Subject: Re: [ckrm-tech] [RFC] [PATCH 0/3] Add group fairness to CFS
Posted by Srivatsa Vaddagiri on Wed, 30 May 2007 17:14:05 GMT
View Forum Message <> Reply to Message

On Sat, May 26, 2007 at 08:41:12AM -0700, William Lee Irwin III wrote:
> The smpnice affair is better phrased in terms of task weighting. It's
> simple to honor nice in such an arrangement. First unravel the
> grouping hierarchy, then weight by nice. This looks like
>
> task nice hier1 hier2 ... hierN
> t_1 w_n1 w_h11 w_h21 ... w_hN1
> t_2 w_n2 w_h12 w_h22 ... w_hN2
> ...
>
> For the example of nice 0 vs. nice 10 as distinct users with 10%
> steppings between nice levels, one would have
>
> task nice hier1
> t_1 1 1
> t_2 0.3855 1
>
> w_1, the weight of t_1, would be
> (w_h11*w_n1/(w_h11*w_n1 + w_h12*w_n2))
> = (1*1/(1 + 1*0.3855..))
> = 0.7217..
> w_2, the weight of t_2, would be
> (w_h12*w_n2/(w_h11*w_n1 + w_h12*w_n2))
> = (1*0.3855../(1 + 1*0.3855..))
> = 0.27826..
> This just so happens to work out to being the same as if t_1 and t_2
> had their respective nice numbers without the scheduler grouping, which
> is basically what everyone wants to happen.
>
> It's more obvious how to extend it to more tasks than levels of
> hierarchy. An example of that follows:
>
> task nice hier1 hier2 ... hierN
> t_1 0.3 0.6 * ... *
> t_2 0.7 0.4 * ... *
>
> hier2 through hierN are ignorable since t_1 and t_2 are both the only
> members at those levels of hierarchy. We then get something just like
> the above example, w_1 = 0.3*0.6/(0.3*0.6+0.7*0.4) = 0.3913.. and
> w2 = 0.7*0.4/(0.3*0.6+0.7*0.4) = 0.6087..
>
> It's more interesting with enough tasks to have more meaningful levels
> of hierarchy.
>

> task nice hier1 hier2
> t_1 0.7 0.6 0.6
> t_2 0.3 0.6 0.4
> t_3 0.7 0.4 0.6
> t_4 0.3 0.4 0.4
>
> where t_1 and t_2 share a hier1 grouping and t_3 and t_4 also share
> a hier1 grouping, but the hier1 grouping for t_1 and t_2 is distinct
> from the hier1 grouping for t_3 and t_4. All hier2 groupings are
> distinct. So t_1 would have pre-nice weight 0.6*0.6, t_2 0.6*0.4,
> t_3 0.6*0.4, and t_4 0.4*0.4 (the numbers were chosen so denominators
> conveniently collapse to 1). Now that the hierarchy is flattened,
> nice numbers can be factored in for t_1's final weight being
> 0.7*0.36/(0.7*0.36+0.3*0.24+0.7*0.24+0.3*0.16) = 0.252/0.54 = 0.467..
> and the others being 0.133.. (t_2), 0.311.. (t_3), and 0.0889.. (t_4).

Hmm ..so do you think this weight decomposition can be used to flatten
the tree all the way to a single level in case of cfs? That would mean we can
achieve group fairness with single level scheduling in cfs ..I am
somewhat skeptical that we can achieve group fairness with a single
level rb-tree (and w/o substantial changes to pick_next_task logic in cfs
that is), but if it can be accomplished would definitely be a great win.

> In such a manner nice numbers obey the principle of least surprise.

--
Regards,
vatsa

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