Network-Attached Secure Disks (NASD)

Garth Gibson, Carnegie Mellon Univ.

http://ww.cs.cmu.edu/afs/cs/project/nasd/WWW/NASD.html
http://www.cs.cmu.edu/Web/Groups/PDL/
garth+@cs.cmu.edu

Eliminate file server performance bottlenecks

Eliminate (parallel) client file system bottlenecks

Shore up network-attached storage security

Executive Summary

Scalable storage based on aggregation

- switched networks and mesh interconnects
- storage striping

Traditional file server is a bottleneck

- stop & copy forwarding from peripheral net to LAN

Network-attached storage removes bottleneck

- transfers involve only client, network, storage

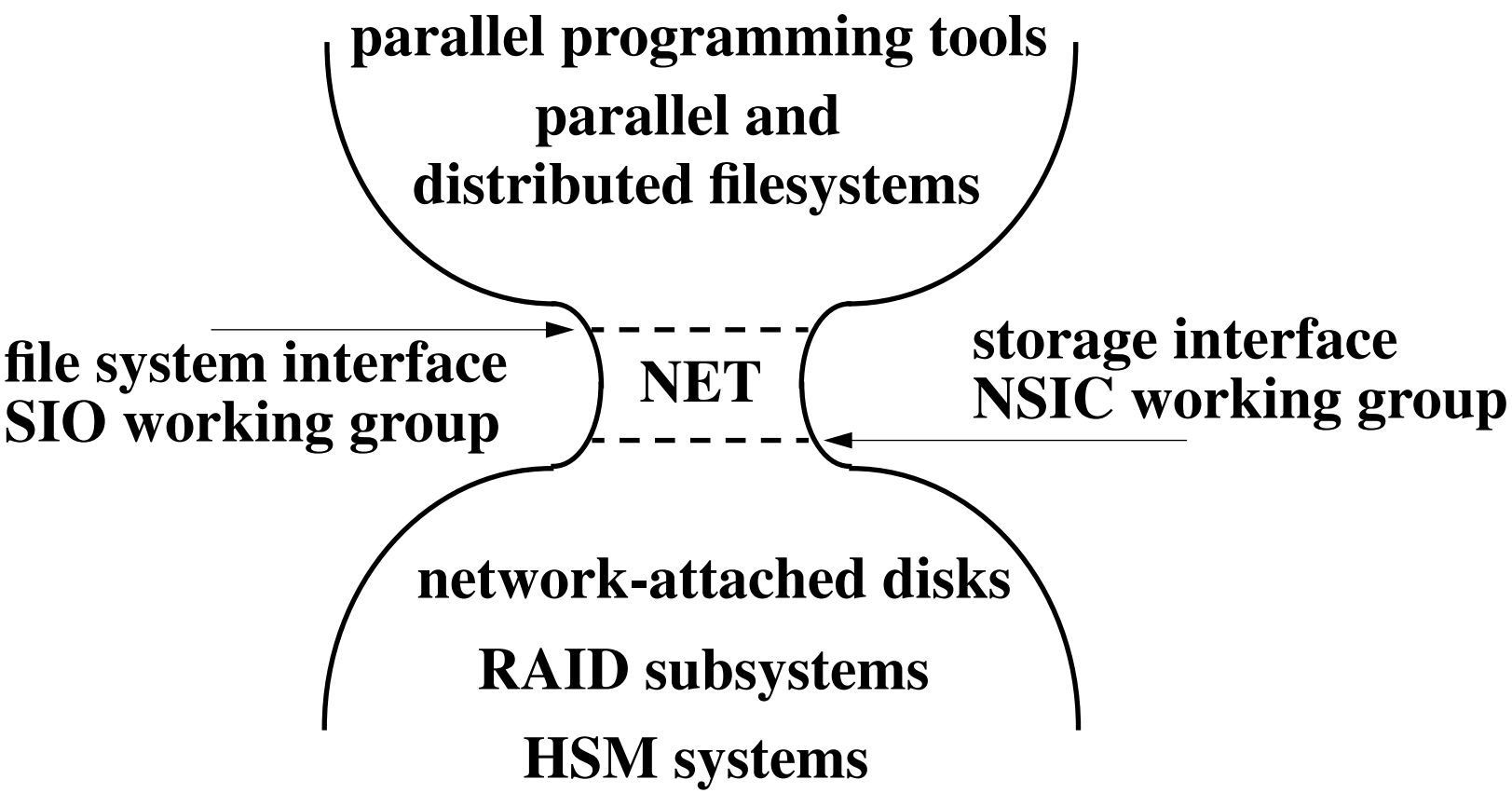
Client file system API has bottlenecks too

- SIO parallel FS low-level API eliminates copying

NASD exposed to insecure network

- NASD participates in LAN authentication protocol

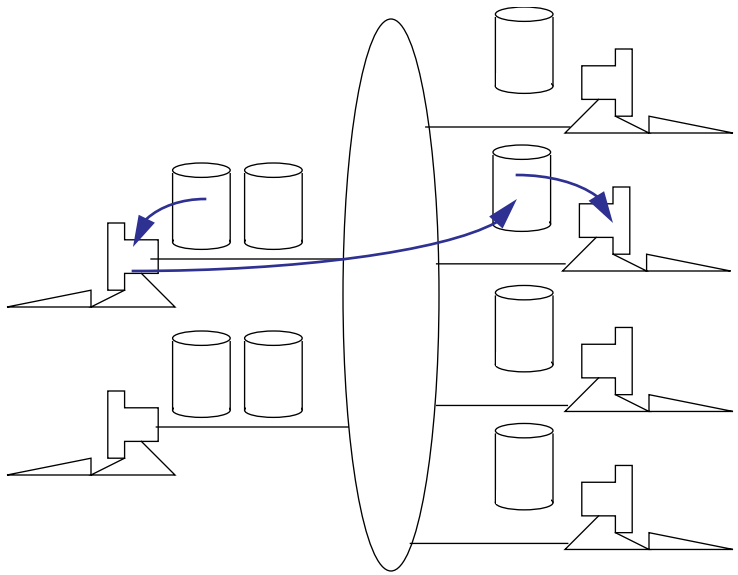
Broad Strategy



Trends: Data bytes travel over LANs

Workstation/compute node a poor & costly server

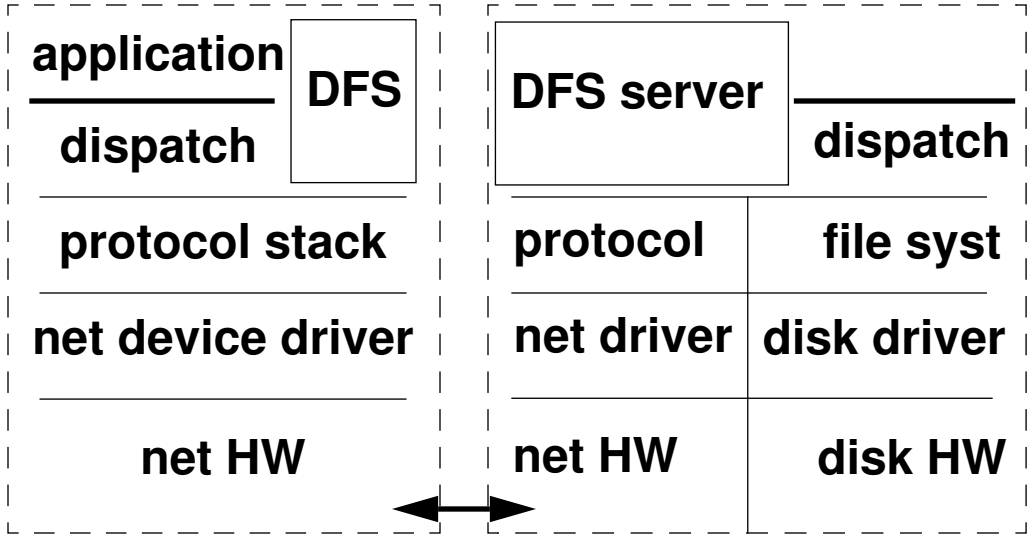
- designed around caching for processor-local work
- induces extra copying
- network bandwidth often close to single disk bandwidth



Today's Distributed File System Abstraction

Client-server partition of local file system

- scale # clients/server
- global file naming
- "efficient" bulk transfer
- partial failure modes, disconnected operation
- expensive state mgmt: atomicity, consistency, accounting, administration



Trends: Storage "hardware" evolving rapidly

Drive data rate rising rapidly (~20% per year)

SCSI physical interconnect evolves to packet LAN

- Firewire, SSA, Fibrechannel

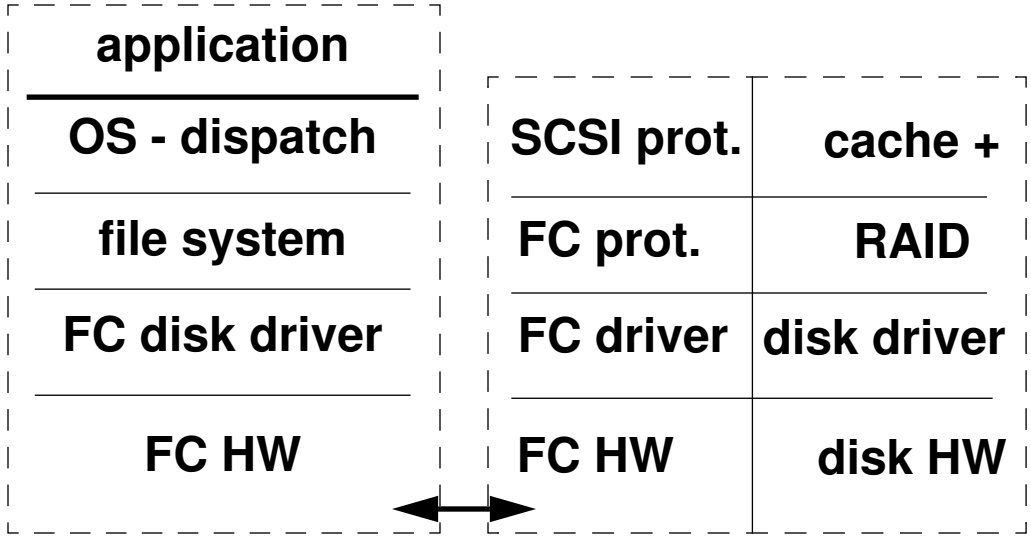
Fast-moving, profit-tight marketplace

- Compete with value-added intelligence for performance
- Exploit VLSI for more local cycles and bytes
- Geometry sensitive disk scheduling, readahead/writebehind
- RAID support to offload parity update computation
- Cost of managing storage per year 3-7X storage cost
- Dynamic mapping for transparent optimizations

Local Storage System is Partitioned too

Local RAID disk more than disk arm & R/W head

- e.g. Fibrechannel
- e.g. RAID subsystem embeds capabilities of workstation computer
- packetized SCSI is transparent to file system
- "cache +" includes dynamic remapping, compression, etc.

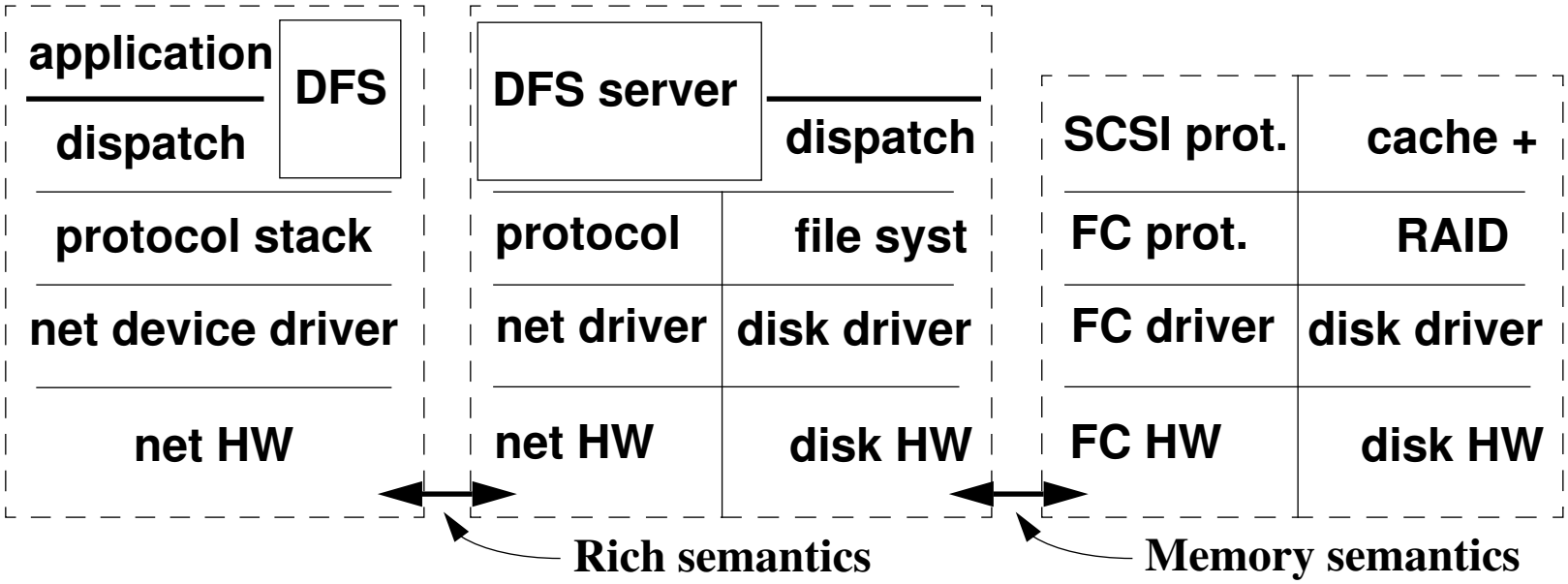


Storage System Structure Revisited

Storage subsystem limited by file server

- higher bandwidth
- better cost-performance
- interoperates more widely

Too many protocol layers, too much copying

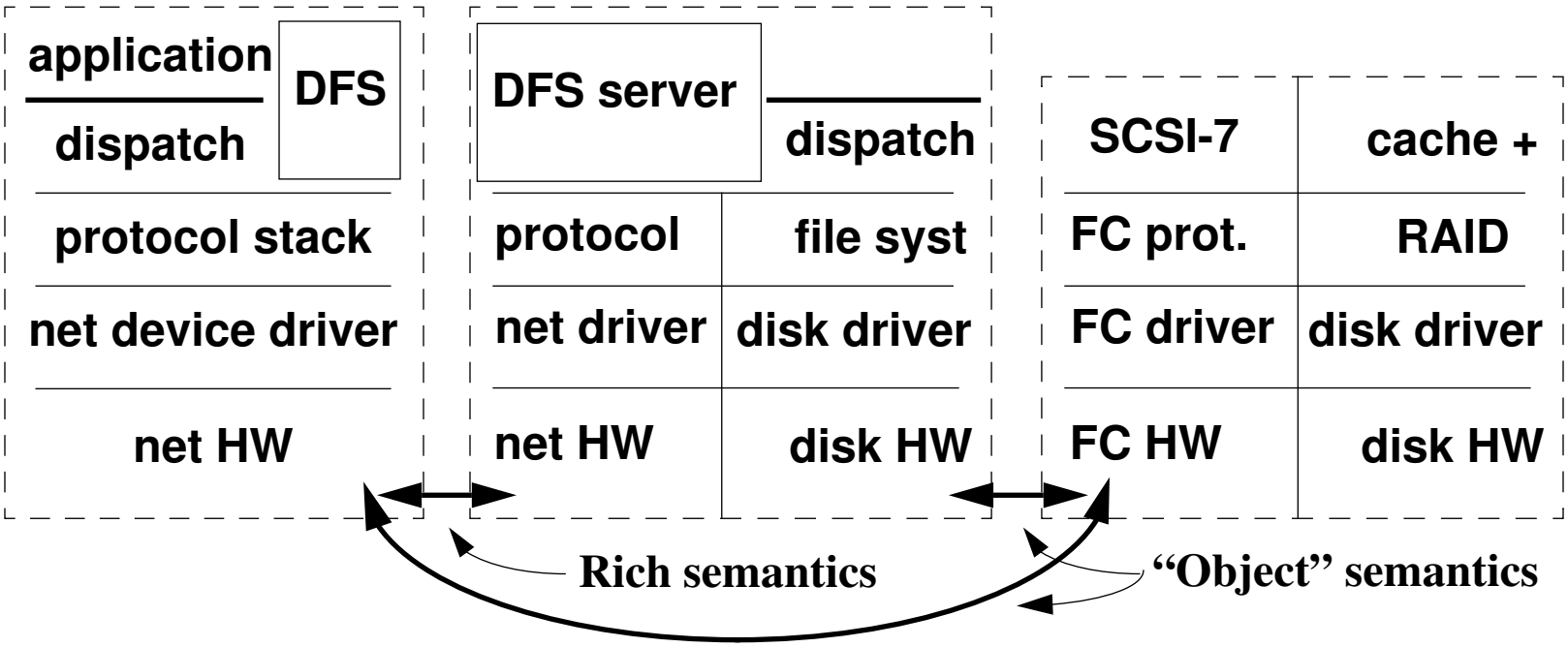


Network-Attached Storage System Structure

Fastpath common operations (read, write, open)

Raise storage subsystem interface to objects

Network "DMA" (copy, scatter/gather)



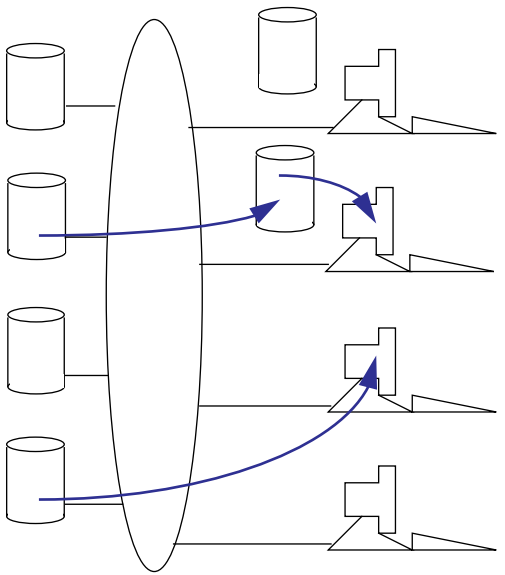
DFS, NASD Functional Partition

Distributed file system remaining functions

- names interpretation and administration
- concurrency and consistency semantics
- access control interpretation and accounting

NASD function

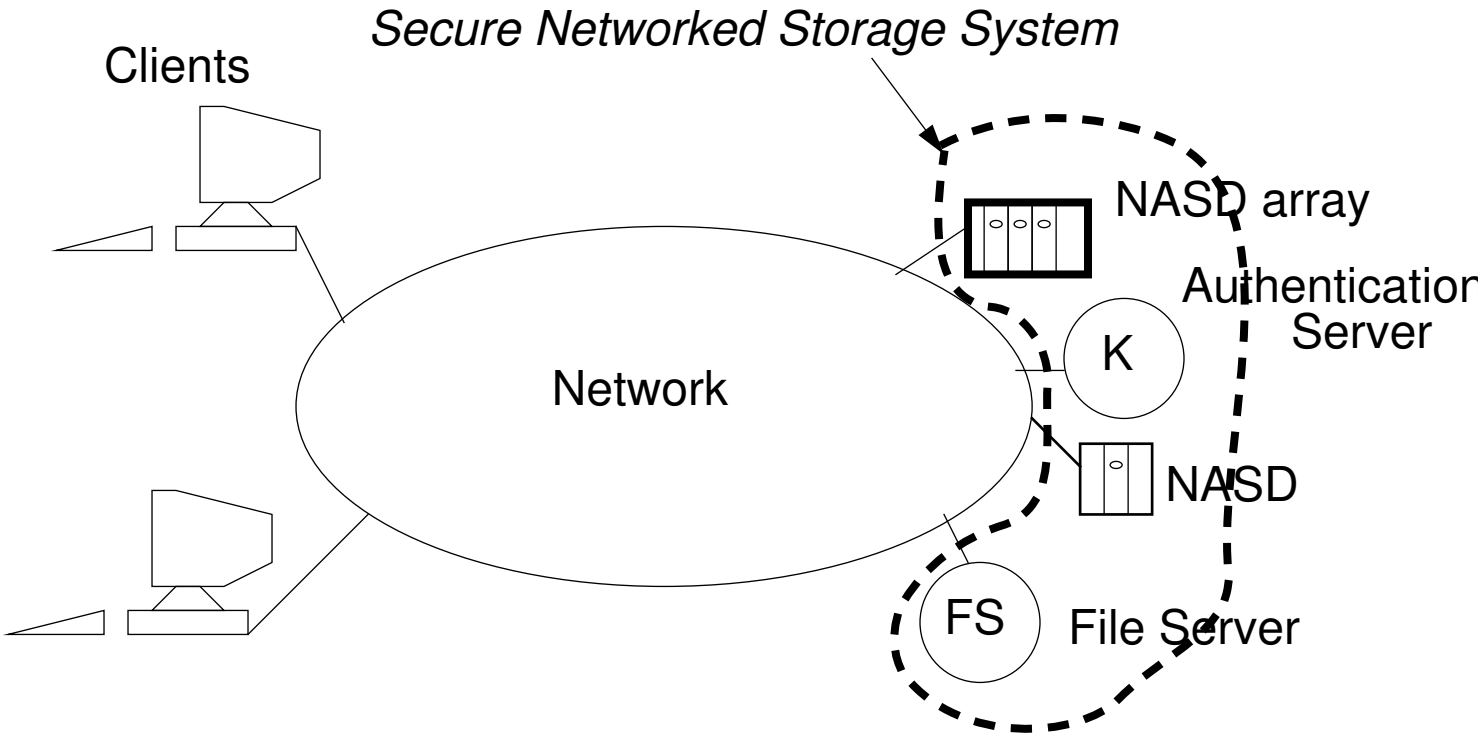
- object storage metadata
- common command processing
- efficient network transfer
- access control enforcement
- storage value-added



NASD Security Model

Arbitrary packets must not be seen as commands

- **private key cryptography; kerberos standard**
- **tamper-resistant encryption chip on drive**



NASD Security Protocol Goals

Security goals:

1) No unauthorized modification of data

2) No unauthorized disclosure of data

3) No unauthorized allocation of resources

4) Limit threat caused by a compromised machine

• Performance goals:

- Minimize number of messages
- Direct request/reply between client and drive

• Flexibility goals:

- Protocol unaware of filesystem implementation

How to Exploit NASD Interface

For existing distributed file systems

- reduce file server load improves cost-performance, scalability
- fixed client FS API limits application benefit

For new parallel file systems

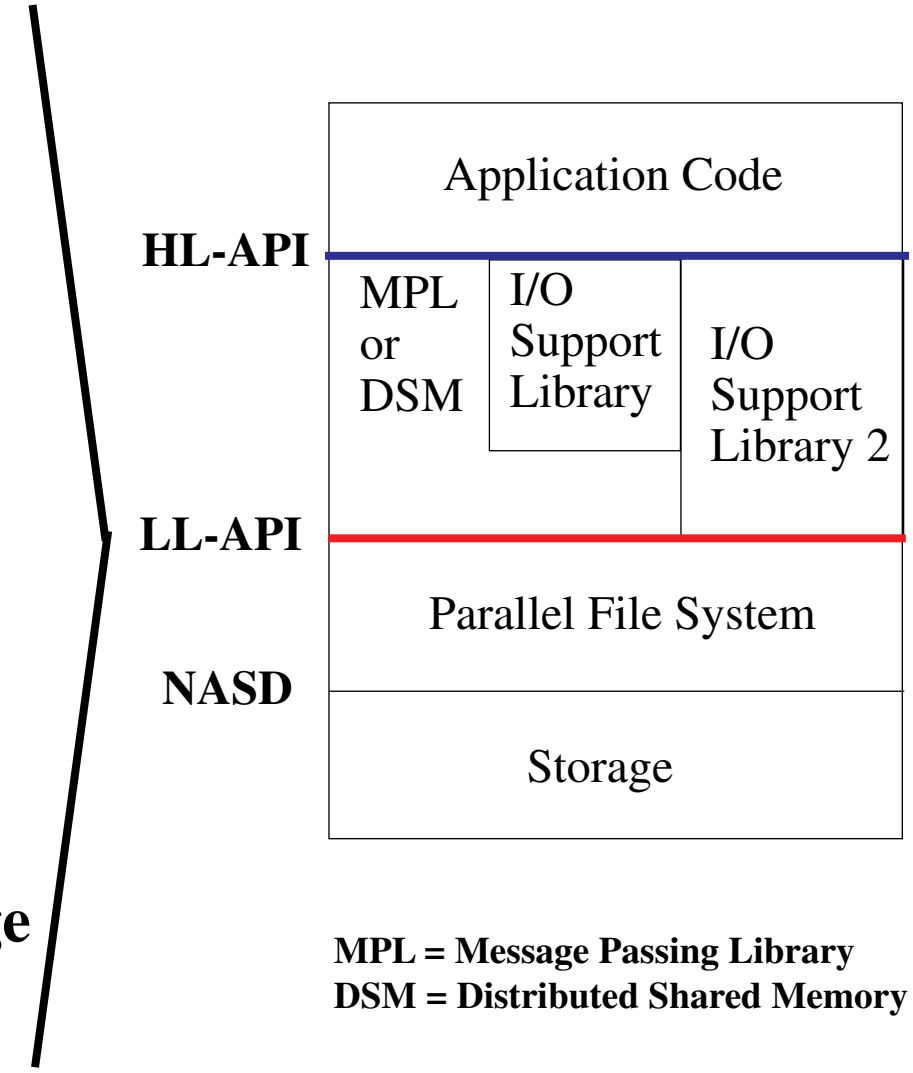
- PFS structures/APIs still developing
- Scalable I/O developing new high-performance standards

SIO focus on low-level API design for client

- portable, flexible, minimal number of functions
- minimal constraints on parallelism, efficiency
- applicable to MPP and LAN-based multicomputer

SIO Parallel File System Low Level API

- **Scatter/Gather**
- **Copy Avoidance**
- **Collective Transfer**
- **Copy-on-write**
- **Client cache control**
- **Hints to/from storage**



MPL = Message Passing Library
DSM = Distributed Shared Memory

Exploiting NASD: Network Support for Parallel Flows

Switch-style scalable storage transfers in multiple streams

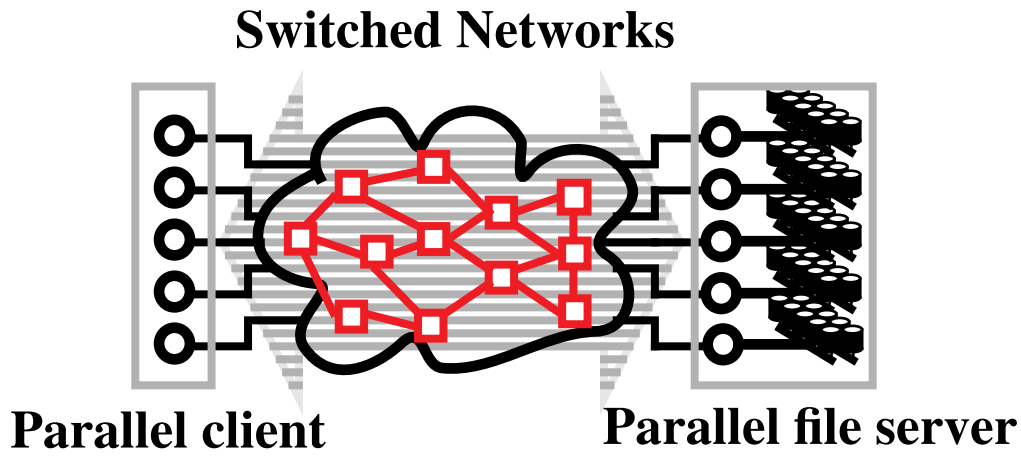
- but networks deal in individual connections

Network support for coordinated routing of multiple streams

- multi-path connections, source routing, load-sensitive

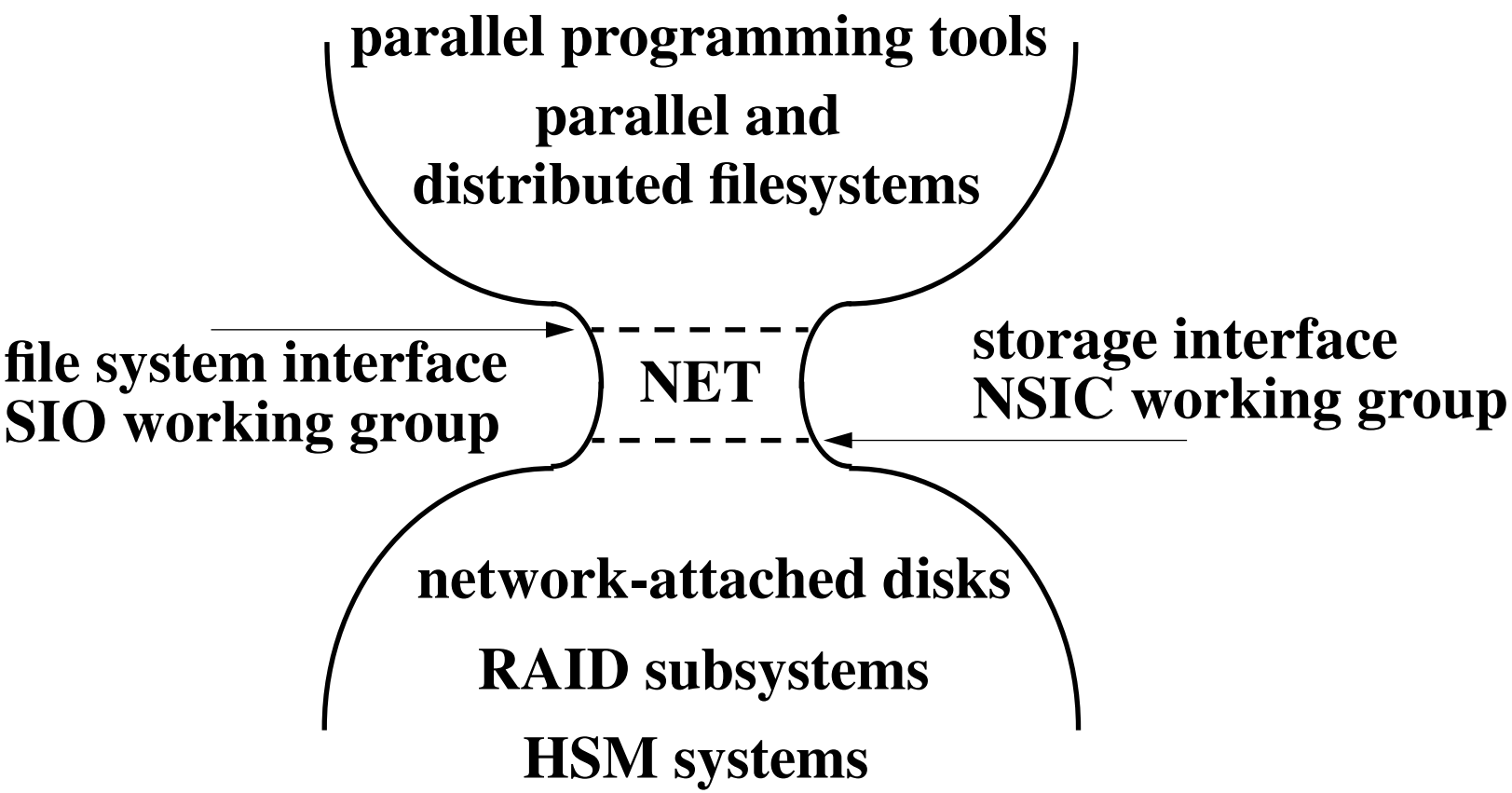
API for negotiating parallel flow service

- enable applications to adapt to bandwidth availability



Summary

Network-attached Secure Disks (NASD)

- **“first class network citizens”**
- **direct data transfer to client**
- **eliminate workstation file server bottleneck**
- **raise drive interface to object storage system**
- **integrate with authentication-based security**
- **repartition file system between client and drive**
- **support multiple client filesystem personalities**
- **network support for storage striped transfers**
- **cost-constrained, transfer-oriented drive architecture**

Points of Leverage