Subject: Re: [ckrm-tech] [RFC][PATCH 5/7] UBC: kernel memory accounting (core)
Posted by Chandra Seetharaman on Mon, 21 Aug 2006 20:55:44 GMT
View Forum Message <> Reply to Message

On Mon, 2006-08-21 at 14:51 +0400, Kirill Korotaev wrote:
> Chandra Seetharaman wrote:
> > Kirill,
> >
> > IMO, a UBC with resource constraint(limit in this case) should behave no
> > different than a kernel with limited memory. i.e it should do
> > reclamation before it starts failing allocation requests. It could even
> > do it preemptively.
> first, please notice, that this thread is not about user memory.
> we can discuss it later when about to control user memory. And
> I still need to notice, that different models of user memory control
> can exist. With and without reclamation.
>
we can talk about it then :)

> > There is no guarantee support which is required for providing QoS.
> where? in UBC? in UBC _there_ are guarentees, even in regard to OOM killer.

I do not see it in the patches you have submitted. May be I overlooked.
Can you please point me the code where guarantee is handled.

>
> > Each controller modifying the infrastructure code doesn't look good. We
> > can have proper interfaces to add a new resource controller.
> controllers do not modify interfaces nor core. They just add
> themself to the list of resources and setup default limits.
> do you think it is worth creating infrastructure for these
> 2 one-line-changes?

Yes, IMO, it is cleaner.

Think of the documentation that explains how to write a controller for
UBC.

With a proper interface it will read something like: One have to call
register_controller(char *name) and on success it returns a unique id
which is the id for the controller.

Vs

With changing lines in the core code: One have to edit the file
filename.c and add a macro to this of macros with an incremented value
for their controller and add the name of their controller to the array
named controller_names[].

I think the first one is cleaner, what do you think ?

<snip>

--

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