
Subject: Re: [ckrm-tech] [RFC] Resource Management - Infrastructure choices
Posted by [Matt Helsley](#) on Tue, 31 Oct 2006 00:26:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Mon, 2006-10-30 at 18:26 +0300, Pavel Emelianov wrote:

> Paul Jackson wrote:
> > Pavel wrote:
> >>>> 3. Configfs may be easily implemented later as an additional
> >>>> interface. I propose the following solution:
> >>>> ...
> >> Resource controller has nothing common with configfs.
> >> That's the same as if we make netfilter depend on procfs.
> >
> > Well ... if you used configfs as an interface to resource
> > controllers, as you said was easily done, then they would
> > have something to do with each other, right ;)?
>
> Right. We'll create a dependency that is not needed.
>
> > Choose the right data structure for the job, and then reuse
> > what fits for that choice.
> >
> > Neither avoid nor encouraging code reuse is the key question.
> >
> > What's the best fit, long term, for the style of kernel-user
> > API, for this use? That's the key question.
>
> I agree, but you've cut some important questions away,
> so I ask them again:
>
> > What if if user creates a controller (configfs directory)
> > and doesn't remove it at all. Should controller stay in
> > memory even if nobody uses it?

Yes. The controller should stay in memory until userspace decides that control of the resource is no longer desired. Though not all controllers should be removable since that may impose unreasonable restrictions on what useful/performant controllers can be implemented.

That doesn't mean that the controller couldn't reclaim memory it uses when it's no longer needed.

<snip>

Cheers,
-Matt Helsley
