Subject: Re: [ckrm-tech] [patch00/05]: Containers(V2)- Introduction
Posted by Chandra Seetharaman on Thu, 21 Sep 2006 00:30:07 GMT
View Forum Message <> Reply to Message

On Wed, 2006-09-20 at 12:57 -0700, Paul Menage wrote:
> On 9/20/06, Chandra Seetharaman <sekharan@us.ibm.com> wrote:
> > > At its most crude, this could be something like:
> > >
> > > struct container {
> > > #ifdef CONFIG_CPUSETS
> > >   struct cpuset cs;
> > > #endif
> > > #ifdef CONFIG_RES_GROUPS
> > >   struct resource_group rg;
> > > #endif
> > > };
> >
> > Won't it restrict the user to choose one of these, and not both.
>
> Not necessarily - you could have both compiled in, and each would only
> worry about the resource management that they cared about - e.g. you
> could use the memory node isolation portion of cpusets (in conjunction
> with fake numa nodes/zones) for memory containment, but give every
> cpuset access to all CPUs and control CPU usage via the resource
> groups CPU controller.
>
> The generic code would take care of details like container
> creation/destruction (with appropriate callbacks into cpuset and/or
> res_group code, tracking task membership of containers, etc.

What I am wondering is that whether the tight coupling of rg and cpuset
(into a container data structure) is ok.

>
> Paul
--

Chandra Seetharaman | Be careful what you choose....
- sekharan@us.ibm.com | .....you may get it.

Subject: Re: [ckrm-tech] [patch00/05]: Containers(V2)- Introduction
Posted by Paul Jackson on Thu, 21 Sep 2006 00:33:17 GMT
View Forum Message <> Reply to Message

Chandra wrote:

> What I am wondering is that whether the tight coupling of rg and cpuset
> (into a container data structure) is ok.

Just guessing wildly here, but I'd anticipate that at best we
(resource groups and cpusets) would share container mechanisms,
but not share the same container instances.

--

I won't rest till it's the best ...
Programmer, Linux Scalability
Paul Jackson <pj@sgi.com> 1.925.600.0401

Subject: Re: [ckrm-tech] [patch00/05]: Containers(V2)- Introduction
Posted by Paul Menage on Thu, 21 Sep 2006 00:34:52 GMT
View Forum Message <> Reply to Message

On 9/20/06, Chandra Seetharaman <sekharan@us.ibm.com> wrote:
>
> What I am wondering is that whether the tight coupling of rg and cpuset
> (into a container data structure) is ok.

Can you suggest a realistic scenario in which it's not? Don't forget
that since the container abstraction is hierarchical, you don't have
to use both at the same level. So you could easily e.g. have a parent
container in which you bound to a set of memory/cpu nodes, but had no
rg limits, and several subcontainers where you configured nothing
special for cpuset parameters (so inherited the parent params) but
tweaked different rg parameters.

Paul

Subject: Re: [ckrm-tech] [patch00/05]: Containers(V2)- Introduction
Posted by Chandra Seetharaman on Thu, 21 Sep 2006 00:50:00 GMT
View Forum Message <> Reply to Message

On Wed, 2006-09-20 at 17:33 -0700, Paul Jackson wrote:
> Chandra wrote:
> > What I am wondering is that whether the tight coupling of rg and cpuset
> > (into a container data structure) is ok.
>
> Just guessing wildly here, but I'd anticipate that at best we
> (resource groups and cpusets) would share container mechanisms,
> but not share the same container instances.

That is what my thinking too.

>
--

---------- ----------
Chandra Seetharaman | Be careful what you choose....
- sekharan@us.ibm.com | .....you may get it.
---------- ----------