Data-intensive computing is easy
Data-intensive computing is hard

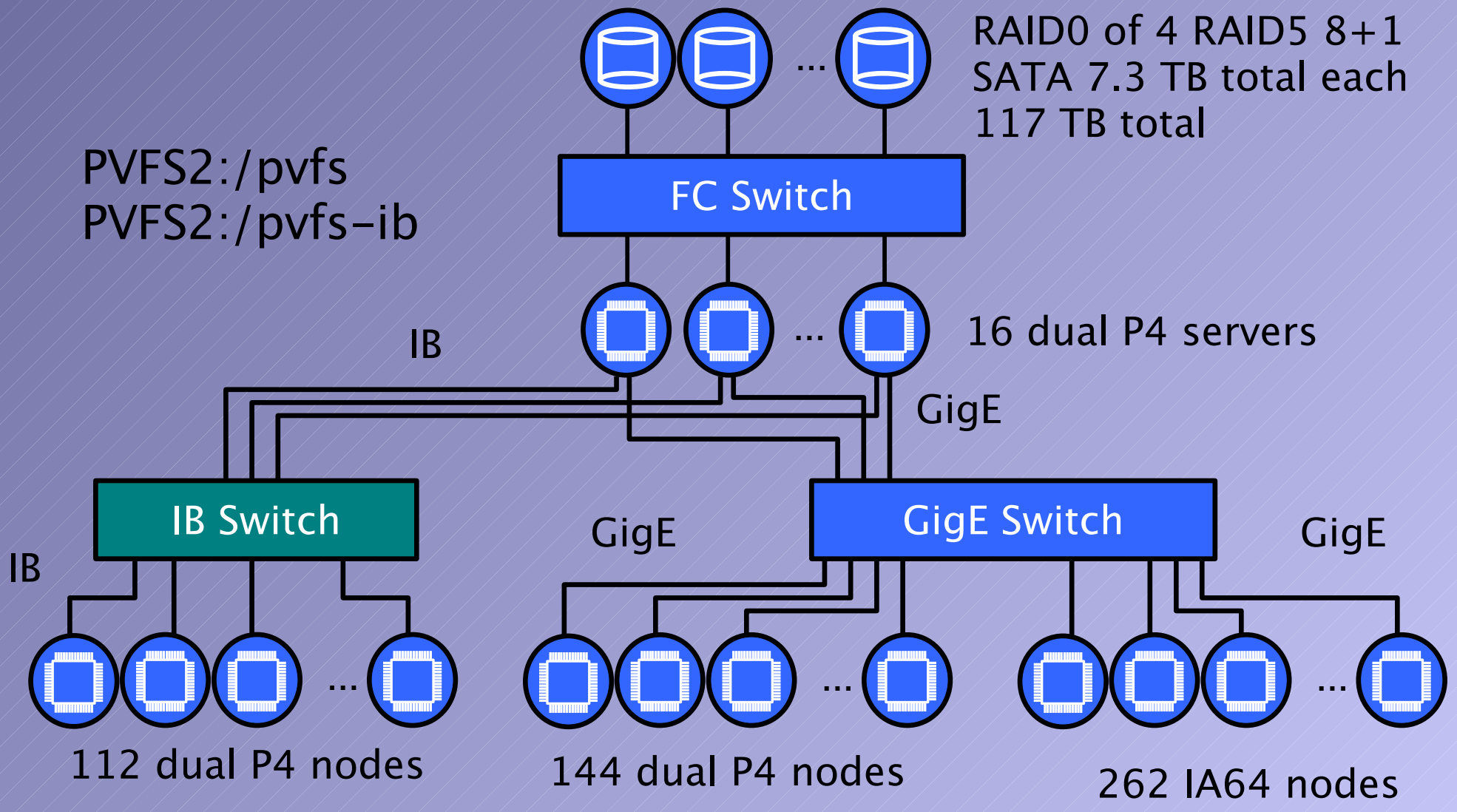
Pete Wyckoff

OSC

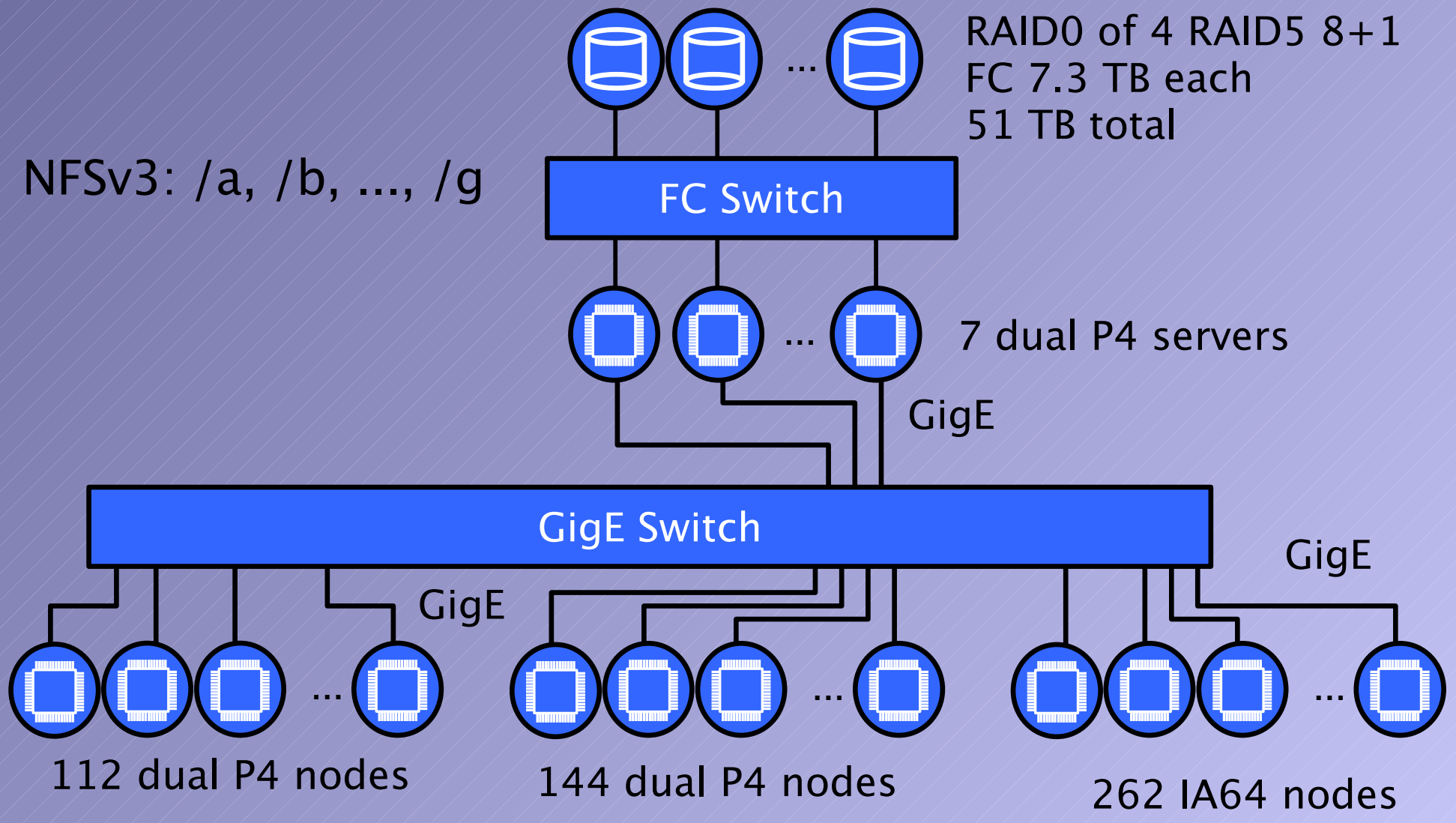
pw@osc.edu

7 Mar 2006

OSC Parallel Storage

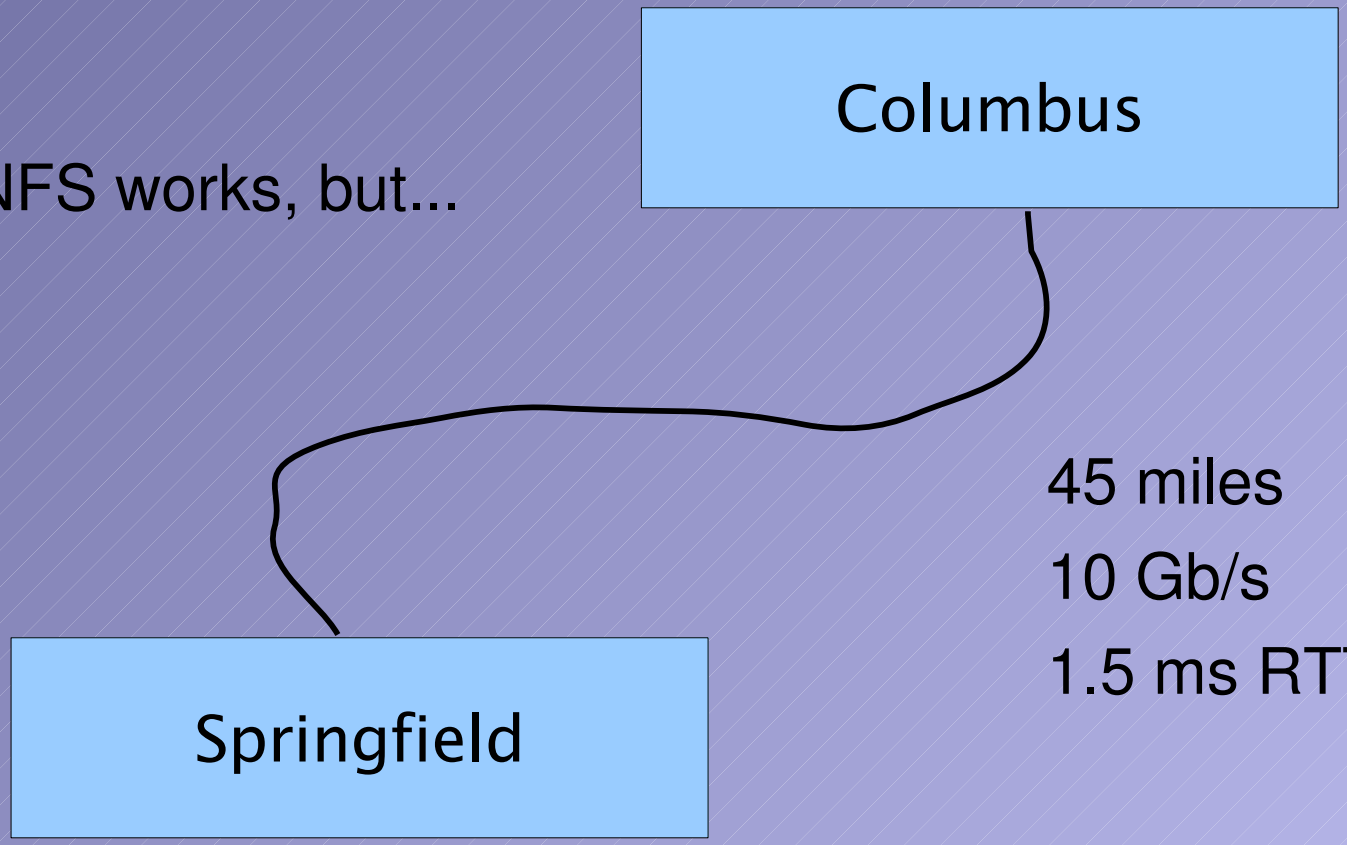


OSC /home Storage



Remote Site

NFS works, but...



45 miles

10 Gb/s

1.5 ms RTT

Access Methods

- Internal
  - MPI to /pvfs
  - MPI to /home (not recommended)
  - POSIX to /pvfs (not recommended)
  - POSIX to /home (frequently discouraged)
  - POSIX to /tmp, cp to /home
- External
  - ssh, sftp
  - globus
  - web portals

Data-intensive computing is easy

as long as you do everything exactly right

- Streaming, big reads or writes
  - lots of disks and hosts, no problem
  - use separate files
  - issue large operations
  - don't mix reads and writes
  - make sure nobody else is using the system
- Use a database
  - if your app is small-data SQL, no problem
- High-speed wide-area
  - also easy if your app is remote copy

Data-intensive computing is hard

- APIs are a mess
  - POSIX: open, close, read, write, seek
  - MPI: rocket scientists only
  - Matlab, python, ...: object-oriented but don't look under the hood
  - HDF5, pnetCDF: OO yet difficult
- API suggestions
  - application specific is most useful
  - but hard to standardize, teach, implement
  - checkpoint is almost working

Even implementing APIs are hard

- Available hardware poorly fits app needs
- Must tune each API for
  - programming language
  - file system (parallel and local)
  - storage network architecture
  - storage servers and disks
  - batch environment

Computing paradigm is wrong

- I/O performance problems are a symptom of the overall disease
- Obsession with data is harmful
- Computing is about orchestrating data motion, not algorithms or science
  - processor-centric designs
  - 4-5 levels of mem cache (write buf, memory buf)
  - disk caching in client, server, controllers, disks
  - explicit cache management instructions (proc, disk)

Good luck “computing” tomorrow

- Still processor centric
- But processors even farther from data
  - multicore unibus
  - cell memory-less procs

- Processing is trivial
- Moving data around is difficult
- Recompute rather than reread
- Build smart storage systems