

The Metadata Problem

Understanding and Addressing the Issues

Anand Babu (AB) Periasamy
Gluster

What is Metadata



- Properties of the data itself
(all file systems need)
 - Name, timestamp, permissions, etc

- Information that the file system needs to function
(not the user)
 - Index location, internal locks, state

Why you should care

- ❑ It's the core problem for large scale file systems
- ❑ The challenge of indexing billions of files
 - Complex algorithms
- ❑ Performance
- ❑ Scalability
- ❑ Reliability

Why you should care

Today structured, unstructured, and objects are stored as files

- ❑ Historically data was mostly structured and transactional
- ❑ Unstructured file data now exploding
- ❑ Databases now struggling with scale and distributed object storage (NoSQL)

Why you should care

- ❑ Single -> double -> triple indirection
- ❑ Journalled file systems
- ❑ Software RAID and volume managers
- ❑ Techniques to index as a tree
 - Rapidly search dynamically changing data
- ❑ Don't address scaling past 16TB and multiple machines
 - Btrfs and zfs are not sufficient

Disk file systems are overwhelmed by the explosive growth of unstructured file data

Metadata Management: the most critical challenge of a file system



- Many solutions to parallel data access
- Metadata access is often a bottleneck
- Virtualization and cloud presenting a whole new set of challenges



❑ Clustered / Parallel file systems

- Lustre, Hadoop

❑ Reasons

- Allowed to parallelize data (but not metadata)
- Designed for large file size environment

❑ Centralized metadata challenges

- Reliability
- Scalability
- Performance bottleneck



- Parallelizes both metadata and data
 - Isilon, HP/Ibrix
- Introduces new challenges
 - Complexity
 - Data coherency
 - Reliability
- The “Google model” (GoogleFS II)



- ❑ No separation of data and metadata
- ❑ If everything is stored as a file, why separate metadata from data
- ❑ Knowledge of the file layout is preserved from the top VFS layer to the block layer
- ❑ Techniques
 - hash + mod
 - elastic hashing

No-Metadata Provides a Simple Foundation

- Storage becomes more reliable and scalable
- Take advantage of true parallelism
- Build a more sophisticated solution

Preparing for Failure

- ❑ Know the answer to the question: what if I lose my metadata?
- ❑ Prevent corruption from propagating
- ❑ File system verification / healing needs to be on line

Type of application / workload

- ❑ Mixed workloads
- ❑ Small vs. large file sizes
- ❑ Virtual machine storage
- ❑ Large scale search analytics
- ❑ REST-based

- ❑ Don't worry about marginal gains at the single system level
- ❑ Linear scalability is paramount
- ❑ Use real applications
- ❑ Benchmark the response to failures
- ❑ Disaster recovery
- ❑ Look for the moving bottleneck

Thank You

**Anand Babu (AB) Periasamy
Gluster**

ab@gluster.com