XFS for Linux

Christoph Hellwig
LST e.V.
hch@lst.de

http://verein.lst.de/~hch/

What exactly is a filesystem?

Organization of disk space

- We have a big disk and want to store small data items on them
- Files and directories used to organize

In UNIX everything is a stream of bytes

- More complex file defintion in other OSes
- Devices are also presented in the filesystem
- Linux takes it to the extreme (lots of virtual filesystems)

So where does XFS fit exactl?

Three generation of UNIX filesystems:

- v7 / sysv / coherent / minix
- ffs / ext2
- vxfs / jfs / xfs (/reiserfs)

Features of the 3rd generation UNIX filesystems

- Intent logging / journaling
- Flexible metadata structures
- Dynamic inode allocations
- Extents

A little bit of history

Back in Stoneage (1993)

- Berkely FFS was state of the art
- IRIX had EFS (FFS + Extents)

Limitations

- Small file sizes (2 GB)
- Small filesystem sizes (8 GB)
- Statically allocated metadata
- Long recovery times
- Very slow operation on big directories
- No extentended attributes
- Not very suitable for media streaming

All this is addressed by XFS

<add marketing blurbs here>

XFS Features (1)

- XFS uses B+ trees extensively instead of linear structures
  - locate free space
  - index directory entries
  - manage file extents
  - keep track of the locations of file index information
- XFS is a fully 64-bit file system
  - 64bit variables for global counters
  - 64bit disk addresses
  - 64bit inode numbers (not useable under Linux)
  - 18 million terabytes theoretical max filesystem size

XFS Features (2)

- Partitioned into Allocation Groups
  - each AG manages its own free space and inodes
  - provides scalability and parallelism within the file system
  - limits the size of the structures needed to track this information
  - allows many internal pointers to be 32-bits
  - AGs typically range in size from 0.5 to 4GB
  - files and directories are not limited to a single AG.

XFS Features (3)

- Sophisticated support utilities
  - fast mkfs (make a file system)
  - dump and restore (utilities for backup)
  - xfsrepair to fix corrupt filesystem
  - xfs_fsr (XFS defragmenter)
  - xfsdb (XFS debug)
  - xfscheck (XFS check)
  - xfs_growfs (enlarges filesystems online)

Porting XFS to Linux (1) - Basics

Why?

- Linux had the same issues (in 1999)
- SGI wants to sell Linux Servers
- SGI wants to be credible in the OSS Community

How?

- Kernel code is not portable
- Either rewrite or add a glue layer
- XFS port started with lots of glue
- More and more native interfaces used

Porting XFS to Linux (2) - Glue layers

Linvfs

- Maps Linux file/inode ops to IRIX vnode/vfs ops
- Nowdays very small

Pagebuf

- Implements an IRIX-like buffercache ontop of the linux pagecache

The support/ directory

- Implements IRIX helpers ontop of linux ones

Porting XFS to Linux (3) - Mismatches

- ioctl vs fnctl
  - XFS has many fcntl on IRIX
  - Linux doesn’t allow fs-specific fcntls
  - Use ioctls instead
- Credentials
  - IRIX passed down credentials to the fs
  - Fs has to do access checking by itself
  - Linux does access checks in the VFS
  - Solution: empty struct cred

Porting XFS to Linux (4) - Refinements

- Direct use of Linux data structures
  - Passing down dentry
  - struct statfs vs statvfs
- Duplicate code removal
  - Linux does _much_ more work in common code
  - About 2000 LOC gone
- Use the generic I/O code
  - Early versions uses pagebuf-based I/O path
  - Now uses generic Linux routines
  - Delalloc was hard to fit into this model

Volume Manager Integration

- Linux filesystems traditionally use fixed size I/O requests
  - Makes volume managers a lot easier
  - Too much overhead
- Linux 2.5 allows variable sized I/O requests
  - Exactly what XFS needs
  - Not properly handled by all drivers for a long time
- Linux 2.4 needs hacks
  - Guess whether a Volume Manager is used
  - The vary_io extension would help XFS

Interesting XFS Features

- Direct I/O
  - Allows to perform non-cached, direct to userspace I/O
  - Ported to Linux together with XFS
  - Independent implementation in Linux 2.4.10
  - XFS ported to generic framework
  - Still advantages over generic implementation
- Delayed Allocation
  - Very important for XFS performance
  - IRIX buffercache rewritten around it
  - Difficult to fit into 2.4 VM
  - 2.5 way of buffer writeout helps a lot
  - Same concept used on 2.4 now too

Data Migration API - DMAPI

- A horrible X/Open standard for HSM
  - still used a lot (Cray/SGI DMF, Veritas, IBM)
- XFS is the only Linux filesystem with DMAPI support
  - there was an OpenXDSM project, but it's dead now
- DMAPI is incompatible with Linux mount semantics
  - need to take special care when mounting
  - thus not part of XFS in Linus' 2.5 tree, only in SGI's tree

Opensourceing XFS

- Licensing questions
  - Opensource or not?
  - Community doesn’t care about proprtary drivers
  - Filesystem API changes a lot
  - Not GPL-compatible code won’t be merged into mainline

- Enncumbrance review
  - Contact as much as possible original hackers
  - Compare with other codebases (SVR4, BSD, ..)
    - keywords search
    - prototype comparism
  - Very few matches found and corrected
    - usually removal of uneeded code

IRIX vs Linux

- Two different codebases
  - core code is kept in sync
- Performance hard to compare due to different hardware
  - probably faster on Linux for smaller I/O
  - probably faster on IRIX for really large I/O
- Guranteed rate I/O only avaible on IRIX
  - becomes possible with Linux 2.6
- Some features were in Linux before IRIX
  - group quotas
  - v2 log format

Currenst Status

- XFS 1.3
  - 4th prerelease is out now
  - native byteorder incore extents
  - support for sector sizes != 512
  - lots of speedups
- Linux 2.4
  - part of Alan Cox’s tree now
  - required patches slowly go to Marcelo
- Linux 2.5
  - lagging a little bit behind 2.4 sometimes
  - XFS takes advantage of lots of 2.5 core changes
- xfsprogs
  - included in all major Linux distributions except Red Hat
  - ported to FreeBSD, Darwin and IRIX (again)

Future

- Case insensitive support
  - big speedup for samba
- Inode reclaim
  - allows to free space occupied by inodes again
- Performance tuning (especially on 2.5)

Why you want to use XFS

- Stable, mature codebase
  - oldest journaling filesystem available on Linux
- Very good performance for large IOS
- Designed for large systems
- DMAPI support
- Good support for ACLs / EAs
- Same disk format as on IRIX

Why you don’t want to use XFS

- No data journaling
- Baselines when hacking fs code :)

- 2000/03/30 Linux XFS source code officially available
- 2000/06/23 Usenix 2000 XFS pre-beta iso Image
- 2000/09/22 XFS Beta Release
- 2001/05/01 XFS 1.0 Release
- 2001/07/10 XFS 1.0.1 Release
- 2001/09/27 Mandrake 8.1 is available with native XFS support.
- 2001/11/16 XFS 1.0.2 Release
- 2002/04/17 XFS 1.1 Release
- 2002/04/18 SuSE 8.0 is available, with XFS filesystem support.
- 2002/09/16 XFS is merged into Linus' 2.5 development tree.
- 2003/02/11 XFS 1.2 Release
- 2003/04/28 XFS is now in Alan Cox's 2.4.21-rc1-ac3 kernel.
- 2003/07/?? XFS 1.3 Release

Ressources

- XFS/Linux homepage
  - http://oss.sgi.com/projects/xfs/