

---

Subject: Re: [PATCH 6/7] containers (V7): BeanCounters over generic process containers

Posted by [Paul Menage](#) on Mon, 12 Feb 2007 18:49:58 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On 2/12/07, Srivatsa Vaddagiri <vatsa@in.ibm.com> wrote:

> On Mon, Feb 12, 2007 at 12:15:27AM -0800, menage@google.com wrote:

> > This patch implements the BeanCounter resource control abstraction

> > over generic process containers.

>

> Forgive my confusion, but do we really need two-levels of resource control

> abstraction here? Why can't resource controllers directly work with containers

> (just like cpu accounting does)?

>

The generic containers patch represents a pretty low-level view of task grouping - it doesn't try to prescribe how to do accounting, nor exactly what API to present to the user (beyond providing a filesystem-based interface).

Resource controllers certainly can be written directly over it, but equally having additional abstractions to provide a common user API and kernel API for multiple resources is a reasonable goal.

I would imagine that each different resource being controlled would be represented as a container subsystem, which is how I structured the ResGroups example patch - ResGroups becomes a library that provides a common set of file manipulations for different resource controllers, each of which is a containers subsystem. The same could potentially be done for BeanCounters if people wanted.

But the main point of the latter four patches in this series is to illustrate to the various folks writing resource controller systems (and other observers) that this patch provides sufficient features to act as a base for their work. I don't presume to claim that one higher-level resource control abstraction is better than another.

Paul

---