Subject: Re: Linux-VServer example results for sharing vs. separate mappings ...
Posted by akpm on Sun, 25 Mar 2007 18:51:09 GMT
View Forum Message <> Reply to Message

On Sun, 25 Mar 2007 15:20:35 +0530 Balbir Singh <balbir@in.ibm.com> wrote:

> Andrew Morton wrote:
> <snip>
> > The problem is memory reclaim. A number of schemes which have been
> > proposed require a per-container page reclaim mechanism - basically a
> > separate scanner.
> >
> > This is a huge, huge, huge problem. The present scanner has been under
> > development for over a decade and has had tremendous amounts of work and
> > testing put into it. And it still has problems. But those problems will
> > be gradually addressed.
> >
> > A per-container recaim scheme really really really wants to reuse all that
> > stuff rather than creating a separate, parallel, new scanner which has the
> > same robustness requirements, only has a decade less test and development
> > done on it. And which permanently doubles our maintenance costs.
> >
>
> The current per-container reclaim scheme does reuse a lot of code. As far
> as code maintenance is concerned, I think it should be easy to merge
> some of the common functionality by abstracting them out as different
> functions. The container smartness comes in only in the
> container_isolate_pages(). This is an easy to understand function.

err, I think I'd forgotten about container_isolate_pages(). Yes, that
addresses my main concern.

> > So how do we reuse our existing scanner? With physical containers. One
> > can envisage several schemes:
> >
> > a) slice the machine into 128 fake NUMA nodes, use each node as the
> >   basic block of memory allocation, manage the binding between these
> >   memory hunks and process groups with cpusets.
> >
> >   This is what google are testing, and it works.
>
> Don't we break the global LRU with this scheme?

Sure, but that's deliberate!

(And we don't have a global LRU - the LRUs are per-zone).

> >

> > b) Create a new memory abstraction, call it the "software zone", which
> > is mostly decoupled from the present "hardware zones". Most of the MM
> > is reworked to use "software zones". The "software zones" are
> > runtime-resizeable, and obtain their pages via some means from the
> > hardware zones. A container uses a software zone.
> >
>
> I think the problem would be figuring out where to allocate memory from?
> What happens if a software zone spans across many hardware zones?

Yes, that would be the tricky part. But we generally don't care what
physical zone user pages come from, apart from NUMA optimisation.

> The reclaim mechanism proposed *does not impact the non-container users*.

Yup. Let's keep plugging away with Pavel's approach, see where it gets us.

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers