Subject: A consideration on memory controller.
Posted by KAMEZAWA Hiroyuki on Mon, 21 Jan 2008 08:07:59 GMT
View Forum Message <> Reply to Message

This mail is about memmory controller feature in my mind.
no patches, just a chitchat.

==
One of my purposes for contributing memory controller is to make applications stable. That is, I'd like to reduce hiccups on the system and guarantee applications some level of performance, throughput, latency, AMAP.

>From my experience in user support, one of causes of unexpected temporal performance regression is File-I/O. iowait. In many case, customers say that there are too many unused page caches, reduce it...
But delay is caused by caching is not correct.
That's just because there are too much dirty buffer and not enough free memory for stable run, just the system is overloaded or parameter tuning was not enough.

Shortage of free memory can delay system temporarlly. A big delay is caused by write-back and a small reason is LRU-scan.
(A bit off-topic.
 For reducing amount of write-back, we can use dirty_ratio. But when we reduce
 dirty_ratio, syslogd hits it and delayed. This delays other applications
 which just doesn't issue I/O but call syslog(). Does anyone have good idea ?)

Kswapd reclaims freeable memory periodically for keeping free memory to be some amount within min <-> low <-> high. But in emergency, an application itself can reclaim memory by itself with calling try_to_free_pages().
This try_to_free_pages() scans LRU and reclaims some amount of memory and delays an application which doesn't I/O just requesting memory.

If memory controller is used, we can limit maximum usage of memory per applications. Workload can be isolated per cgroup.
This is good one progress. But maybe I need more features for my purpose....maybe.

One consideration is...
Now, memory controller can tamper LRU/relcaim handling but cannot do free memory. For guaranteing amount of usable memory for an applicatons, using VM is the best answer. But sometimes it can't be used.
I'm wondering whether we can add free-memory controller or not. It will gather free memory for some cgroup with low <-> min <-> high + page-order setup and work as buffer within cgroup <-> system workload.
But I'm not sure this idea is good or not ;)

BTW, I and YAMAMOTO-san is now considering followings for next series.

- back ground reclaim (Maybe it's better to wait for RvR's LRU set merge.)
- guarantee some amount of memory not to be reclaimed by global LRU.
- per cgroup swappiness.
- swap controller. (limit swap usage...maybe independet from memory
  controller.)

belows are no patch, no plan topics.

- limit amount of mlock.
- limit amount of hugepages.
- more parameters for page reclaim.
- balancing on NUMA (if we can find good algorythm...)
- dirty_ratio per cgroup.

- multi-level memory controller.

If you have feature-lists against memory controller, I'd like to see.

Note:
In last year, limit size of page-cache was posted but denied. It is said that free memory is bad memory. Now, I never think anything just for limitig page-cache will be accepted.

Thanks,
-Kame

____