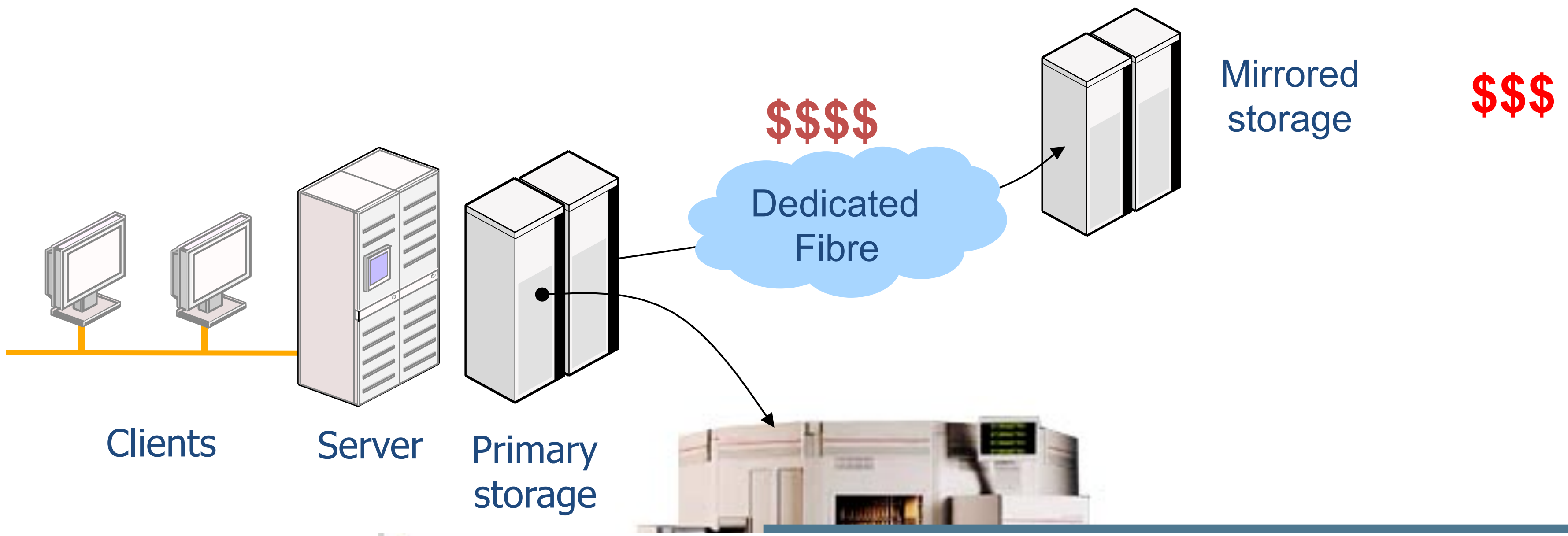
Deduplication File System
&
Course Review

Kai Li

Topics

- Deduplication File System
- Review

Storage Tiers of A Traditional Data Center



US bank loses details of 4.5 million customers

Social security numbers and birthdates are among the data lost by the Bank of New York Mellon Corp

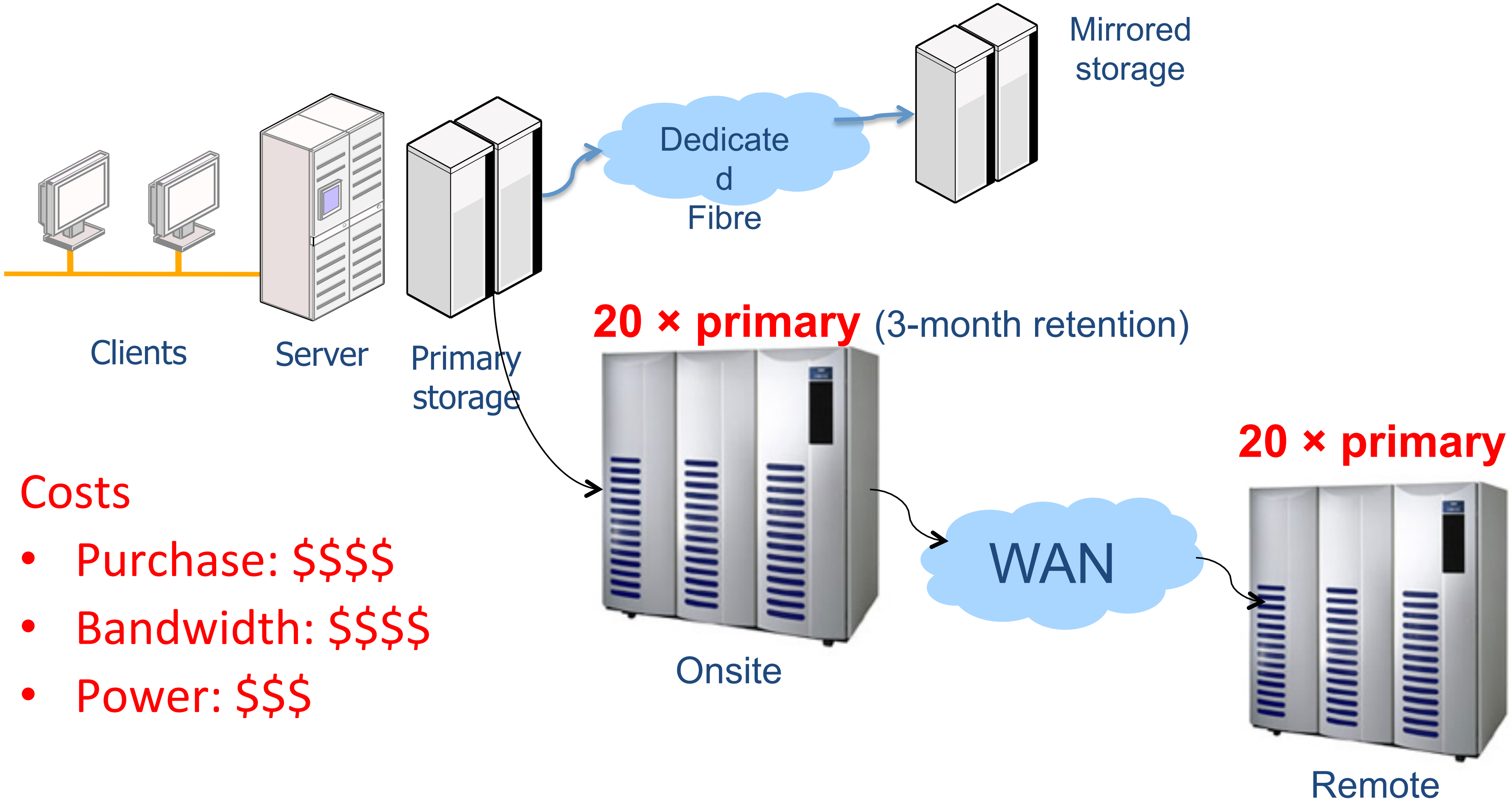
Written by Neon Kelly
Computing, 02 Jun 2008

The details of over 4.5 million customers have gone missing at the Bank of New York Mellon

The Bank of New York Mellon Corporation has admitted to misplacing the details of 4.5 million customers, following the loss of a data tape earlier this year.

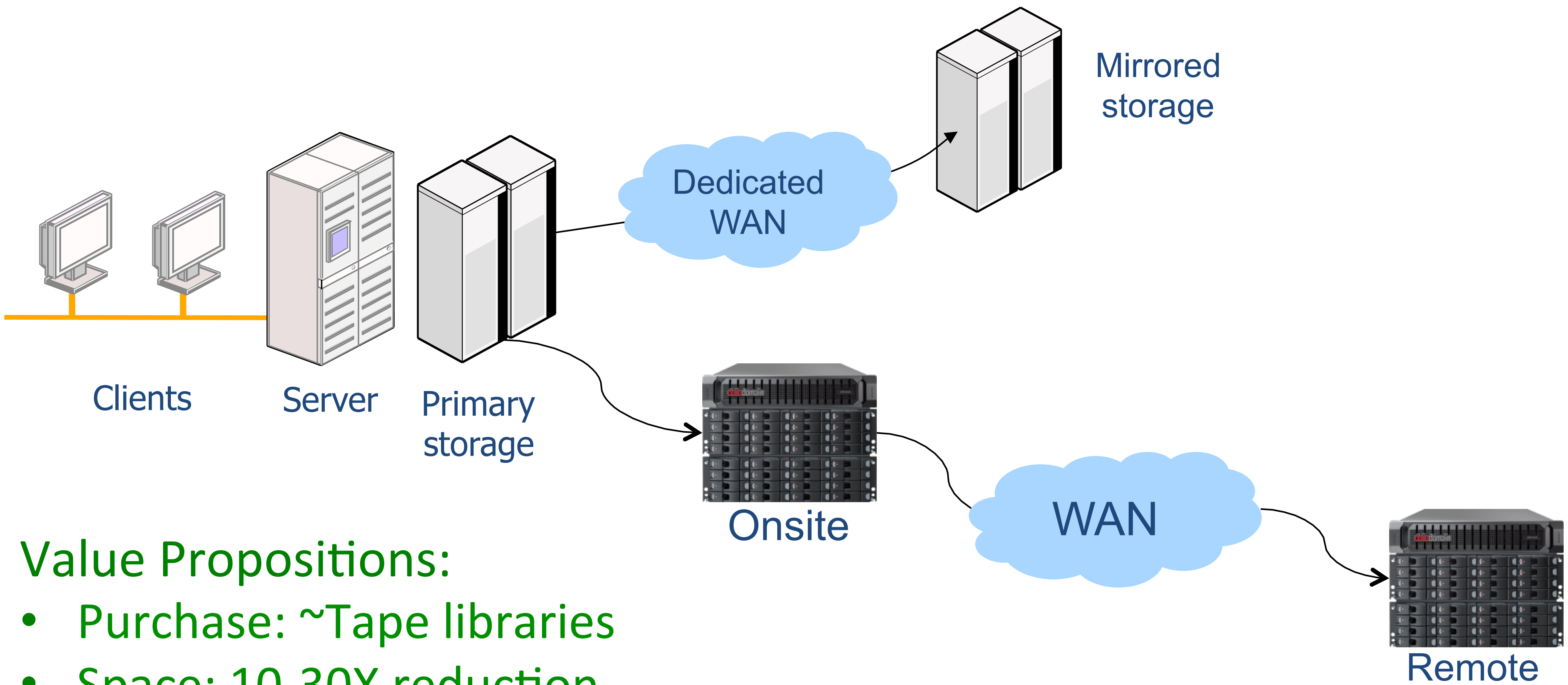
The backup tape went missing on 27 February while being transported to an off-site archive by a third-party vendor. The lost data includes the names, birthdates and social security numbers of customers of the Bank of NY Mellon and the People's United Bank in Bridgeport, Connecticut.

Replacing Tape Library using Disk Storage?



Costs
- Purchase: $$$$
- Bandwidth: $$$$
- Power: $$$

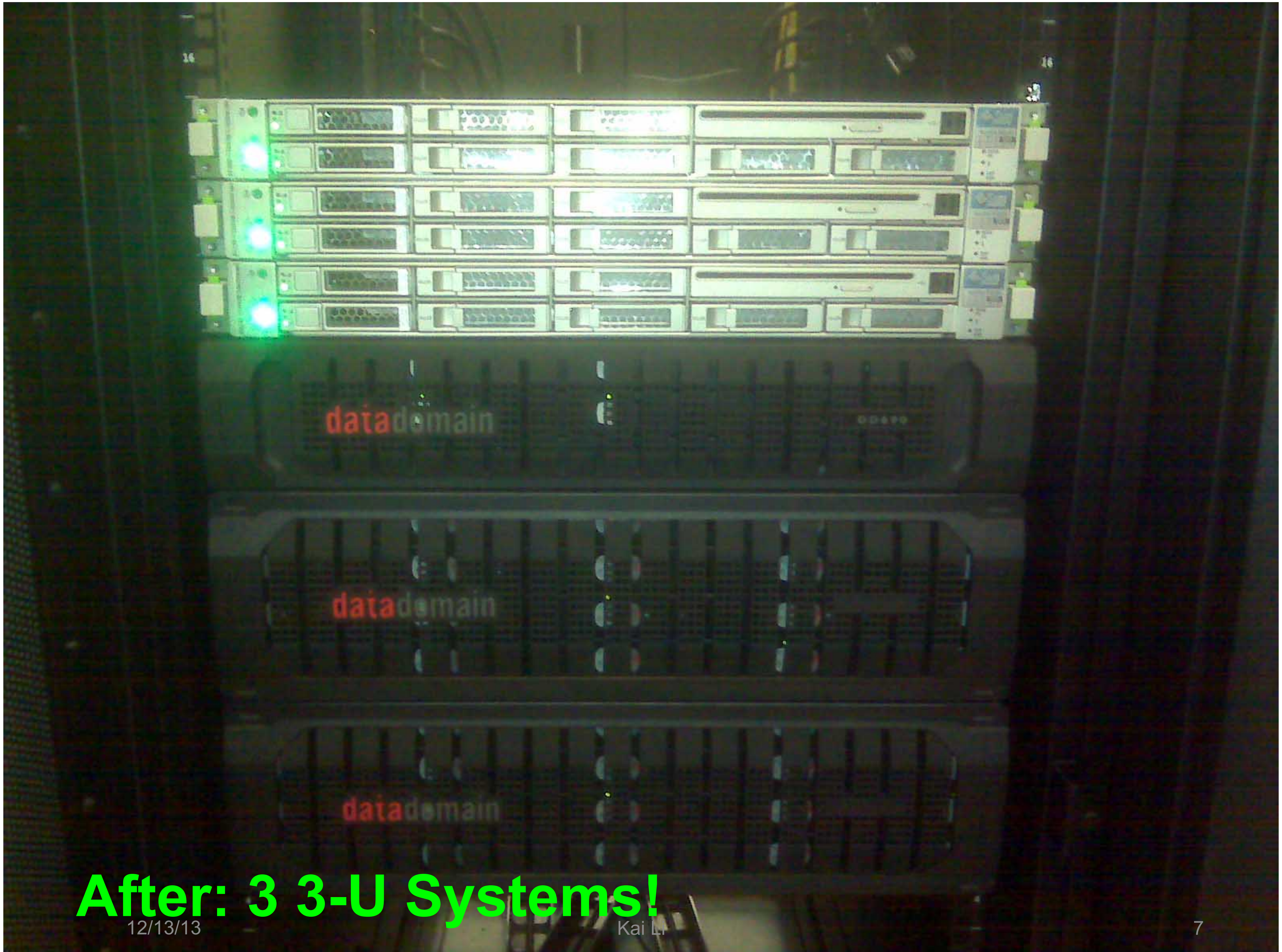
Vision: **Deduplication Storage** Eco-System



Value Propositions:

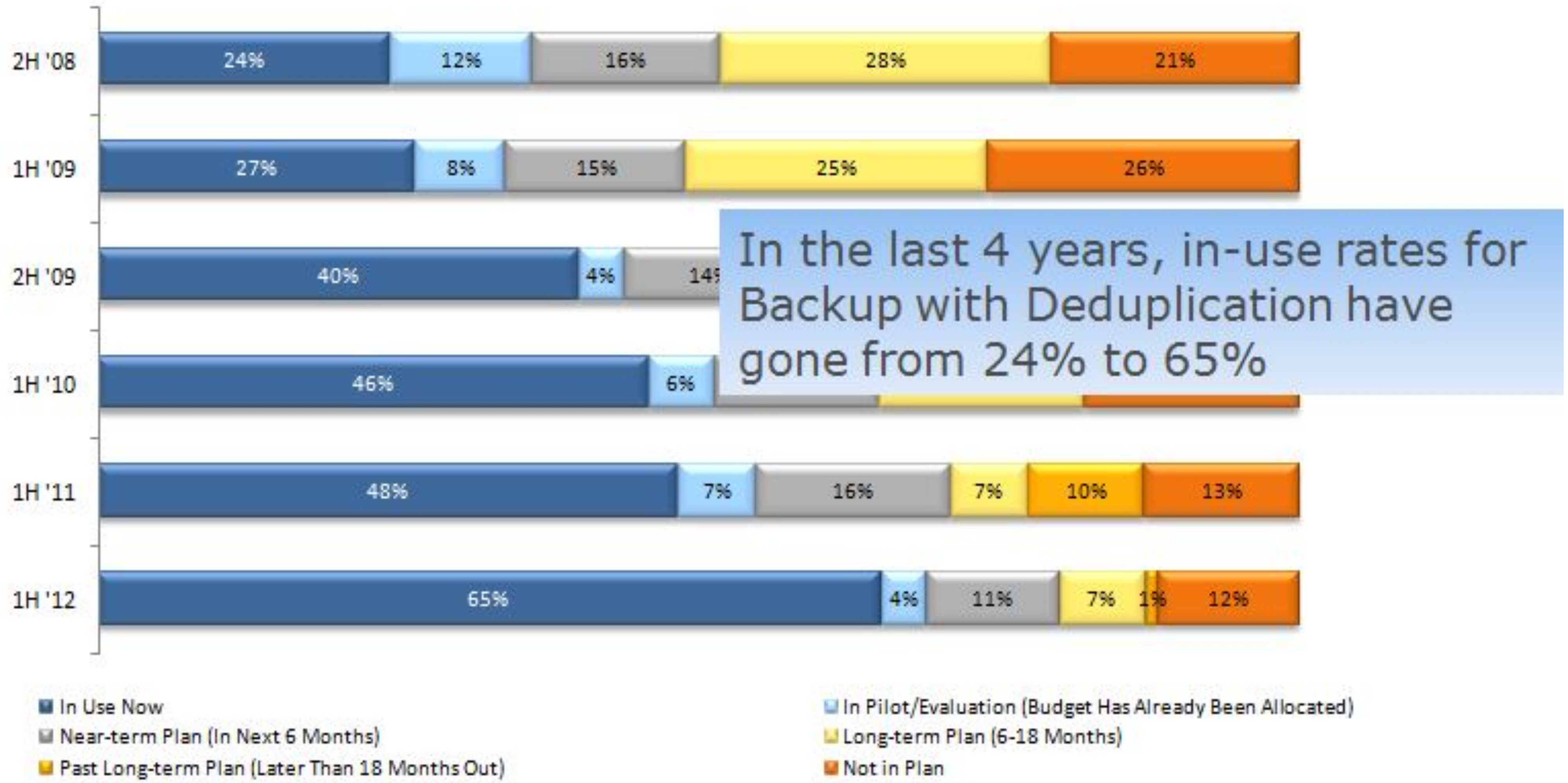
- Purchase: ~Tape libraries
- Space: 10-30X reduction
- WAN BW: 10-50X reduction
- Power: ~10X reduction

Before: 17 Tape Libraries


datadomain
datadomain
datadomain
After: 3 3-U Systems!

DD Protected 26EB & Replaced 33M Tapes

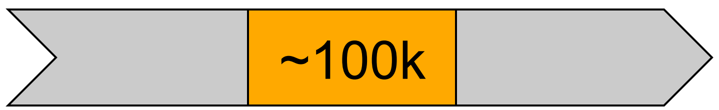
- EMC Blog by C. Gordon (4/26/2013)



2H '08, n=127; 1H '09, n=147; 2H '09, n=182; 1H '10, n=146; 1H '11, n=31; 1H '12, n=181. TheInfoPro, Wave 16 Storage Study – 1H 2012, published June 2012 (ww.theinfopro.com)

Local vs. Global Redundancies

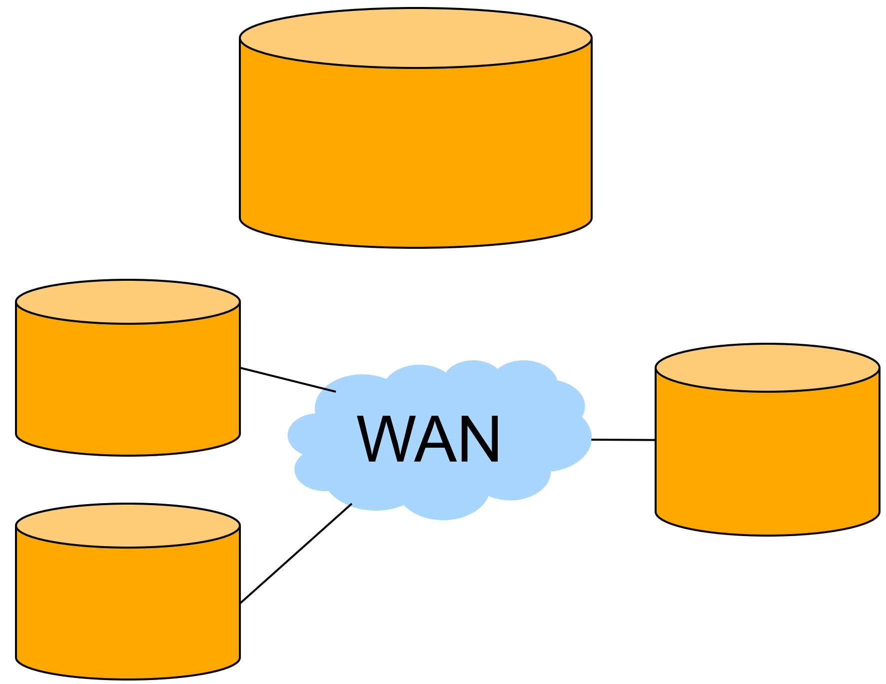
“Local” compression



Encode a sliding window [Ziv&Lempel77]

~2-3X compression

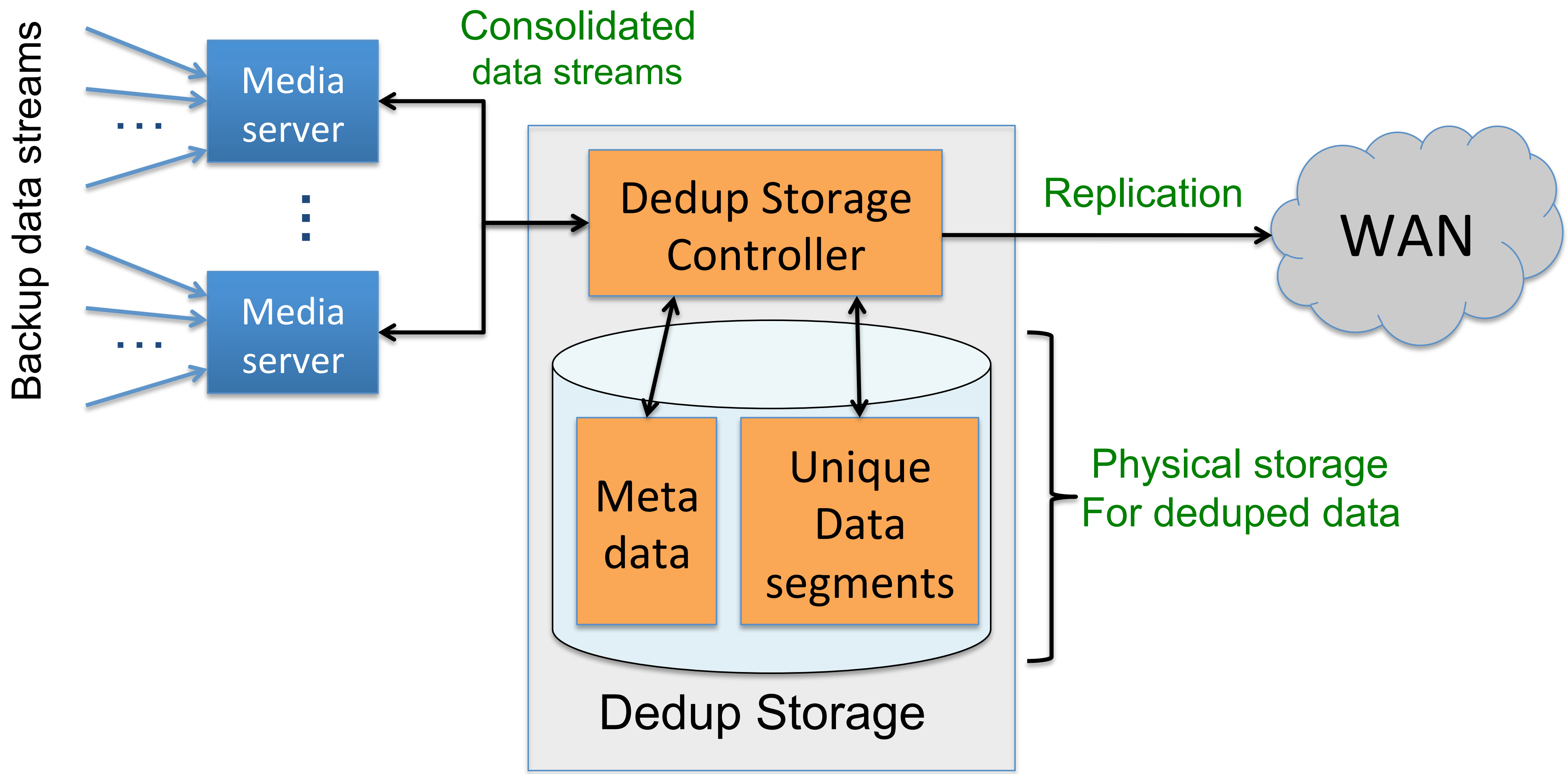
Deduplication



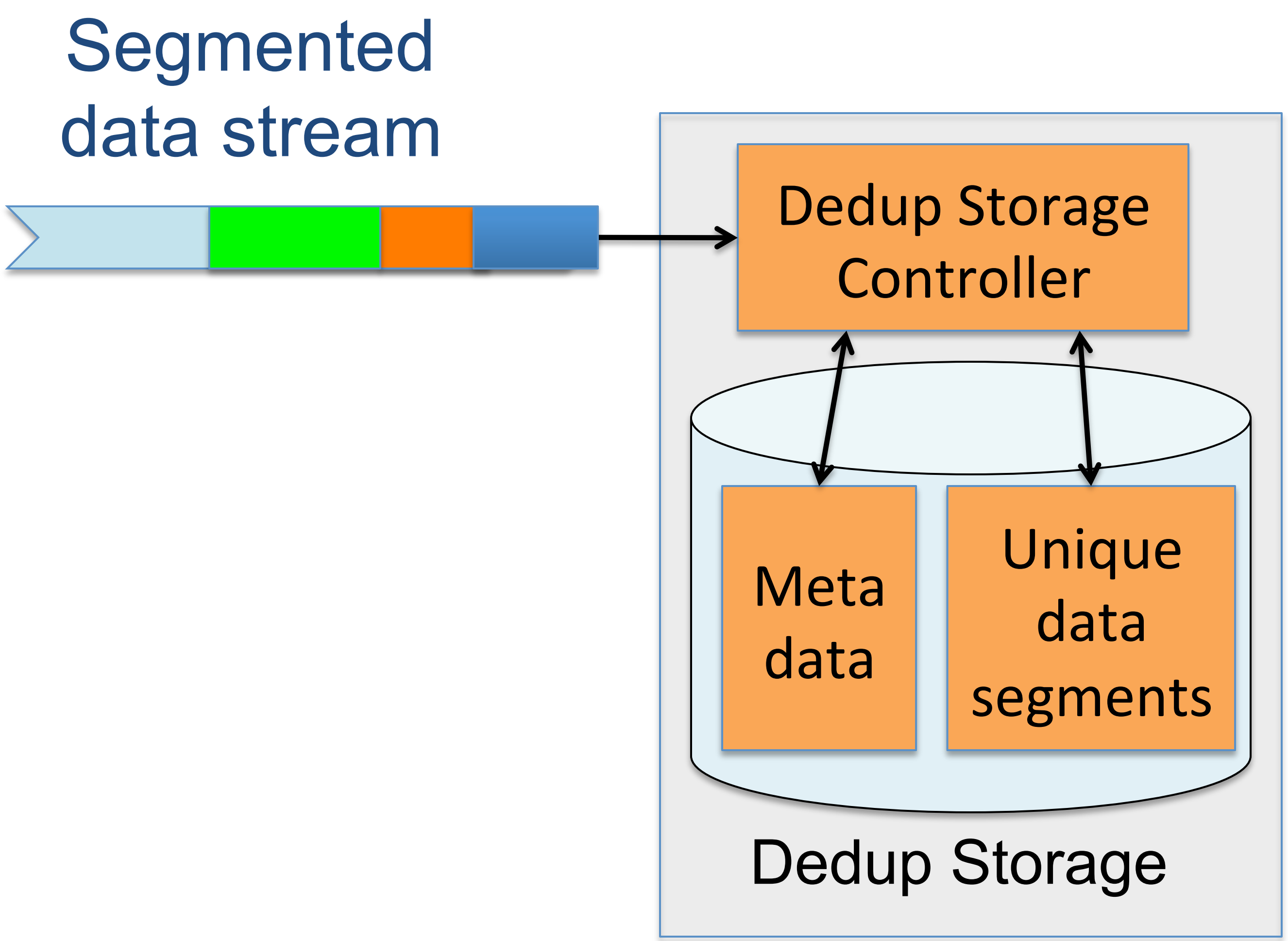
~10-30X compression

Larger “windows” have more redundancies

Dedup Storage System for Backups



How Does It Work?

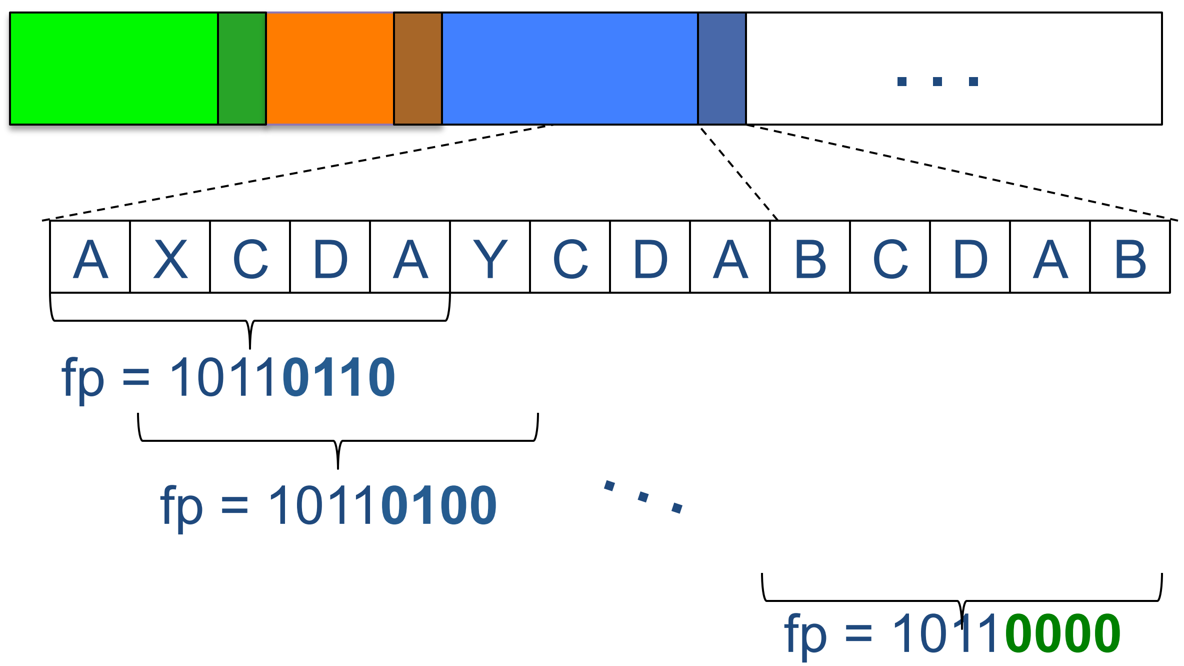


Fixed vs. Variable Segmentation

- Fixed size

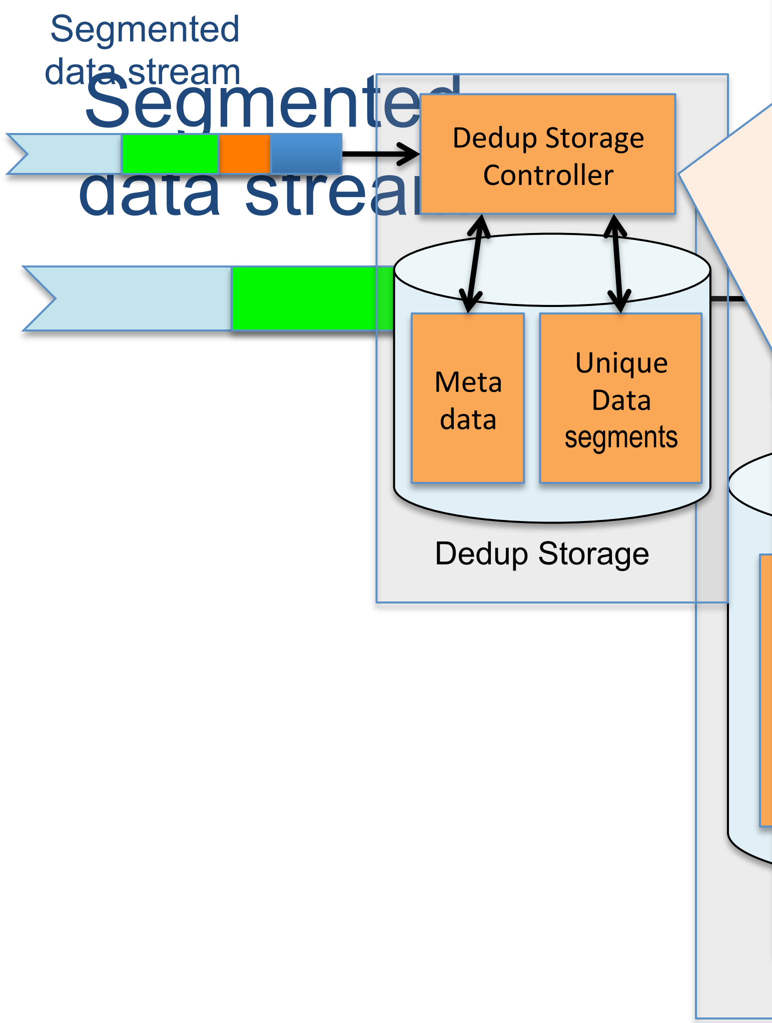
Cannot handle deletes, shifts

- Content-based, variable size



No problem w/ deletes, shifts [Manber93, Brin94]

More Details



For each segment

- Compute a strong fingerprint
- Use an index to lookup
  - If unique
    - **Locally compress the segment**
    - store segment
    - store meta data
  - If duplicate
    - store meta data

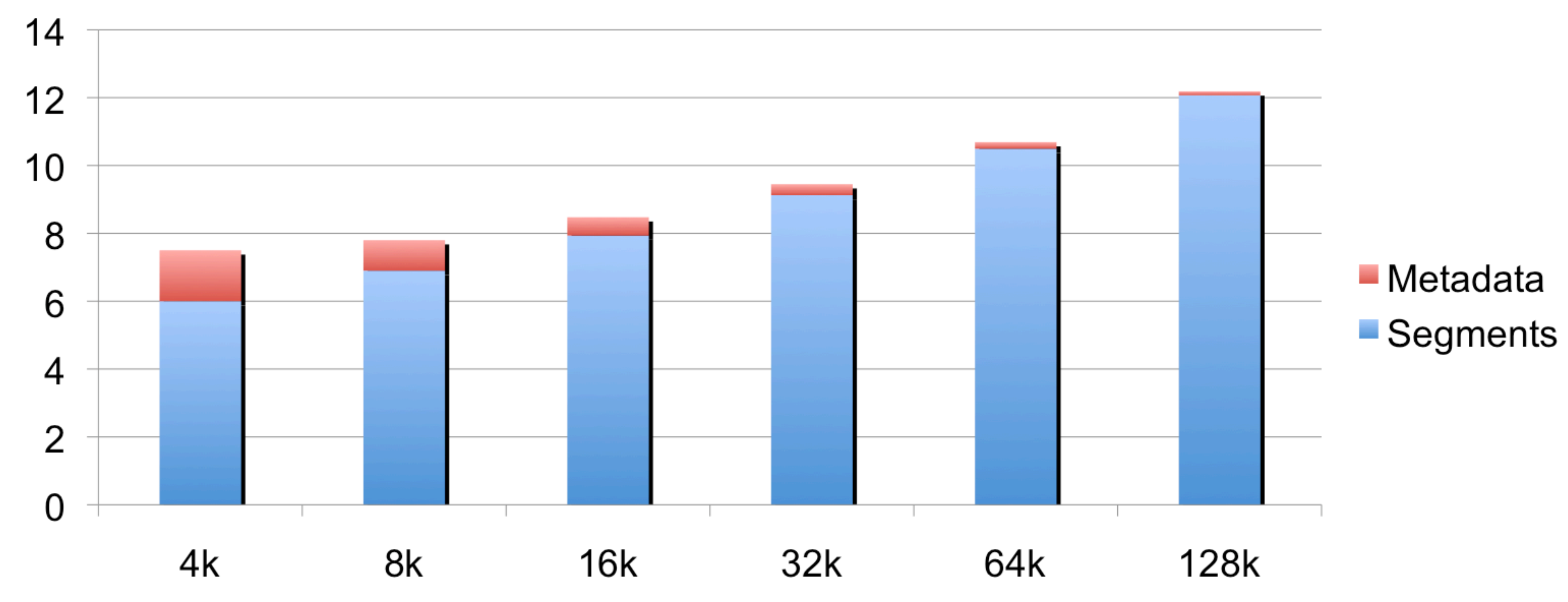
Design Challenges

- Very reliable and self-healing
  - Corrupting a segment may corrupt multiple files
  - NVRAM to store log (transactions)
  - Invulnerability features:
    - Frequent verifications
    - Metadata reconstruction from self-describing containers
    - Self-correction from RAID-6
- High-speed high-compression at low HW cost
  - Speed challenge: data 2X/18 months and 24 hours/day
  - Compression: reduce cost
  - Use commodity server hardware

High Deduplication Ratio

- High deduplication factor → hardware cost
  - Smaller segments achieve higher compression ratios?
  - Smaller segments result in higher ratio of metadata to physical segments
- Data Domain's approach
  - Use the sweet spots of segment sizes
  - Multiple local compression algorithms

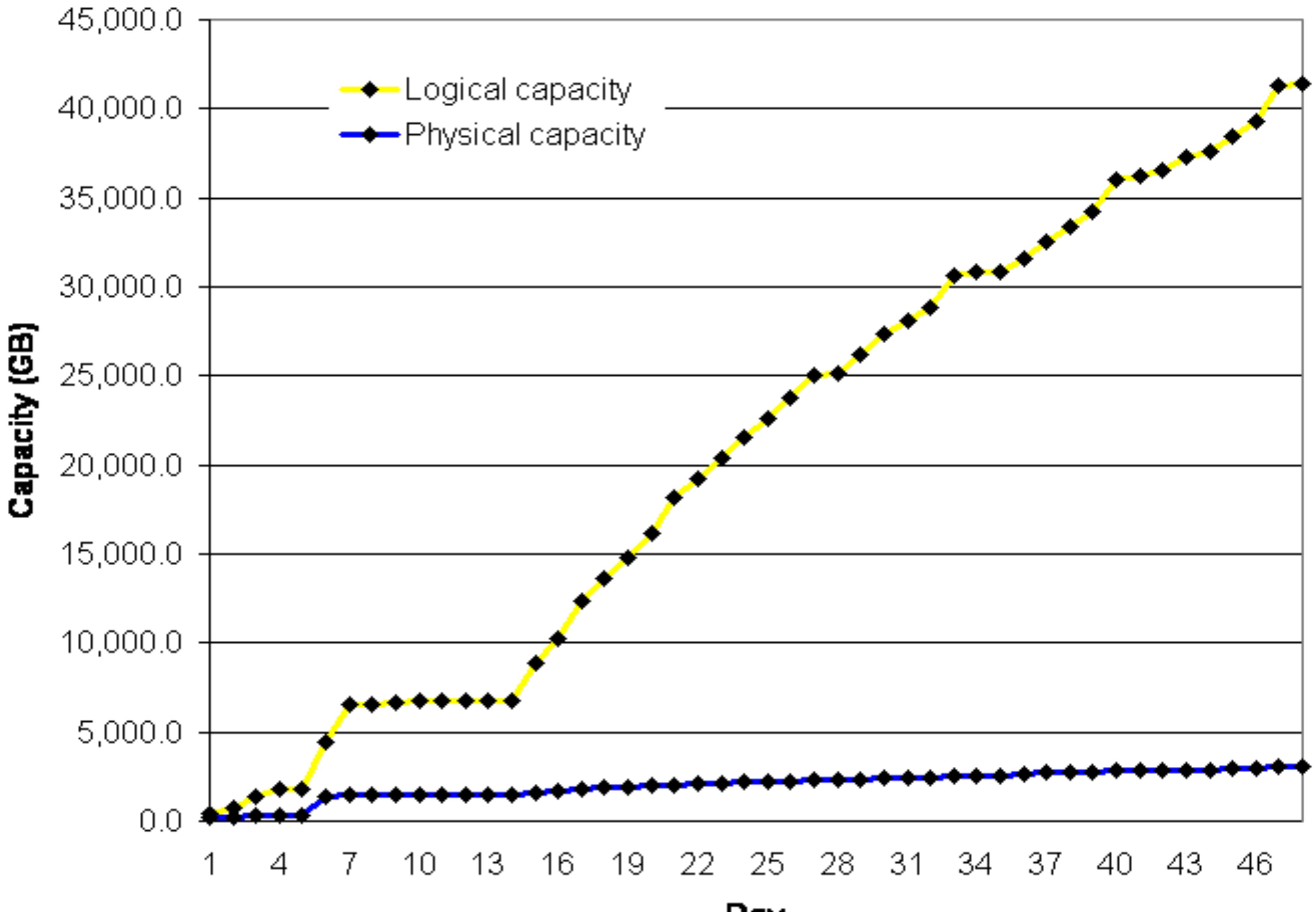
Segment Sizes for Backup Data



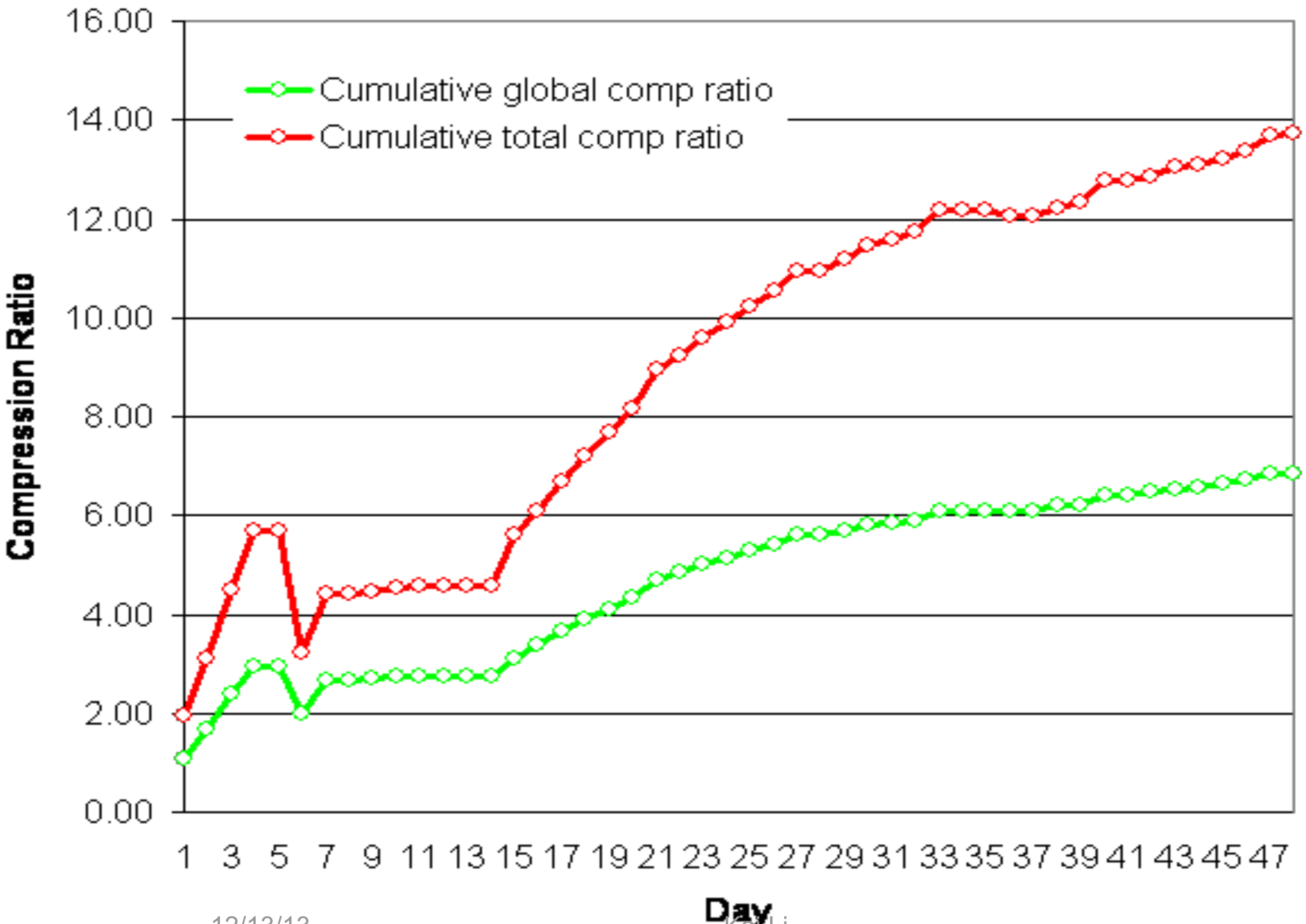
Rule of thumb: 2X segment size will

- increase space for unique segments by 15%
- decrease metadata by about 50%
- deduce disk I/Os for writes and reads

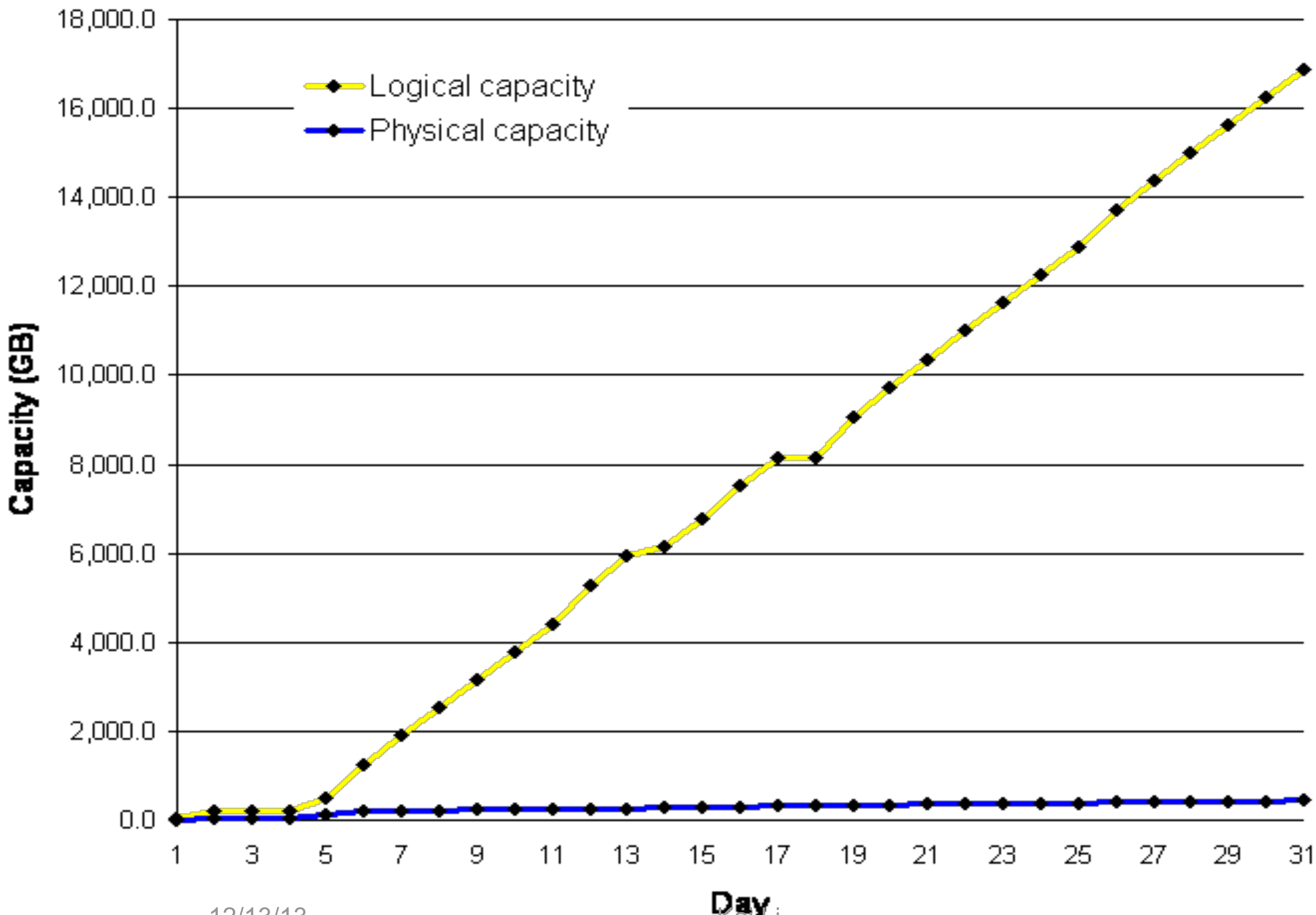
Real World Example at Datacenter A



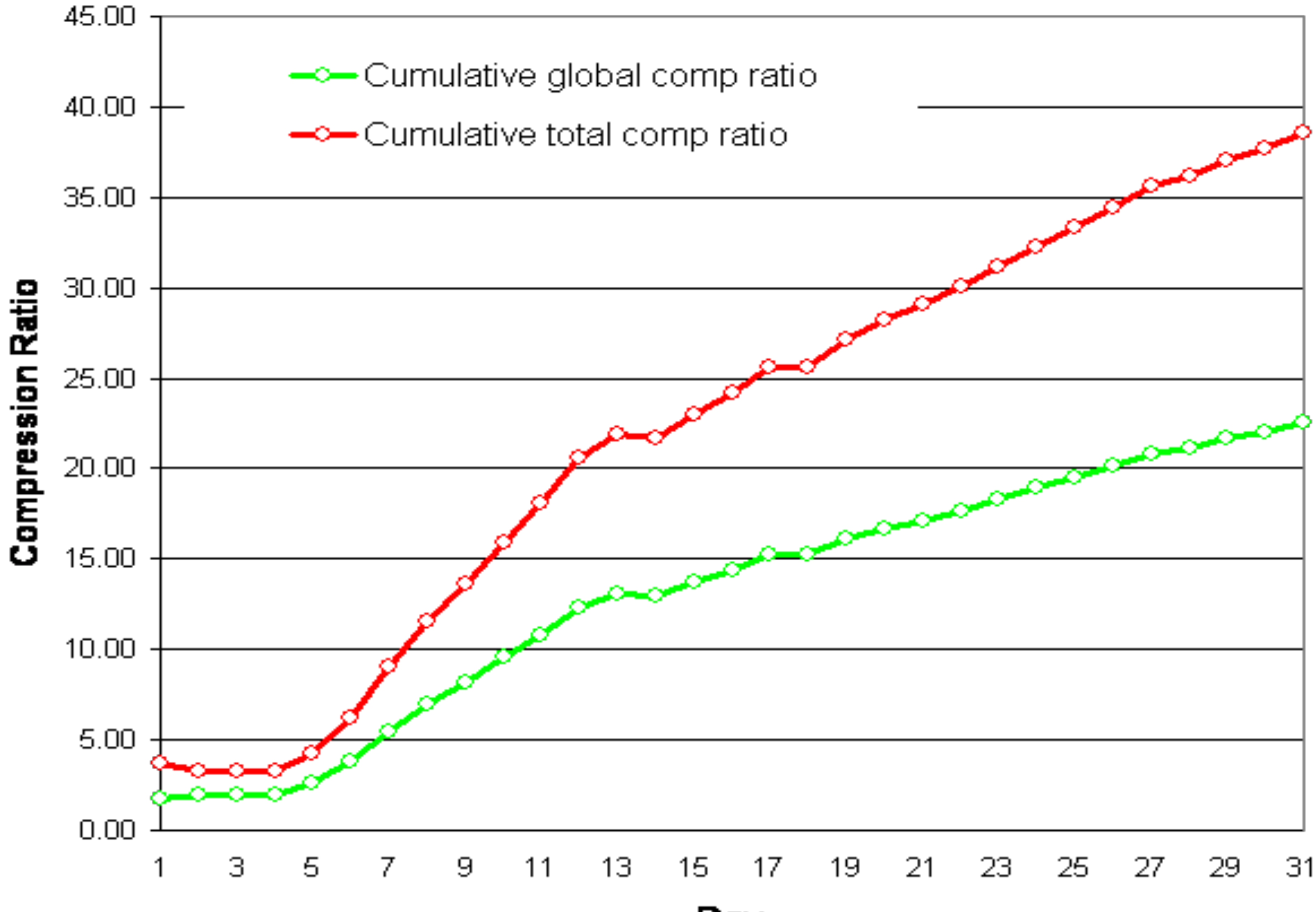
Real World Compression at Datacenter A



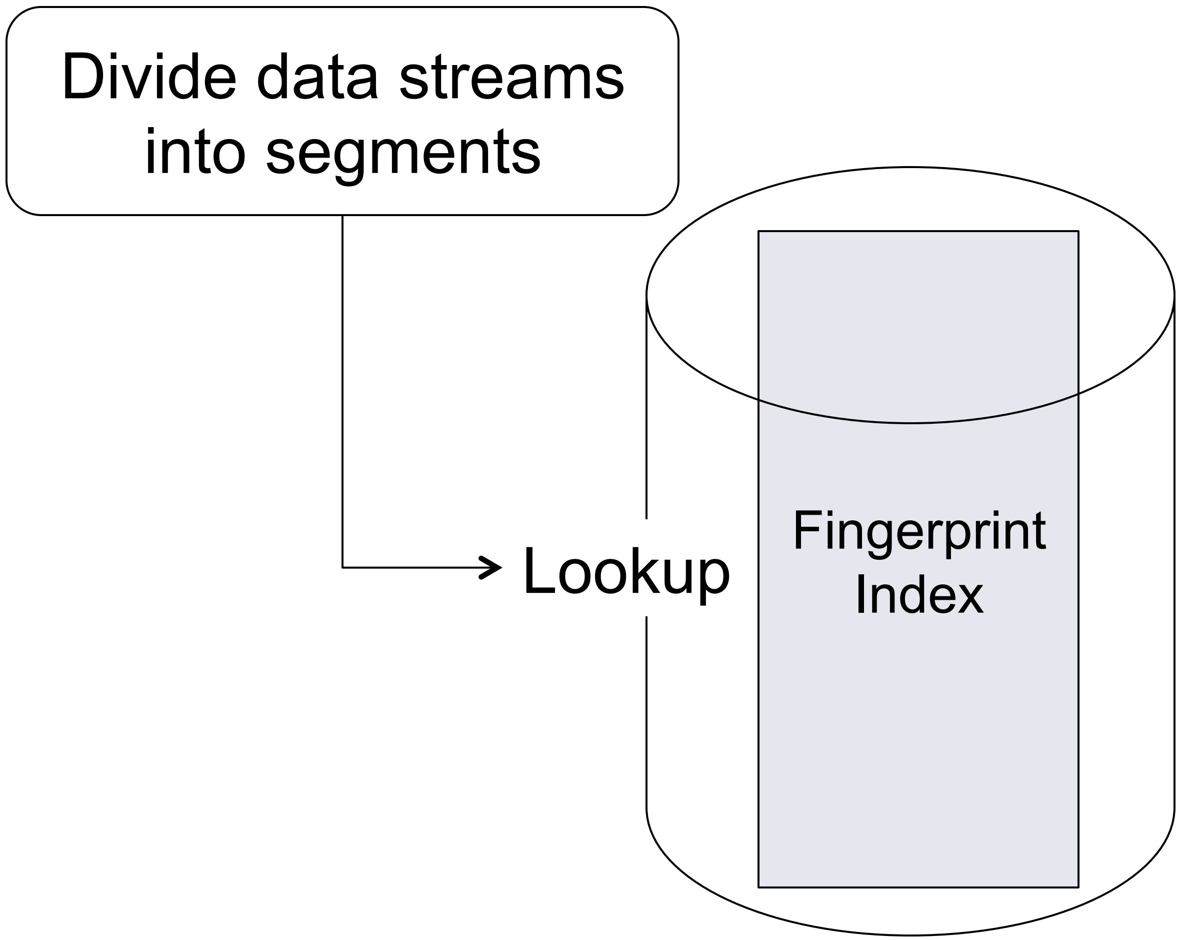
Real World Example at Datacenter B



Real World Compression at Datacenter B

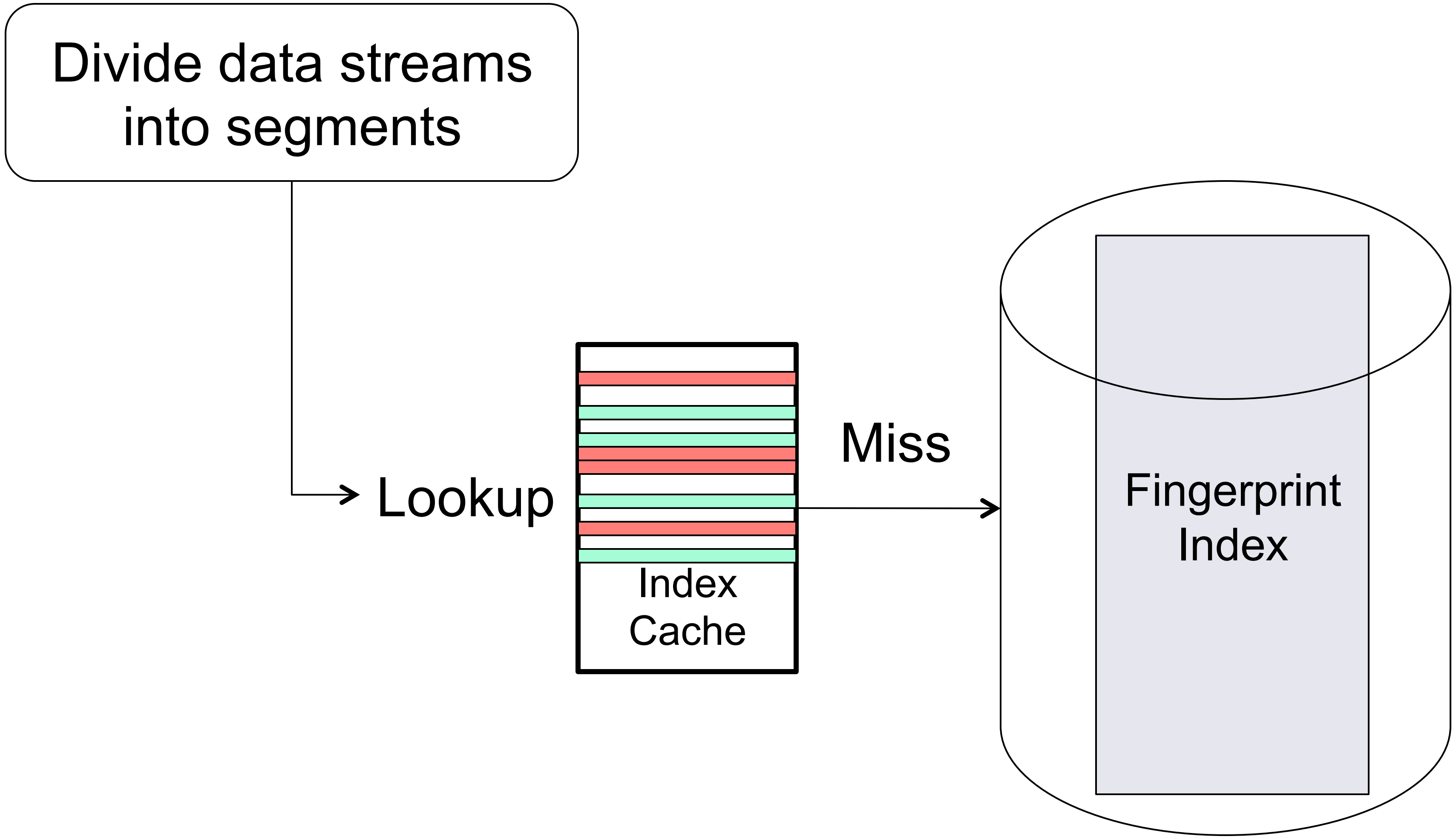


High Throughput Is Challenging



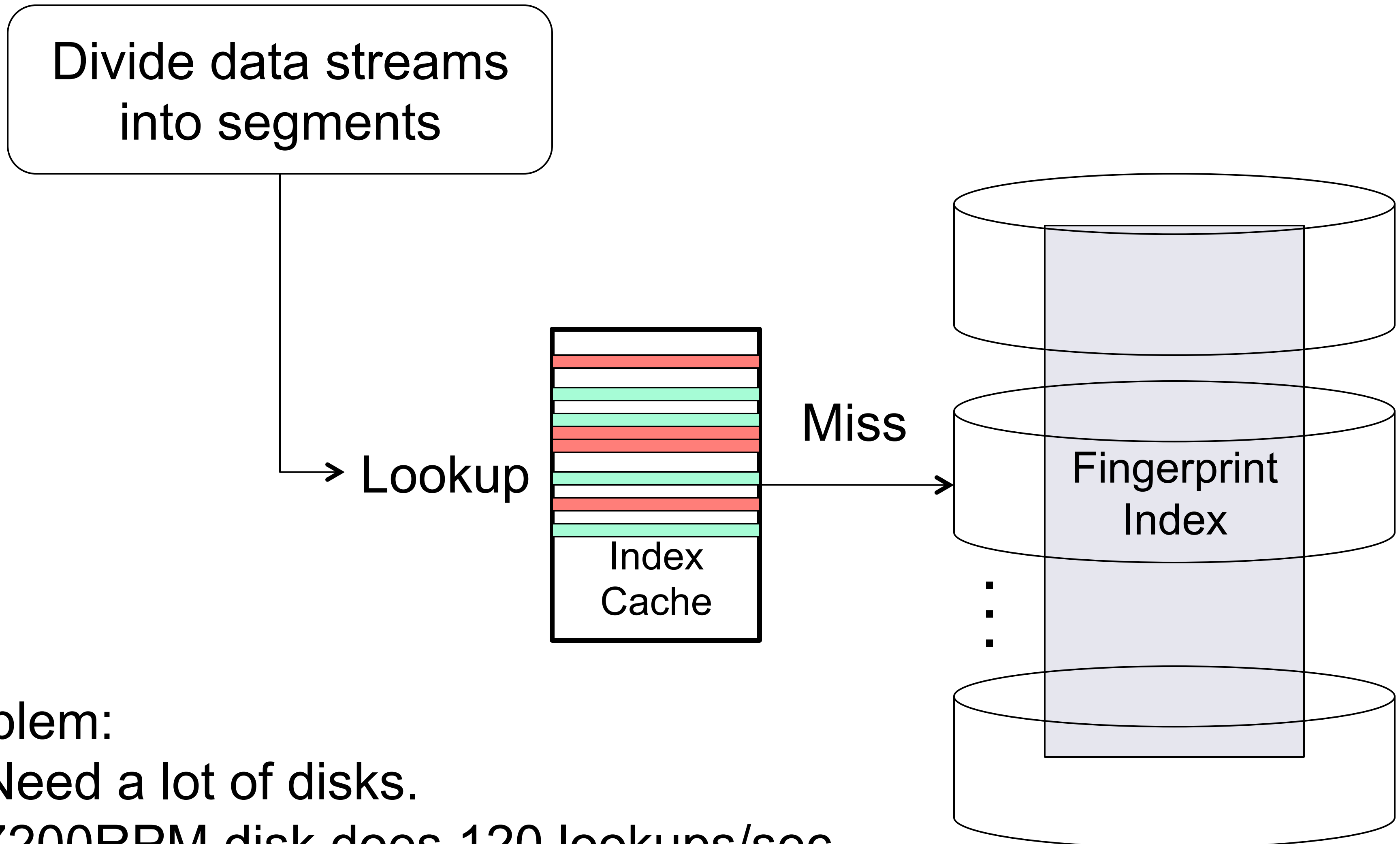
Index size for 160TB w/ 8KB segments
= (160TB/8KB) * 24B
= **480GB!**

Caching?



Problem: **No locality**.

Parallel Index Need Many Disks [Venti02]



Problem:

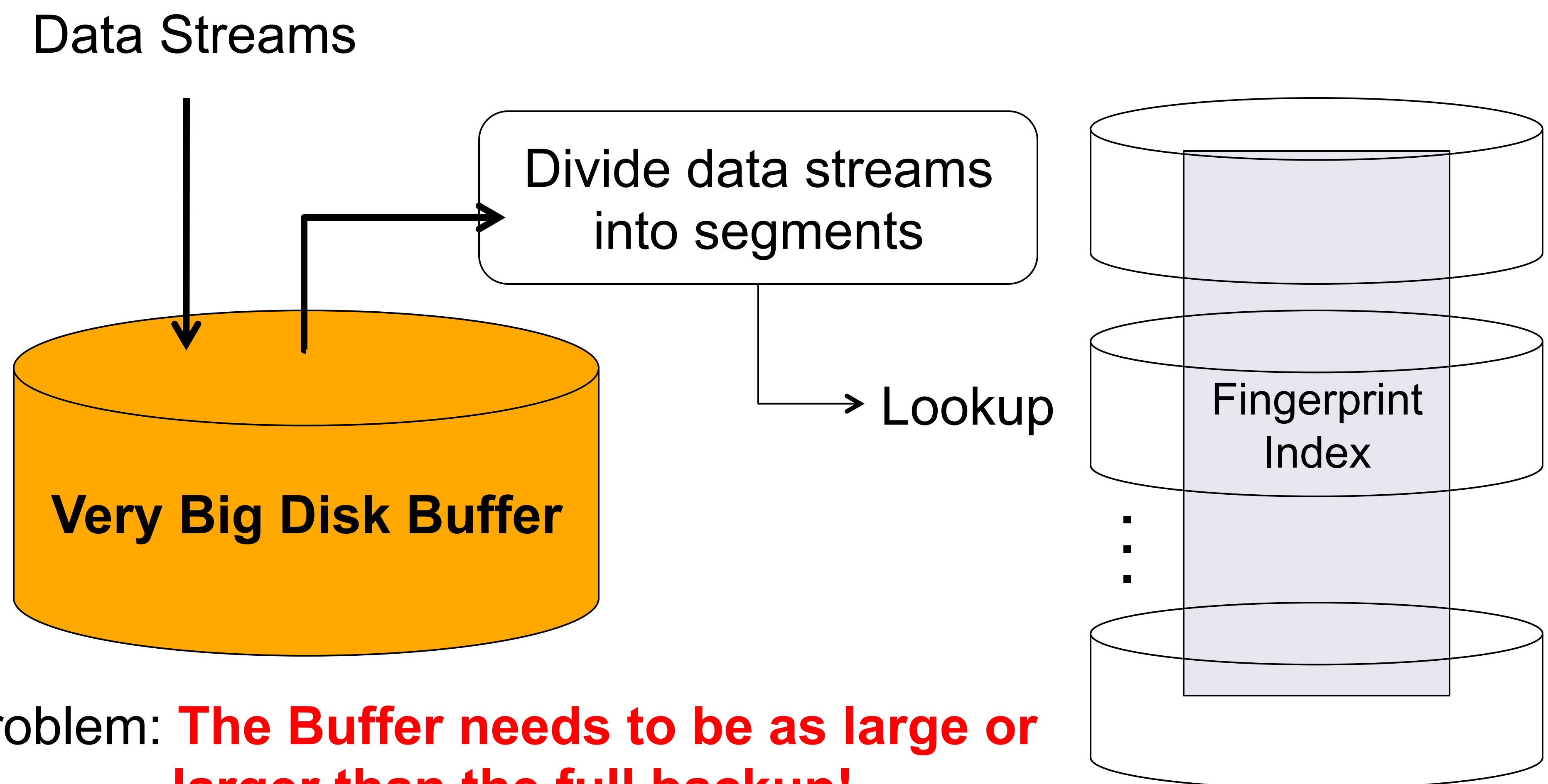
Need a lot of disks.

7200RPM disk does 120 lookups/sec.

1MB/sec with 8KB segment per disk

1GB/sec needs 1,000 disks!

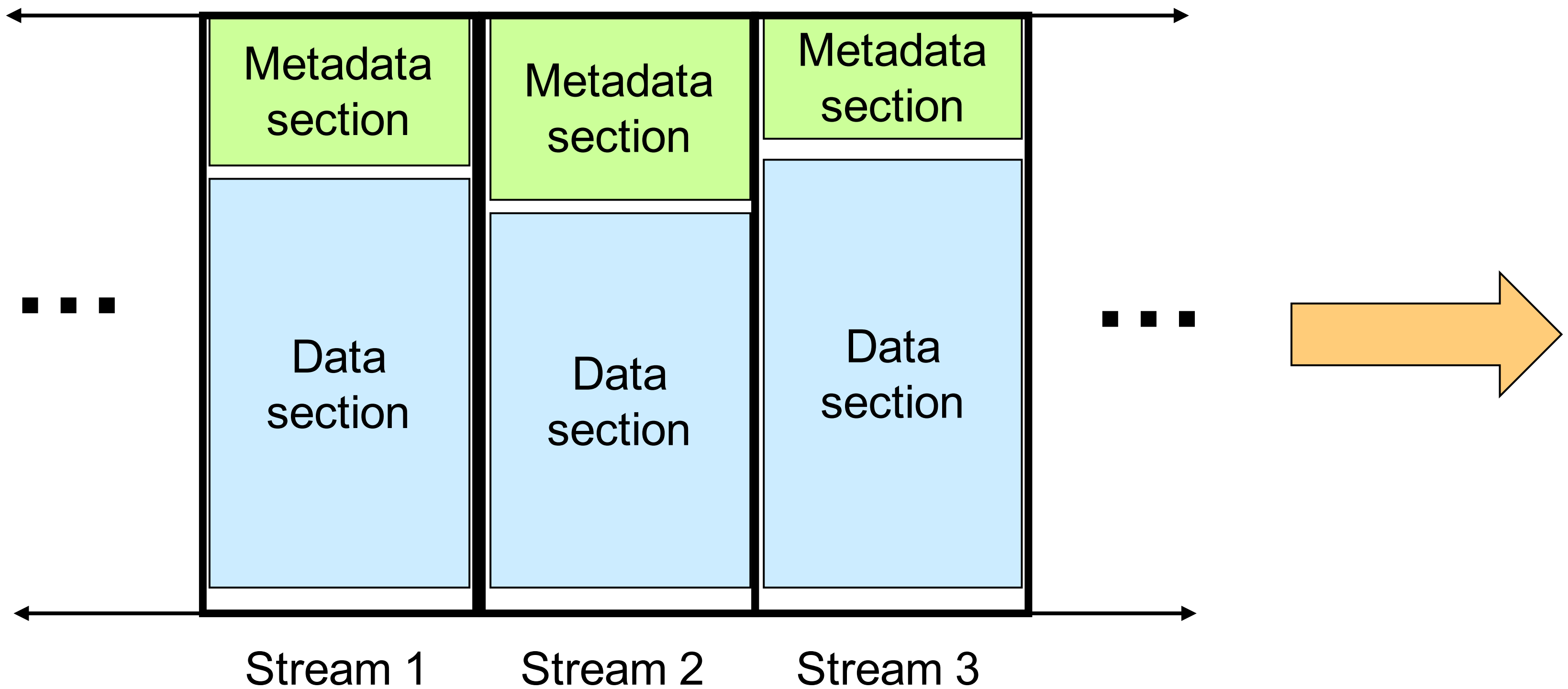
Staging Needs More Disks



Problem: **The Buffer needs to be as large or larger than the full backup!**
Big delay for replication

Stream Informed Segment Layout

- Log structured layout (inspired by LFS)
- Fixed size large containers to create locality
- A container
  - Segments from the same stream
  - Metadata (index data)



A Combination of Techniques

- Stream Informed Segment Layout
- A sophisticated cache for the fingerprint index
  - Summary data structure for new data
  - “locality-preserved caching” for old data
- Parallelized software systems to leverage multicore processors

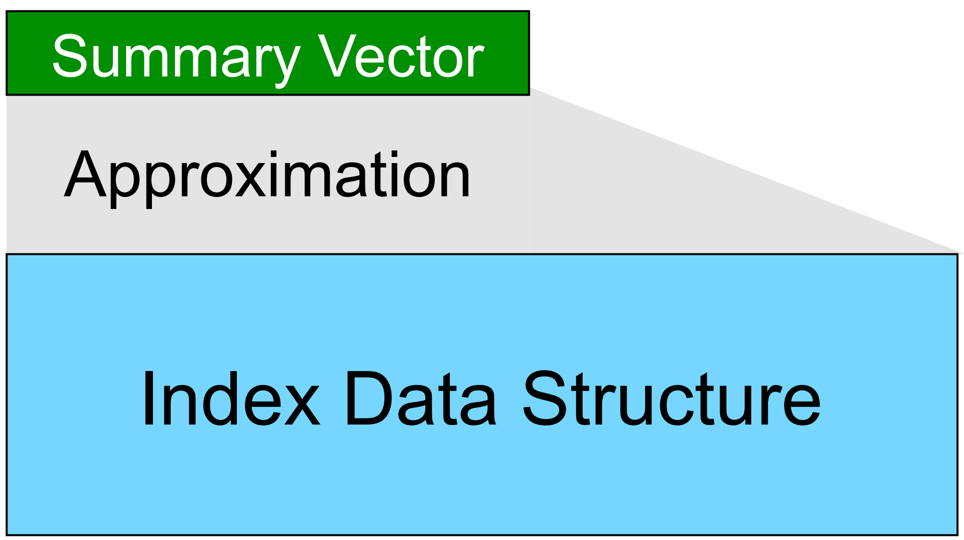
Benjamin Zhu, Kai Li and Hugo Patterson. Avoiding the Disk Bottleneck in the Data Domain Deduplication File System. In Proceedings of The 6th USENIX Conference on File and Storage Technologies (FAST’08). February 2008

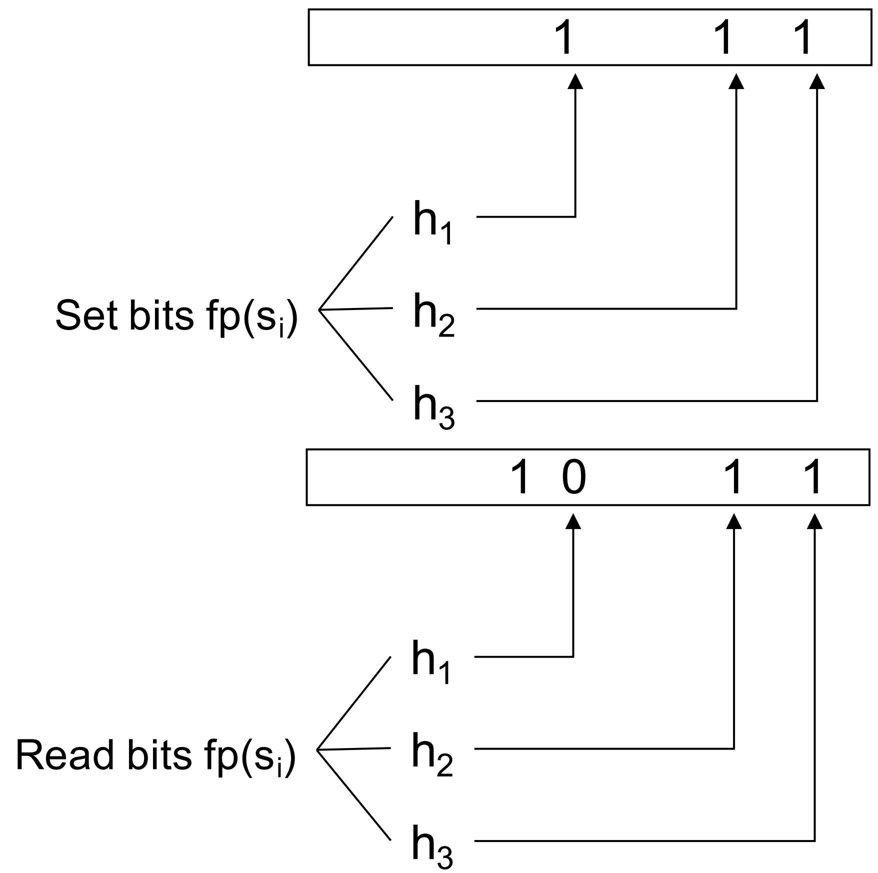
Summary Data Structure

Goal: Use minimal memory to test for new data

⇒ Summarize what segments have been stored, with Bloom filter (Bloom'70) in RAM

⇒ If Summary Vector says no, it's new segment

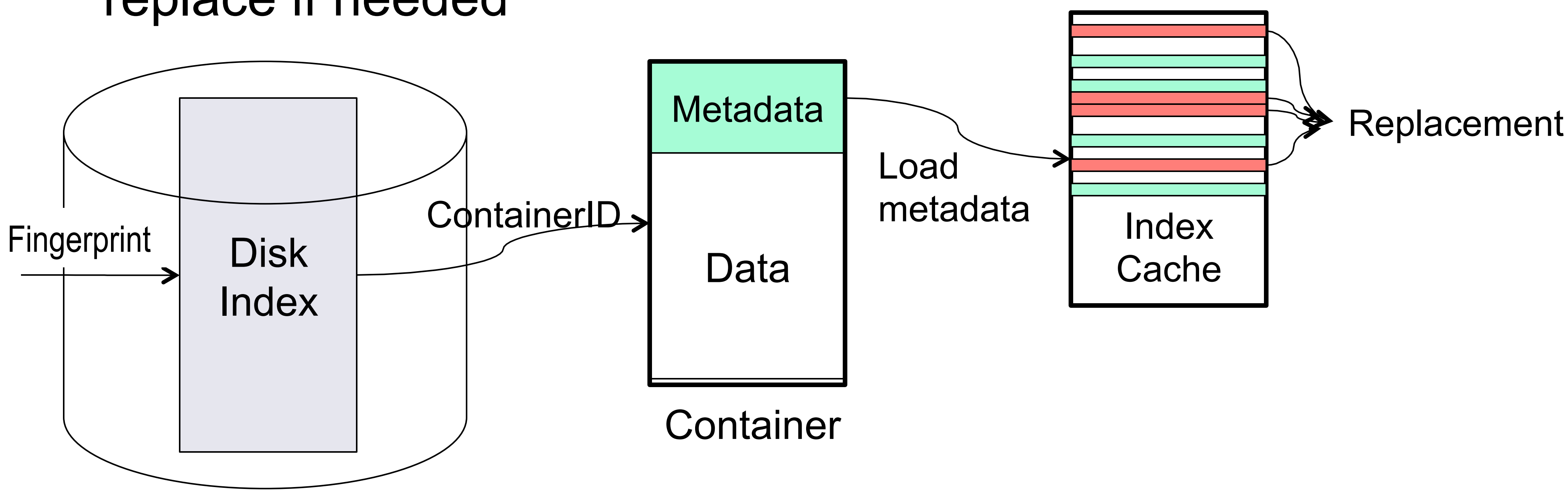




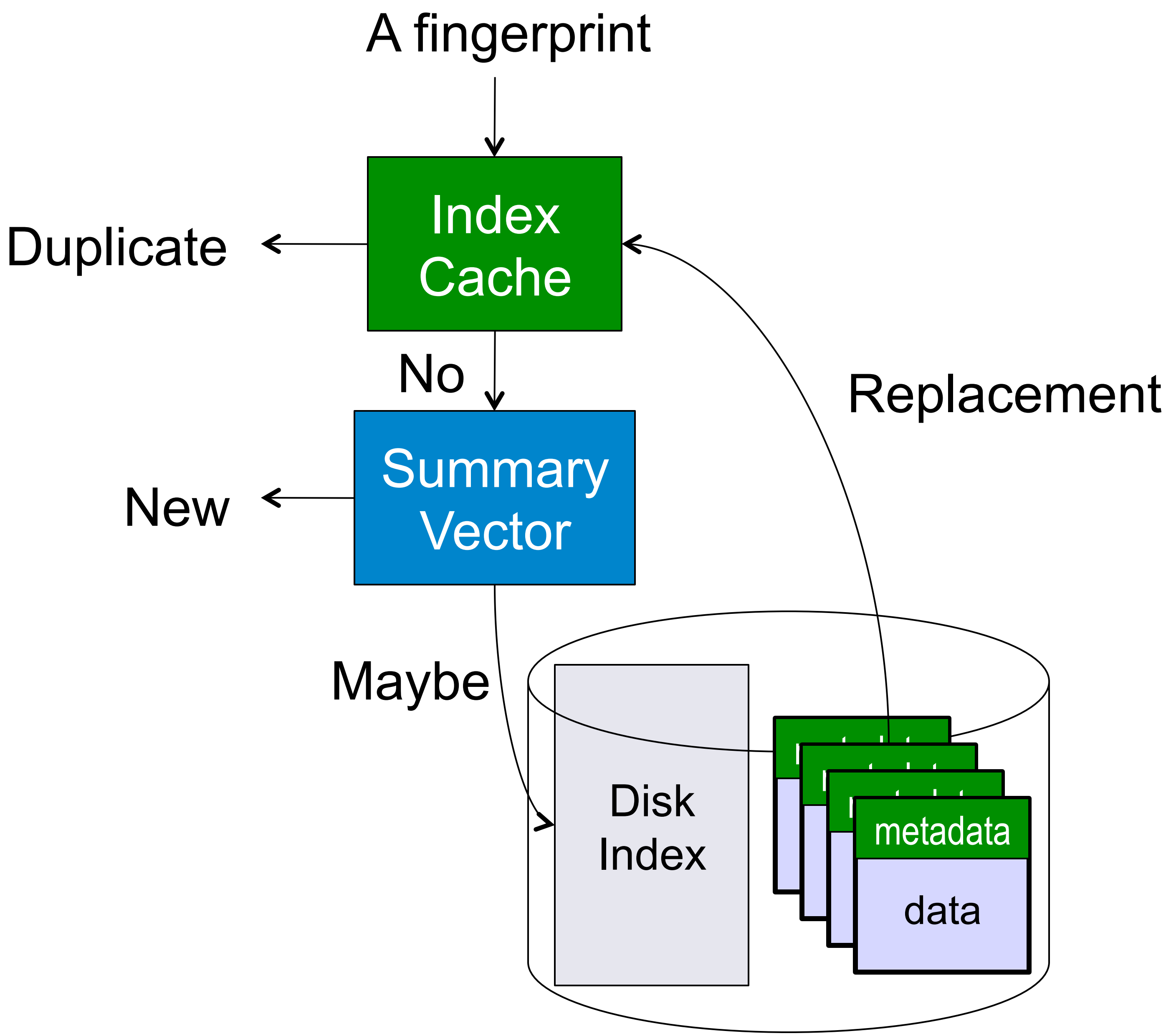
Locality Preserved Caching

Algorithm

- Disk Index has all <fingerprint, containerID> pairs
- On a miss, lookup Disk Index to find containerID
- Load the container into memory
- Load the metadata of a container into Index Cache, replace if needed



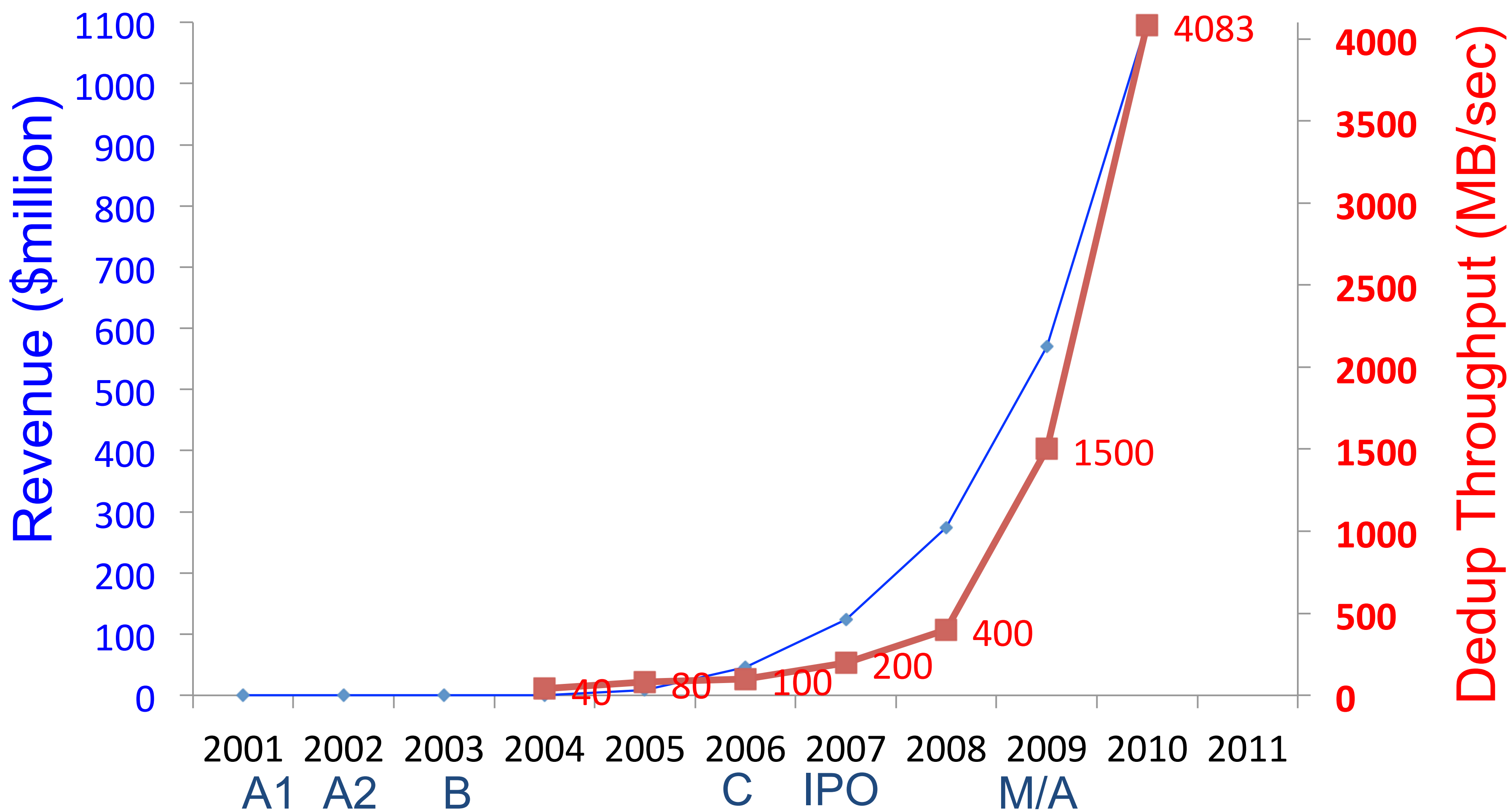
Putting Them Together



Disk I/O Reduction Results

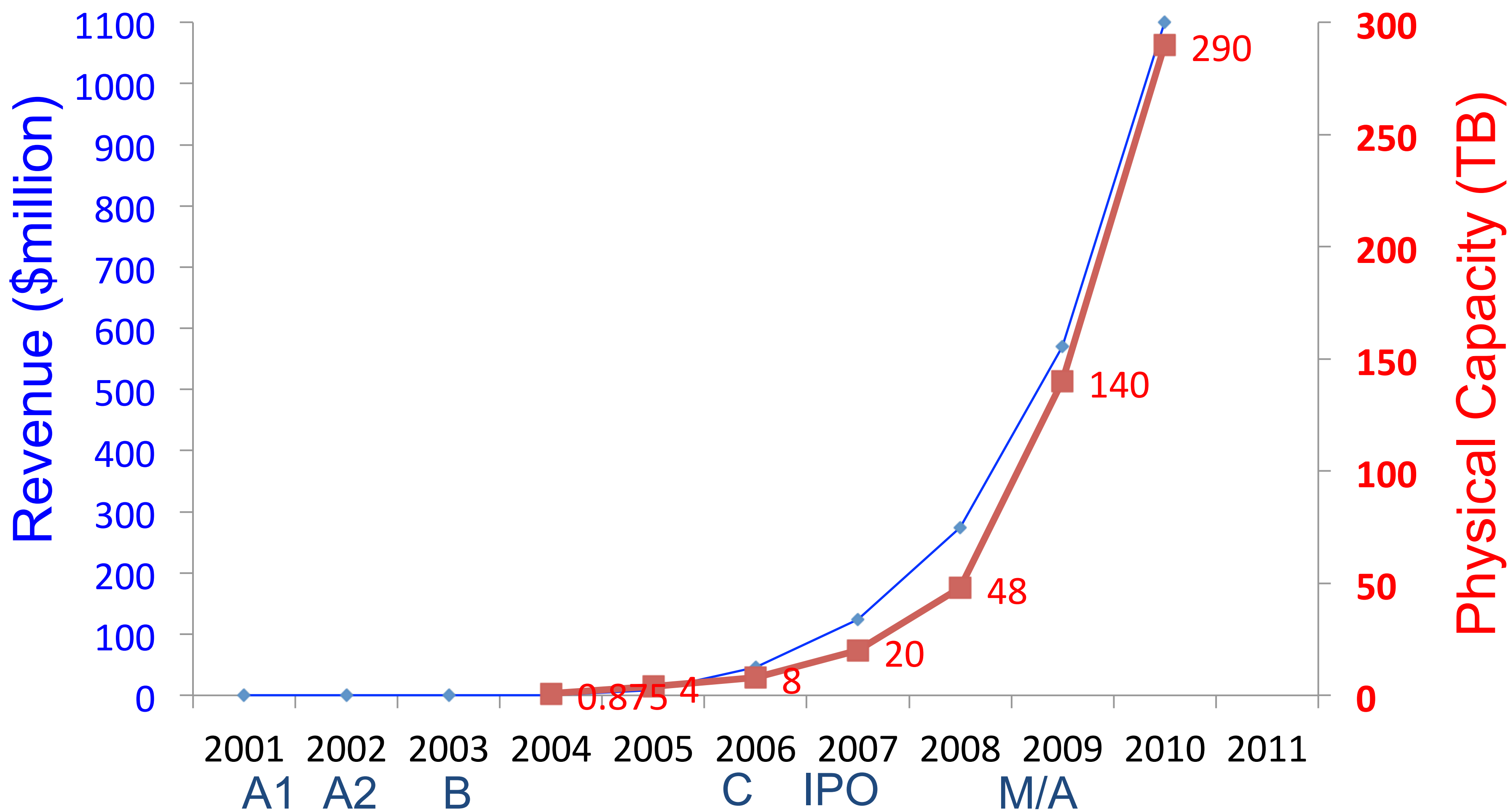
	Exchange data (2.56TB) *135-daily full backups*		Engineering data (2.39TB) *100-day daily inc, weekly full*	
	# disk I/Os	% of total	# disk I/Os	% of total
No summary, No SISL/LPC	328,613,503	100.00%	318,236,712	100.00%
Summary only	274,364,788	83.49%	259,135,171	81.43%
SISL/LPC only	57,725,844	17.57%	60,358,875	18.97%
Summary & SISL/LPC	**3,477,129**	**1.06%**	**1,257,316**	**0.40%**

Deduplication Throughput



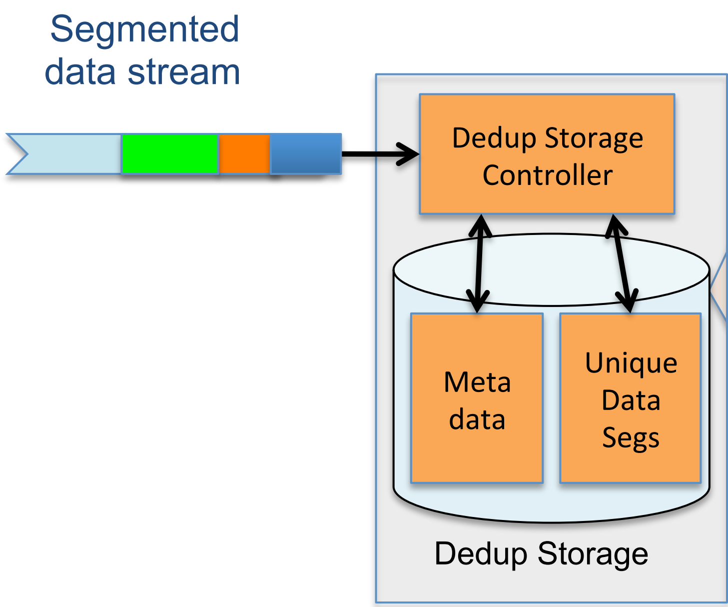
Improved by ~100X in 6 years

Physical Capacity



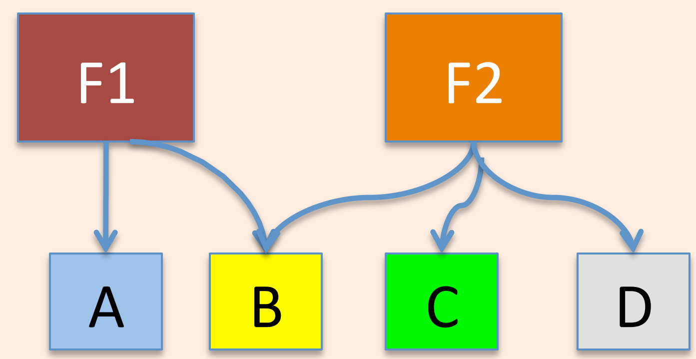
Increased by ~330X in 6 years

Even More Details

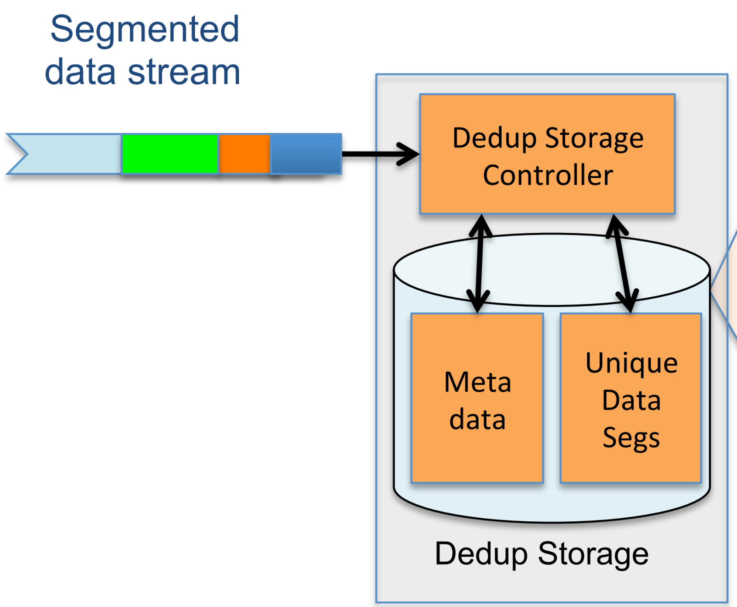


Concurrent Garbage Collection

- A segment may be shared by multiple files
- Backups need to be deleted sometime
- GC cannot interfere with backups

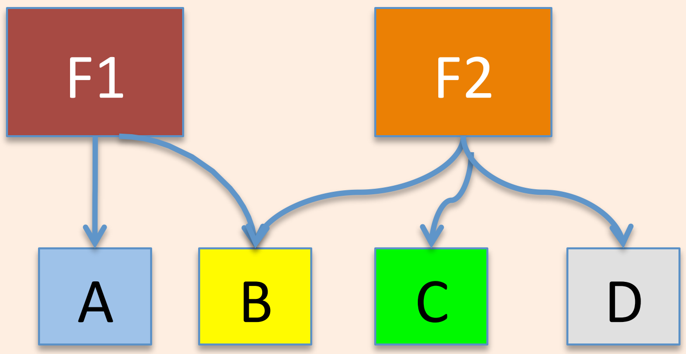


More Details



Concurrent Garbage Collection

- A segment may be shared by multiple files
- Backups need to be deleted sometime
- GC cannot interfere with backups



Summary

- Dedup storage systems
  - Replace tape libraries
  - Near-line and archival storage
  - Primary storage
- More Redundancy in larger windows
- Locality preserved caching
  - Reduce cost
  - Achieve high throughput

Review Topics

- OS structure
- Process management
- CPU scheduling
- I/O devices
- Virtual memory
- Disks and file systems
- General concepts

Operating System Structure

- Abstraction
- Protection and security
- Kernel structure
  - Layered
  - Monolithic
  - Micro-kernel
- Virtualization
  - Virtual machine monitor

Process Management

- Implementation
  - State, creation, context switch
  - Threads and processes
- Synchronization
  - Race conditions and inconsistencies
  - Mutual exclusion and critical sections
  - Semaphores: P() and V()
  - Atomic operations: interrupt disable, test-and-set.
  - Monitors and Condition Variables
  - Mesa-style monitor
- Deadlocks
  - How deadlocks occur?
  - How to prevent deadlocks?

CPU Scheduling

- Allocation
  - Non-preemptible resources
- Scheduling -- Preemptible resources
  - FIFO
  - Round-robin
  - STCF
  - Lottery

I/O Devices

- Latency and bandwidth
- Interrupts and exceptions
- DMA mechanisms
- Synchronous I/O operations
- Asynchronous I/O operations
- Message passing

Virtual Memory

- Mechanisms
  - Paging
  - Segmentation
  - Page and segmentation
  - TLB and its management
- Page replacement
  - FIFO with second chance
  - Working sets
  - WSClock

Disks and File Systems

- Disks
  - Disk behavior and disk scheduling
  - RAID4, RAID5 and RAID6
- Flash memory
  - Write performance
  - Wear leveling
  - Flash translation layer
- Directories and implementation
- File layout
- Buffer cache
- Transaction and journaling file system
- NFS and Stateless file system
- Snapshot
- Deduplication file system

Major Concepts

- Locality
  - Spatial and temporal locality
- Scheduling
  - Use the past to predict the future
- Layered abstractions
  - Synchronization, transactions, file systems, etc
- Caching
  - TLB, VM, buffer cache, etc

Operating System as Illusionist

Physical reality

- Single CPU
- Interrupts
- Limited memory
- No protection
- Raw storage device

Abstraction

- Infinite number of CPUs
- Cooperating sequential threads
- Unlimited virtual memory
- Each address has its own machine
- Organized and reliable storage system

Future Courses in Systems

- Spring
  - COS 461: computer networks
  - COS 598C: Analytics and systems of big data
- Fall:
  - COS 432: computer security
  - COS 561: Advance computer networks or (or COS 518: Advanced OS)
  - Some grad seminars in systems