prefer to your last version in
> the way of computing
> idle time (http://thread.gmane.org/gmane.linux.kernel/1194838). And
> iowait can be
> computed in the similar way.

No. The idea that idle time can only be true system-wide is wrong. As a
matter of fact, that first series of mine is totally wrong wrt that (and
then I changed).

A cgroup is idle when none of its tasks are in the runqueue. What is the
problem that you see with this?

As for iowait, that one seemed a bit trickier, so we decided to leave it
out at least for now.

>
> As to steal time, "Steal time is the percentage of time a virtual CPU
> waits for a real
> CPU while the hypervisor is servicing another virtual processor".
> Speaking from the
> point of view of resource controlling(isolation), cgroup is a
> lightweight method towards
> virtualiztion, so I think obeying its primitive meaning is more
> appropriate: the time not

> servicing me including time stolen by the tasks of other cgroup.

And that's exactly what I've done.

Steal time is runqueue time, until you are chosen to run.

In a summary: If you are not running, you can be either idle or stolen.
if you are in the runqueue, you are stolen.
If you are not, you are idle.
