
Subject: Re: [PATCH 4/7] chained slab caches: move pages to a different cache when a cache is destroyed.

Posted by [Glauber Costa](#) on Wed, 22 Feb 2012 14:50:58 GMT

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

On 02/22/2012 03:40 AM, Suleiman Souhlal wrote:

> On Tue, Feb 21, 2012 at 3:34 AM, Glauber Costa<glommer@parallels.com> wrote:

>> In the context of tracking kernel memory objects to a cgroup, the
>> following problem appears: we may need to destroy a cgroup, but
>> this does not guarantee that all objects inside the cache are dead.
>> This can't be guaranteed even if we shrink the cache beforehand.

>>

>> The simple option is to simply leave the cache around. However,
>> intensive workloads may have generated a lot of objects and thus
>> the dead cache will live in memory for a long while.

>

> Why is this a problem?

>

> Leaving the cache around while there are still active objects in it
> would certainly be a lot simpler to understand and implement.

>

Yeah, I agree on the simplicity. The chained stuff was probably the hardest one in the patchset to get working alright. However, my assumptions are as follow:

1) If we bother to be tracking kernel memory, it is because we believe its usage can skyrocket under certain circumstances. In those scenarios, we'll have a lot of objects around. If we just let them flowing, it's just wasted memory that was created from the memcg, but can't be reclaimed on its behalf.

2) We can reclaim that, if we have, as a policy, to always start shrinking from those when global pressure kicks in. But then, we move the complication from one part to another.
