Alternative Implementations of Cluster File Systems

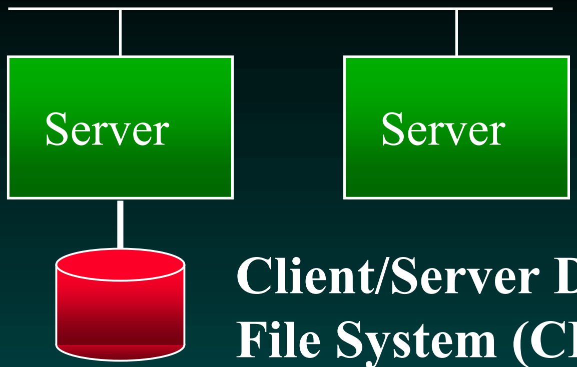
Yoshitake Shinkai
Fujitsu Laboratories LTD.
Jim Williams
Amdahl Corporation

Background:
Explosion of Internet

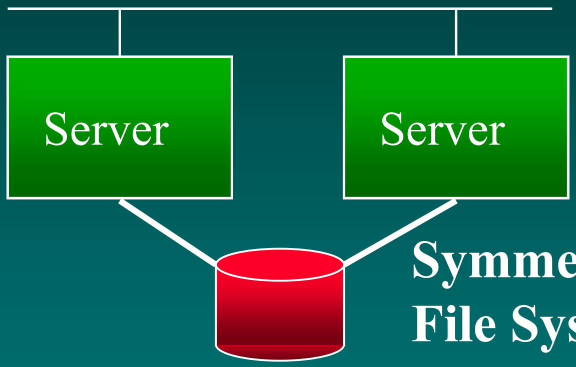
- Cluster Systems
- Storage Area Networks

Cluster File Systems

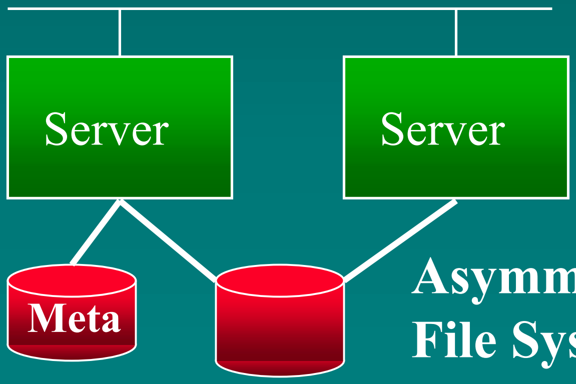
Architectural Models



Client/Server Distributed File System (CDFS)



Symmetric Shared File System (SSFS)



Asymmetric Shared File System (ASFS)

	CDFS	SSFS	ASFS
Simplicity	✦✦		✦
Performance		~✦✦	✦✦✦
Extensibility		✦	✦✦
Scalability		✦	
Reliability		✦✦	✦✦✦

ASFS can be superior through enhancements

HAMFS

Asymmetric Shared File System

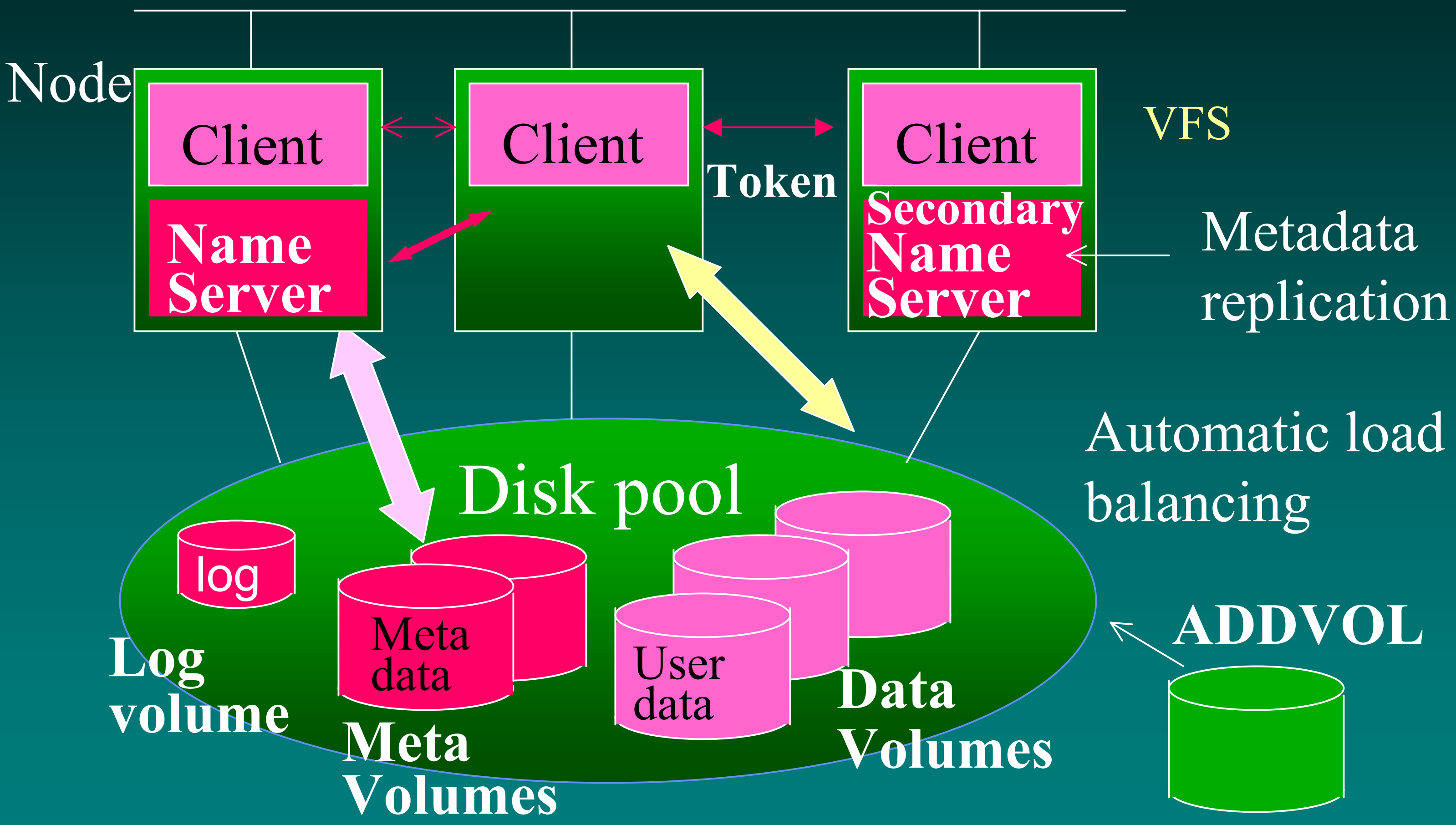
HAMFS : Research Project

SafeFILE : Product Version

- 24x7 operation
- Highly available
- High performance
- No special hardware requirements
- Easily managed

HAMFS: Highly Available Multi-server File System

HAMFS Configuration

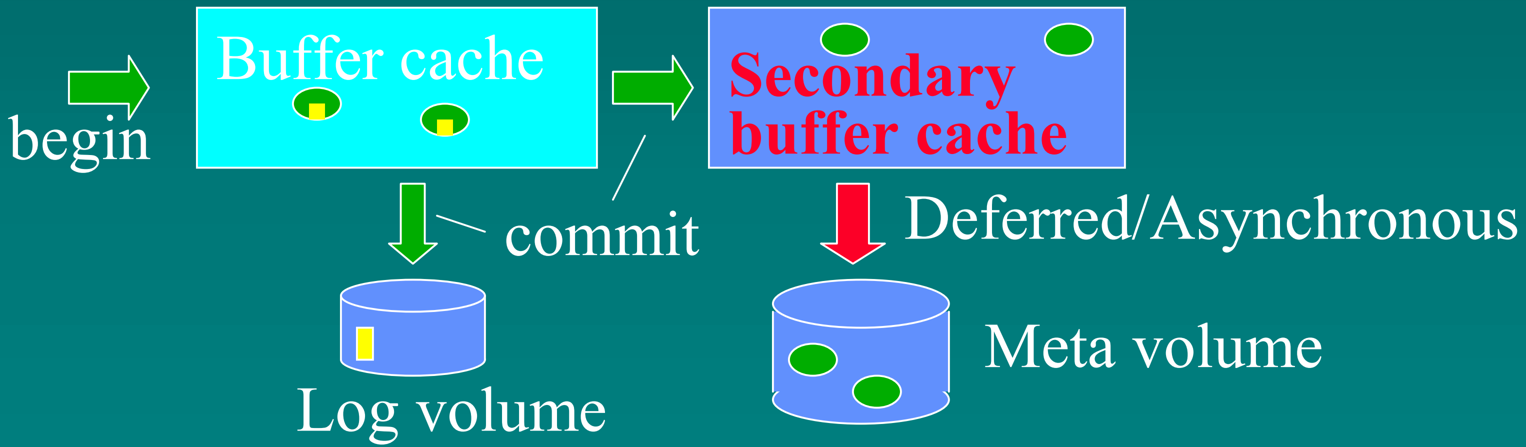


Product Features

- Token Management
  - Fine grain tokens
  - Token escalation.
- Space Reserve Function
  - Contiguous space allocation
  - Minimized communication overhead with Btree.
- Improved Logging
  - Straight-forward development
  - Good performance and availability.

Improved Logging
byte-range-log

- Metadata update is done as an atomic transaction for easier maintenance and improved performance.
- Responds immediately after writing a small log update.
  - Extracts only modified byte-range of data (Byte-range-log)
- Automatic deadlock detection and retry.



Improved Logging
Early Commit

Offsets extra overhead in cluster environments

- Transfers log data to secondary Name Sever instead of writing on dedicated log volume.
  - Write though secondary buffer cache (Secondary Name Server)
- UPS used to protect data from a power failure

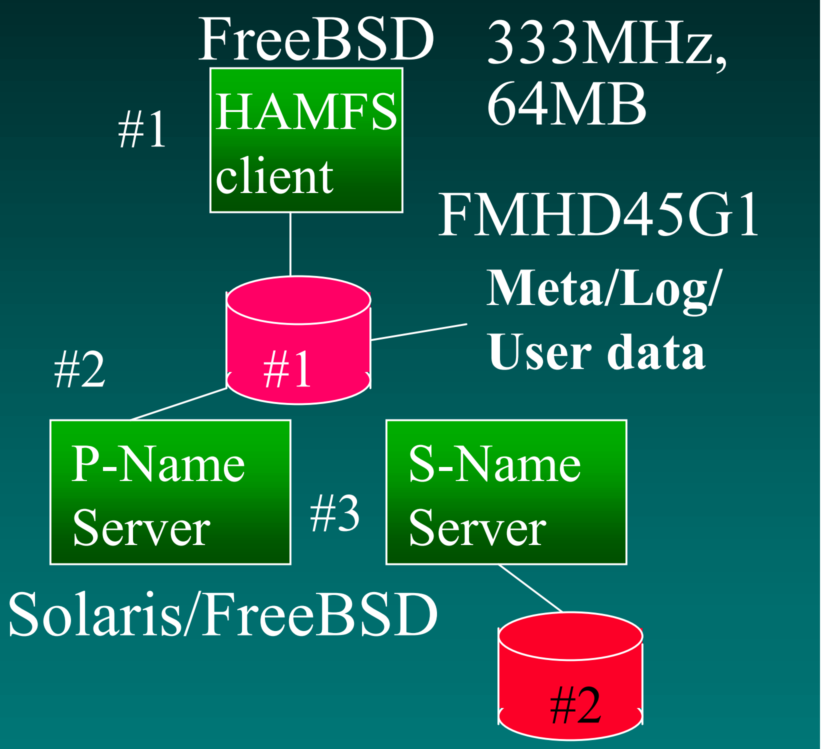
Measurement Methods

- Configuration

  Buffer cache (0.5MB)

  Secondary Buffer cache (1MB)

  100Mbps Ethernet



- Short file access

  Lat_fs program in lmbench (Create 1000 files and delete them all)

- Large file access

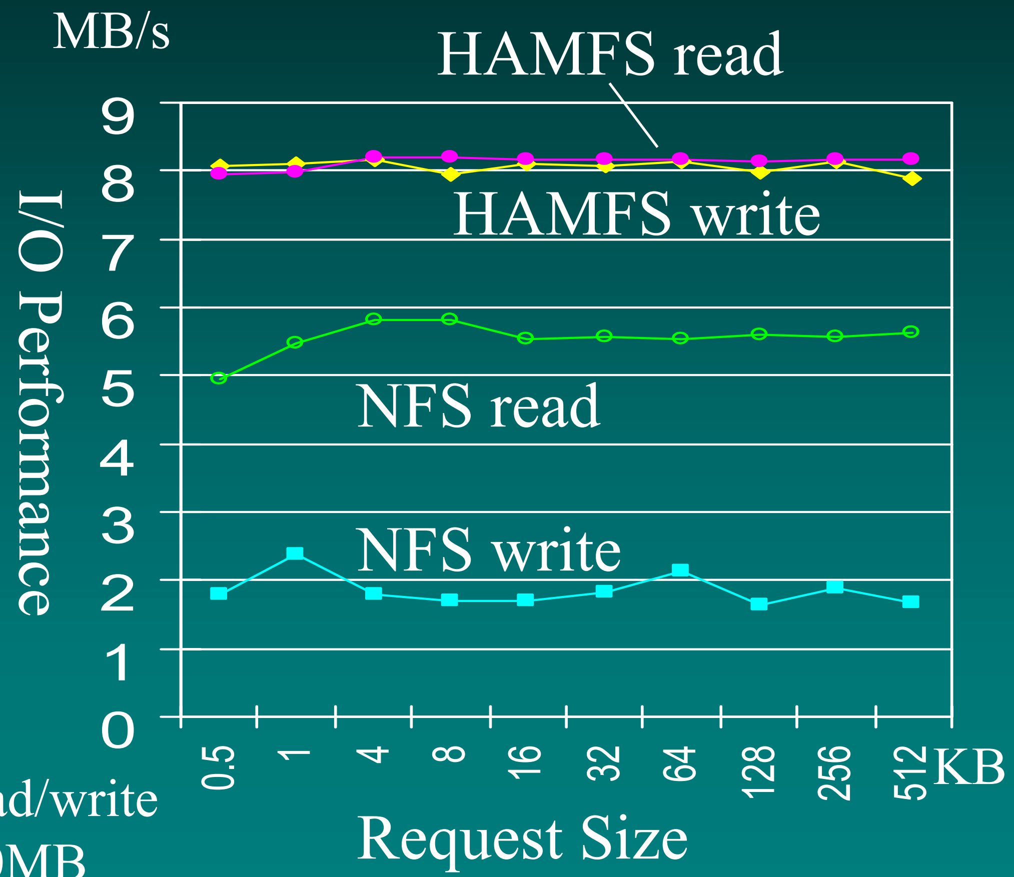
  Create a 100MB file and read it

UFS : PC#1, DISK#1

NFS-V3 : PC#1/2, DISK#1

HAMFS : PC#1/2/3, DISK#1/2

Measurement Results
Large File Access

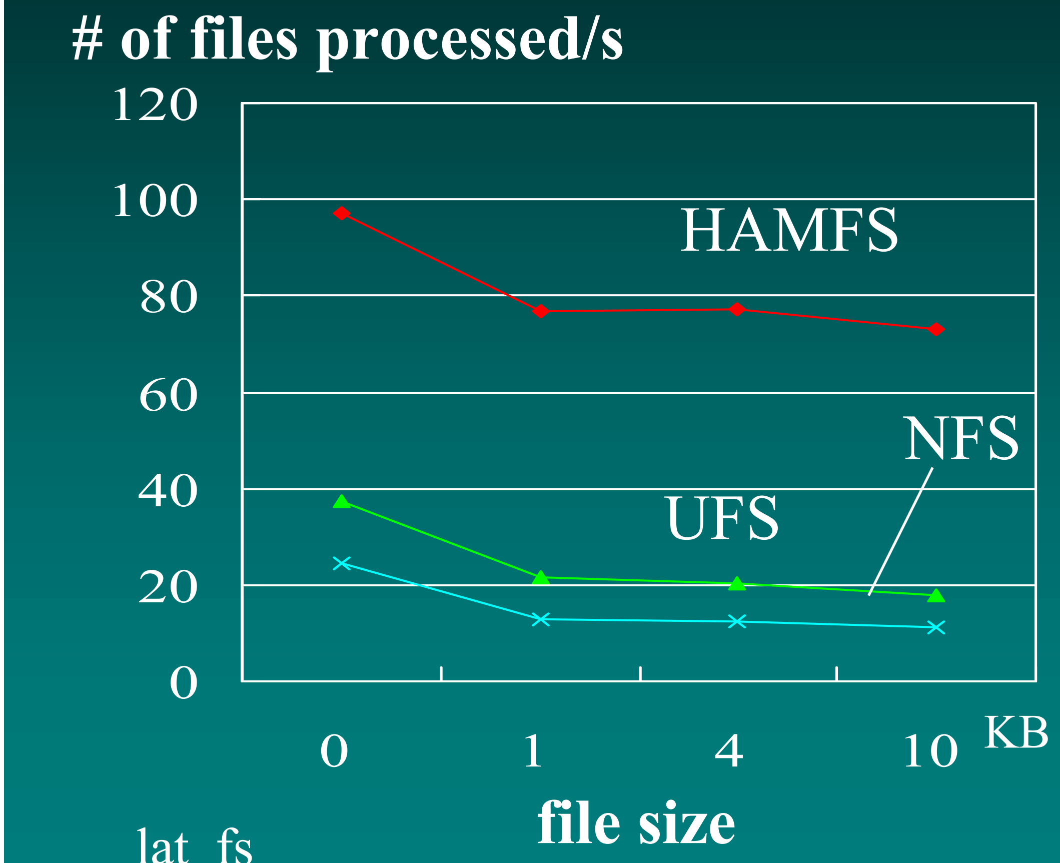


Read/write
100MB

- ASFS derives maximum disk potential
- Superior to SSFS due to -
  - Easier space allocation
  - More efficient caching

Driving the disk with enough data is essential

Measurement Results
Short File Access

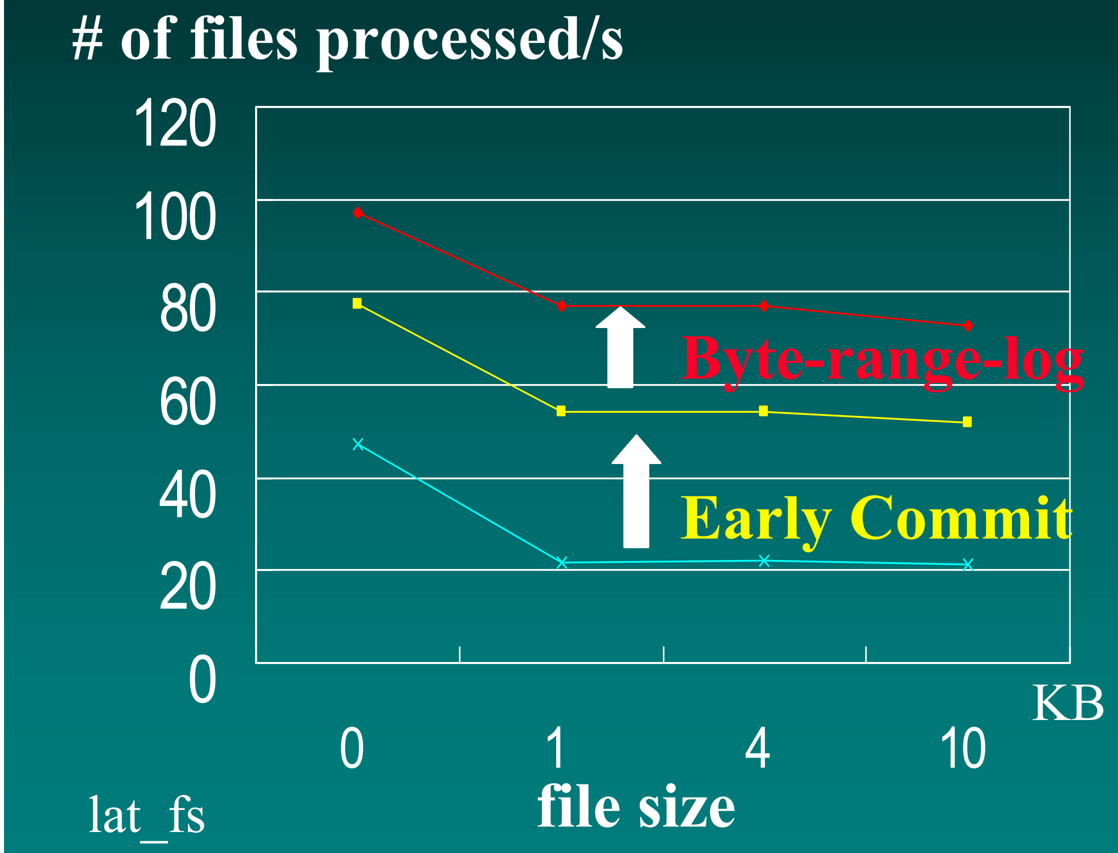


lat_fs

CDFS suffers from greater communication overhead

ASFS can outperform local file systems

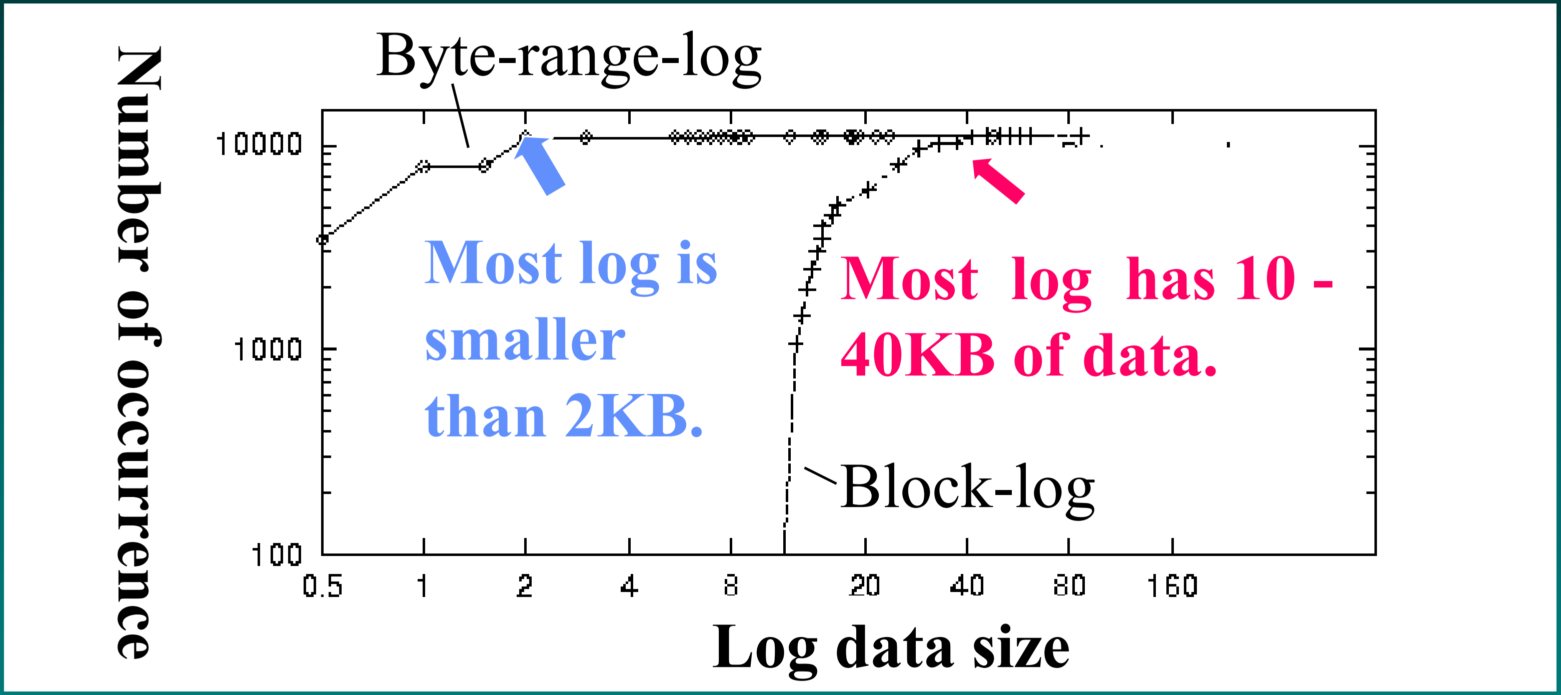
Measurement Results
Effects of Logging



Efficient logging achieves improved performance and availability

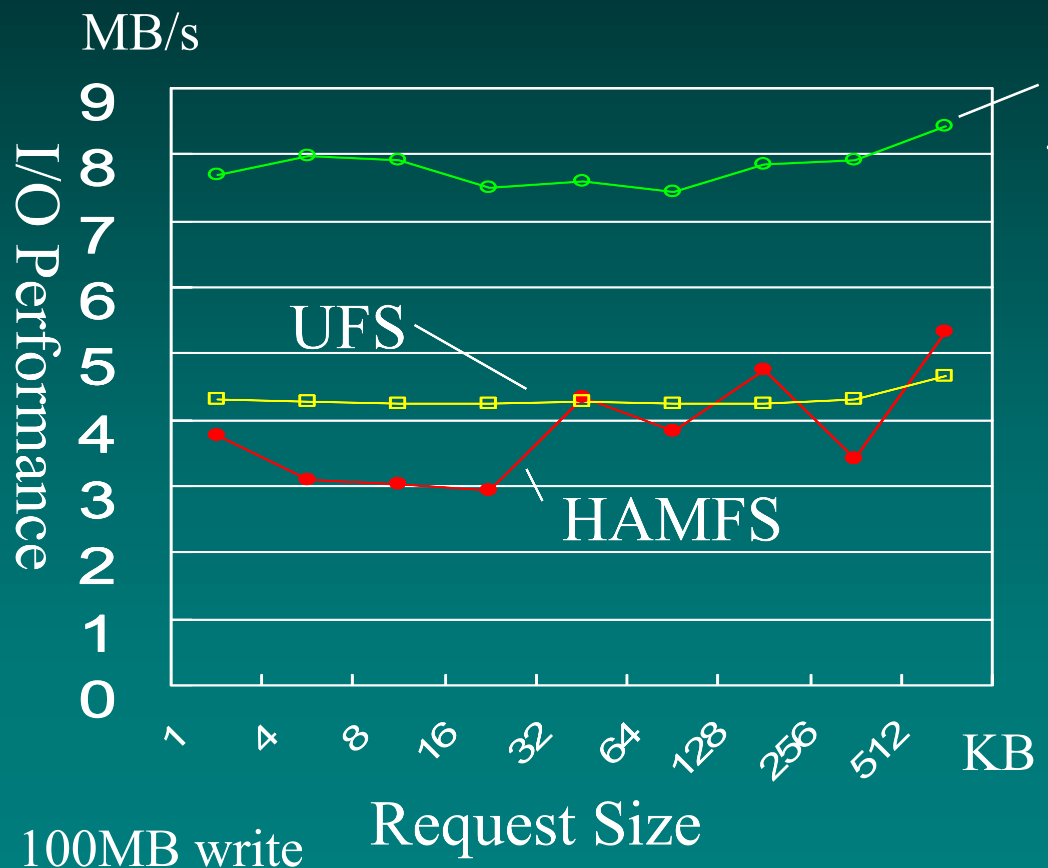
It is difficult for other file systems types to adopt these techniques.

Measurement Results
Log Size Distribution



Byte-range log reduces dramatically the size of log generated

Measurement Results
Shared Environment

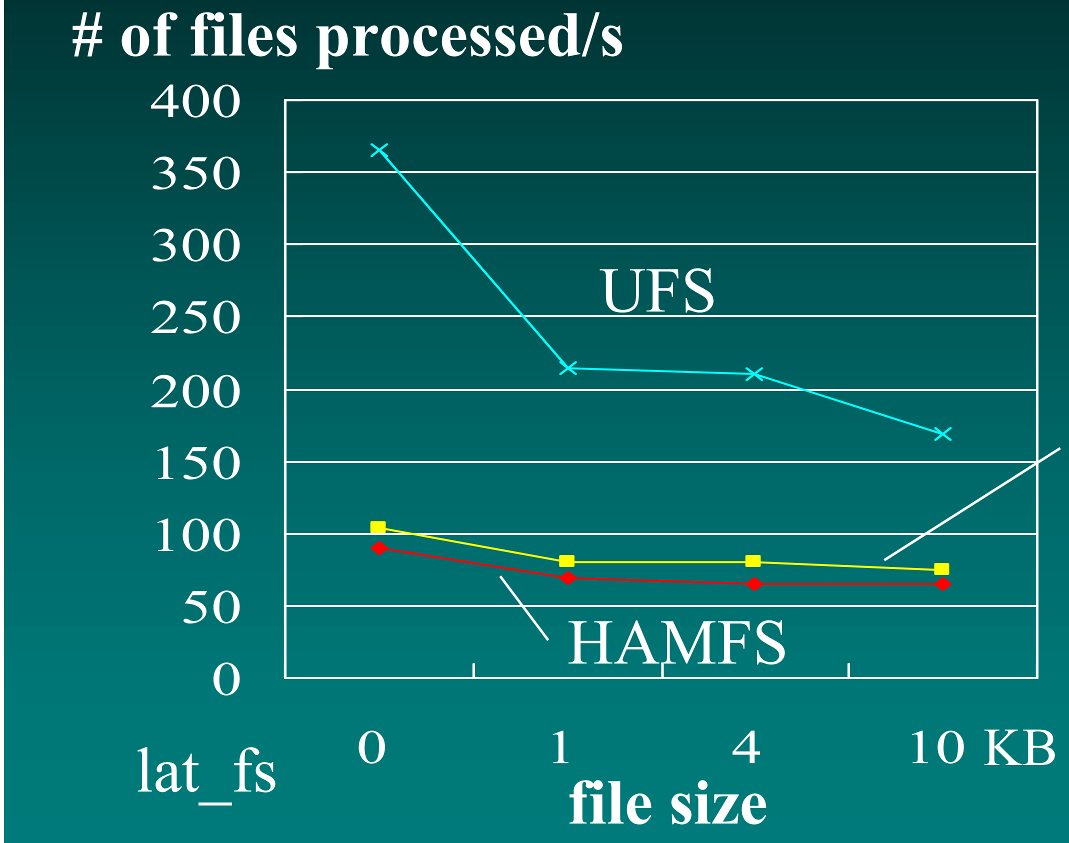


2 UFS partition accessed from a single node

Tag queuing across multiple node is critical

100MB write

Measurement Results
High performance Disks



HAMFS without Early Commit

High speed disk:DF-F350 (FJ-RAID Array)

Reducing communication overhead is important.

Adapting to underlying disk topology is necessary.

Conclusions

Asymmetric Shared File Systems have significant benefits -

- Benefits from new disk technologies (SANs, 4Gbps FC,Ultra-320 SCSI)
- Good performance and availability.
- Easily extensibility and simpler to implement.

But, tag queuing across multiple nodes and dynamic adaptation to underlying disk topology may be required.

Improving future scalability might also be a challenge.