
Subject: Re: [PATCH][for -mm] Fix and Enhancements for memory cgroup [6/6] add force reclaim interface

Posted by [KAMEZAWA Hiroyuki](#) on Wed, 10 Oct 2007 00:41:38 GMT

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

On Wed, 10 Oct 2007 00:14:53 +0530

Balbir Singh <balbir@linux.vnet.ibm.com> wrote:

> KAMEZAWA Hiroyuki wrote:

> > This patch adds an interface "memory.force_reclaim".

> > Any write to this file will drop all charges in this cgroup if

> > there is no task under.

> >

> > %echo 1 > /...../memory.force_reclaim

> >

>

> Looks like a good name, do you think system administrators would

> find force_empty more useful?

>

good name :) I'll use it.

> > +static void

> > +mem_cgroup_force_reclaim_list(struct mem_cgroup *mem, struct list_head *list)

> > +{

> > + struct page_cgroup *pc;

> > + struct page *page;

> > + int count = SWAP_CLUSTER_MAX;

> > + unsigned long flags;

> > +

> > + spin_lock_irqsave(&mem->lru_lock, flags);

> > +

>

> Can we add a comment here stating that this routine reclaims just

> from the per cgroup LRU and not from the zone LRU to which the

> page belongs.

>

Ok.

> > + while (!list_empty(list)) {

> > + pc = list_entry(list->prev, struct page_cgroup, lru);

> > + page = pc->page;

> > + if (clear_page_cgroup(page, pc) == pc) {

> > + css_put(&mem->css);

> > + res_counter_uncharge(&mem->res, PAGE_SIZE);

> > + list_del_init(&pc->lru);

> > + kfree(pc);

> > + } else

```

>> + count = 1; /* race? ...do relax */
>> +
>> + if (--count == 0) {
>> + spin_unlock_irqrestore(&mem->lru_lock, flags);
>> + cond_resched();
>> + spin_lock_irqsave(&mem->lru_lock, flags);
>> + count = SWAP_CLUSTER_MAX;
>> + }
>> + }
>> + spin_unlock_irqrestore(&mem->lru_lock, flags);
>> +}
>> +
>> +int mem_cgroup_force_reclaim(struct mem_cgroup *mem)
>> +{
>> + int ret = -EBUSY;
>> + while (!list_empty(&mem->active_list) ||
>> + !list_empty(&mem->inactive_list)) {
>> + if (atomic_read(&mem->css.cgroup->count) > 0)
>> + goto out;
>> + mem_cgroup_force_reclaim_list(mem, &mem->active_list);
>> + mem_cgroup_force_reclaim_list(mem, &mem->inactive_list);
>> + }
>> + ret = 0;
>> +out:
>> + css_put(&mem->css);
>
> We do a css_put() here, did we do a css_get() anywhere?
>

```

Good catch. it is a BUG. I'll fix.

Thanks,
-Kame

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