LStore: Logistical Storage

Alan Tackett

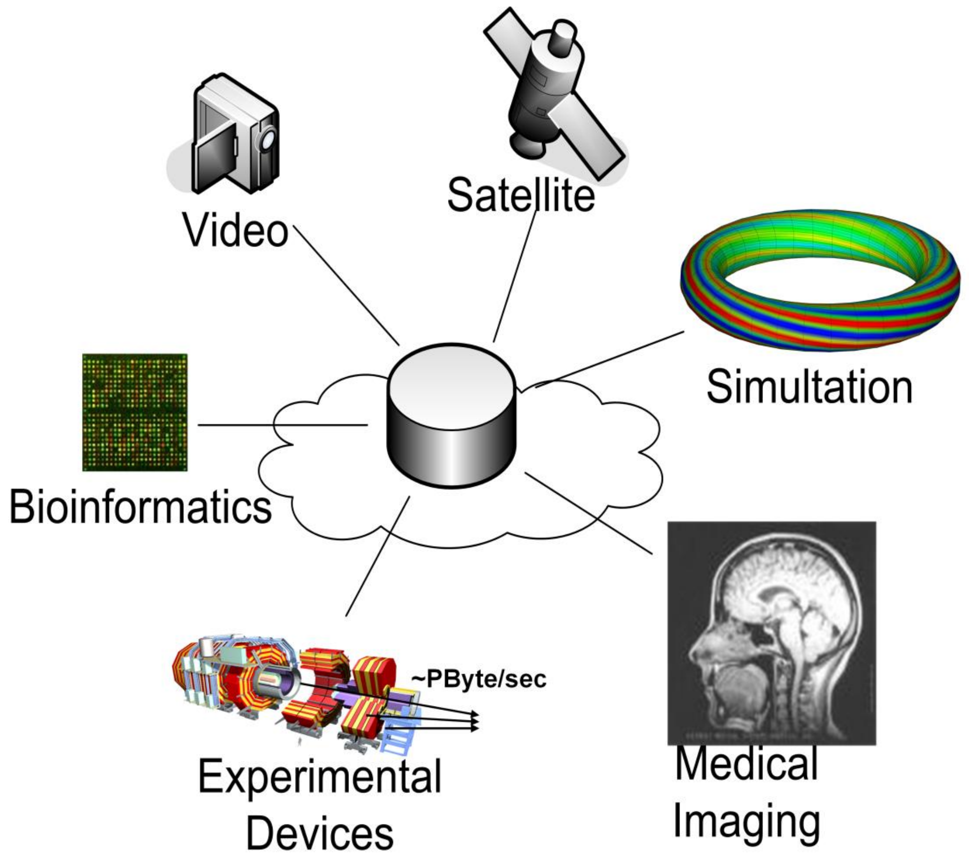
Overview

- Logistical Networking
- LStore Architecture
- Lessons Learned

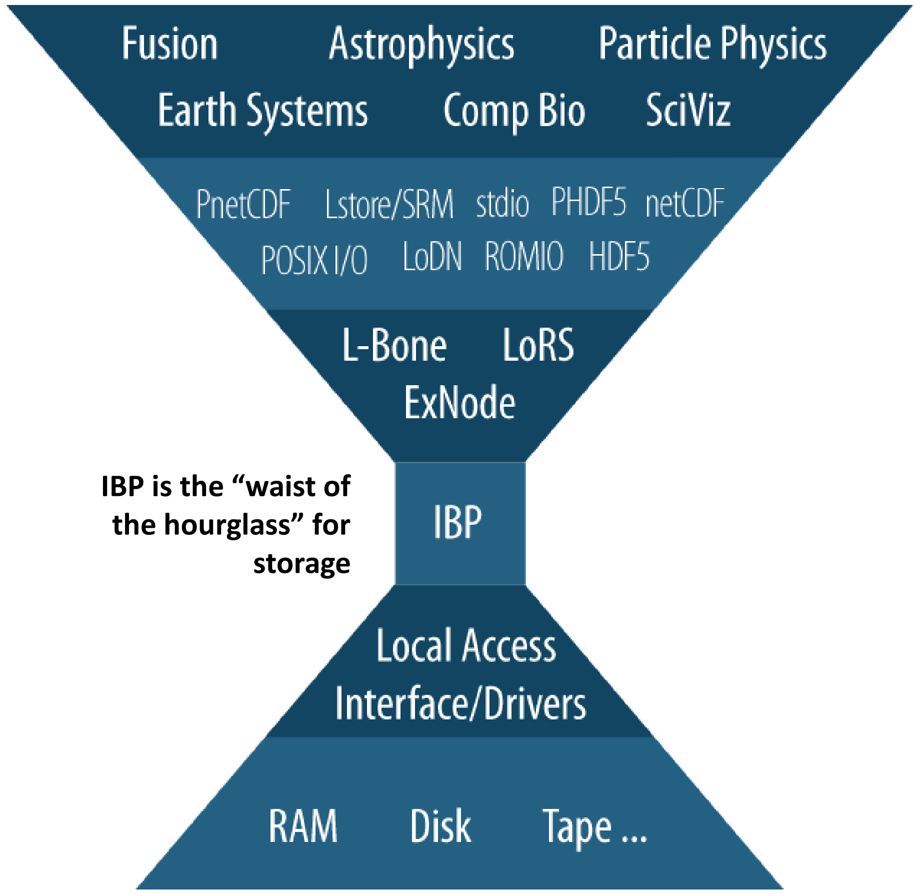
Logistical Netowking (LN) provides a “bits are bits” Infrastructure

- Standardize on what we have an adequate common model for
  - Storage/buffer management
  - Coarse-grained data transfer
- Leave everything else to higher layers
  - End-to-end services: checksums, encryption, error encoding, etc.
- Enable autonomy in wide area service creation
  - Security, resource allocation, QoS guarantees
- **Gain the benefits of interoperability!**

One structure to serve them all!



LoCI Tools* : Logistical Computing and Internetworking Lab

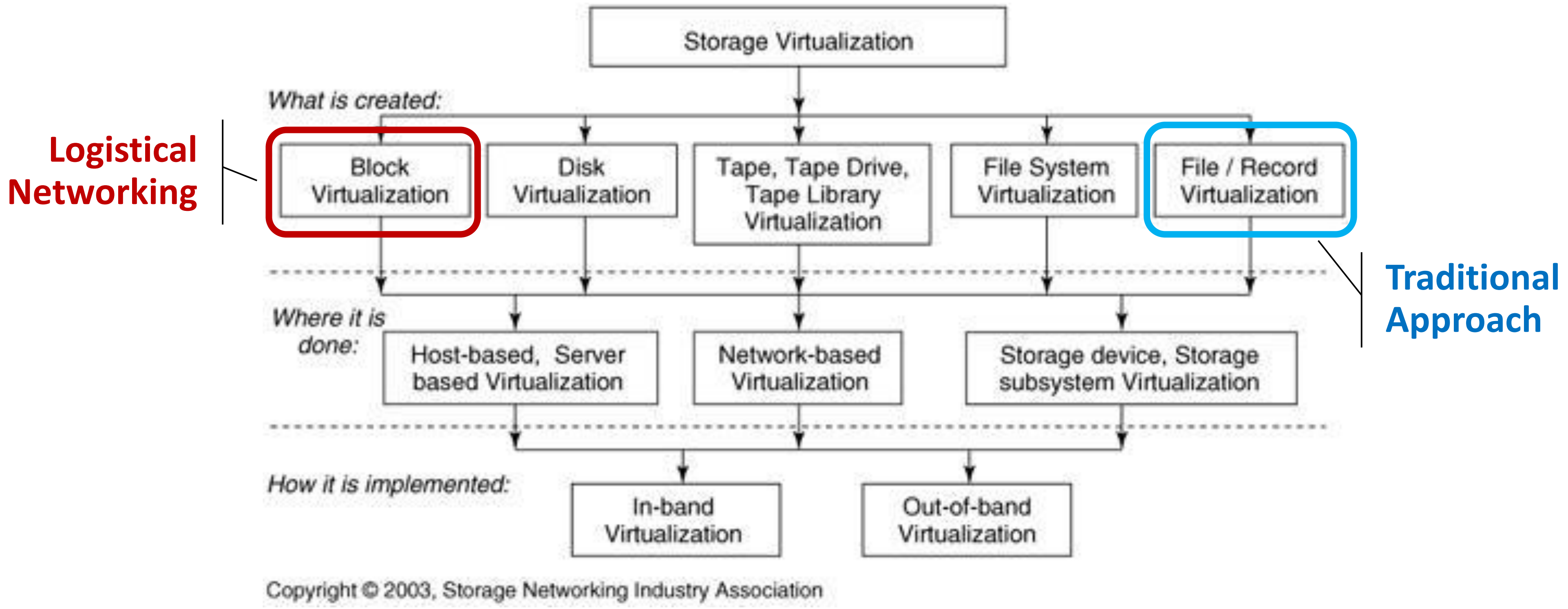


*http://loci.cs.utk.edu

- IBP Internet Backplane Protocol
  - Middleware for managing and using remote storage
  - Allows advanced space and **TIME** reservation
  - Supports multiple connections/depot
  - User configurable block size
  - Designed to support large scale, distributed systems
  - Provides global "*malloc()*" and "*free()*"
  - End-to-end guarantees
  - Capabilities
    - Each allocation has separate Read/Write/Manage keys

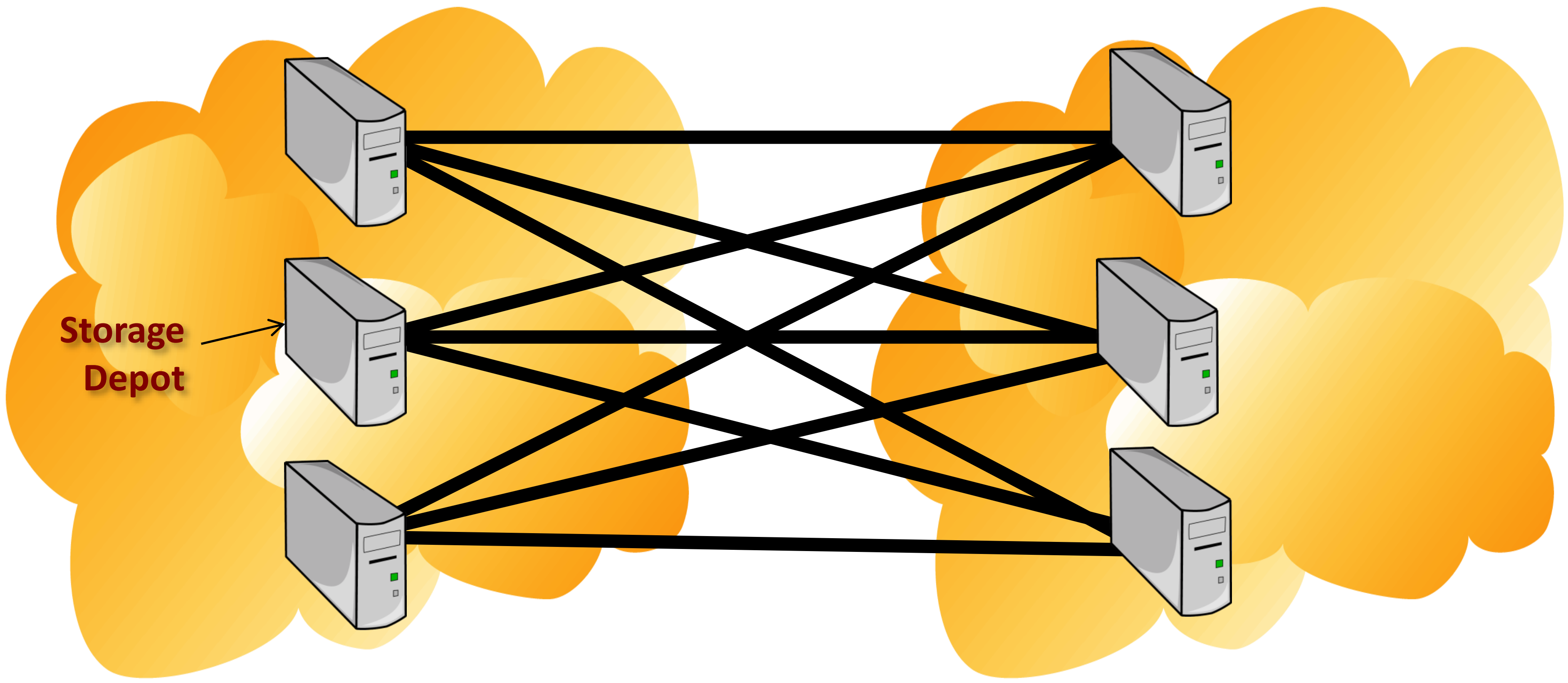
What makes LN Different?

Based on a highly generic abstract block for storage (IBP)


Logistical Networking
Traditional Approach
Storage Virtualization
What is created:
Block Virtualization
Disk Virtualization
Tape, Tape Drive, Tape Library Virtualization
File System Virtualization
File / Record Virtualization
Where it is done:
Host-based, Server based Virtualization
Network-based Virtualization
Storage device, Storage subsystem Virtualization
How it is implemented:
In-band Virtualization
Out-of-band Virtualization
Copyright © 2003, Storage Networking Industry Association

What makes LN Different?

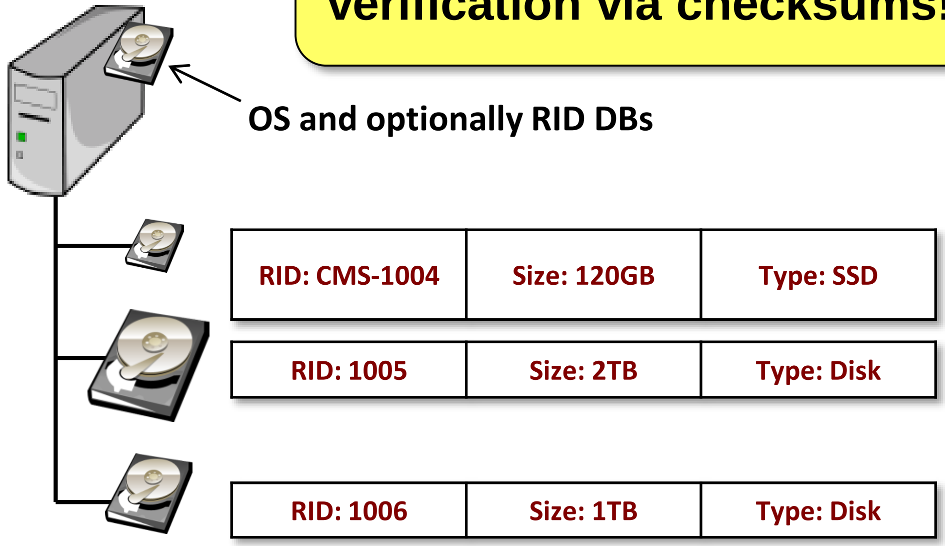
Based on a highly generic abstract block for storage (IBP)



IBP Server (Depot)

- Runs the ibp_server process
- Resource
  - Unique ID
  - Separate data and metadata partitions
  - Optionally can import metadata to SSD
- Typically JBOD disk configuration
- Heterogeneous disk sizes and types
- Don't have to use dedicated disks

Supports byte level R/W verification via checksums!



- **Current depots have 36 10TB drives and can sustain 15Gb/s (disk check summing to protect against bit rot) to 20Gb/s (no disk chksum)**
- **150Gb/s is the currently highest sustained transfer rate and was network limited.**

IBP Functionality

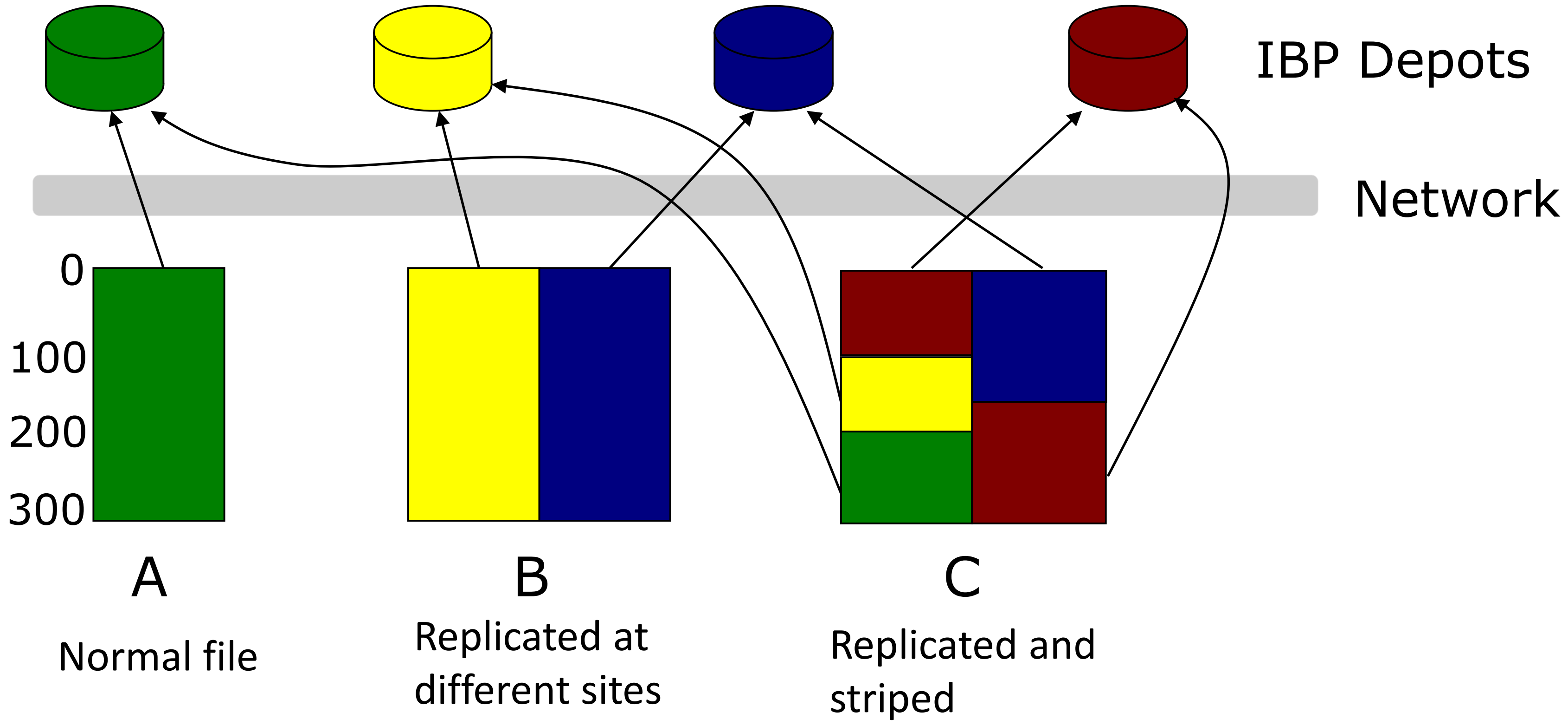
- Allocate
  - Reserve space for a limited time
  - Can create allocation **Aliases** to control access
  - Enable block level disk checksums to detect silent read errors
- Manage allocations
  - INC/DEC allocation reference count (when 0 allocation is removed)
  - Extend duration
  - Change size
  - Query duration, size, and reference counts
- Read/Write
  - Optionally use network checksums for end-to-end guarantees
  - Depot-Depot Transfers
  - Data can be either **pushed** or **pulled** to allow firewall traversal
  - Supports both append and random offsets
  - Pheobus support – I2 overlay routing to improve performance
- Depot Status
  - Number of resources and their resource ID's
  - Amount of free space
  - Software version

What is a “capability”?

- Controls access to an IBP allocation
- 3 separate caps/allocation
  - Read
  - Write
  - Manage - delete or modify an allocation
- Alias or Proxy allocations supported
  - End user never has to see true capabilities
  - Can be revoked at any time

exNode: File metadata object

- Analogous to a disk I-node and contains
  - Allocations
  - How to assemble file
  - Fault tolerance encoding scheme
  - Encryption keys



LStore Architecture

- **Exnode**
  - Collection of data blocks and segments to provide different views of data
  - Different segments can be used for versioning, replication, optimized access (row vs. column), etc.
  - **Segment**
    - Collection of blocks with a predefined structure.
    - Type: Linear, LUN, RAID5, Generalized Reed-Solomon, Log, caching, etc.
    - Can be stacked with other segments
  - **Resource Service**
    - Data placement(stripe across depots vs. across disks) and lookup
    - Boolean query expression
  - **Data Service**
    - Performs the actual data operations
    - Currently only IBP is supported
- **Object Service**
  - Metadata operations
  - Full support for streaming operations to minimize latency effects. Most operations can be implemented with a single call (ls -l, mkdir, find, etc)

Lstore Command Line Tools

The Logistical Input/Output (LIO) command line tools are designed to replicate the normal Linux File System Tools

Linux

lio_cp	lio_mkdir
lio_du	lio_mv
lio_find	lio_rm
lio_fsck	lio_rmdir
lio_ln	lio_touch
lio_ls	

LIO Specific

lio_fuse	Load the FUSE mount
lio_get	Equivalent to cat
lio_getattr	Get a file attribute
lio_inspect	Checks integrity/repairs a file
lio_put	Pipe from stdin to a file
lio_setattr	Set a file attribute
lio_signature	Prints exnode container structure
lio_warm	Extends the file expiration date

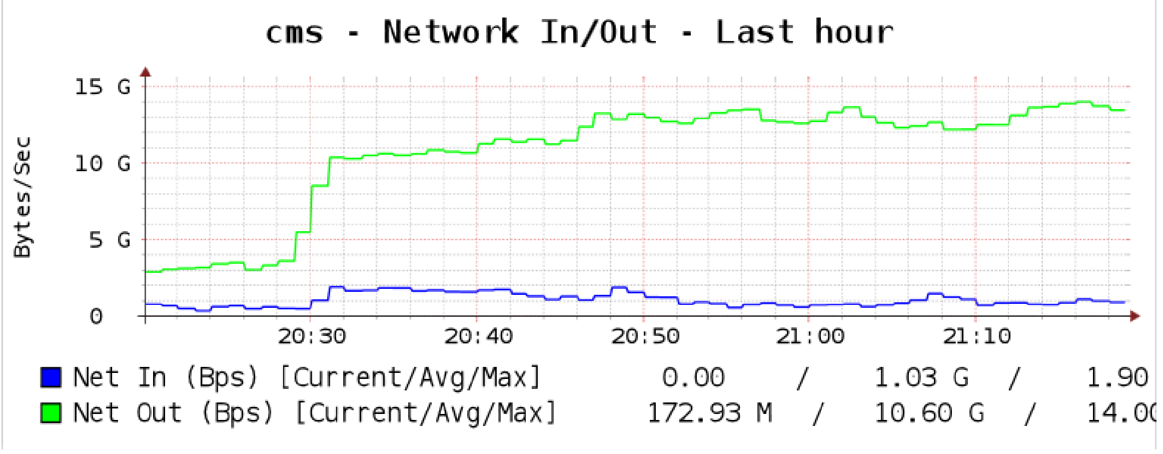
Linux FUSE client

- Hard and Symbolic link supported.
- Full support for *xattr* interface.

- Tape backup and restore of both native LStore and extended attributes.
- Can be used as a Tape backup disk cache.
- Integration with Teradactyl backup software.

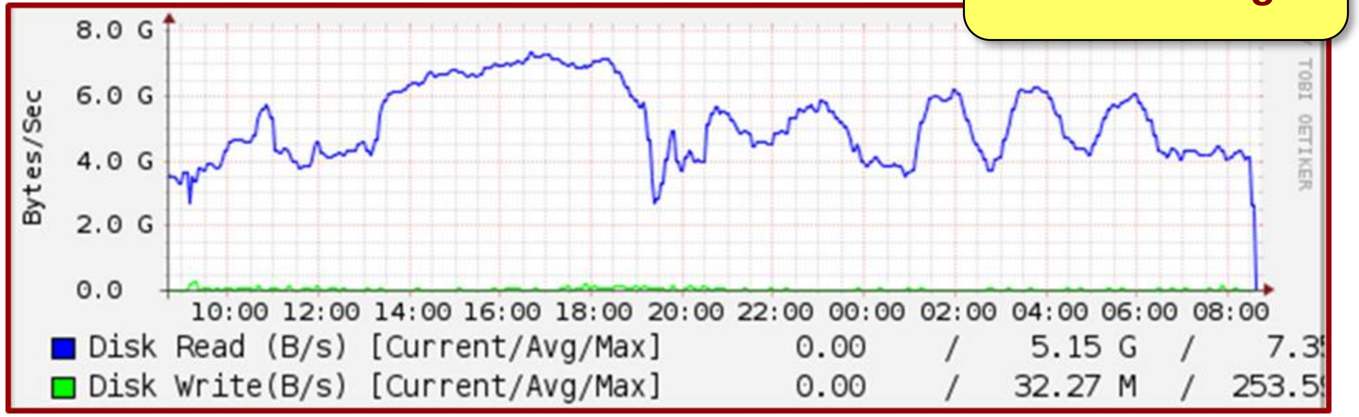
LStore Usage at VU

- CMS-HI has 10PB of native disk space
  - Routinely read over 1PB/week using the FUSE mount for production jobs and sustain 120Gb/s read rates.
- 600TB of shared space at VU
  - Vanderbilt TV News Archive has 400+ TB of space



```
■ Net In (Bps) [Current/Avg/Max]        0.00      /    1.03 G   /    1.90
■ Net Out (Bps) [Current/Avg/Max]     172.93 M    /   10.60 G   /   14.00
```

24 hour plot of CMS-HI usage



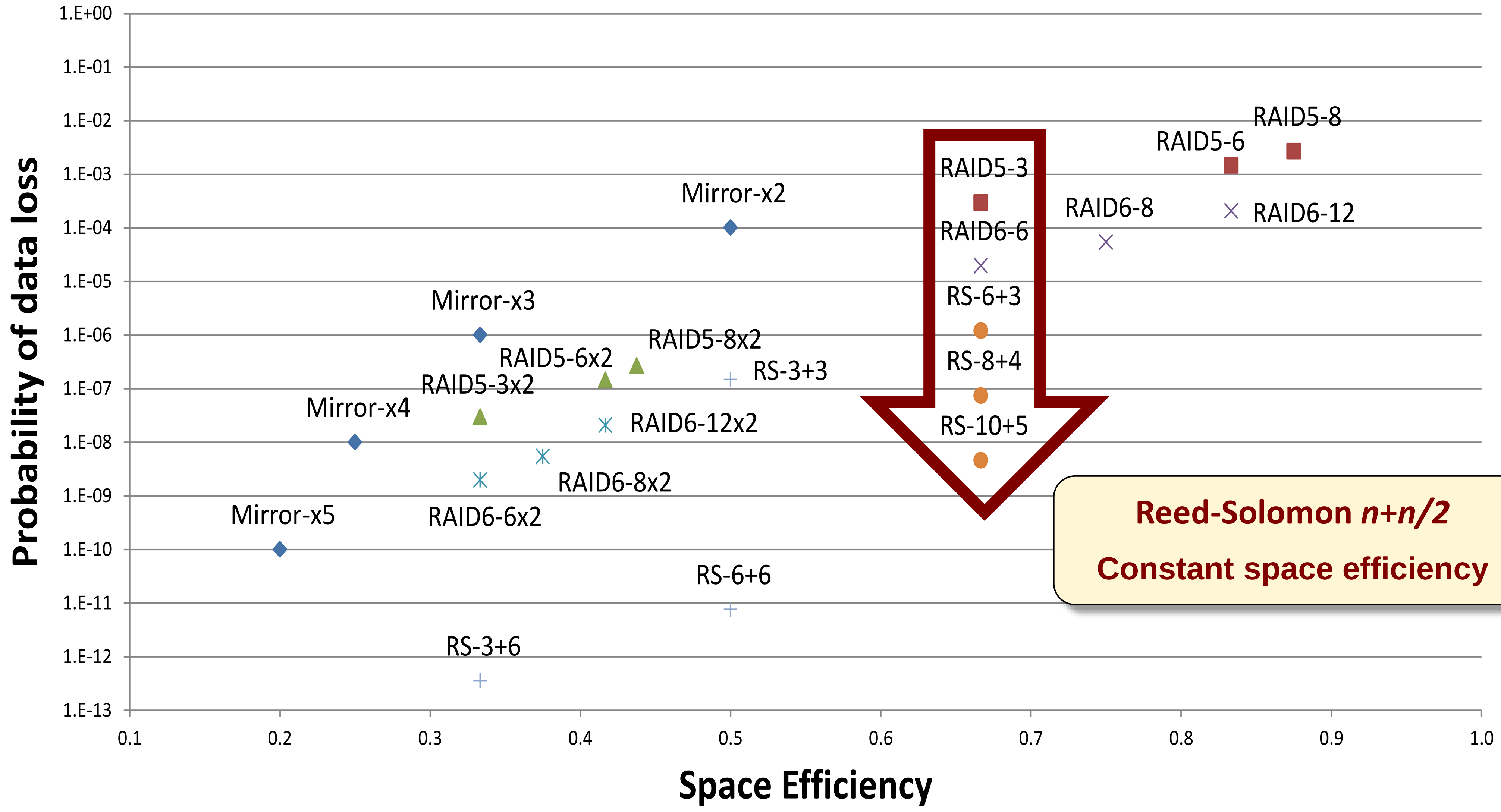
```
■ Disk Read (B/s) [Current/Avg/Max]       0.00    /    5.15 G   /    7.3
■ Disk Write(B/s) [Current/Avg/Max]       0.00    /   32.27 M   /  253.5
```

Hardware Lessons

- Mechanical Vibrations
  - Disappearing drives
  - Broken electrical traces
  - Pre-mature drive failure
- Buggy Firmware
  - Drive hard locks requiring a power cycle
- Drive fragmentation
- Drive failures (6-8% AFR)
  - Replaced 50+ drives in the last 3 months

- Data integrity
  - Bit rot – Errors not detected by the drive.
  - Unrecoverable bit errors
    - typically 10^{14} or 10^{15} bits
    - 1PB = 8×10^{15} bits
  - Journal replay
    - Metadata recovery is primary goal with data being secondary
    - Most common form of data corruption

Probability of data loss vs Space efficiency

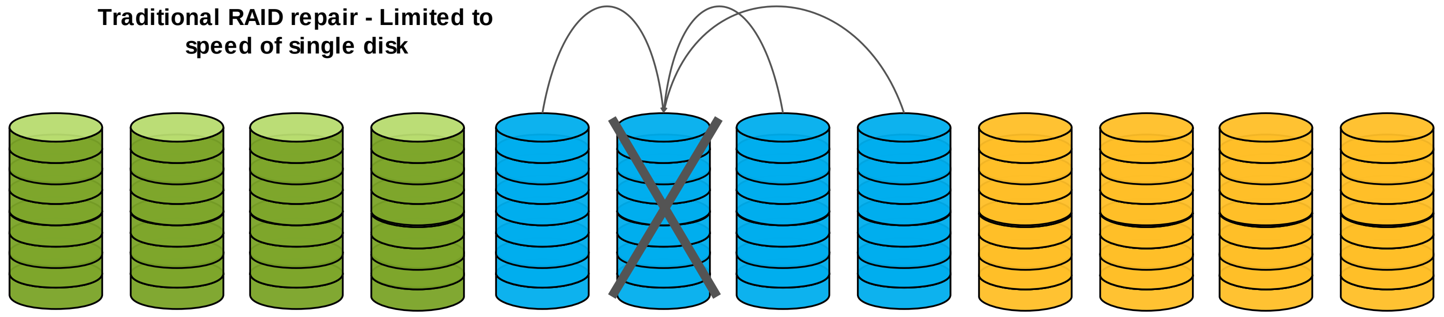


Reed-Solomon *n+n/2*

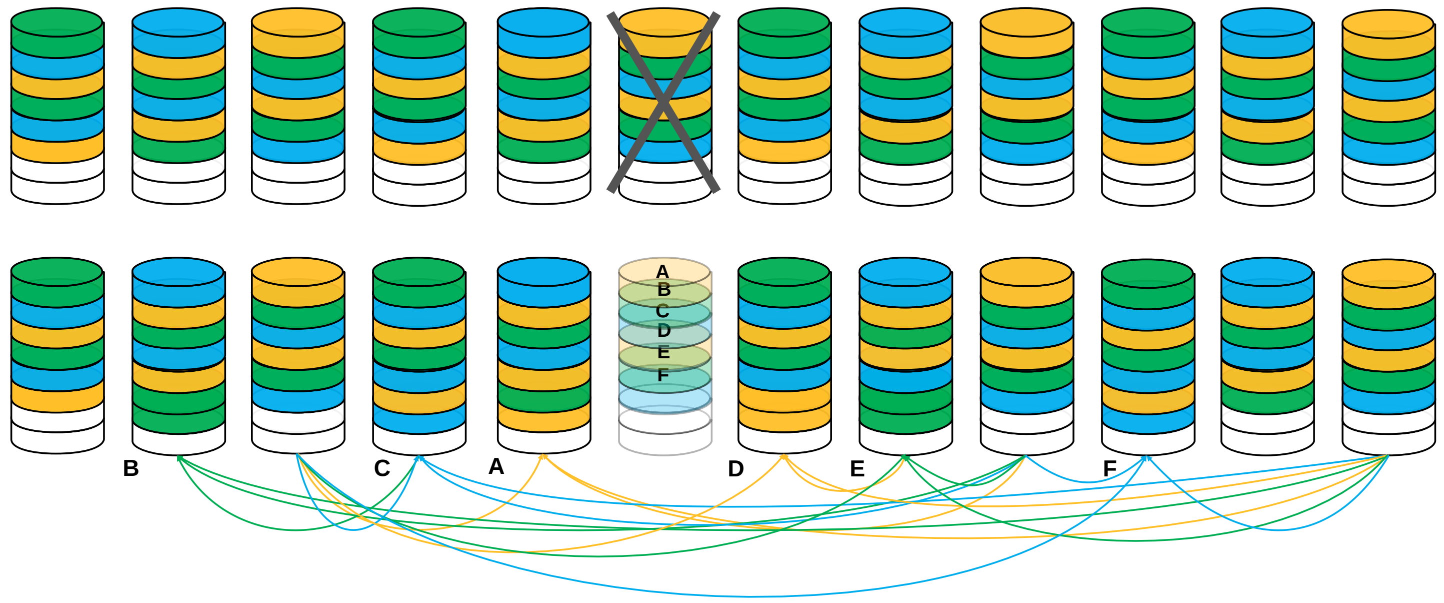
Constant space efficiency

Distributed or De-clustered

Traditional RAID repair - Limited to speed of single disk



Distributed RAID has logical RAID arrays using many disks



The Logical arrays residing on the failed physical disk are repaired to remaining disks. Only data used is repaired. Many more disks take part in the repair.

Questions?

- LN provides a generic block level storage abstraction
- LStore provides a highly scalable, fault tolerant file system