
Subject: Subject: [PATCH 0/2] dm-ioband: I/O bandwidth controller v1.3.0:
Introduction

Posted by [Ryo Tsuruta](#) on Fri, 11 Jul 2008 11:14:11 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi everyone,

This is the dm-ioband version 1.3.0 release.

Dm-ioband is an I/O bandwidth controller implemented as a device-mapper driver, which gives specified bandwidth to each job running on the same physical device.

- Can be applied to the kernel 2.6.26-rc5-mm3.

- Changes from 1.2.0 (posted on Jul 4, 2008):

- I/O smoothing take #2

This feature makes I/O requests of each group issued smoothly. Once a certain group has used up its tokens, all I/O requests to the group will be blocked until all the other groups used up theirs. This feature is to minimize this blocking time and to issue I/O requests at a constant rate according to the weight, without decreasing throughput.

We have tested various ideas to achieve this feature and we have chosen the most effective ways as follows:

- Shorten the epoch period of dm-ioband, each of which every ioband group will get new tokens. On the other hand, the leftover tokens for the past few epochs can be taken over to the next epoch so that it can keep the fairness between the groups even when the I/O loads of some groups are changing.
 - Make a new epoch immediately when a group with large weight used up its tokens even though there remain a lot of in-flight I/Os.

To gain the throughput, dm-ioband will recharge tokens to all the groups without waiting their I/O completion if possible.

- Make the I/O requests which user process have just made be handled ahead of the blocked I/O requests, since it would make sense that you assume the groups which issued these blocked I/O requests will have small weights.
 - Make the number of I/O requests which can be queued in dm-ioband smaller, so it will prevent all the I/O request of each

group from being issued at the same time when a new epoch gets made.

- TODO

- Implementing cgroup support for dm-ioband is in progress. This feature makes it be able to handle asynchronous I/O requests properly.

I added a new benchmark result on the dm-ioband webpage. This result shows that dm-ioband can control a bandwidth even when an unbalanced

I/O load is applied.

<http://people.valinux.co.jp/~ryov/dm-ioband/benchmark/partition3.html>

Thanks,

Ryo Tsuruta

Linux Block I/O Bandwidth Control Project

<http://people.valinux.co.jp/~ryov/bwctl/>

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>
