Petal: Distributed Virtual Disks

Systems Research Center
Digital Equipment Corporation

Edward K. Lee
Chandramohan A. Thekkath

3/11/97

Motivation

- Large-scale storage systems are expensive to manage.
- In 1994, $50B spent on storage hardware, but $150B spent to manage storage.
- Labor costs estimated at $2 to $7 per megabyte per year.
- Storage hardware costs expected to decrease faster than storage management costs.

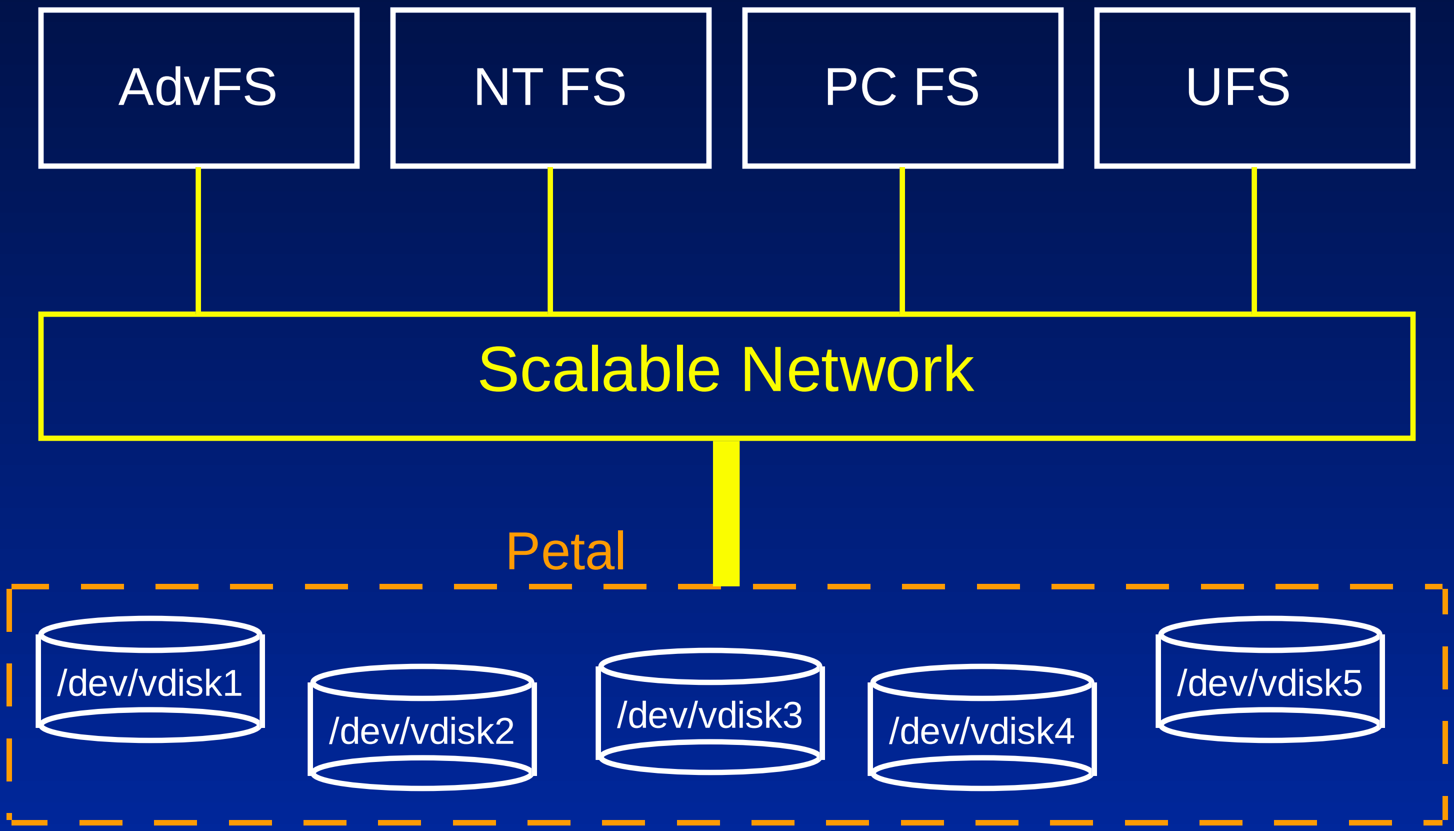
Existing Solutions

- Each controller or storage server appears as a separate storage system to a system administrator.
- Often, low-level components of the system must also be managed.

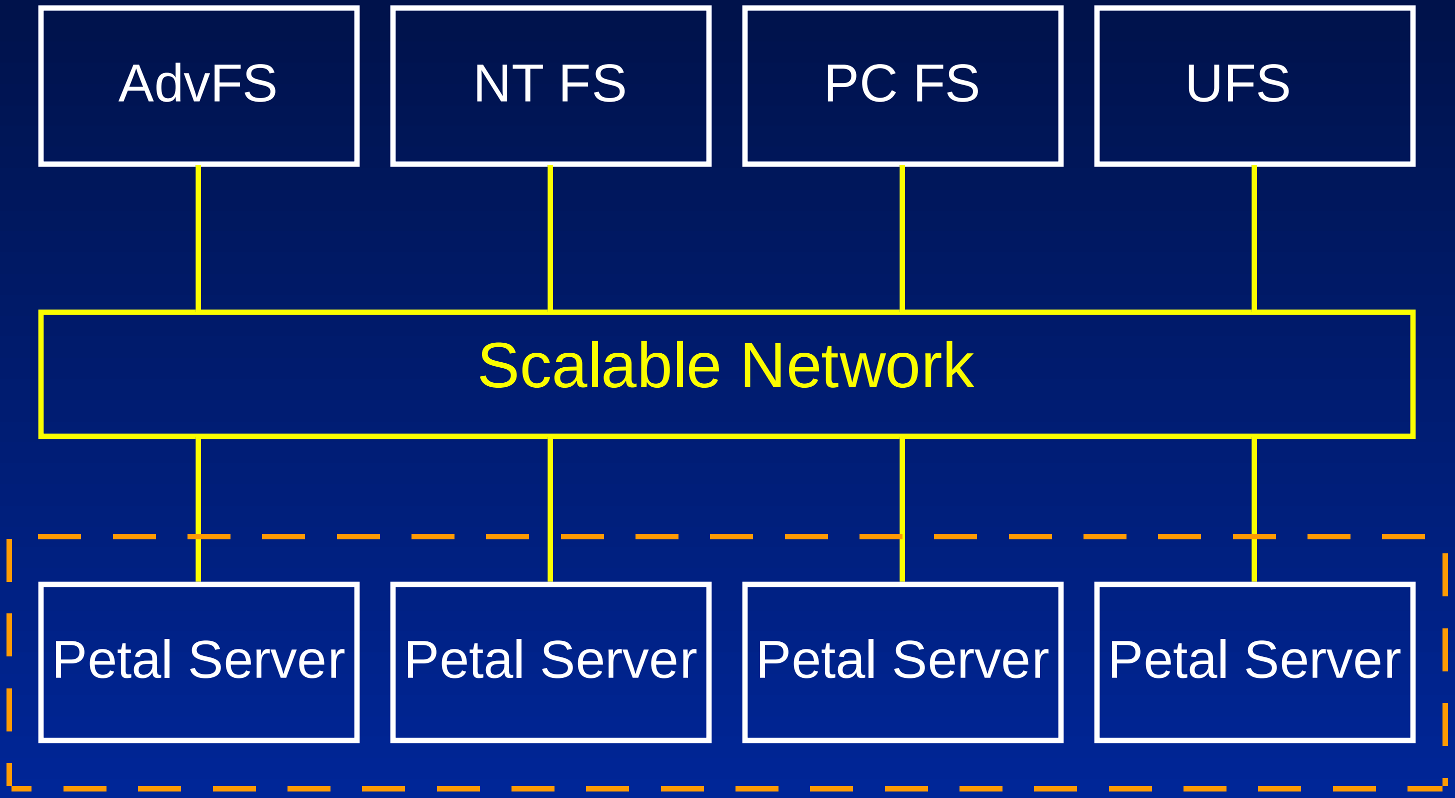
Petal

- Distributed block-level storage system.
- Automatically handle component failures.
- Automatic load and capacity balancing.
- Support for fast online backup.
- Heterogeneous systems and applications.
- Incremental online expansion.
- Survive site failures.

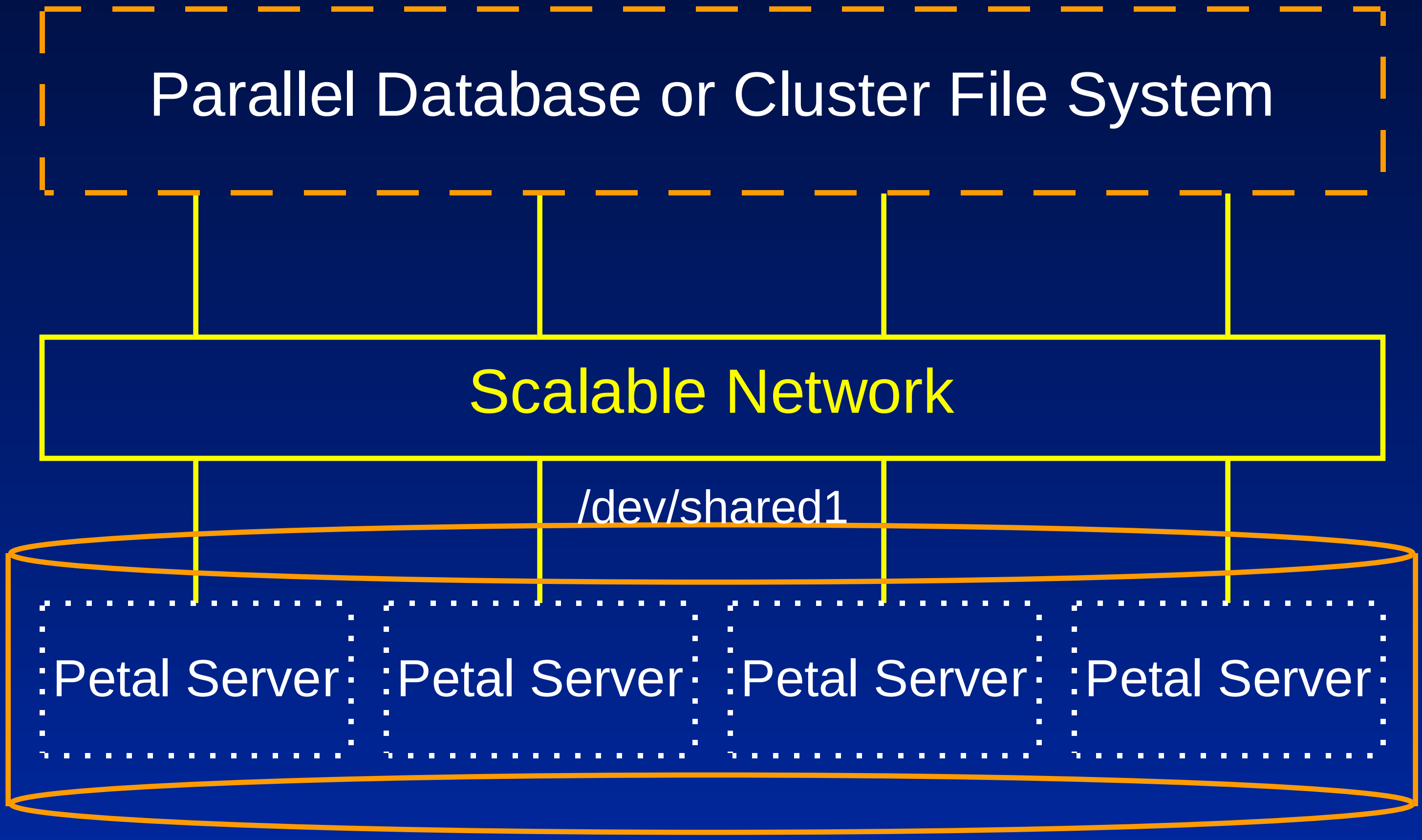
Logical System View



Physical System View



Physical System View



Related Work

- Disks: Logical Disk, Swift, AutoRAID, RAID-II, TickerTAIP, Loge, Mime.
- File Systems: xFS, Zebra, Echo, AFS, parallel file systems.
- Differences with Petal:
  - » Degree of distribution.
  - » Level of fault-tolerance.
  - » Ease of management.

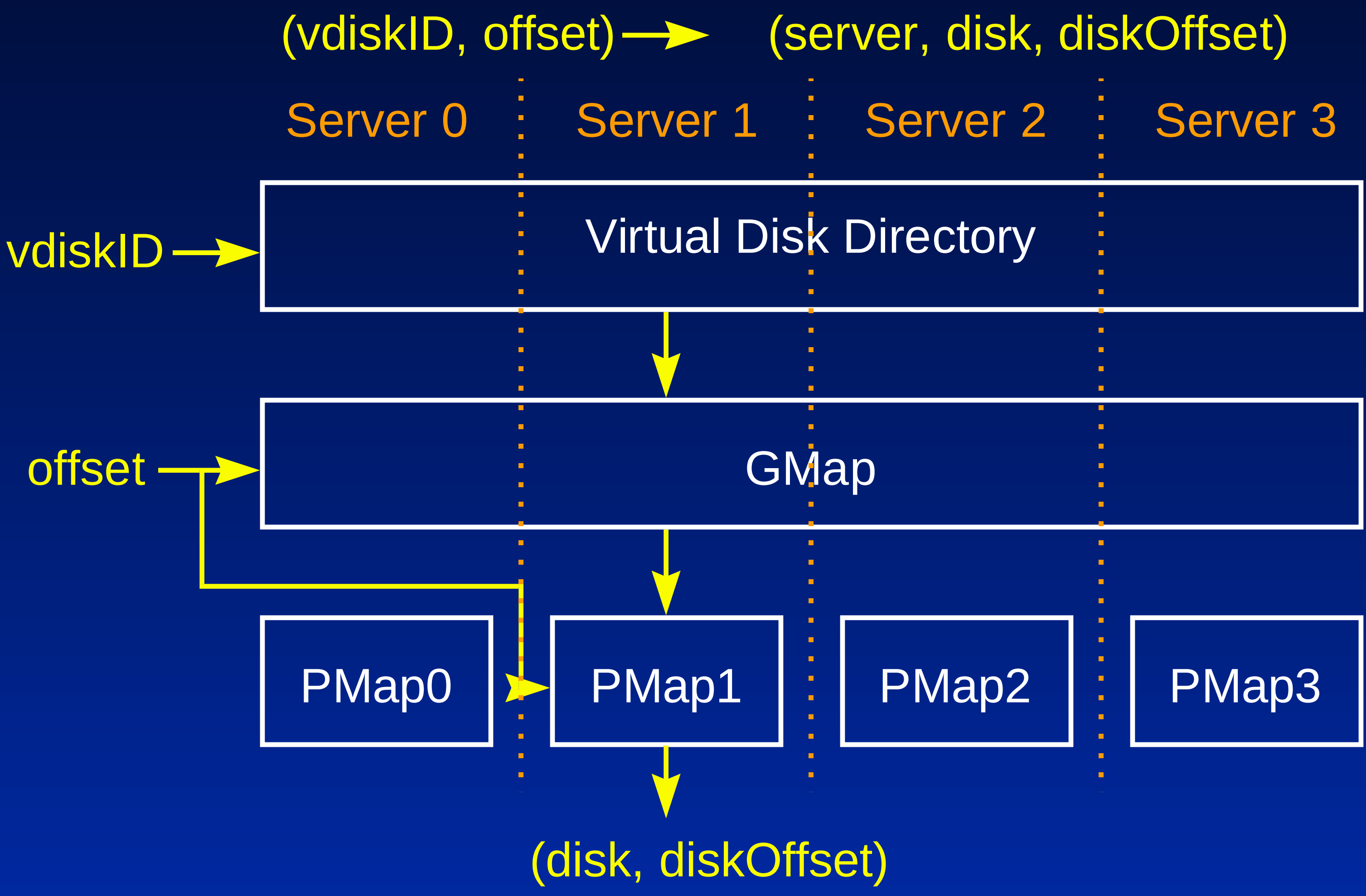
Outline

- Design Overview
- Performance Measurements
- Summary & Future Directions

Virtual Disks

- Each disk provides 2^64 byte address space.
- Created and destroyed on demand.
- Allocates disk storage on demand.
- Snapshots via copy-on-write.
- Online incremental reconfiguration.

Virtual to Physical Translation



Global State Management

- Based on Leslie Lamport’s Paxos algorithm.
- Global state is replicated across all servers.
- Consistent in the face of server & network failures.
- A majority is needed to update global state.
- Any server can be added/removed in the presence of failed servers.

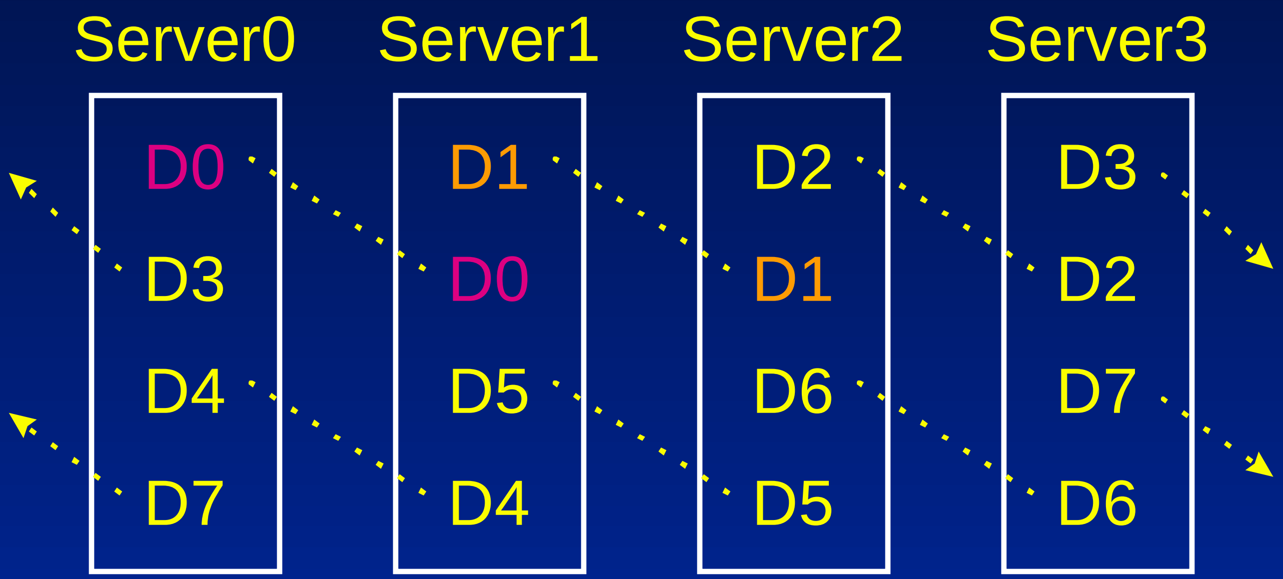
Fault-Tolerant Global Operations

- Create/Delete virtual disks.
- Snapshot virtual disks.
- Add/Remove servers.
- Reconfigure virtual disks.

Data Placement & Redundancy

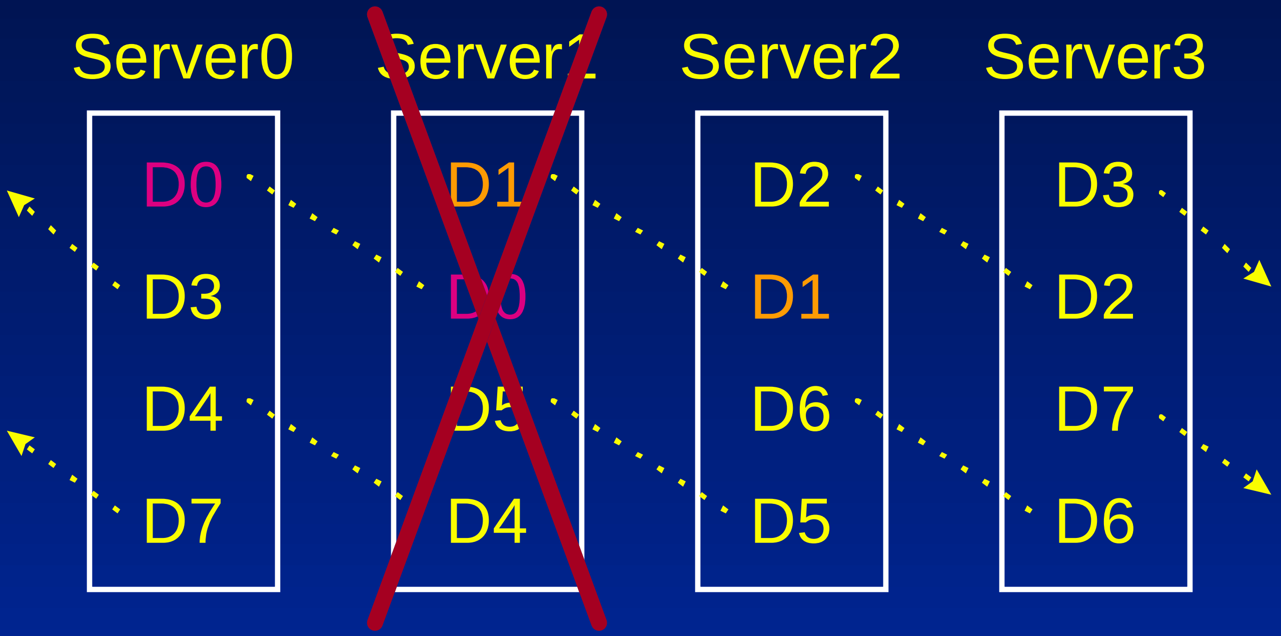
- Supports non-redundant and chained-declustered virtual disks.
- Parity can be supported if desired.
- Chained-declustering tolerates any single component failure.
- Tolerates many common multiple failures.
- Throughput scales linearly with additional servers.
- Throughput degrades gracefully with failures.

Chained Declustering



Server0	Server1	Server2	Server3
D0	D1	D2	D3
D3	D0	D1	D2
D4	D5	D6	D7
D7	D4	D5	D6

Chained Declustering

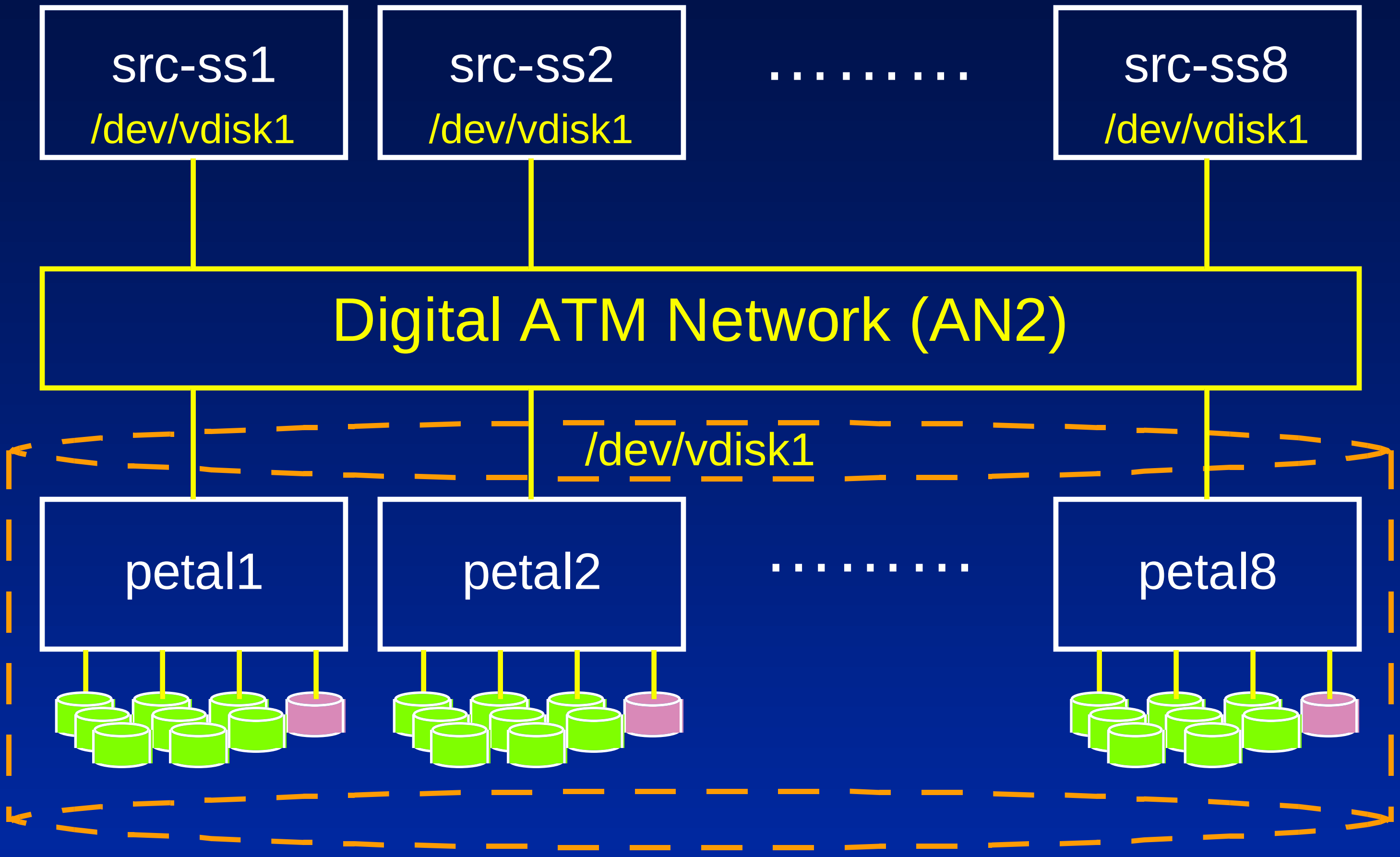


Server0	Server1	Server2	Server3
D0	D1	D2	D3
D3	D0	D1	D2
D4	D5	D6	D7
D7	D4	D5	D6

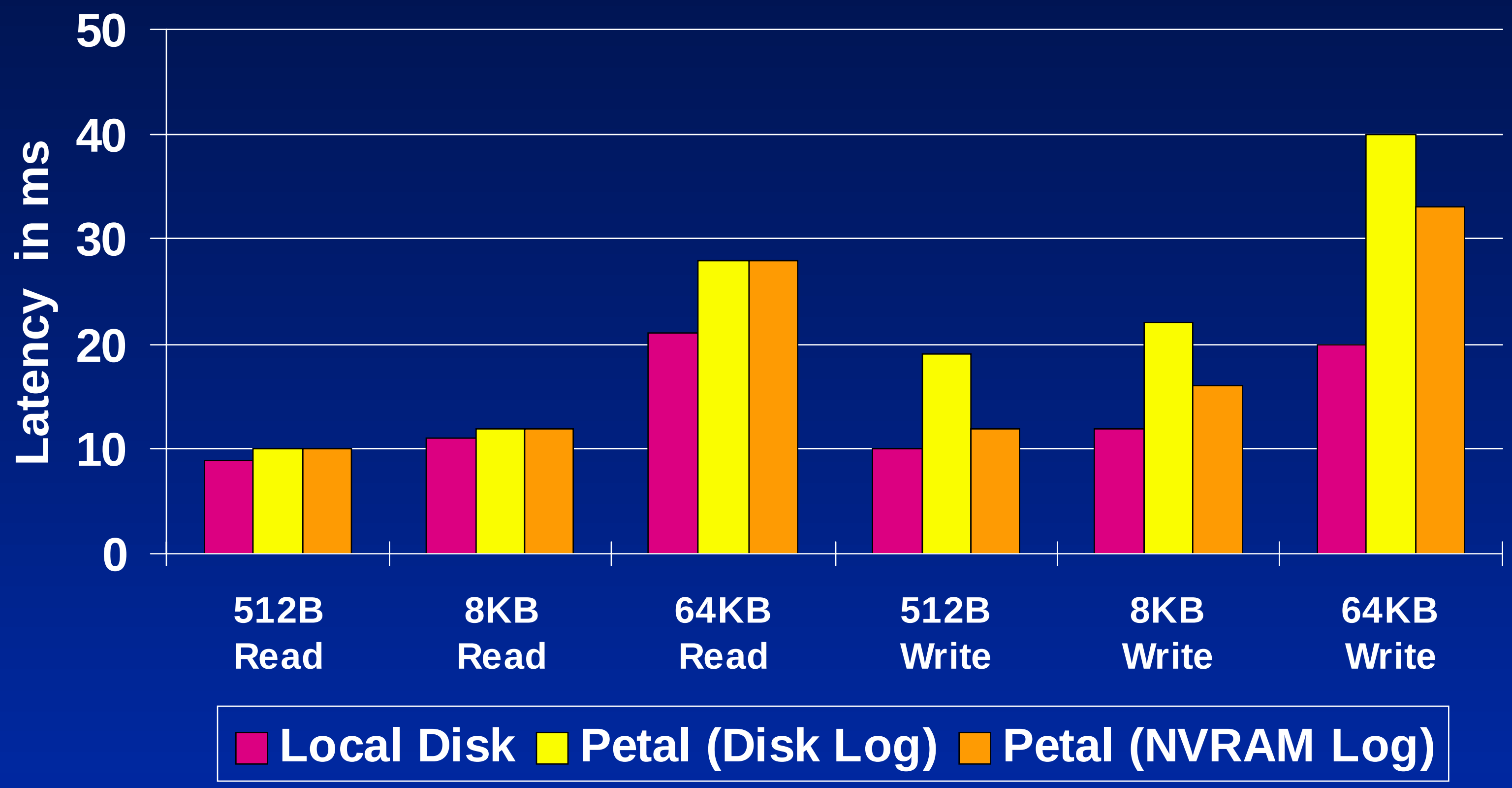
The Prototype

- Digital ATM network.
  - » 155 Mbit/s per link.
- 8 AlphaStation Model 600.
  - » 333 MHz Alpha running Digital Unix.
- 72 RZ29 disks.
  - » 4.3 GB, 3.5 inch, fast SCSI (10MB/s).
  - » 9 ms avg. seek, 6 MB/s sustained transfer rate.
- Unix kernel device driver.
- User-level Petal servers.

The Prototype



Client Request Latency



Aggregate Throughput

	Throughput	CPU Util
512B Read	3.8 MB/s (7510 IO/s)	44 %
8KB Read	53.7 MB/s	48 %
64KB Read	115.2 MB/s	17 %
512B Write	0.89 MB/s (1739 IO/s)	33 %
8KB Write	23.1 MB/s	68 %
64KB Write	49.3 MB/s	85 %

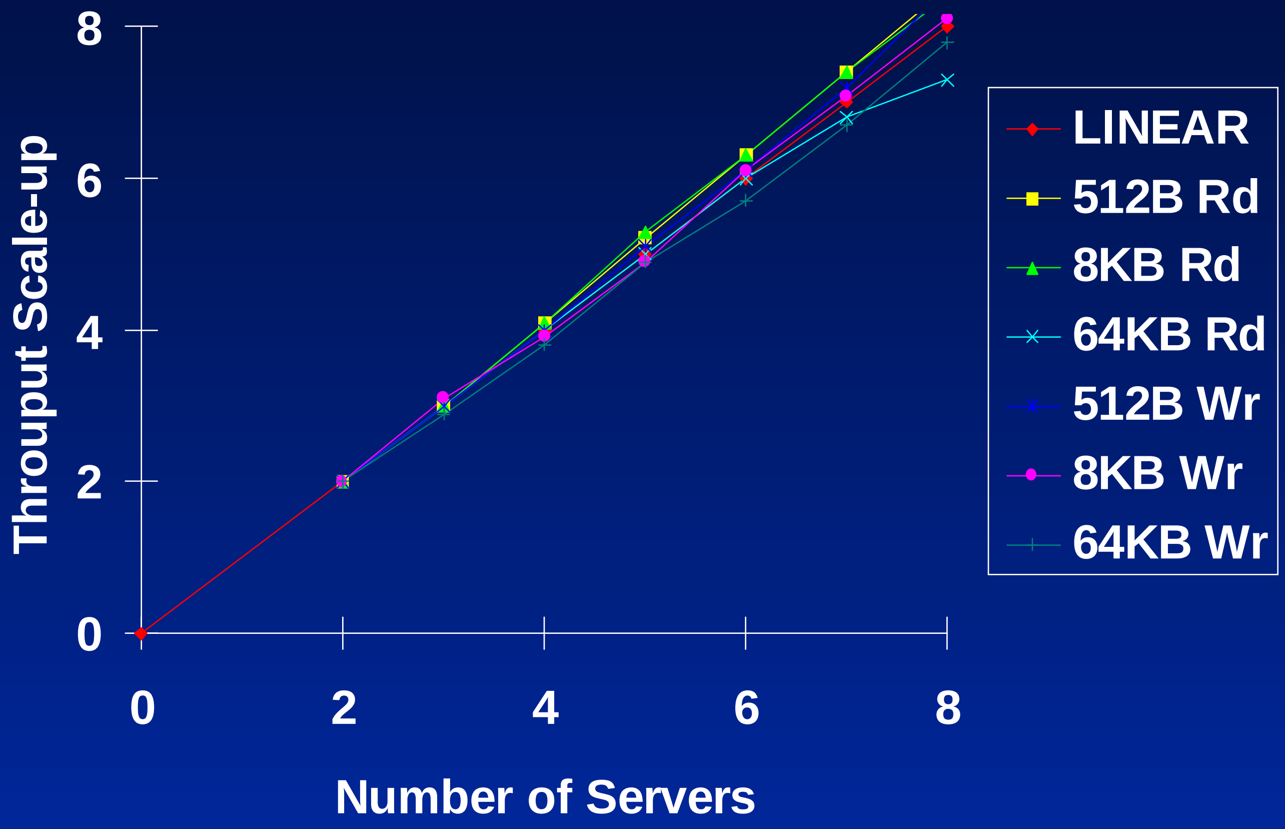
Chain-declustered virtual disk, 8 servers.
Random requests.

Failure Mode Performance

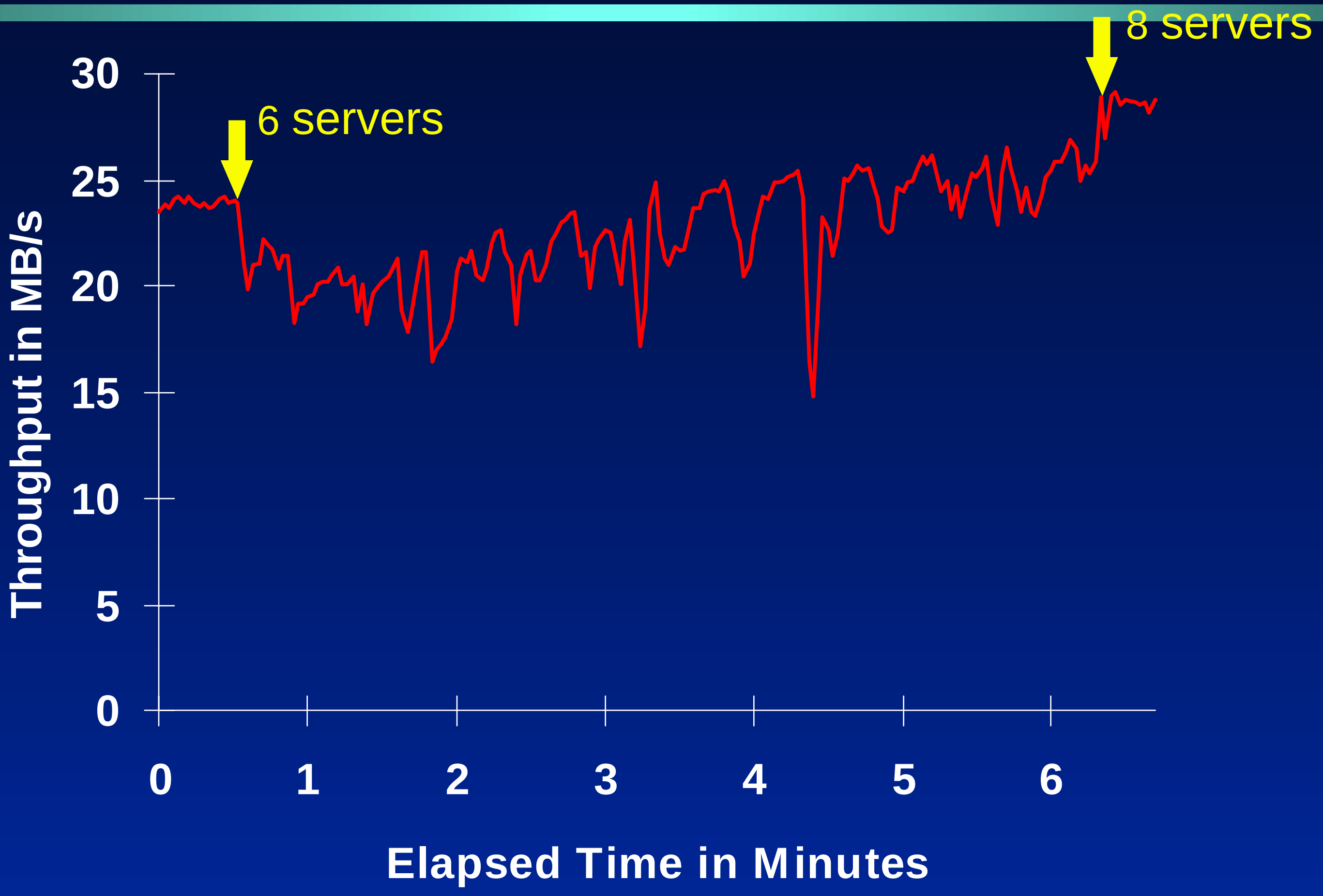
1 out of 8 servers failed
7/8 = 88%

	Failed	Normal	% of Normal
512B Read	3.4 MB/s	3.8 MB/s	87 %
8KB Read	47.1 MB/s	53.7 MB/s	88 %
64KB Read	106.7 MB/s	115.2 MB/s	93 %
512B Write	0.88 MB/s	0.89 MB/s	99 %
8KB Write	22.9 MB/s	23.1 MB/s	99 %
64KB Write	48.4 MB/s	49.3 MB/s	98 %

Throughput Scaling



Virtual Disk Reconfiguration



virtual disk w/ 1GB of allocated storage
8KB reads & writes

Modified Andrew Benchmark

Elapsed Time in Seconds				
	UFS		AdvFS	
	RZ29	Petal	RZ29	Petal
Create Directories	0.9	1.4	0.28	0.28
Copy Files	4.1	4.4	3.6	3.7
Directory Status	4.3	4.1	4.2	4.6
Scan Files	5.1	5.2	5.2	5.3
Compile	41.1	41.8	40.0	40.6

Chain-declustered virtual disk.

Summary

- Latency comparable to a local disk.
- Up to 115 MB/s on reads 49 MB/s on writes.
- Automatically tolerates and recovers from component and communication failures.
- Automatically distributes load in the face of component failures.
- Incremental online expansion.

Future Directions

- Improved modularity and performance.
- AltaVista.
- Oracle Parallel Server.
- Cluster File System.