

Re: True fsync() in Linux (on IDE)

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2004-03/4405.html>

From: Matthias Andree (matthias.andree_at_gmx.de)

Date: 03/18/04

Date: Thu, 18 Mar 2004 12:34:53 +0100
To: Linux Kernel <linux-kernel@vger.kernel.org>

On Thu, 18 Mar 2004, Jens Axboe wrote:

> *Chris and I have working real fsync() with the barrier patches. I'll
> clean it up and post a patch for vanilla 2.6.5-rc today.*

This is good news.

The barrier stuff is long overdue^UI'm looking forward to this.

I'm using the term "TCQ" liberally although it may be inexact for older (parallel) ATA generations:

All these ATA fsync() vs. write cache issues have been open for much too long – no reproaches, but it's a pity we haven't been able to have data consistency for data bases and fast bulk writes (that need the write cache without TCQ) in the same drive for so long. I have seen Linux introduce TCQ for PATA early in 2.5, then drop it again. Similarly, FreeBSD ventured into TCQ for ATA but appears to have dropped it again as well.

May I ask that the information whether a particular driver (file system, hardware) supports write barriers be exposed in a standard way, for instance in the Kconfig help lines?

If I recall correctly from earlier patches, the barrier stuff is 1. command model (ATA vs. SCSI) specific and 2. driver and hardware specific and 3. requires that the file system knows how to use this properly.

Given that file systems have certain write ordering requirements if they are to be recoverable after a crash, I suspect Linux has `_not_` been able to guarantee on-disk consistency for any time for years, which means that a crash in the wrong moment can kill the file system itself if the drive has reordered writes – only ext3 without write cache seems to behave better in this respect (data=ordered).

I would like to have a document that shows which file system, which

Linux-Kernel: Re: True fsync() in Linux (on IDE)

chipset driver for PATA, which chipset driver for ATA, which low-level SCSI host adaptor driver, which file system support write barrier. We will probably also need to check if intermediate layers such as md and dm-mod propagate such information.

Given the necessary information, I can hack together a HTML document to provide this information; this offer has however not seen any response in the past. I am however not acquainted with the drivers and need information from the kernel hackers. Without such support, such a documentation effort is doomed.

BTW, I should very much like to be able to trace the low-level write information that goes out to the device, possibly including the payload – something like tcpdump for the ATA or SCSI commands that are sent to the driver. Is such a facility available?

--

Matthias Andree

Encrypt your mail: my GnuPG key ID is 0x052E7D95

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@vger.kernel.org

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>