GJournal

Paweł Jakub Dawidek
<pjd@FreeBSD.org>

Journaling in FreeBSD (1/2)

- a tendon of Achilles for many years
- many failed attempts of journaling implementation for UFS in FreeBSD (SUN proved it's possible)

Journaling in FreeBSD (2/2)

- many available file system for FreeBSD, none of the ports support journaling:
  - ext2fs (journaling in ext3fs)
  - XFS (read-only)
  - ReiserFS (read-only)
  - HFS+ (read-write, but without journaling)
  - NTFS (read-only)

GEOM Journal

- block level journaling – below file system level
- both data and metadata journaling
- circular journal on the same or another provider
- file system independent
- minimum knowledge about file system is needed (5 lines of UFS-specific code in gjournal)

GJournal – not only UFS

- no need for synchronization of RAID1/RAID3 after a power failure or system crash
- speeds up operations on small files

UFS+GJournal

- file can be deleted, but still open
- directory can be removed, but still open
- after a power failure we have orphaned inode (with no name)

FS_GJOURNAL flag

- newfs(8)/tunefs(8)-time option
- added fs_unrefs counter to the super block
- added cg_unrefs counter to the cylinder group structure
- after a crash, on boot, fsck only looks for orphaned objects, which is much faster than regular fsck (TB file systems are checked in seconds/minutes instead of many hours)

Usage

```
# gjournal label da0
# newfs -J /dev/da0.journal
# mount -o async /dev/da0.journal /mnt
```

- yes, the 'async' option is safe for gjournaled file systems

How does it work

- every 10 seconds file system is synchronized, writes are suspended and journal is closed
- consistent journal is copied to the data provider

- during those 10 seconds, gjournal tries to optimize I/O traffic by combining Ios, skipping duplicated writes, etc.

Write Cache

- “turn off WC when you use journaling!”
- not with gjournal...

BIO_FLUSH

- new I/O request – BIO_FLUSH
- means “empty your write cache please”

Performance (1/4)

- copying one large file

UFS:	8s
UFS+SU:	8s
UFS+gjournal(1):	16s
UFS+gjournal(2):	14s

Performance (2/4)

- copying eight big files in parallel

UFS:	120s
UFS+SU:	120s
UFS+gjournal(1):	184s
UFS+gjournal(2):	165s

Performance (3/4)

- untaring eight src.tgz

UFS:	791s
UFS+SU:	650s
UFS+gjournal(1):	333s
UFS+gjournal(2):	309s

Performance (4/4)

- 'grep -r' on two src/ directories

UFS:	84s
UFS+SU:	138s
UFS+gjournal(1):	102s
UFS+gjournal(2):	89s

Status

- GJournal was committed to the HEAD recently
- one (probably more) remaining issue we plan to work with Kris after the 6.2-RELESE
- gjournal(8) manual page coming soon (after EuroBSDCon)

The End

Questions?