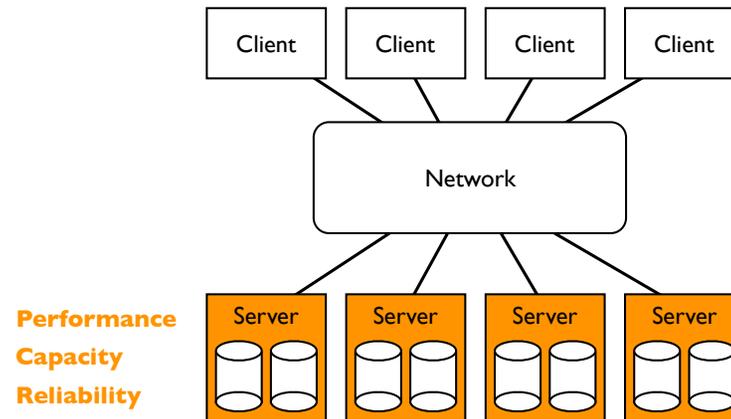


# Wisconsin HaRD

Andrea C. Arpaci-Dusseau  
Remzi H. Arpaci-Dusseau  
University of Wisconsin, Madison

# Currently: DOS

The Era of DOS: Disk-Only Storage



# And then came Flash

NGST: Next Great Storage Technology

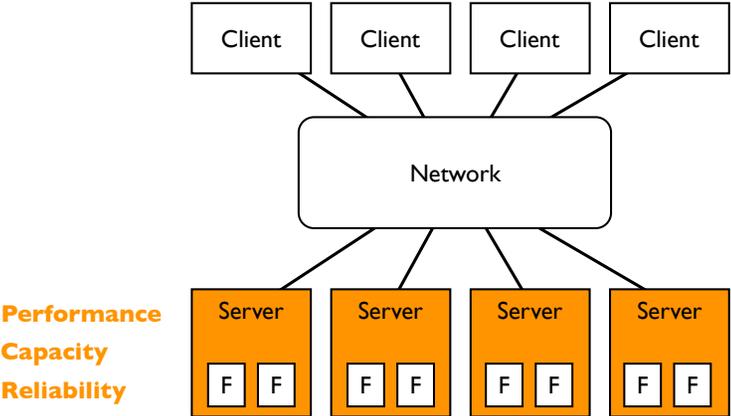
But this time, for real

- Found market in laptops, portable electronics, etc.
- Costs came down....



# Whither Flash?

FOSL? (Flash-Only Storage Layer?)



# FOSL: Why not?

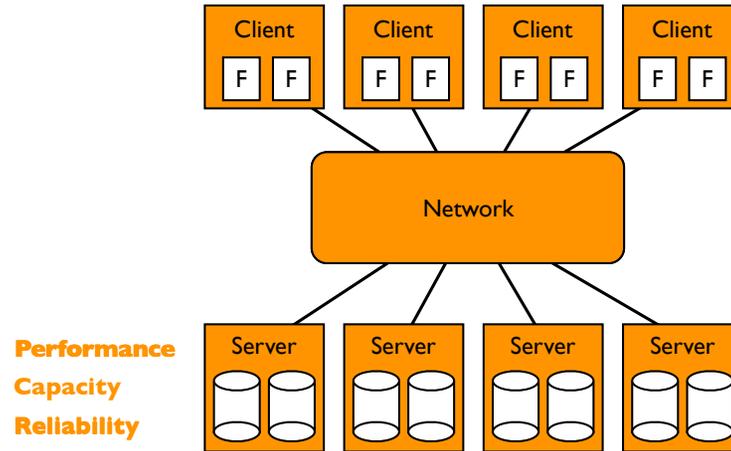
Server-side disk replacement?

Reasons against:

- Network latency & queueing
- Capacity loss / cost increase
- And it doesn't work too well [Narayanan '09]

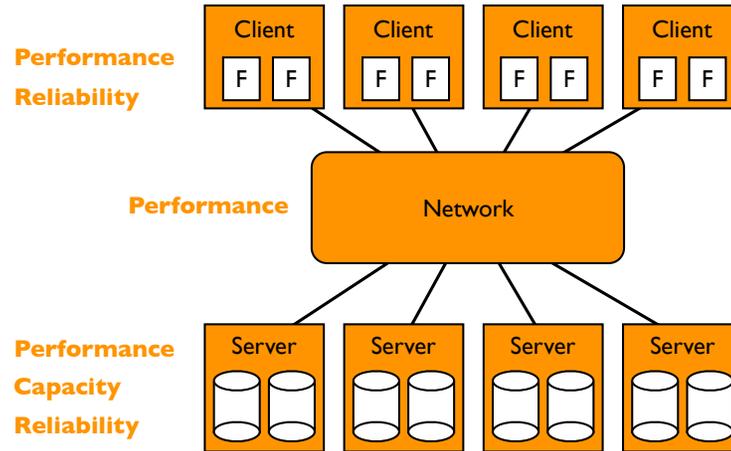
# Alternative: Client flash

Client-side flash [Kleiman '08]



# Alternative: Client flash

Client-side flash [Kleiman '08]



# Wisconsin **HaRD**

**H**ierarchically-**R**edundant, **D**ecoupled Storage

Why hierarchically redundant?

- Copies across both clients and servers

Why decoupled?

- Performance demands disentanglement of clients and servers

# HaRD Focus: **Performance**

Performance, Performance, Performance

- Most relevant to HEC
- Without it, what is the point?
- Plenty of problems to solve...

Reliability, Capacity: Left for future, others?

# Outline

Application Trends

HaRD performance issues

- Client-side
- Server-side
- Network

Conclusions and RFC

# Application Trends: Then

When DFS's developed:

- 10s to 100s of clients, talking to few servers
- Workloads: Desktop-based
- Result: NFS, AFS, CIFS, etc.
- Focus: Cache consistency, directory hierarchy, etc.

# Application Trends: Now

Not just desktop apps anymore...

- Data processing (e.g., MapReduce)
- Parallel scientific (checkpointing, etc.)
- “Cloud” (e.g., virtual machine images)
- Photo storage (e.g., facebook)

Sometimes, a mismatch ...

# Application Trends: Implications

No “one size fits all” solution

- Must be tailored to workload, environment

Addition of flash ala HaRD: Opportunity

- Re-architecting a chance to build a more flexible storage substrate

# Outline

Application Trends

HaRD performance issues

- Client-side
- Server-side
- Network

Conclusions and RFC

# Client-side Issues

## Flash-based RAID

- Wear-leveling may lead to non-independent failure modes
- How to rethink RAID in light of this?

# Client-side Issues

## File system integration

- Hard to deploy new file systems
- FUSE is a good basis, but slow?
- How to speed up/harden FUSE?

# Server-side Issues

## In-memory metadata: Costly

- e.g., one pointer per block does not scale
- How to transform (inefficient) disk metadata into (efficient) in-memory form?
- When can reorganizing on-disk data help?

# Network Issues

## Exposed control

- Current traffic: Implicit  
(can lead to bursts & performance loss)
- HaRD traffic: Explicit data flow  
(expose control of data movement)

## Also...

- Orchestrated transfers
- Bulk-oriented protocols

# Putting it all together: Prototypes

By hand

- Parallel checkpointing FS
- MapReduce FS
- Photostore

Vision: Towards “storage automation” ...

# Educational Focus: Undergrads

Wisconsin resource: Undergraduates

- #1 in producing Fortune 500 CEOs  
(tied with Harvard) [Felicelli '08]

Undergrad systems hacking lab

- Weekly meeting of 10-20 selected undergrads
- Each focused on individual/team projects

# Conclusions

Wisconsin HaRD: Client-side flash integration

- Client, server, and network performance
- Reliability, capacity for future...

Work has begun:

- Simulation, prototyping
- + seeking other funds!

Mail to [remzi@cs.wisc.edu](mailto:remzi@cs.wisc.edu) with comments