Subject: Re: [PATCH 02/11] memcg: Reclaim when more than one page needed.
Posted by Glauber Costa on Tue, 26 Jun 2012 09:23:40 GMT
View Forum Message <> Reply to Message

On 06/26/2012 01:17 PM, David Rientjes wrote:
> On Tue, 26 Jun 2012, Glauber Costa wrote:
>
>>> Nope, have you checked the output of /sys/kernel/slab/.../order when
>>> running slub? On my workstation 127 out of 316 caches have order-2 or
>>> higher by default.
>>>
>>
>> Well, this is still on the side of my argument, since this is still a majority
>> of them being low ordered.
>
> Ok, so what happens if I pass slub_min_order=2 on the command line? We
> never retry?

Well, indeed. The function has many other RETRY points, but I believe
we'd reach none of them without triggering oom first.

>> The code here does not necessarily have to retry -
>> if I understand it correctly - we just retry for very small allocations
>> because that is where our likelihood of succeeding is.
>>
>
> Well, the comment for NR_PAGES_TO_RETRY says
>
> /*
> * We need a number that is small enough to be likely to have been
> * reclaimed even under pressure, but not too big to trigger unnecessary
> * retries
> */
>
> and mmzone.h says
>
> /*
> * PAGE_ALLOC_COSTLY_ORDER is the order at which allocations are deemed
> * costly to service. That is between allocation orders which should
> * coalesce naturally under reasonable reclaim pressure and those which
> * will not.
> */
> #define PAGE_ALLOC_COSTLY_ORDER 3
>
> so I'm trying to reconcile which one is correct.
>

I am not myself against reverting back to costly order. The check we

have here, of course, is stricter than costly order, so there is nothing to reconcile. Now if we really need a stricter check or not, is the subject of discussion.