pNFS

NAS Industry Conference

October, 2004

Brent Welch

welch@panasas.com

panasas

Abstract

- **Scalable I/O problem**
  - 1000's of clients accessing shared storage
- **Asymmetric, Out-of-band solutions offer scalability**
  - Control path (open/close) different from Data Path (read/write)
  - Until now, no standard solutions
- **pNFS extends NFSv4**
  - Minimum extension to allow out-of-band I/O
  - Standards-based scalable I/O solution

panasas

pNFS

- **Extension to NFSv4**
  - NFSv4 is a great base, and it allows extension
  - Fewest additions that enable parallel I/O from clients
  - Avoid entanglements with other features
- **Layouts are the key additional abstraction**
  - Describe where data is located and how it is organized
  - Client I/O operations can bypass the file server
  - Client I/O can access storage devices in parallel (data striping)
- **Generalized support for asymmetric, out-of-band solutions**
  - Files: Clean way to do filer virtualization, eliminate botteneck
  - Objects: Standard way to do object-based file systems
  - Blocks: Standard way to do block-based SAN file systems

Scalable I/O Problem

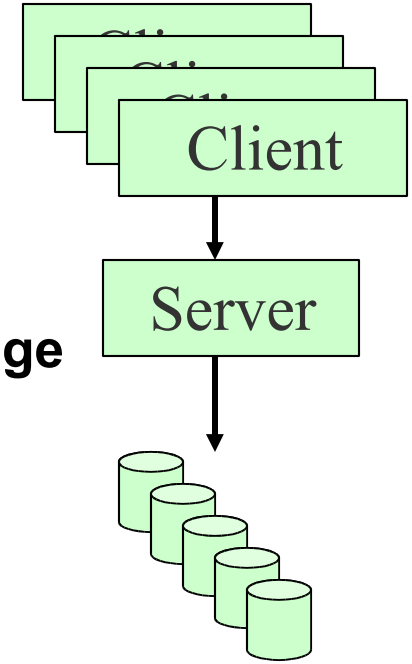
- **Storage for 1000's of active clients => lots of bandwidth**
- **Scaling capacity through 100's of TB and into PB**
- **File Server model has good sharing and manageability**
  - but it is hard to scale
- **Many other proprietary solutions**
  - GPFS, CXFS, StorNext, Panasas, Sustina, Sanergy, …
  - Everyone has their own client
  - Like to have a standards based solution => pNFS

Scaling and the Client

- **Gary Grider's rule of thumb for HPC**
  - 1 Gbyte/sec for each Teraflop of computing power
  - 2000 3.2 GHz processors => 6TF => 6 GB/sec
  - One file server with 48 GE NICs? I don't think so.
  - 100 GB/sec I/O system in '08 or '09 for 100 TF cluster
- **Making movies**
  - 1000 node rendering farm, plus 100's of desktops at night
- **Oil and Gas**
  - 100's to 1000's of clients
  - Lots of large files (10's of GB to TB each)
- **EDA, Compile Farms, Life Sciences …**
  - Everyone has a Linux cluster these days

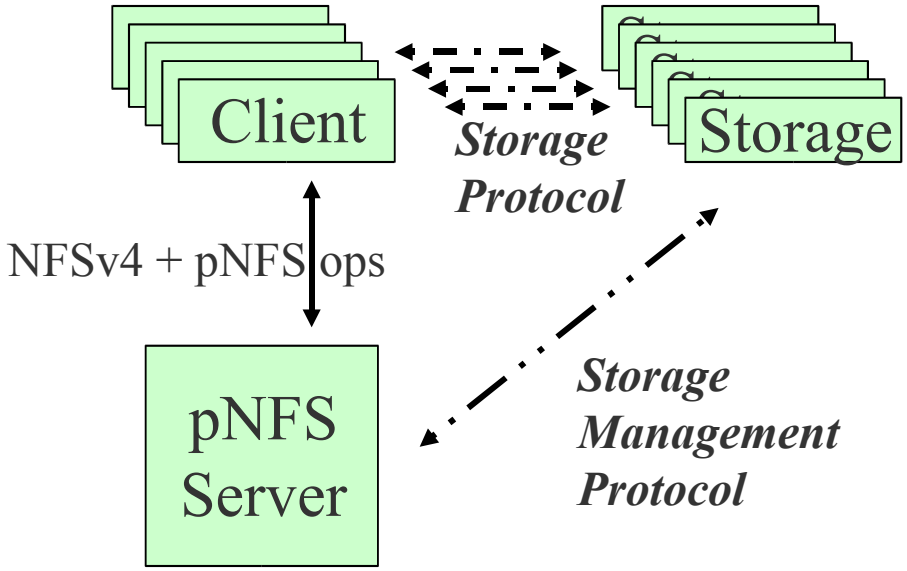
Scaling and the Server

- **Tension between sharing and throughput**
  - File server provides semantics, including sharing
  - Direct attach I/O provides throughput, no sharing
- **File server is a bottleneck between clients and storage**
  - Pressure to make server ever faster and more expensive
  - Clustered NAS solutions, e.g., Spinnaker
- **SAN filesystems provide sharing and direct access**
  - Asymmetric, out-of-band system with distinct control and data paths
  - Proprietary solutions, vendor-specific clients
  - Physical security model, which we'd like to improve



Asymmetric File Systems

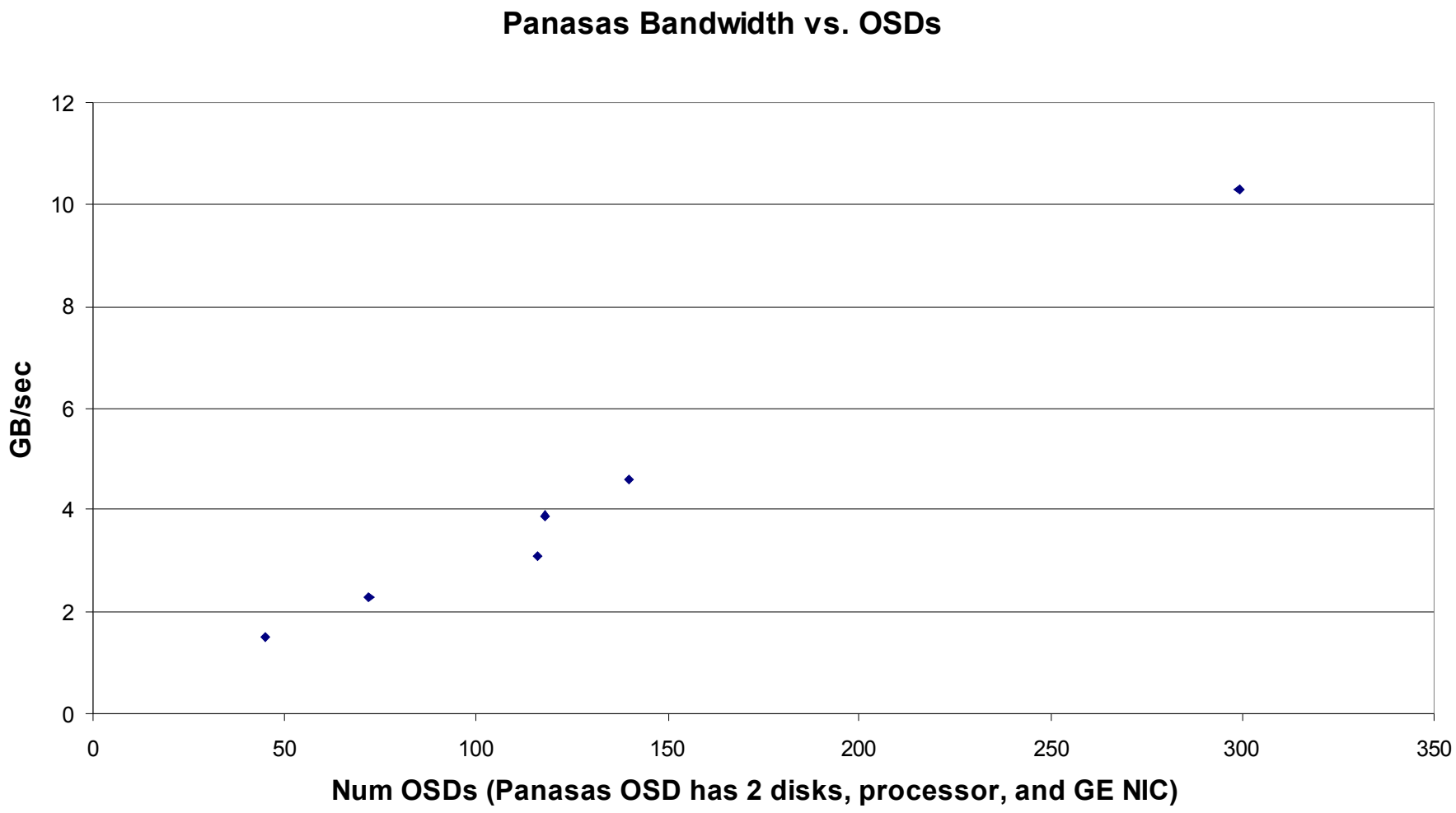
- **Control Path vs. Data Path ("out-of-band" control)**
- **Metadata servers (Control)**
  - File name space
  - Access control (ACL checking)
  - Sharing and coordination
- **Storage Devices (Data)**
  - Clients access storage directly
- **SAN Filesystems**
  - CXFS, EMC Hi Road, Sanergy, SAN FS
- **Object Storage File Systems**
  - Panasas, Lustre



“Out-of-band” Value Proposition

- **Out-of-band allows a client to use more than one storage address for a given file, directory or closely linked set of files**
  - **Parallel I/O direct from client to multiple storage devices**
- **Scalable capacity: file/dir uses space on all storage: can get big**
- **Capacity balancing: file/dir uses space on all storage: evenly**
- **Load balancing: dynamic access to file/dir over all storage: evenly**
- **Scalable bandwidth: dynamic access to file/dir over all storage: big**
- **Lower latency under load: no bottleneck developing deep queues**
- **Cost-effectiveness at scale: use streamlined storage servers**
- **pNFS standard leads to standard client SW: share client support $$$**

Scalable Bandwidth



panasas

pNFS

- **Extension to NFSv4**
  - NFS is THE file system standard
  - Fewest additions that enable parallel I/O from clients
- **Layouts are the key additional abstraction**
  - Describe where data is located and how it is organized
  - Clients access storage directly, in parallel
- **Generalized support for asymmetric solutions**
  - Files: Clean way to do filer virtualization, eliminate botteneck
  - Objects: Standard way to do object-based file systems
  - Blocks: Standard way to do block-based SAN file systems

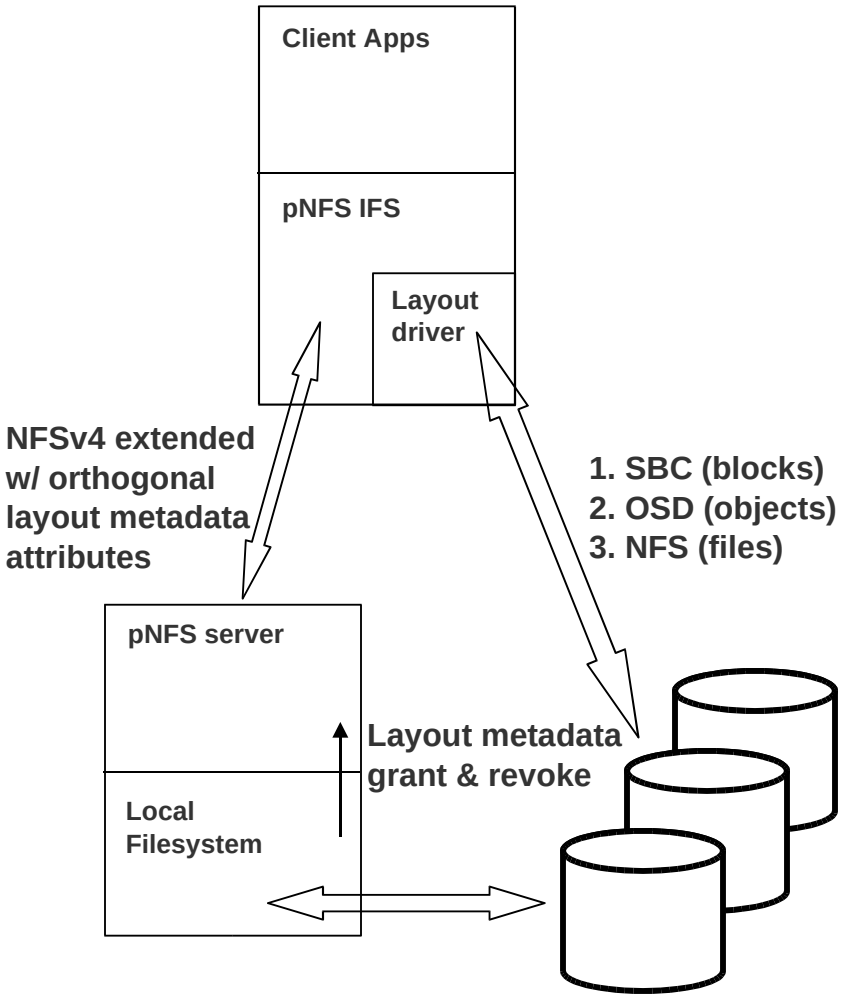
panasas

pNFS Ops Summary

- **GETDEVINFO**
  - Maps from opaque device ID used in layout data structures to the storage protocol type and necessary addressing information for that device
- **LAYOUTGET**
  - Fetch location and access control information (i.e., capabilities)
- **LAYOUTCOMMIT**
  - Commit write activity. New file size and attributes visible on storage.
- **LAYOUTRELEASE**
  - Give up lease on the layout
- **CB_LAYOUTRETURN**
  - Server callback to recall layout lease

Multiple Data Server Protocols

- **BE INCLUSIVE !!**
  - Broaden the market reach
- **Three (or more) flavors of out-of-band metadata attributes:**
  - BLOCKS:
    SBC/FCP/FC or SBC/iSCSI… for files built on blocks
  - OBJECTS:
    OSD/iSCSI/TCP/IP/GE for files built on objects
  - FILES:
    NFS/ONCRPC/TCP/IP/GE for files built on subfiles
- **Inode-level encapsulation in server and client code**



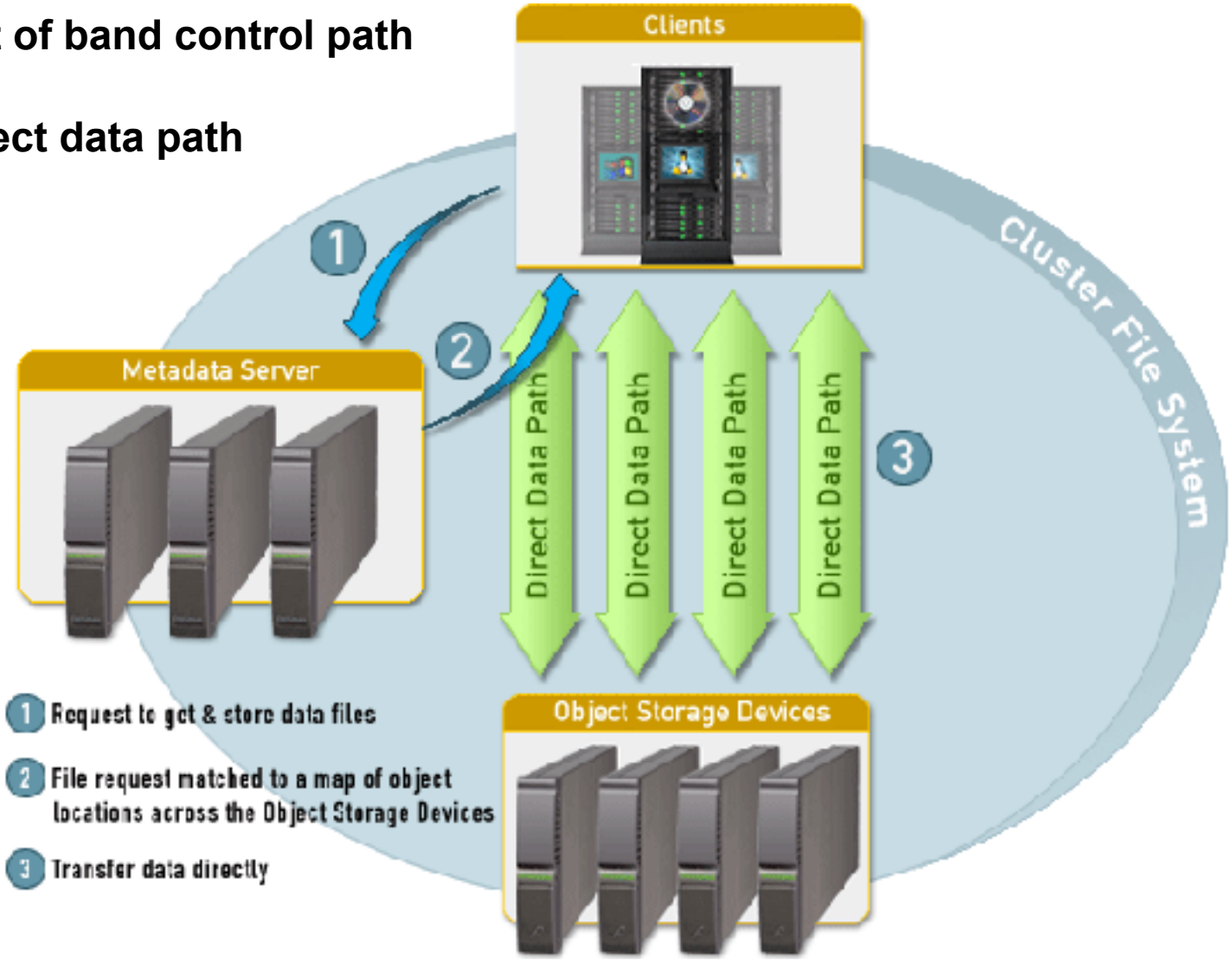
panasas

Object Storage

- **Object interface is midway between files and blocks**
  - Create, Delete, Read, Write, GetAttr, SetAttr, …
  - Objects have numeric ID, not pathnames
- **Clean security model based on shared, secret device keys**
  - Metadata manager generates capabilities
  - Clients present capabilities with each operation
  - Object Storage Device (OSD) checks capability on each access
  - <object id, data range, operation(s), expire time, cap version> signed with device key
- **Based on NASD and OBSD research out of CMU (Gibson et. al)**
- **SNIA T10 standards based. V1 complete, V2 in progress.**

Object Storage File System

- Out of band control path
- Direct data path



panasas

Status

- **pNFS ad-hoc working group**
  - Dec ’03 Ann Arbor, April ’04 FAST, Aug ’04 IETF, Sept ’04 Pittsburgh
- **IETF**
  - Initial pitch at Seoul ’04
  - Planned addition to NFSv4 charter, D.C. ’04 in November
- **RFC**
  - draft-gibson-pnfs-problem-statement-01.txt July 2004
  - Requirements RFC for November
  - Ops RFC for November

panasas

Backup

panasas

Symmetric File Systems

- **Distribute storage among all the clients**
  - GPFS (AIX), GFS, PVFS (User Level)
- **Reliability Issues**
  - Compute nodes less reliable because of the disk
  - Storage less reliable, unless replication schemes employed
- **Scalability Issues**
  - Stealing cycles from clients, which have other work to do
- **Coupling of computing and storage**
  - Like early days of engineering workstations, private storage