

---

Subject: Re: [PATCH 00/23] slab+slub accounting for memcg  
Posted by [Glauber Costa](#) on Wed, 02 May 2012 15:14:37 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On 04/30/2012 06:43 PM, Suleiman Souhlal wrote:

>> I am leaving destruction of caches out of the series, although most  
>> > of the infrastructure for that is here, since we did it in earlier  
>> > series. This is basically because right now Kame is reworking it for  
>> > user memcg, and I like the new proposed behavior a lot more. We all seemed  
>> > to have agreed that reclaim is an interesting problem by itself, and  
>> > is not included in this already too complicated series. Please note  
>> > that this is still marked as experimental, so we have so room. A proper  
>> > shrinker implementation is a hard requirement to take the kmem controller  
>> > out of the experimental state.  
> We will have to be careful for cache destruction.  
> I found several races between allocation and destruction, in my patchset.  
>  
> I think we should consider doing the uncharging of kmem when  
> destroying a memcg in mem\_cgroup\_destroy() instead of in  
> pre\_destroy(), because it's still possible that there are threads in  
> the cgroup while pre\_destroy() is being called (or for threads to be  
> moved into the cgroup).

I found some problems here as well.

I am trying to work on top of what Kamezawa posted for pre\_destroy()  
rework. I have one or two incorrect uncharging issues to solve, that's  
actually what is holding me for posting a new version.

expected soon

---