Subject: Re: [RFC] [PATCH] memory controller background reclamation
Posted by yamamoto on Mon, 22 Oct 2007 23:44:30 GMT
View Forum Message <> Reply to Message

hi,

```
> > @@ -250,6 +256,69 @@ unsigned long mem_cgroup_isolate_pages(u
> >   return nr_taken;
> >  }
> > 
> > +static int
> > +mem_cgroup_need_reclaim(struct mem_cgroup *mem)
> > +{
> > + struct res_counter * const cnt = &mem->res;
> > + int doreclaim;
> > + unsigned long flags;
> > +
> > + /* XXX should be in res_counter */
> > + /* XXX should not hardcode a watermark */
>
> We could add the following API to resource counters
>
> res_counter_set_low_watermark
> res_counter_set_high_watermark
> res_counter_below_low_watermark
> res_counter_above_high_watermark
>
> and add
>
> low_watermark
> high_watermark
>
> members to the resource group. We could push out data
> upto the low watermark from the cgroup.
```

it sounds fine to me.

```
> > +static void
> > +mem_cgroup_reclaim(struct work_struct *work)
> > +{
> > + struct mem_cgroup * const mem =
> > +     container_of(work, struct mem_cgroup, reclaim_work);
> > + int batch_count = 128; /* XXX arbitrary */
> > +
> > + for (; batch_count > 0; batch_count--) {
> > +  if (!mem_cgroup_need_reclaim(mem))
> > +   break;
> > +  /*
```

> > +   * XXX try_to_free_foo is not a correct mechanism to
> > +   * use here. eg. ALLOCSTALL counter
> > +   * revisit later.
> > +   */
> > + if (!try_to_free_mem_cgroup_pages(mem, GFP_KERNEL))
>
> We could make try_to_free_mem_cgroup_pages, batch aware and pass that
> in scan_control.

in the comment above, i meant that it might be better to introduce
something like balance_pgdat rather than using try_to_free_mem_cgroup_pages.
with the current design of cgroup lru lists, probably it doesn't
matter much except statistics, tho.

YAMAMOTO Takashi