Subject: Re: [RFC][for -mm] memory controller enhancements for reclaiming take2 [5/8] throttling simultaneous
Posted by Balbir Singh on Tue, 04 Dec 2007 13:27:22 GMT
View Forum Message <> Reply to Message

KAMEZAWA Hiroyuki wrote:
> On Mon, 3 Dec 2007 09:24:18 -0500
> Rik van Riel <riel@redhat.com> wrote:
>
>> On Mon, 3 Dec 2007 18:39:21 +0900
>> KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com> wrote:
>>
>>> Add throttling direct reclaim.
>>>
>>> Trying heavy workload under memory controller, you'll see too much
>>> iowait and system seems heavy. (This is not good.... memory controller
>>> is usually used for isolating system workload)
>>> And too much memory are reclaimed.
>>>
>>> This patch adds throttling function for direct reclaim.
>>> Currently, num_online_cpus/(4) + 1 threads can do direct memory reclaim
>>> under memory controller.
>> The same problems are true of global reclaim.
>>
>> Now that we're discussing this RFC anyway, I wonder if we
>> should think about moving this restriction to the global
>> reclaim level...
>>
> Hmm, I agree to some extent.
> I'd like to add the same level of parameters to memory controller AMAP.
>

The CKRM memory controller had the following parameters for throttling

Watermarks

shrink_at
shrink_to

and

num_shrinks
shrink_interval

Number of times shrink can be called in a shrink_interval.

> But, IMHO, there are differences basically.

>
> Memory controller's reclaim is much heavier than global LRU because of
> increasing footprint , the number of atomic ops....
> And memory controller's reclaim policy is simpler than global because
> it is not kicked by memory shortage and almost all gfk_mask is GFP_HIGHUSER_MOVABLE
> and order is always 0.
>
> I think starting from throttling memory controller is not so bad because
> it's heavy and it's simple. The benefit of this throttoling is clearer than
> globals.
>

I think global throttling is good as well, sometimes under heavy load I
find several tasks stuck in reclaim. I suspect throttling them and avoid
 this scenario. May be worth experimenting an thinking about it deserves
more discussion.

> Adding this kind of controls to global memory allocator/LRU may cause
> unexpected slow down in application's response time. High-response application
> users may dislike this. We may need another gfp_flag or sysctl to allow
> throttling in global.
> For memory controller, the user sets its memory limitation by himself. He can
> adjust parameters and the workload. So, I think this throttoling is not so
> problematic in memory controller as global.
>
> Of course, we can export "do throttoling or not" control in cgroup interface.
>

I think we should export the interface.

>
> Thanks,
> -Kame
>

--
 Warm Regards,
 Balbir Singh
 Linux Technology Center
 IBM, ISTL

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