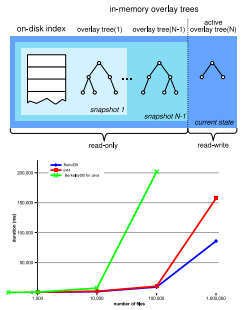
XTREEMFS

file systems for the masses

XtreemOS — Enabling Linux for the Grid

ZIB

BabuDB Metadata Storage



Efficient Key-Value Store

- implements key-value store with LSM-Trees

Snapshots

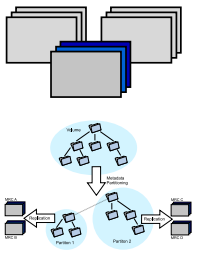
- can create and materialize snapshots asynchonously

Software

- available as Java library babudb.googlecode.com

BABUDB

Metadata Server



Master/Slave Replication

- simple through replicated BabuDB store backend

Master Failover

- master election using distributed lease negotiation with FaTLease [1]

Partitioning

- split the directory tree into subtrees and distribute them onto several servers

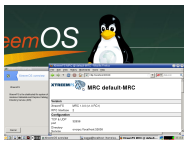
The XtreemOS Project

European Research Project (2006-2010)

- 19 partners from academia and industry from Europe and China

Results

- releases availble for download at www.xtreemos.eu



Software

Multi-Platform

- client for Linux, Windows and OS X
- servers for Linux and Solaris (Java 1.6+)

Download

www.xtreemfs.org

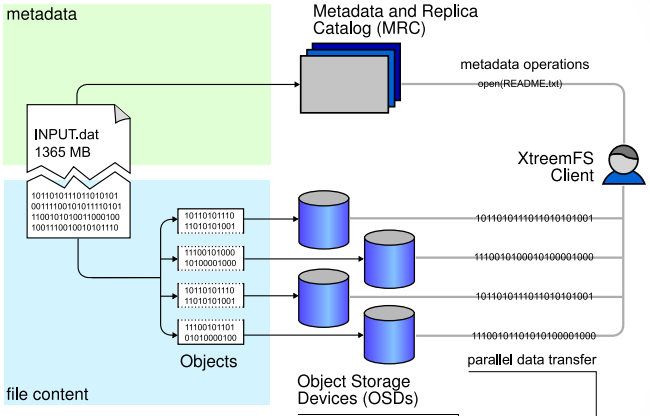
Internet Filesystem

Ready for the WAN

- all components can be distributed and connected via WAN links
- client side caching

SSL & X.509

- authentication based on certificates
- encryption of all network traffic



Architecture

Object-based File System

- metadata: directory tree, file names, xattrs...
- objects: chunks of file data (content)

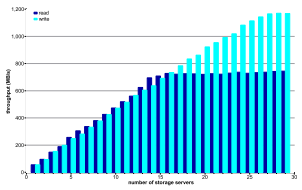
POSIX-compliant

- interface: regular file system interface
- semantics: similar to a local file system = no need to modify applications

Extensible

- plug-in architecture for authentication, authorization, user mappings, replica selection and placement

Scalable I/O with Striping

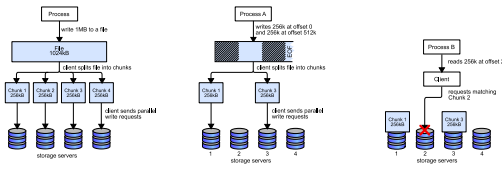


scale the system by adding OSDs to increase:

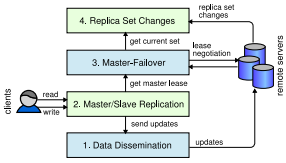
- I/O bandwidth
- IOPS
- storage capacity

several striping schemes:

- RAID0 - for performance
- RAID5 - for data safety



Read/Write File Replication



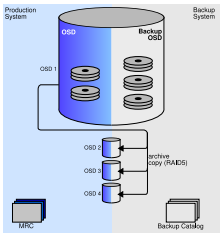
Master/Slave with Failover

- no consensus in critical path
- layered architecture for simpler code

Sequential Consistency

- local file system semantics = replication is transparent to users and applications
- allow user to relax consistency

Integrated Backup Architecture



Snapshots for Consistent Backups

- consistent metadata snapshots
- copy-on-write modification of file content

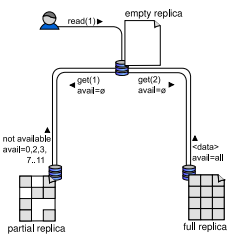
Incremental Backups

- only modified objects are copied
- fast restoration of different backup versions

Scalable Backup

- backup system automatically scales with production system
- scalable capacity
- scalable throughput

Read-only File Replication



Efficient and Transparent Caching

- no overhead for coordination of replicas

Partial Replicas

- contain only a subset of the file's data
- data is loaded and prefetched on demand
- saves bandwidth and disk storage
- quicker application startup

Modular Strategies

- to optimize bandwidth usage and minimize latency experienced by the users

P2P Technology for Maximum Bandwidth

- OSDs exchange information on available objects for load balancing and to increase bandwidth

References

[1] F. Hupfeld, B. Kolbeck, J. Stender, M. Högqvist, T. Cortes, J. Malo, J. Marti. "FaTLease: Scalable Fault-Tolerant Lease Negotiation with Paxos.". In: Proceedings of the International Symposium on High Performance Distributed Computing (HPDC) 2008.

[2] J. Stender, B. Kolbeck, F. Hupfeld, E. Cesario, E. Focht, M. Hess, J. Malo, J. Marti. "Striping without Sacrifices: Maintaining POSIX Semantics in a Parallel File System". 1st USENIX Workshop on Large-Scale Computing (LASCO '08), Boston, 2008

Information Society Technologies

XtreemOS is an Integrated Project supported by the European Commission's IST program #FP6-033576.

Kolbeck, Stender, Langner, Lorenz
Zuse Institute Berlin - © 2009