

NFS Extensions for Parallel Storage

* Peter Honeyman

Center for Information Technology Integration

University of Michigan

Ann Arbor

Introductions

- ◆ Organizers
 - ◆ Garth Gibson
 - ◆ Peter Honeyman
- ◆ Local arrangements
 - ◆ Charles Antonelli
 - ◆ Karen Kitchen



Introductions

- ◆ CITI NFSv4 experts
 - ◆ Andy Adamson
 - ◆ Bruce Fields
 - ◆ Jim Rees
- ◆ NFSv4 WG chairs
 - ◆ Brian Pawlowski
 - ◆ Spencer Shepler



Goals

- ◆ Extend NFS to enable seamless, high-performance access to parallel storage
- ◆ Strawman architecture
- ◆ Prototype implementations
- ◆ Standards process
- ◆ Interoperability testing



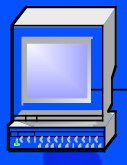
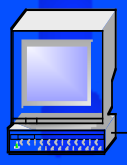
Agenda

- ◆ This morning: presentations
- ◆ This afternoon: discussions
- ◆ This evening: seminars
- ◆ Tomorrow: heir of the dog (@ CITI)
- ◆ This month: comprehensive workshop report will be widely circulated



CITI prototype

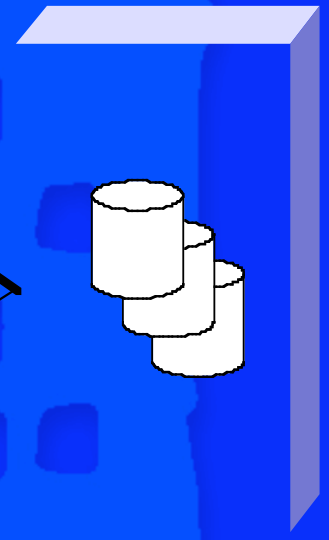
**NFSv4
Clients**



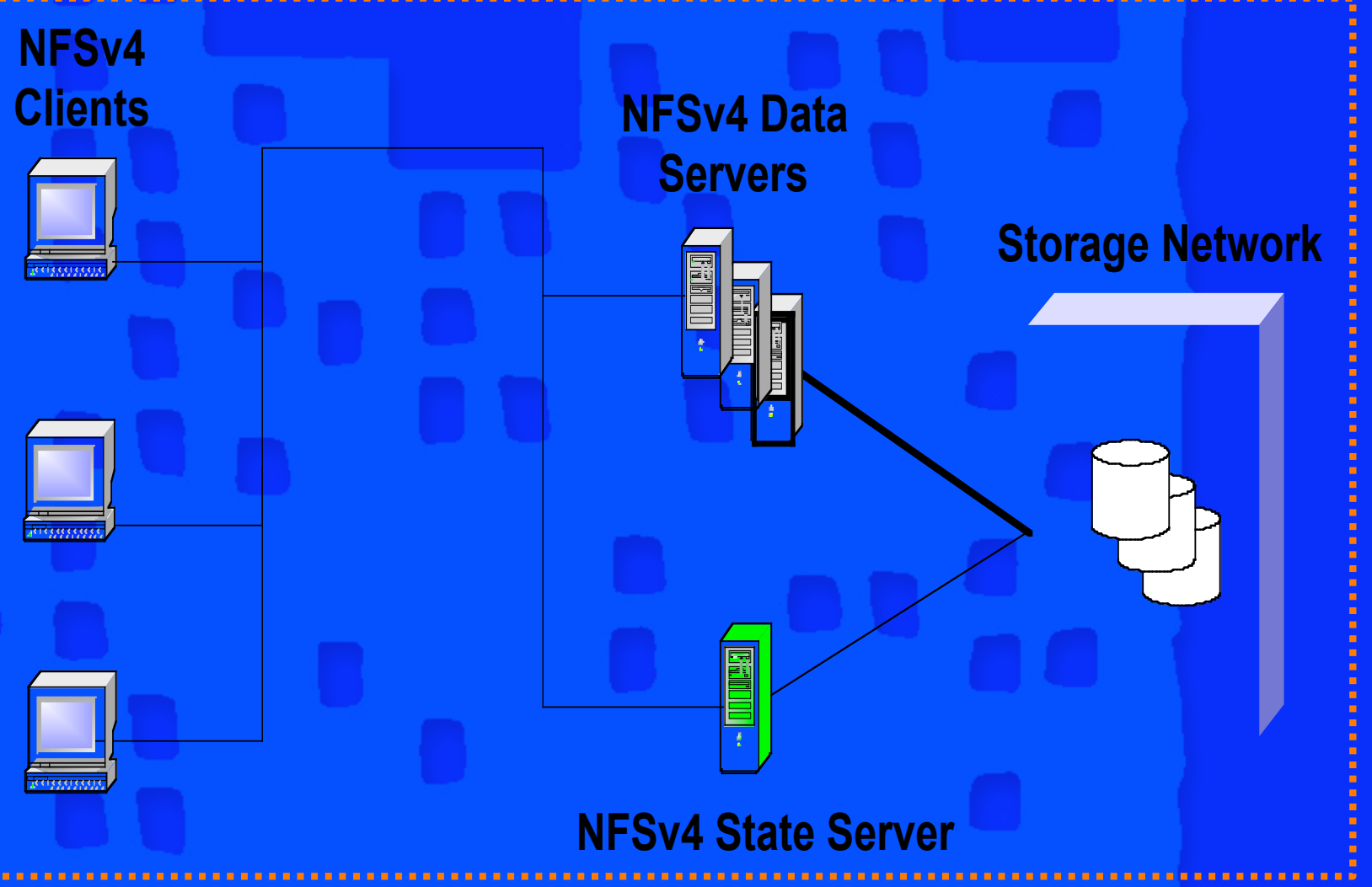
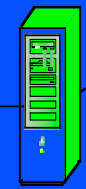
**NFSv4 Data
Servers**



Storage Network



NFSv4 State Server



Lessons learned

- ◆ By recapitulating the mechanism of underlying storage we can not help but run slower than the underlying storage
- ◆ Complex interference patterns depending on implementation details of NFS and underlying storage state management



The state

- Client name
- Client verifier
- Client IP address
- File ID
- State owner ID
- Access bits
- Deny bits
- Lock type
- Start byte
- End byte



Server-to-server protocol

- ◆ DISTRIBUTE_SHARE_STATE_ARGS
- ◆ DISTRIBUTE_LOCK_STATE_ARGS
- ◆ DISTRIBUTE_DELEGATION_STATE_ARGS
- ◆ INVALIDATE_STATE_ARGS



File location

- ◆ FILE_LOCATIONS attribute
 - ◆ List of data servers
 - ◆ Time-to-live parameter
 - ◆ Per data server root pathname
 - ◆ Per data server supported operations
 - ◆ Per data server lease maintenance indicator
- ◆ Extends FS_LOCATIONS



Thank you for your attention! Questions?!

www.citi.umich.edu

