
Subject: Re: [RFC][PATCH 2/7] RSS controller core
Posted by [Andrew Morton](#) on Sun, 11 Mar 2007 12:51:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

> On Sun, 11 Mar 2007 15:26:41 +0300 Kirill Korotaev <dev@sw.ru> wrote:

> Andrew Morton wrote:

> > On Tue, 06 Mar 2007 17:55:29 +0300

> > Pavel Emelianov <xemul@sw.ru> wrote:

> >

> >

> >>+struct rss_container {

> >>+ struct res_counter res;

> >>+ struct list_head page_list;

> >>+ struct container_subsys_state css;

> >>+};

> >>+

> >>+struct page_container {

> >>+ struct page *page;

> >>+ struct rss_container *cnt;

> >>+ struct list_head list;

> >>+};

> >

> >

> > ah. This looks good. I'll find a hunk of time to go through this work

> > and through Paul's patches. It'd be good to get both patchsets lined

> > up in -mm within a couple of weeks. But..

> >

> > We need to decide whether we want to do per-container memory limitation via

> > these data structures, or whether we do it via a physical scan of some

> > software zone, possibly based on Mel's patches.

> i.e. a separate memzone for each container?

Yep. Straightforward machine partitioning. An attractive thing is that it
100% reuses existing page reclaim, unaltered.

> imho memzone approach is inconvenient for pages sharing and shares accounting.

> it also makes memory management more strict, forbids overcommitting

> per-container etc.

umm, who said they were requirements?

> Maybe you have some ideas how we can decide on this?

We need to work out what the requirements are before we can settle on an
implementation.

Sigh. Who is running this show? Anyone?

You can actually do a form of overcommitment by allowing multiple containers to share one or more of the zones. Whether that is sufficient or suitable I don't know. That depends on the requirements, and we haven't even discussed those, let alone agreed to them.
