General Parallel File System (GPFS) Native RAID

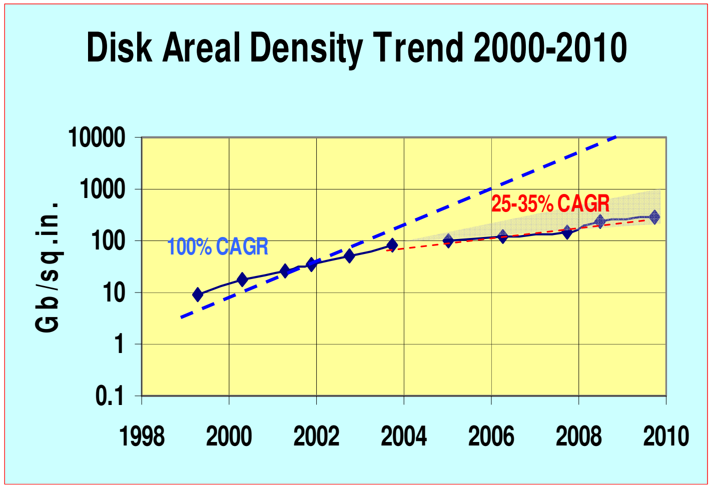
For 100,000-Disk Petascale Systems

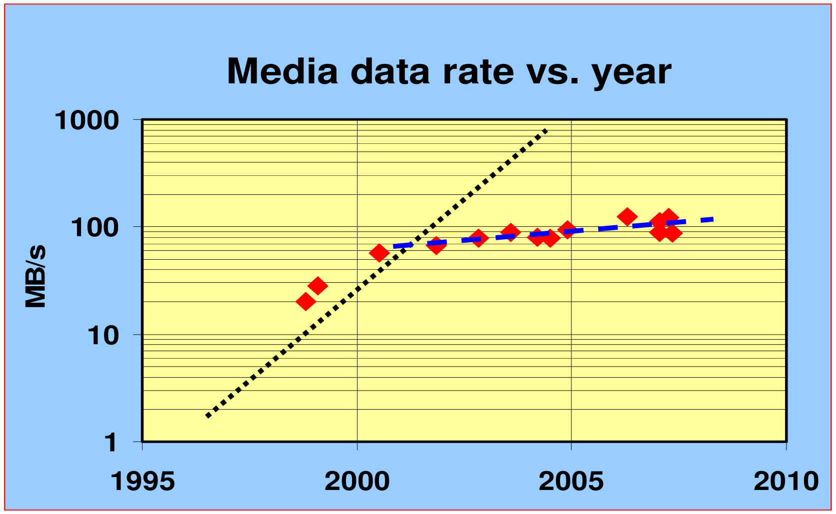
Veera Deenadhayalan

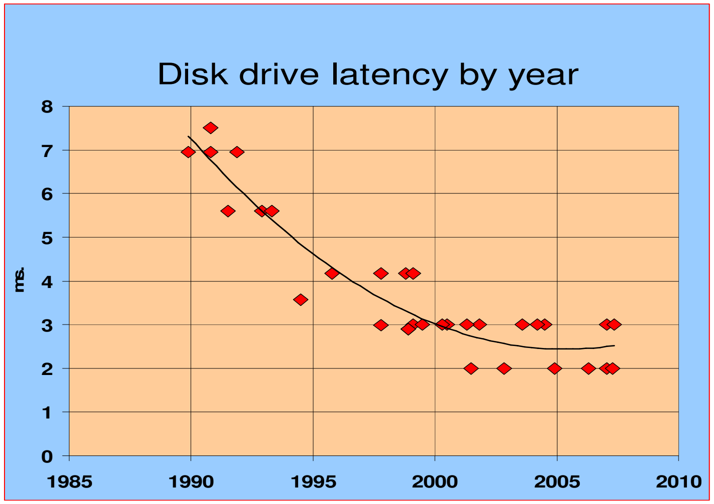
IBM Almaden Research Center

Hard Disk Rates Are Lagging

- There have been recent inflection points in disk technology – in the wrong direction
- In spite of these trends, programs like HPCS aim to maintain performance increases







Too many moving parts

Design Challenge

- *How to design a reliable HPC storage system using 100K+ disk drives?*

System	Year	Peak PFLOPS	# Cores	Storage Capacity	# Disks
ASCI Purple	2005	0.1	12k	2 PB	10K
HPCS (target)	2012	16	500K+	40+ PB	100K+
Exascale (target)	2018	1000	~150M	~1,000 PB	~200K-1M

40x 20x 10x

How to do it at acceptable performance and cost?

Outline

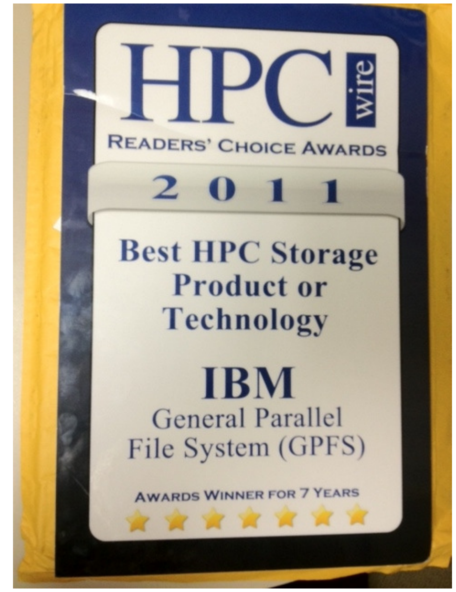
- Background
  - GPFS
  - Parallel Computing
  - RAID

- Challenges with Traditional RAID and Disk Drives

- Solutions in GPFS Native RAID to address the challenges

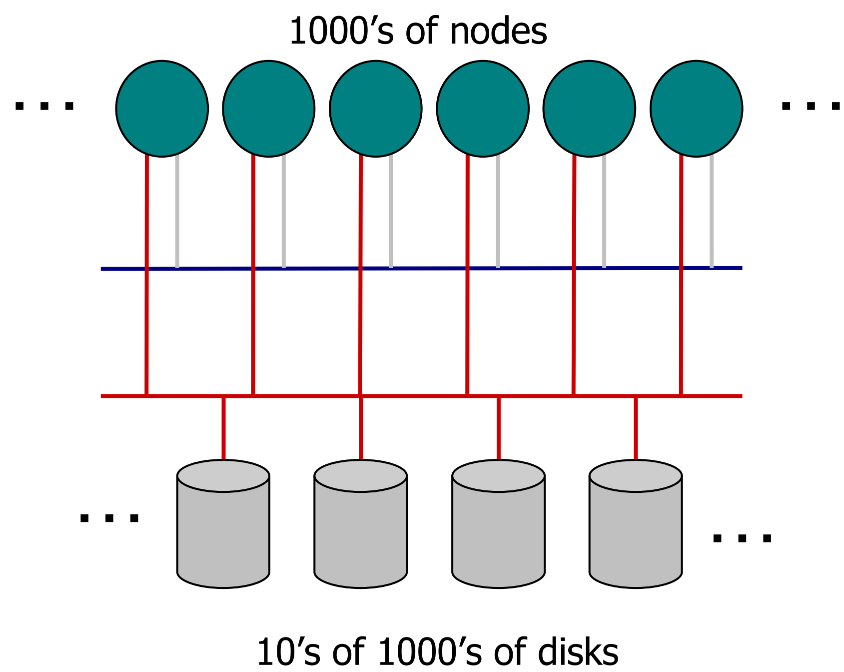
GPFS – General Parallel File System

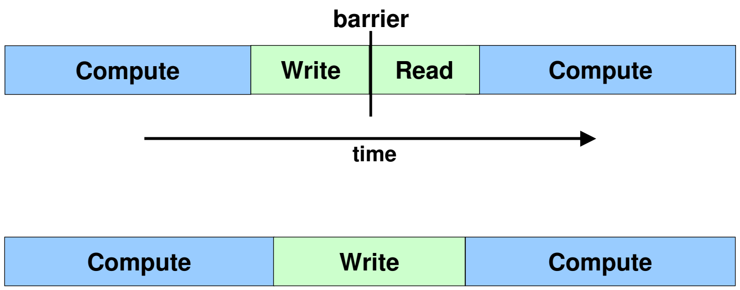
- Originated at IBM Almaden Research Laboratory
- IBM Research continues to deliver new GPFS features
- Applications
  - Supercomputers
    - Aerospace, Automotive, Life Sciences, Defense, Multimedia, National Labs, Universities, Weather modeling, etc.
  - Scalable NAS



Supercomputer Storage Use Cases

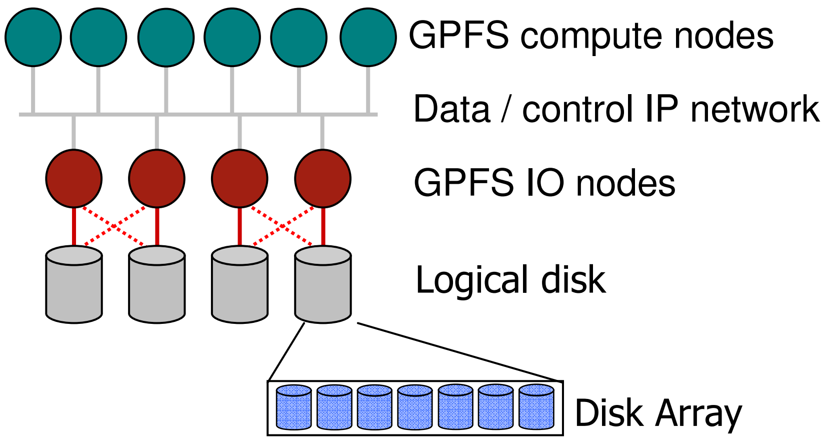
- "A supercomputer is a device for turning compute-bound problems into I/O-bound problems"
  - Ken Batcher, Emeritus Professor, Kent State University
- Share storage within a computation and across workflows
- Checkpoint
- Any time spent doing file I/O is time not spent computing.
- Uses parallel file systems
  - For the single-system image that simplifies programming
  - For POSIX semantics that hides the complexities of clustering
  - For high-throughput and load balancing



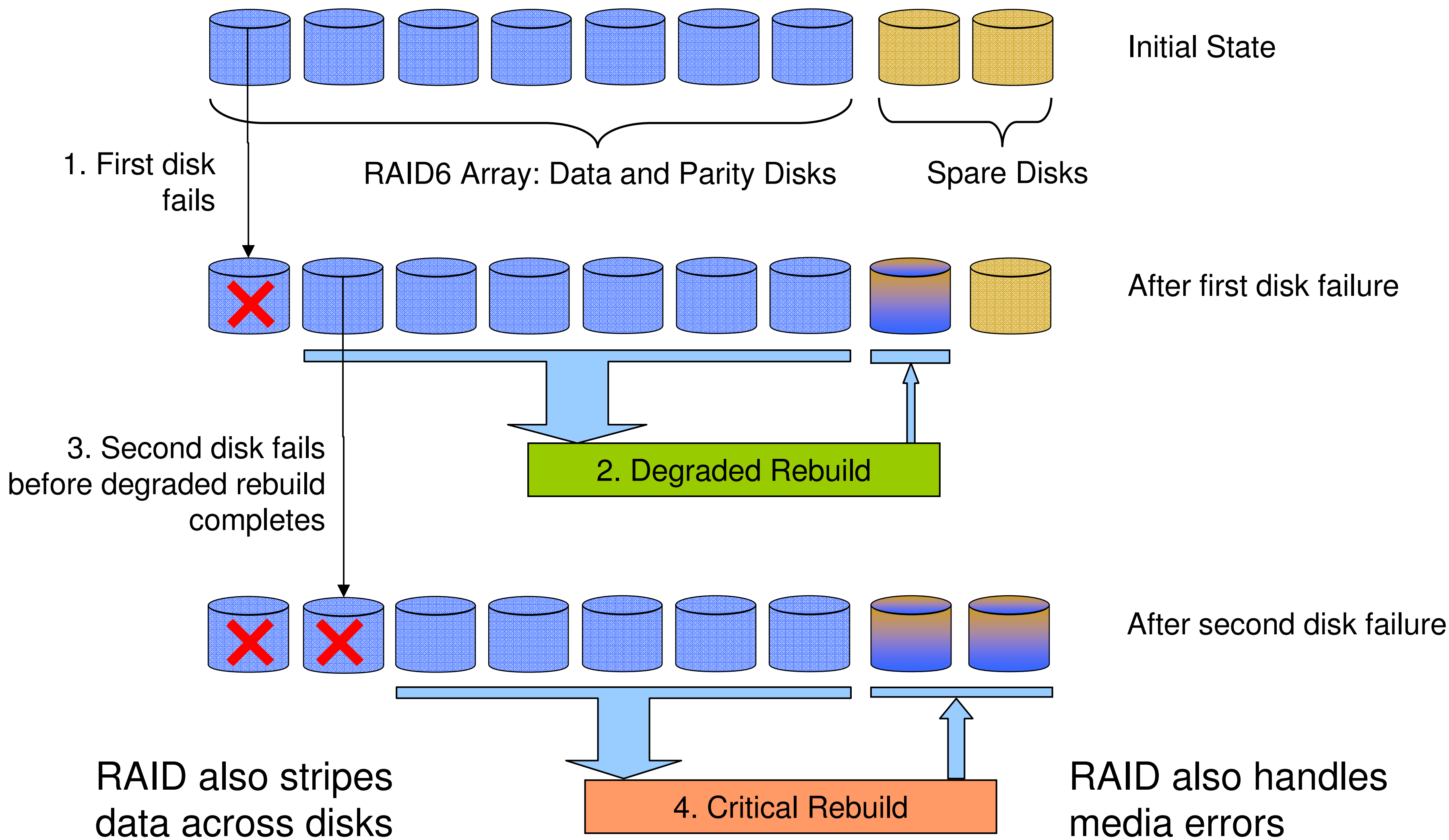


GPFS – High Throughput

- Wide Striping
  - Both data and metadata striped across many disks
  - Files striped block by block across all disks … for throughput and load balancing



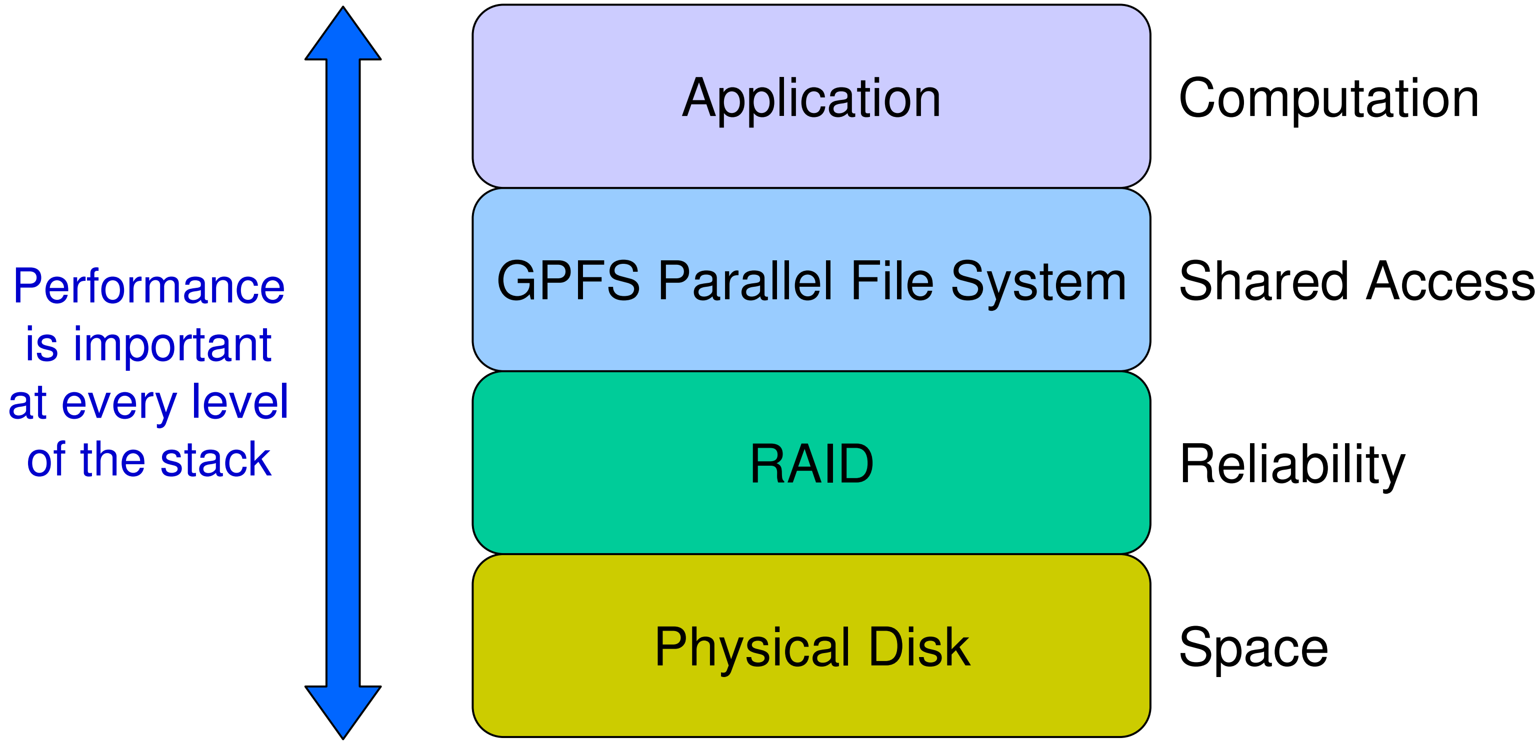
RAID Array Concepts – Disk Failure



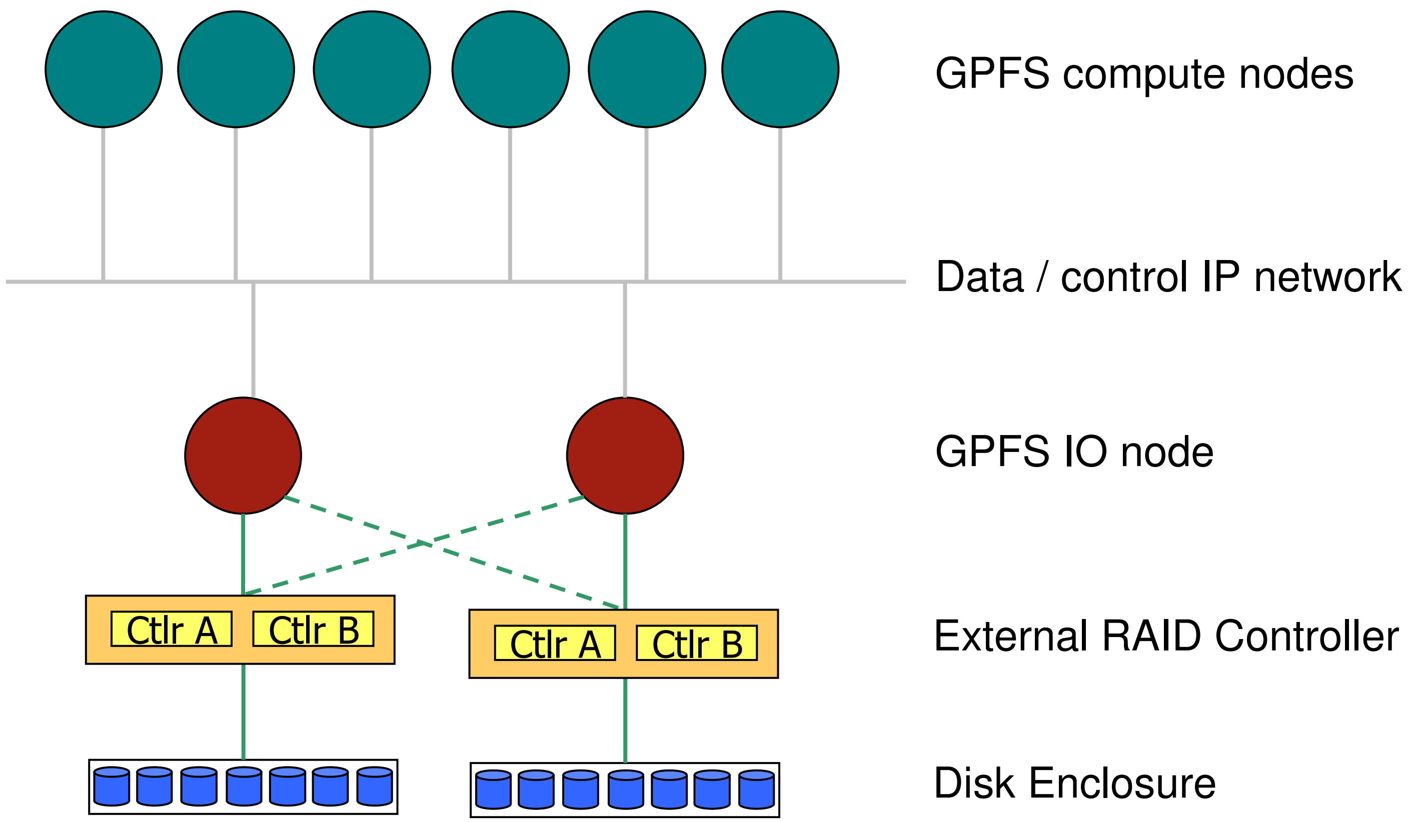
RAID also stripes data across disks

RAID also handles media errors

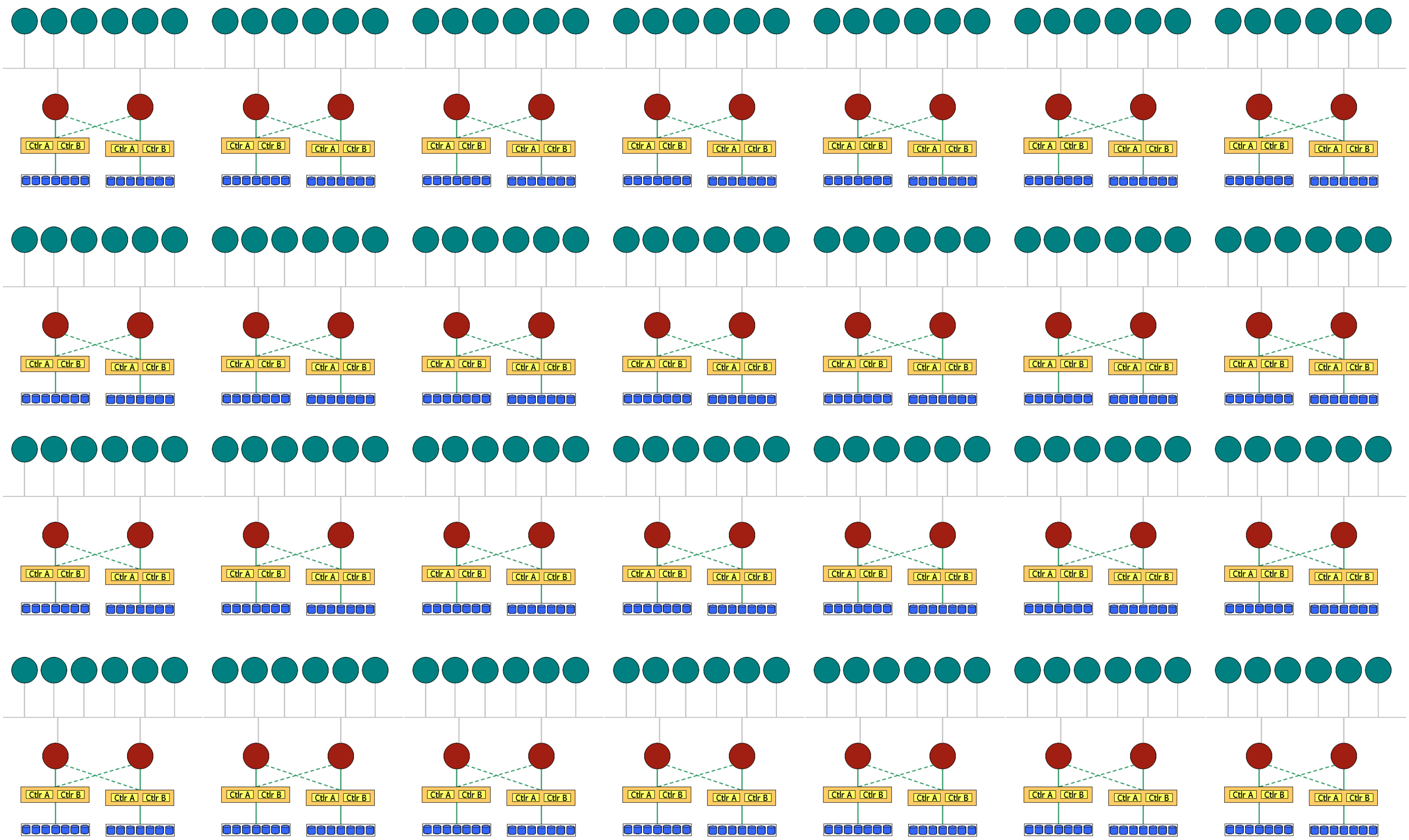
Simplified Storage Stack



Traditional Building Block Uses External RAID Controllers

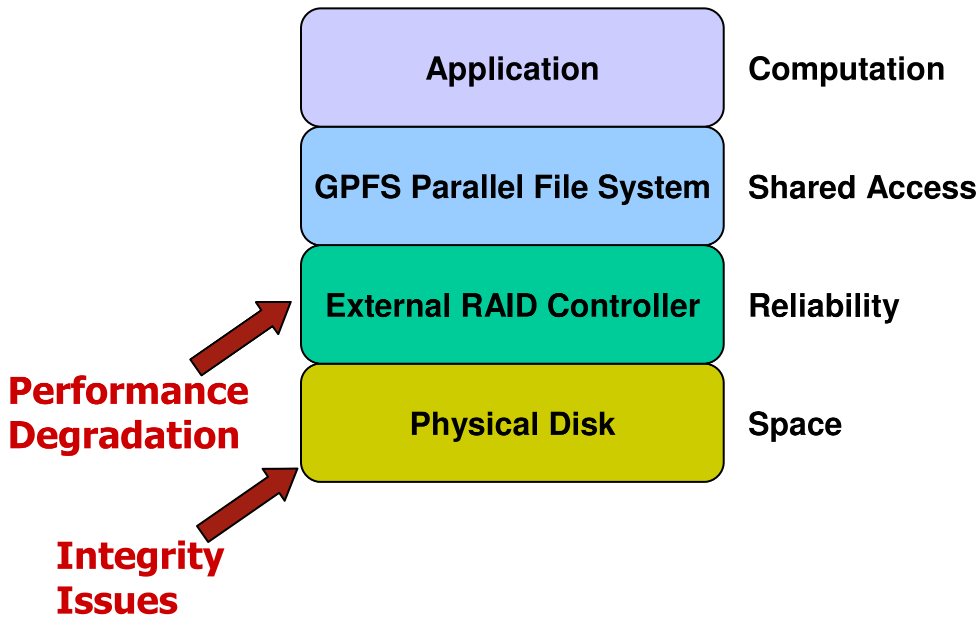


A Cluster Using Traditional RAID

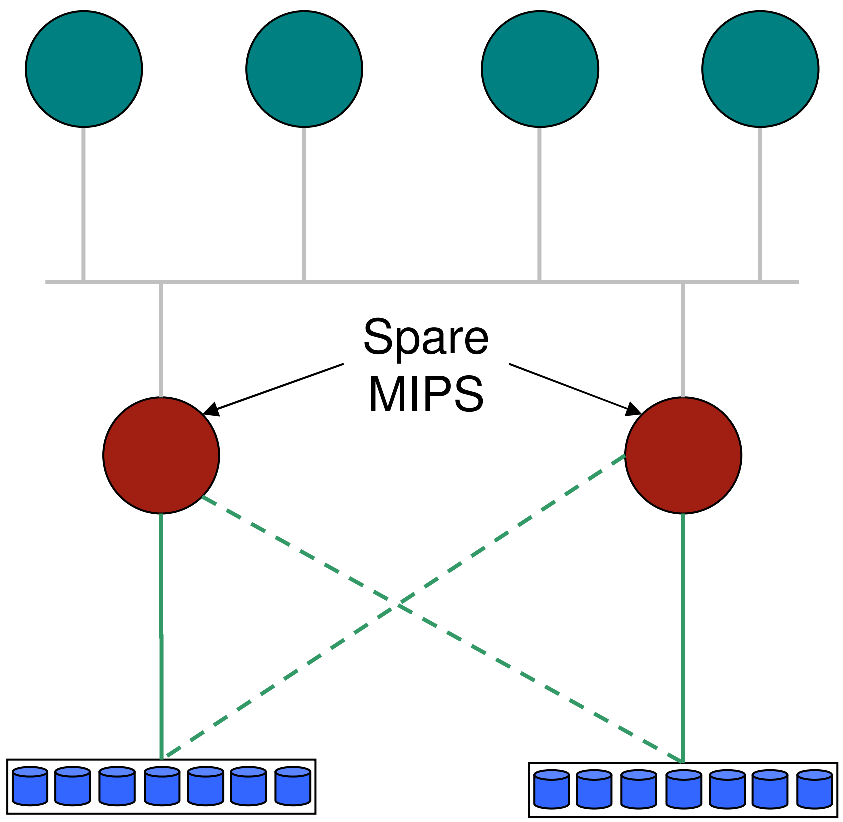


Problems with Traditional RAID and Disks

- Performance: Traditional RAID rebuild significantly affects performance
  - With 100,000 disks, disk drive failures are expected to happen on a daily basis.
  - Disks are getting bigger and hence, take longer to rebuild
- “Silent” data corruption in disk drives



Our Solution Uses GPFS Native RAID



GPFS compute nodes

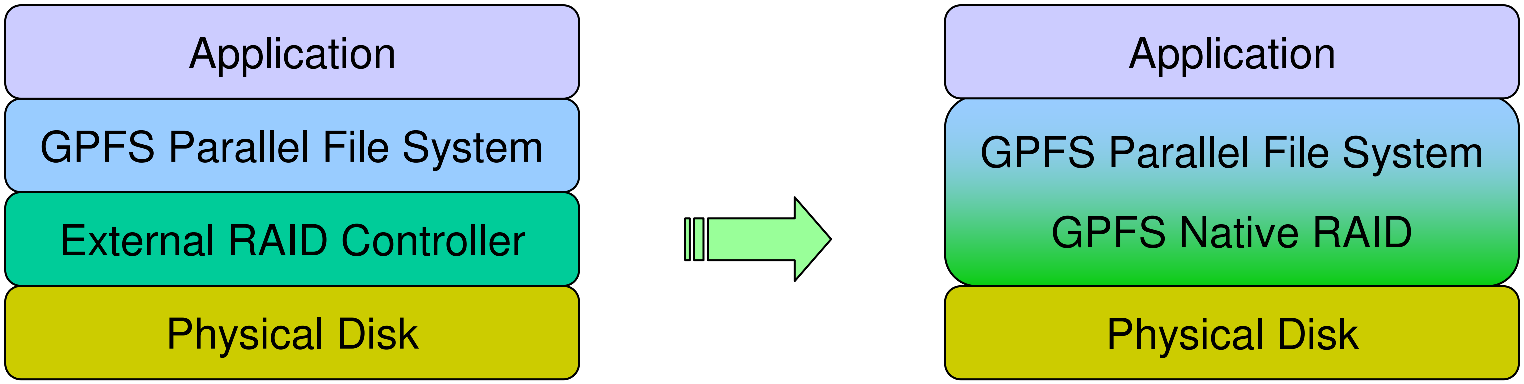
Data / control IP network

GPFS IO node w/ Native RAID

No External RAID Controller

Disk Array

Why Native RAID?



1. Higher performance
   – Use Declustered RAID to minimize performance degradation during rebuild

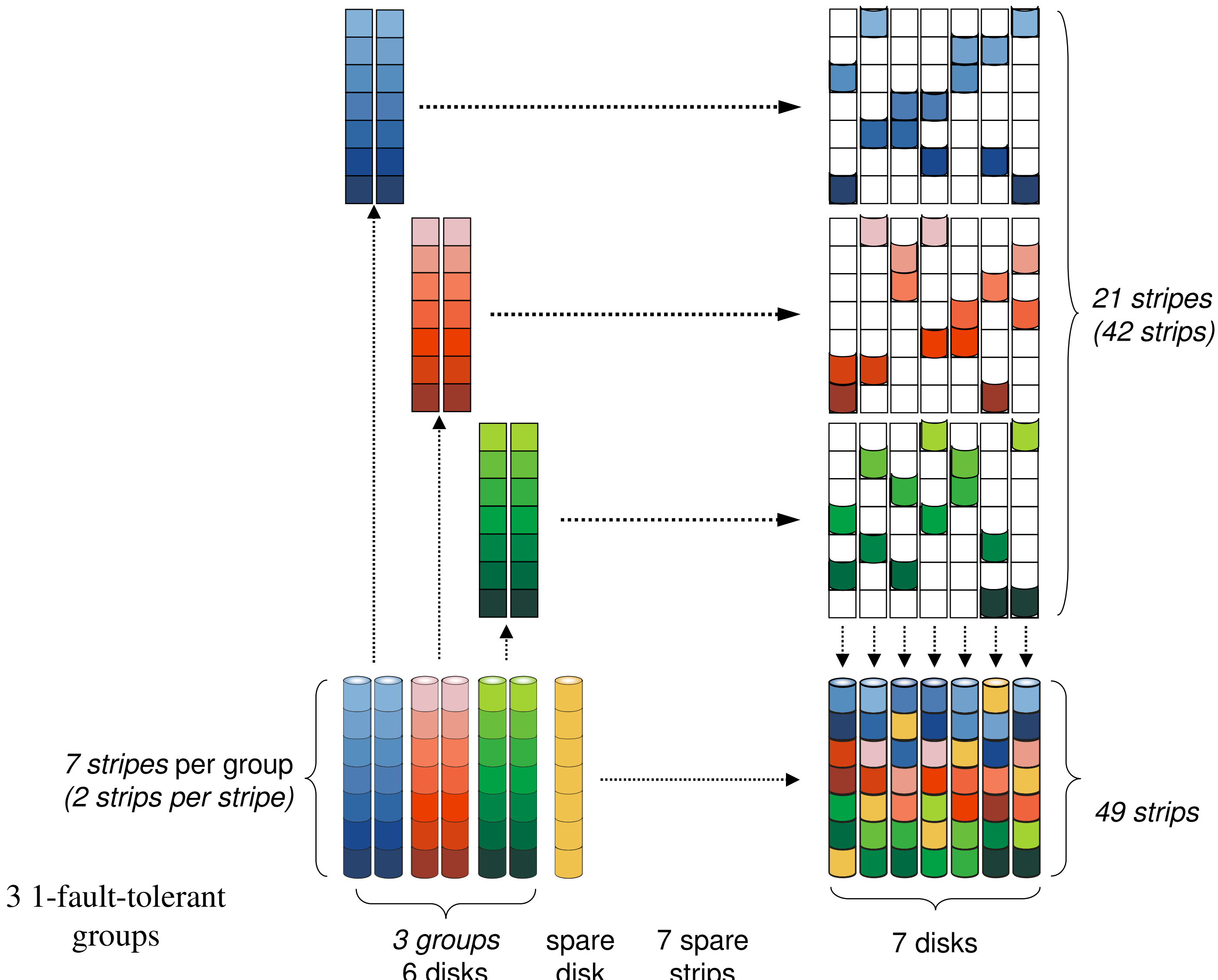
2. Extreme data integrity
   – Use end-to-end checksums and version numbers to detect, locate and correct silent disk corruption

Why Native RAID?

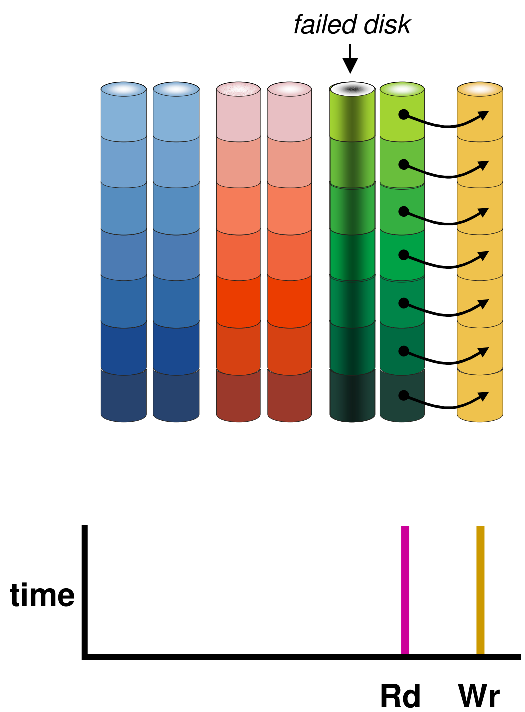
1. Higher performance
   - Use Declustered RAID to minimize performance degradation during rebuild

2. Extreme data integrity
   - Use end-to-end checksums and version numbers to detect, locate and correct silent disk corruption

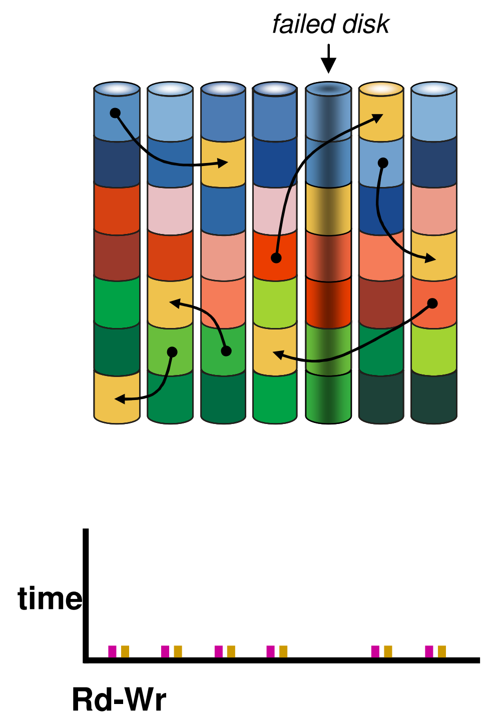
Declustered RAID1 Example



Declustered RAID Rebuild Example – Single Fault



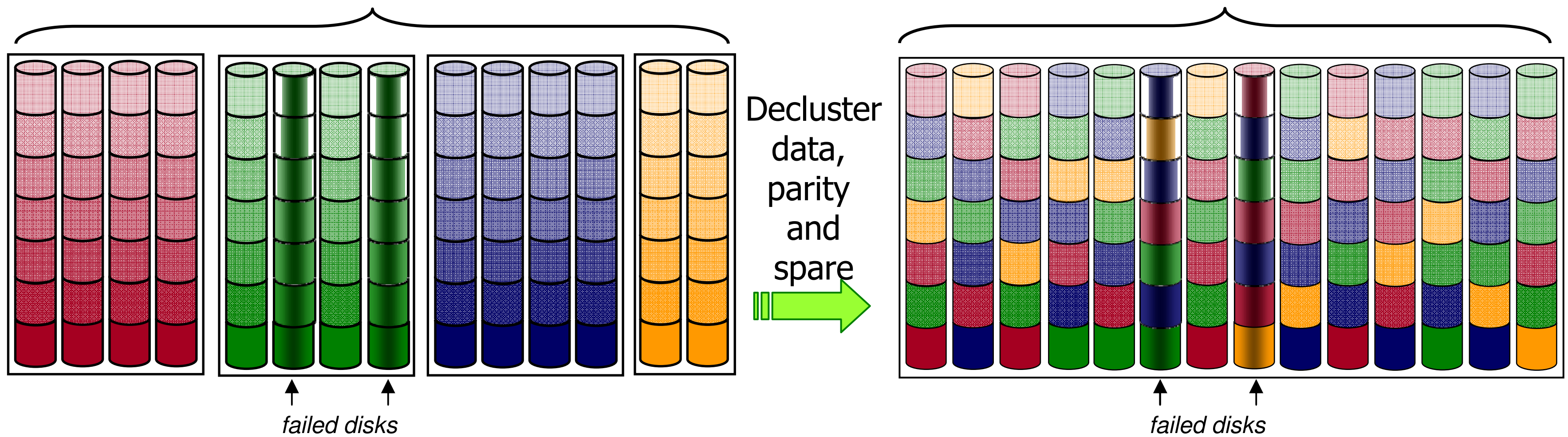
Rebuild activity confined to just a few disks – slow rebuild, disrupts user programs

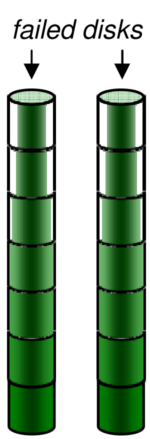


Rebuild activity spread across many disks, faster rebuild or less disruption to user programs

Declustered RAID6 Example

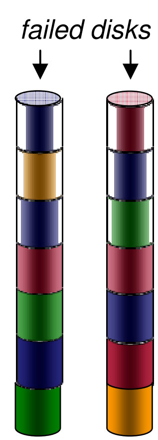
14 physical disks / 3 traditional RAID6 arrays / 2 spares — **14 physical disks / 1 declustered RAID6 array / 2 spares**





Number of faults per stripe		
Red	Green	Blue
0	2	0
0	2	0
0	2	0
0	2	0
0	2	0
0	2	0
0	2	0

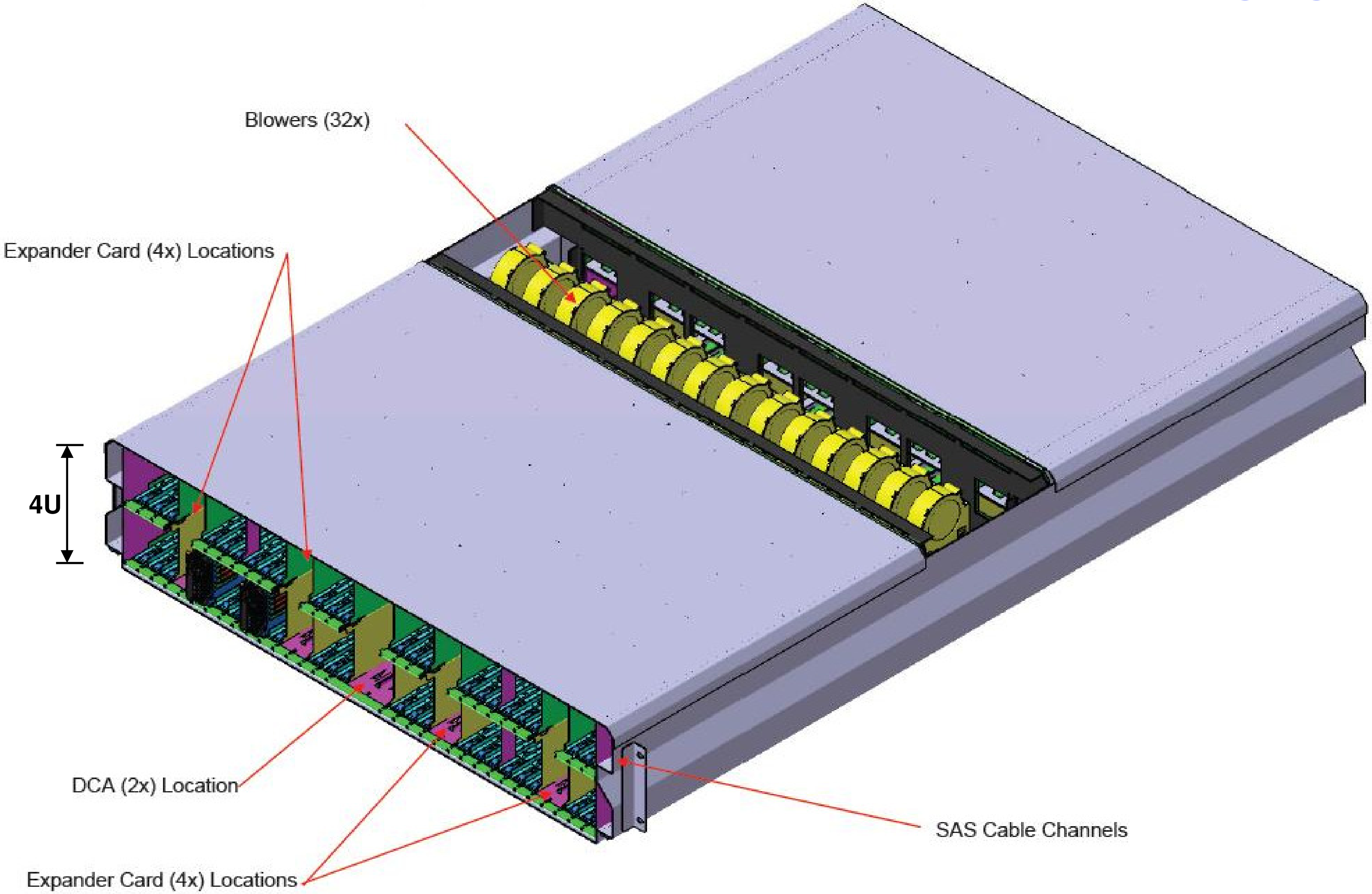
Number of stripes with 2 faults = 7



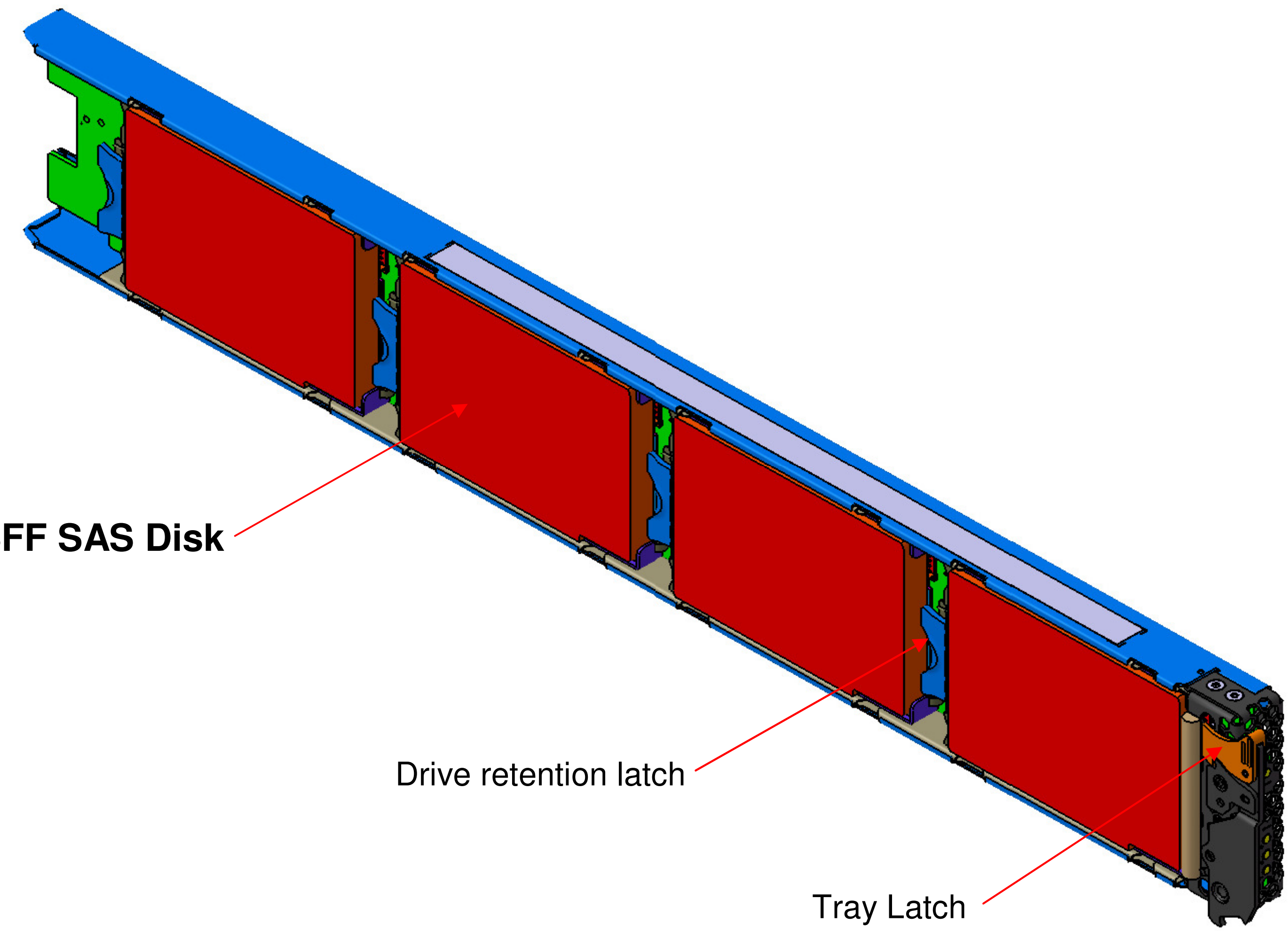
Number of faults per stripe		
Red	Green	Blue
1	0	1
0	0	1
0	1	1
2	0	0
0	1	1
1	0	1
0	1	0

Number of stripes with 2 faults = 1

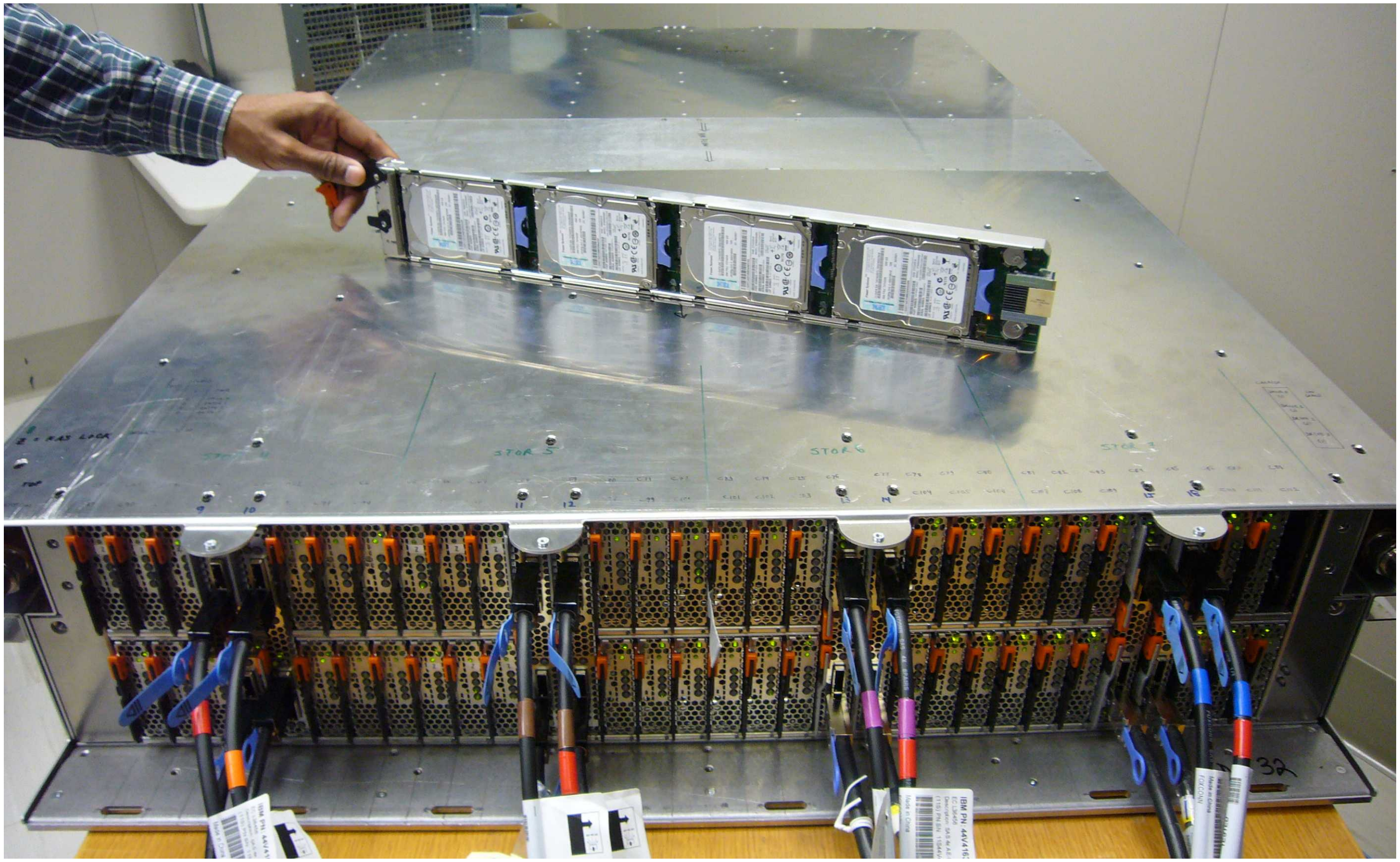
Power 775 Disk Enclosure – 384 Disks Dense Packaging



Power 775 Disk Enclosure Disk Carrier (4 disks)

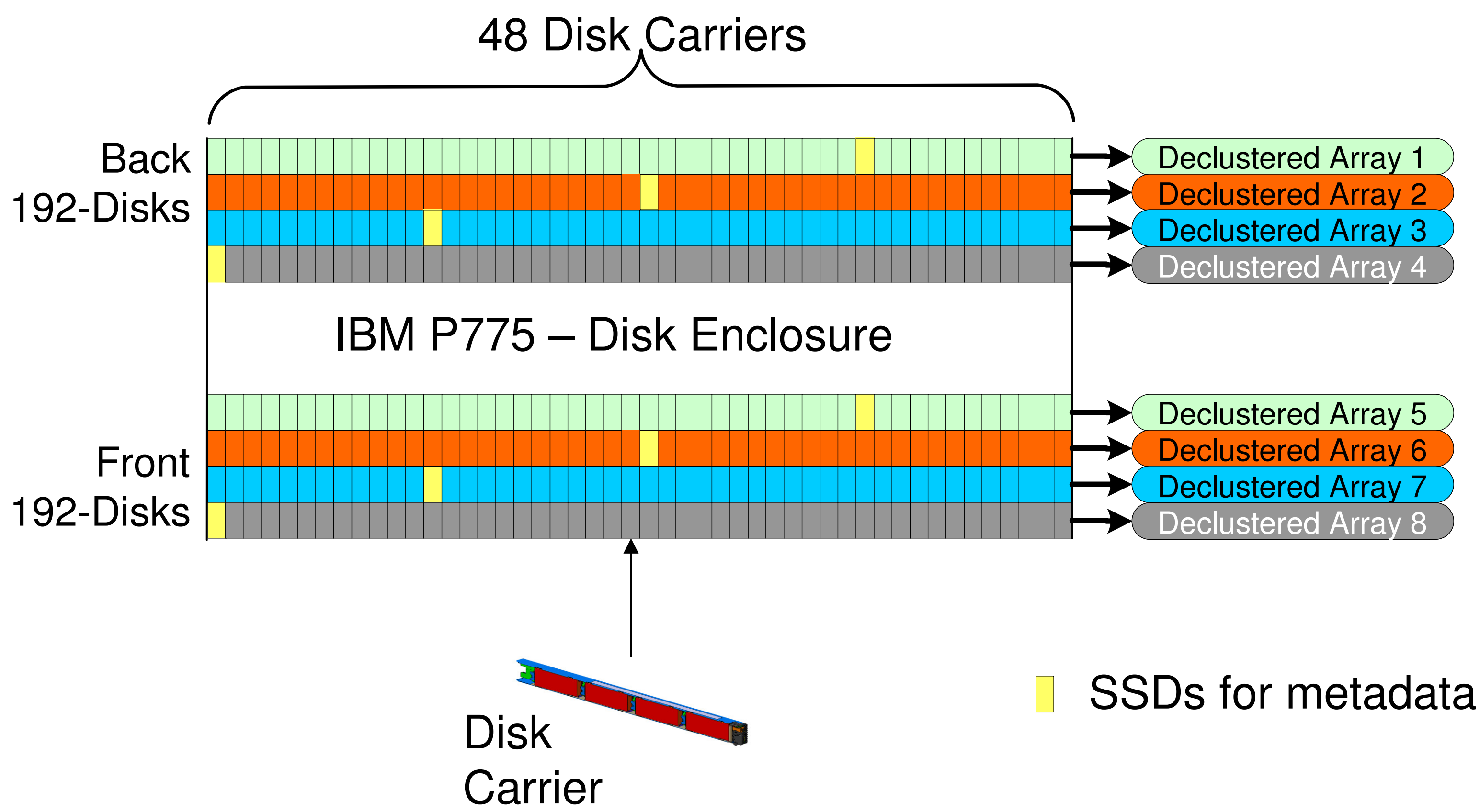


P775 Disk Enclosure in Almaden Lab



Dense Disk Enclosure – 384 disks per enclosure

Declustered RAID Arrays on P775 Disk Enclosure



GPFS Native RAID on P775-Disk Enclosure

3-fault tolerant RAID

- 8 data + 3 parity strips
- 4 way replication
- distributed across 47-drive declustered array

When one disk is down (most common case)

- Rebuild slowly with minimal impact to client workload

When three disks are down (rare case)

- Fraction of stripes that have three failures = 11/47 * 10/46 * 9/45 = 1%
- Quickly get back to non-critical (2 failure) state vs. rebuilding all stripes for conventional RAID

- GPFS Native RAID also supports RAID6 and 3 way replication

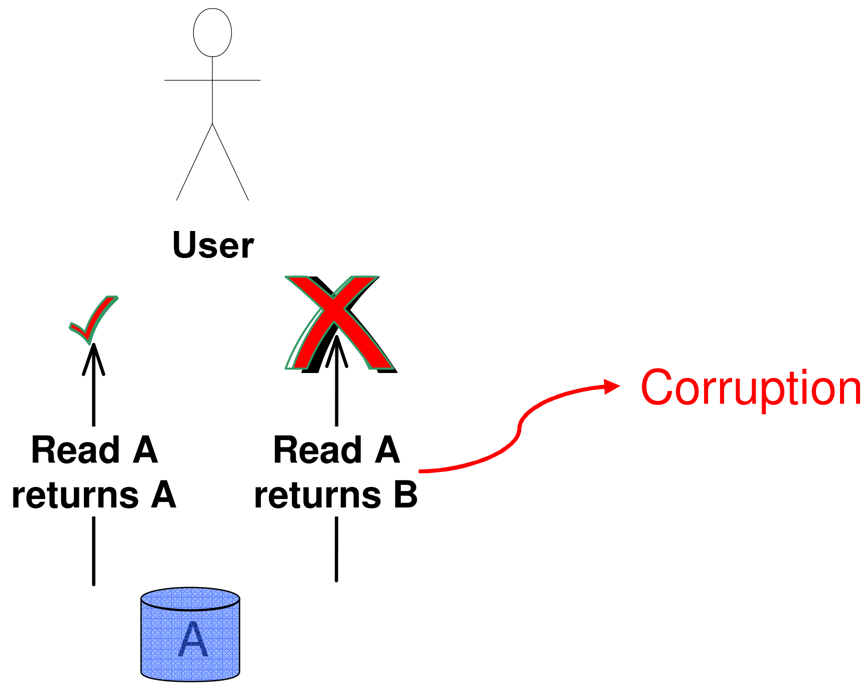
GPFS Native RAID uses declustered RAID to address the rebuild performance issue with traditional RAID.

Why Native RAID?

1. Higher performance
   – Use Declustered RAID to minimize performance degradation during rebuild

2. Extreme data integrity
   – Use end-to-end checksums and version numbers to detect, locate and correct silent disk corruption

Data Integrity Requirement



- Old adage: no data is better than bad data
- The widely publicized 5-9's model reliability and availability, assumes 100% integrity.

Undetected Disk Errors Are Different From Media Errors!

- Also referred to as silent disk errors.

- An analysis of data corruption in the storage stack, *FAST'08,* Bairavasundaram, et al.

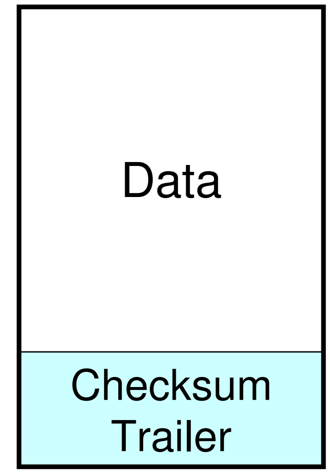
- Evaluating the Impact of Undetected Disk Errors in RAID Systems, *DSN'09*, Rozier, et al.
  - Estimates that a 1000 disk system will experience corruption every 5 years

- Write errors leading to undetected bad data:
  - Far Off-track Writes
  - Near Off-track writes
  - Dropped Writes

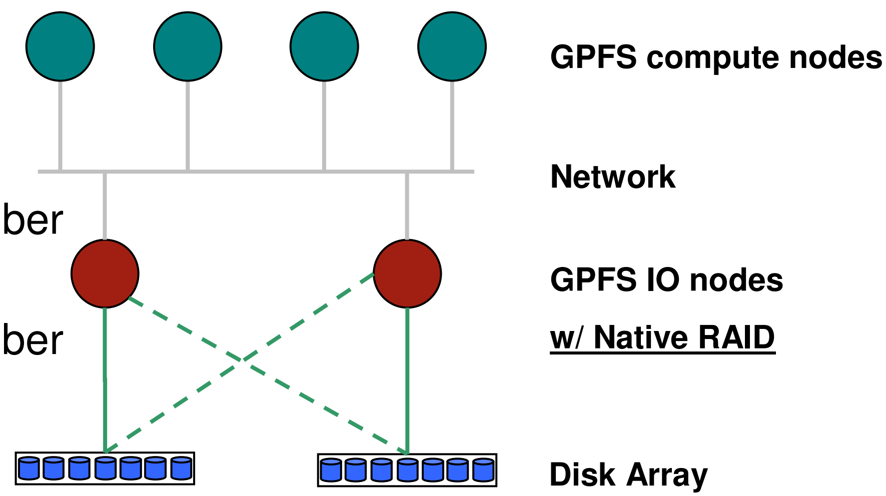
- Undetected Read Errors

- Can be transient or persistent

Checksums and Version Numbers



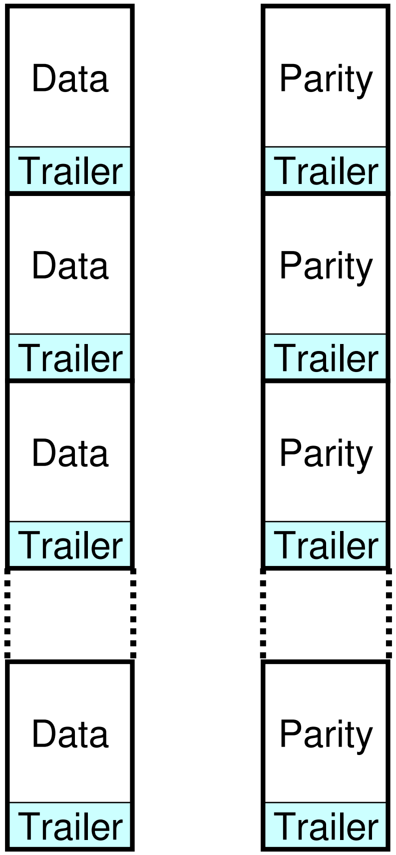
- Checksums in data trailer detects corruption
- Only a validated checksum can protect against dropped writes
  - Old data matches old checksum
- We use version numbers in metadata to validate checksum trailers
- End-to-end checksums
  - Write operation
    - From compute node to IO node
    - From IO node to disk with version number
  - Read operation
    - From disk to IO node with version number
    - From IO node to compute node



Version Numbers and End-to-end Checksum

- Both data and parity strips are divided into buffer chunks.

- Each buffer chunk trailer contains the required information to detect corruption and dropped write:
  - Array Unique Identifier
  - Virtual stripe number
  - Buffer index within stripe
  - Checksum of buffer payload
  - Checksum of buffer trailer
  - Buffer version number – for detecting dropped writes – also stored elsewhere in metadata



GPFS Native RAID uses version numbers and end-to-end checksums to detect, locate and correct silent disk corruption

Integrity Management

- Rebuild
  - Selectively rebuild portions of a disk that was temporarily unavailable.
  - Restore full redundancy after disk failures
  - Restore redundancy for the most affected data first (in the order of 3, 2, & 1 failures)
- Rebalance
  - When a failed disk is replaced with a spare disk, redistribute the free space
- Scrub
  - Verify checksum of data and parity/mirror
  - Verify consistency of data and parity/mirror
  - Fix problems found on disk
- Schedule this activity opportunistically but in the background
  - At full disk speed when no user activity
  - At configurable rate when the system is busy

Disk Management

- Analyze disk errors to determine:
  - connectivity problem
  - media error
  - complete drive failure

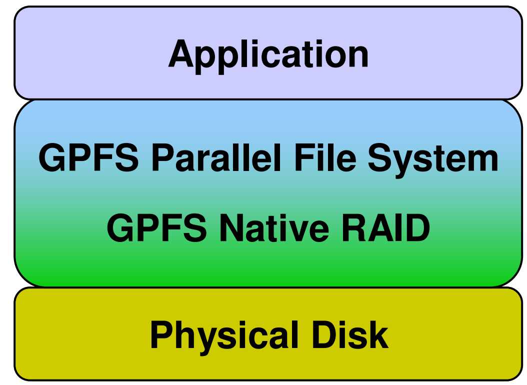
- Take corrective actions such as:
  - correcting media and checksum errors by recomputing and rewriting corrupted data
  - power-cycling unresponsive disks

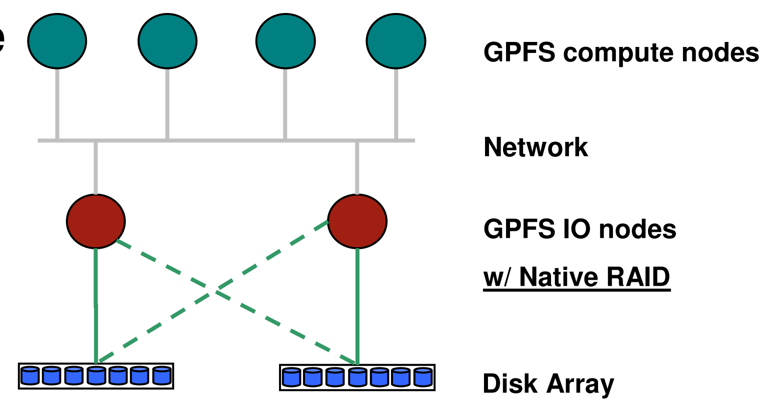
- Maintain per disk "health records" of performance and error rates
  - decide when a disk should be removed from service
  - request disk replacements based on a configurable service policy

- Control disk enclosure error lights, carrier solenoids and power to facilitate disk replacement.

GPFS Native RAID - Summary

- Engineered for high-performance supercomputers
  - Scales to 100k+ storage devices with a single file system
- Offers higher performance and better reliability in spite of unreliable components and subsystems
- Offers lower hardware costs by taking advantage of flourishing multi-core processors
- First customer ship 11/11 on IBM's Power 775 server
- Like GPFS, capable of supporting a wide range of applications and systems





Thank you!

Questions?