Subject: Re: [ckrm-tech] [RFC] [PATCH 0/3] Add group fairness to CFS
Posted by Srivatsa Vaddagiri on Mon, 28 May 2007 16:39:19 GMT
View Forum Message <> Reply to Message

On Fri, May 25, 2007 at 10:14:58AM -0700, Li, Tong N wrote:
> Nice work, Vatsa. When I wrote the DWRR algorithm, I flattened the
> hierarchies into one level, so maybe that approach can be applied to
> your code as well. What I did is to maintain task and task group weights
> and reservations separately from the scheduler, while the scheduler only
> sees one system-wide weight per task and does not concern about which
> group a task is in. The key here is the system-wide weight of each task
> should represent an equivalent share to the share represented by the
> group hierarchies. To do this, the scheduler looks up the task and group
> weights/reservations it maintains, and dynamically computes the
> system-wide weight *only* when it need a weight for a given task while
> scheduling. The on-demand weight computation makes sure the cost is
> small (constant time). The computation itself can be seen from an
> example: assume we have a group of two tasks and the group's total share
> is represented by a weight of 10. Inside the group, let's say the two
> tasks, P1 and P2, have weights 1 and 2. Then the system-wide weight for
> P1 is 10/3 and the weight for P2 is 20/3. In essence, this flattens
> weights into one level without changing the shares they represent.

What do these task weights control? Timeslice primarily? If so, I am not
sure how well it can co-exist with cfs then (unless you are planning to
replace cfs with a equally good interactive/fair scheduler :)

I would be very interested if this weight calculation can be used for
smpnice based load balancing purposes too ..

--
Regards,
vatsa

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers