Subject: Re: [PATCH 0/5] blk-throttle: writeback and swap IO control
Posted by Andrea Righi on Wed, 23 Feb 2011 08:32:06 GMT
View Forum Message <> Reply to Message

On Tue, Feb 22, 2011 at 07:03:58PM -0500, Vivek Goyal wrote:
> > I think we should accept to have an inode granularity. We could redesign
> > the writeback code to work per-cgroup / per-page, etc. but that would
> > add a huge overhead. The limit of inode granularity could be an
> > acceptable tradeoff, cgroups are supposed to work to different files
> > usually, well.. except when databases come into play (ouch!).
>
> Agreed. Granularity of per inode level might be accetable in many
> cases. Again, I am worried faster group getting stuck behind slower
> group.
>
> I am wondering if we are trying to solve the problem of ASYNC write throttling
> at wrong layer. Should ASYNC IO be throttled before we allow task to write to
> page cache. The way we throttle the process based on dirty ratio, can we
> just check for throttle limits also there or something like that.(I think
> that's what you had done in your initial throttling controller implementation?)

Right. This is exactly the same approach I've used in my old throttling controller: throttle sync READs and WRITEs at the block layer and async WRITEs when the task is dirtying memory pages.

This is probably the simplest way to resolve the problem of faster group getting blocked by slower group, but the controller will be a little bit more leaky, because the writeback IO will be never throttled and we'll see some limited IO spikes during the writeback. However, this is always a better solution IMHO respect to the current implementation that is affected by that kind of priority inversion problem.

I can try to add this logic to the current blk-throttle controller if you think it is worth to test it.

-Andrea

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