Contribution ID: **288** Type: **not specified**

IO: Durability, Errors and Documentation

Wednesday 11 September 2019 12:07 (20 minutes)

Postgres (and many other databases) have, until fairly recently, assumed that IO errors would a) be reliably signalled by fsync/fdatasync/... b) repeating an fsync after a failure would either result in another failure, or the IO operations would succeed.

That turned out not to be true: See also https://lwn.net/Articles/752063/

While a few improvements have been made, both in postgres and linux, the situation is still pretty bad.

From my point of view, a large part of the problem is that linux does not document what error and durability behaviour userspace can expect from certain operations.

Problematic areas for the kernel:
- The regular behaviour of durability fs related syscalls are not documented. One extreme example of that is sync_file_range (look at the warning section of the manpage)
- FS behaviour when encountering IO errors is poorly, if at all, documented. For example: there still is no documentation about the error behaviour of fsync, ext4's errors= operation reads as if it applied to all IO errors, but only applies to metadata errors.
- There is very little consistency for error behaviour between filesystems. To the degree that XFS will return different data after writeback failed than ext4.
- There is no usable interface to query / be notified of IO errors
- the rapid development of thin provisioned storage has increased the likelihood of IO errors drastically, as large parts of the IO stack treat out-of-space on the block level as an IO error

It seems worthwhile to work together to at least partially clean this up.

I agree to abide by the anti-harassment policy

Yes

Primary authors: FREUND, Andres (EnterpriseDB / PostgreSQL); Mr VONDRA, Tomas (Postgresql)

Presenters: FREUND, Andres (EnterpriseDB / PostgreSQL); Mr VONDRA, Tomas (Postgresql)

Session Classification: Databases MC