

Re: [RFC] JBD ordered mode rewrite

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2008-03/msg03346.html>

- *From:* Mingming Cao <cmm@xxxxxxxxxx>
 - *Date:* Fri, 07 Mar 2008 16:08:20 -0800
-

On Fri, 2008-03-07 at 16:52 -0700, Andreas Dilger wrote:

On Mar 06, 2008 18:42 +0100, Jan Kara wrote:

Below is my rewrite of ordered mode in JBD. Now we don't have a list of data buffers that need syncing on transaction commit but a list of inodes that need writeout during commit. This brings all sorts of advantages such as possibility to get rid of journal heads and buffer heads for data buffers in ordered mode, better ordering of writes on transaction commit, simplification of some JBD code, no more anonymous pages when truncate of data being committed happens. The patch has survived some light testing but it still has some potential of eating your data so beware :) I've run dbench to see whether we didn't decrease performance by different handling of truncate and the throughput I'm getting on my machine is the same (OK, is lower by 0.5%) if I disable the code in truncate waiting for commit to finish... Also the throughput of dbench is about 2% better with my patch than with current JBD.
Any comments or testing most welcome.

Looks like a very good patch – thanks for your effort in moving this beyond the "hand-waving" stage that it's been in for the past few years.

I'm looking at what implications this has for delayed allocation in ext4, because the vast majority of file data will be unmapped in that case and a journal commit in ordered mode will no longer cause the data to be flushed to disk.

I *think* is OK, because the `pdflushd` will now be totally in charge of flushing the dirty pages to disk, instead of this previously being done by ordered mode in the journal.

I missed something here, just trying to understand: if a journal commit in ordered mode will no longer cause the data to be flushed to disk, how could we ensure the ordering? Are you suggesting with delayed allocation the journalling mode falls back to writeback mode?

Re: [RFC] JBD ordered mode rewrite

I know there have been some bugs in this area in the past, but I guess it isn't much different than running in writeback mode. That said, I don't know how many users run in writeback mode unless they are running a database, and the database is doing a lot of explicit fsync of file data so there may still be bugs lurking...

Some care is still needed here because