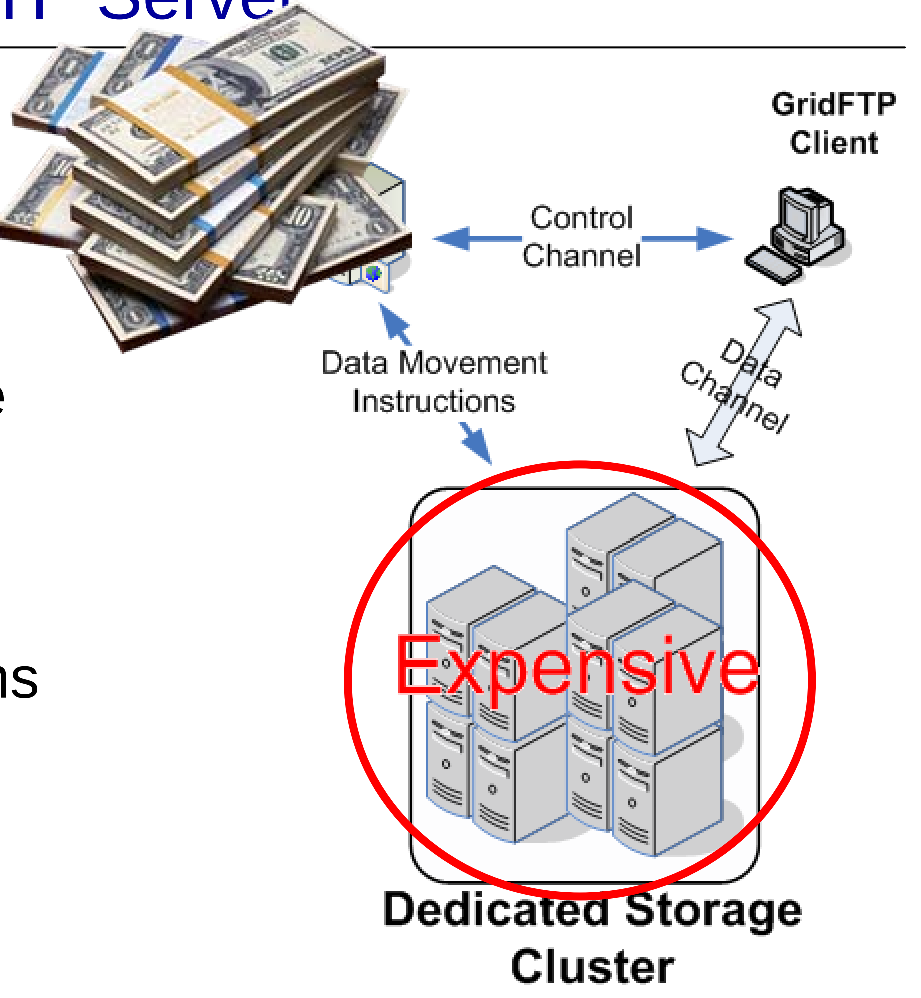
Exploring Data Reliability Tradeoffs in Replicated Storage Systems

Abdullah Gharaibeh Matei Ripeanu

NetSysLab
The University of British Columbia

Motivating Example: GridFTP Server

- A high-performance data transfer protocol
- Widely used in data-intensive scientific communities
- Typical deployments employ cluster-based storage systems



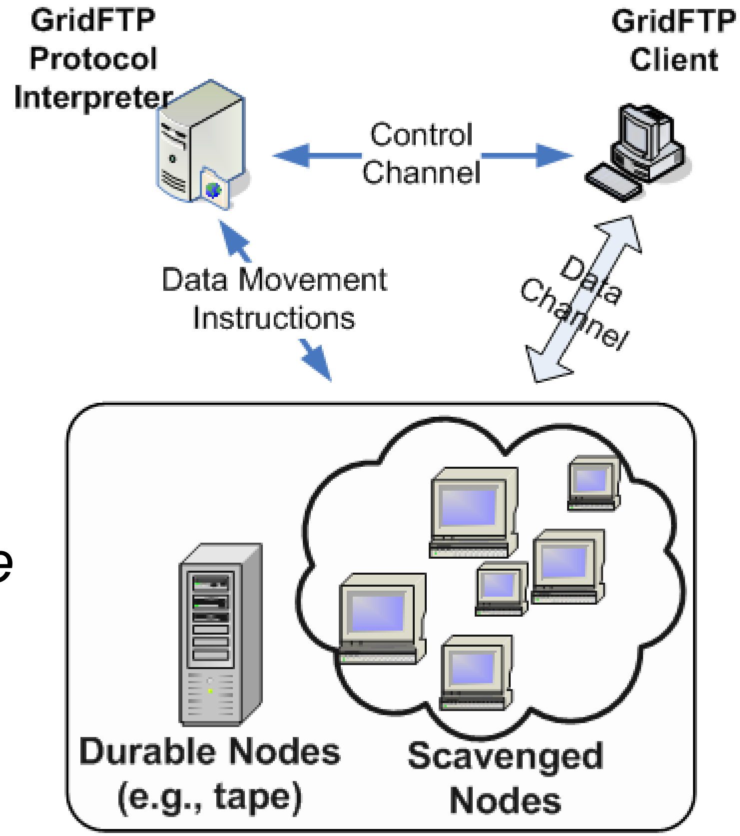
Motivation: ***reduce the cost of GridFTP server while maintaining performance and reliability***

The Solution in a Nutshell

A hybrid architecture: combines scavenged and dedicated, low bandwidth storage

Features:

- *Low cost*
- *Reliable*
- *High performance*



Hybrid Architecture

Outline

- The Opportunity
- The Solution

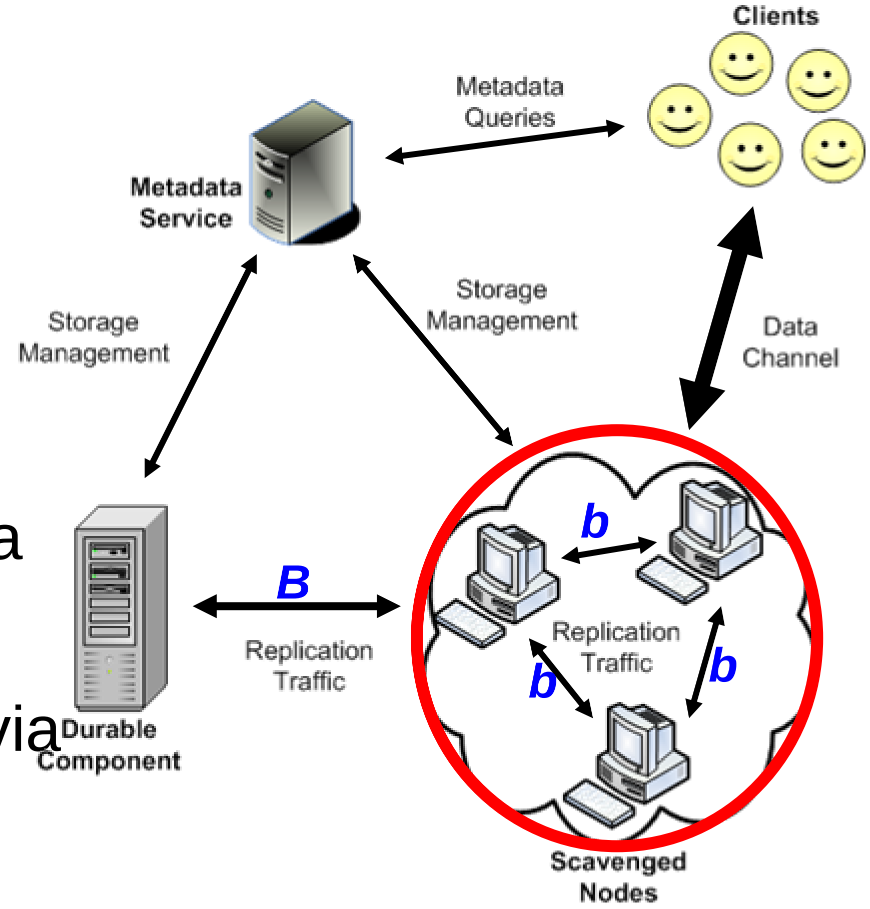
The Opportunity

- Scavenging idle storage
  - High percentage of available idle space (e.g., ~50% at Microsoft, ~60% at ORNL)
  - Well-connected machines

- Decoupling the two components of data reliability, durability and availability
  - Durability is more important than availability
  - Relax availability to reduce overall reliability overhead

The Solution: Internal Design

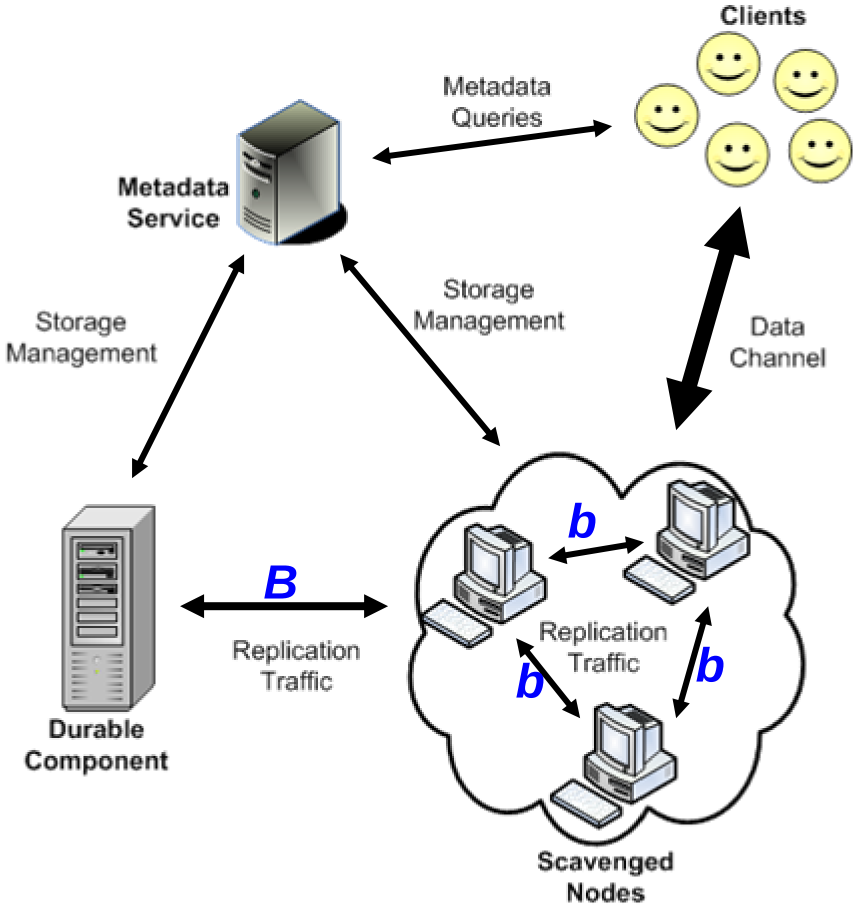
- Scavenged nodes:
  - Maintain ***n*** replicas
  - Replication bandwidth ***b***Mbps
- Durable component:
  - Durably maintain one replica
  - Replication bandwidth ***B***Mbps
- Logically centralized metadata service
- Clients access the system via the scavenged nodes only



*=> Object is **available** when at least one replica exist at the scavenged nodes*

Features Revisited

- Low cost
  - Idle resources
  - low-cost durable component
- Reliable
  - Supports full durability
  - Configurable availability
- High-performance
  - Aggregates multiple I/O channels
  - Decouples data and metadata management



Outline

- Availability Study
- Performance Evaluation: GridFTP Server

Availability Study

Questions:

- ➢ What is the advantage of having a durable component?
- ➢ What is the impact of parameter constraints (e.g., replication level and bandwidth) on availability and overhead?
- ➢ What replica placement scheme enables maximum availability?

To address these questions:

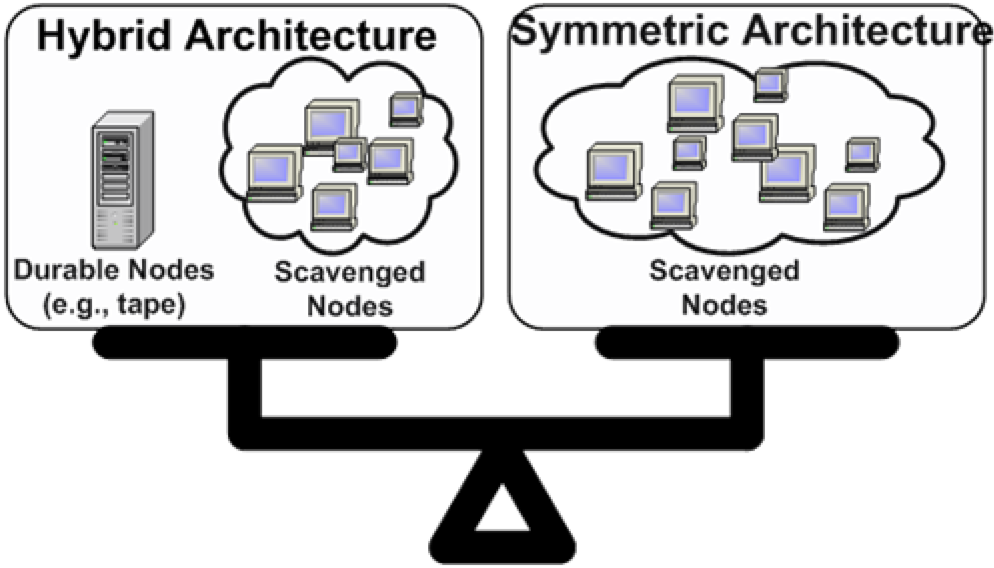
- ➢ analytical model
- ➢ low-level simulator

What is the advantage of adding a durable component?

- **Evaluate the durability of the symmetric architecture**

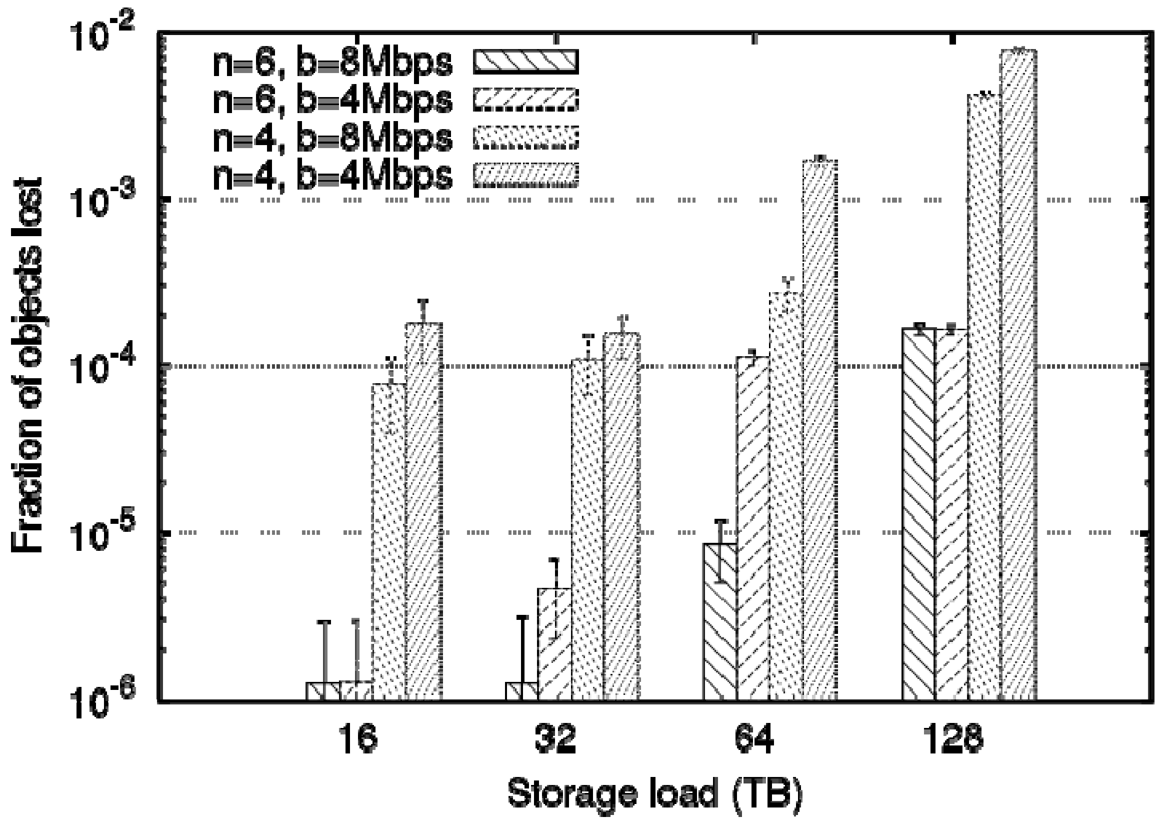
- **Compare the replication overhead**

- **Evaluate the availability of the hybrid architecture**



Durability of Symmetric Architecture

- Durability decreases when increasing storage load
- Minimum configuration to support full durability
  => n = 8
  b = 8Mbps

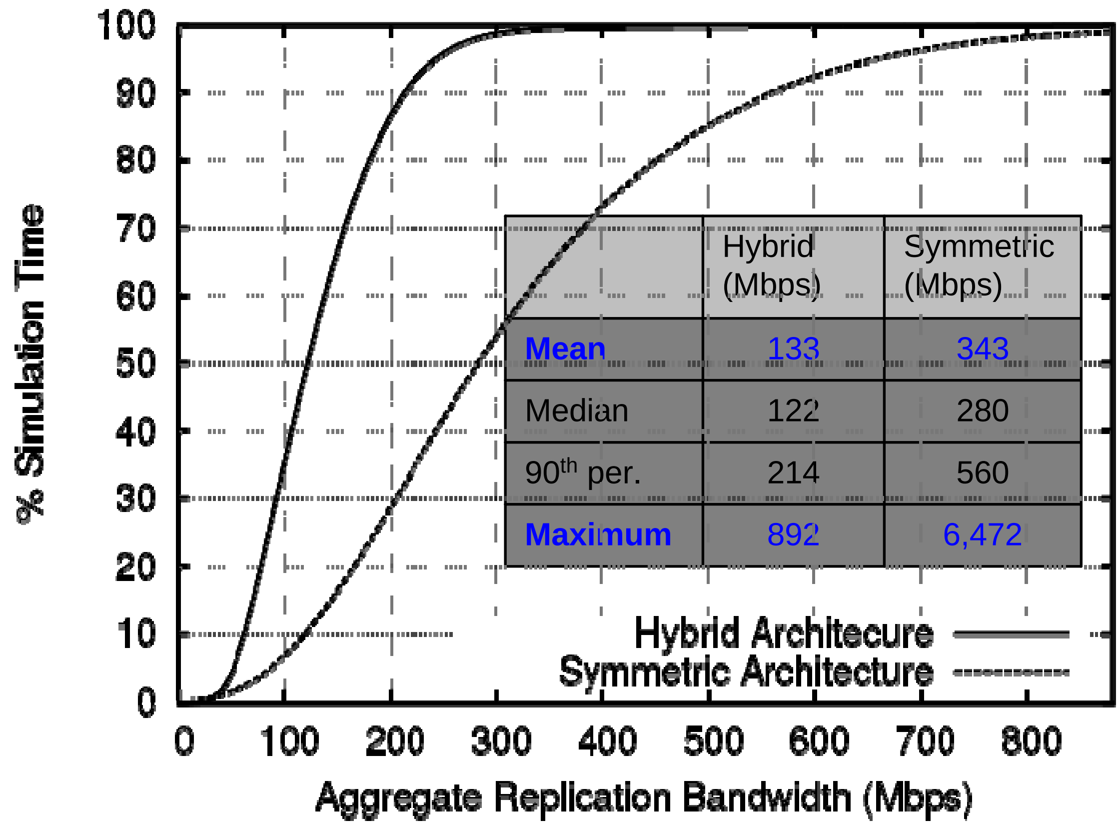


n **= replication level,** ***b*** **= replication bandwidth**

Overhead: Hybrid vs. Symmetric Architecture

Advantages of adding durable component:

- Reduces amount of replication traffic ~ 2.5 times
- Reduces the peak bandwidth ~ 7 times
- Reduces replication traffic variability
- Increases storage efficiency 50%

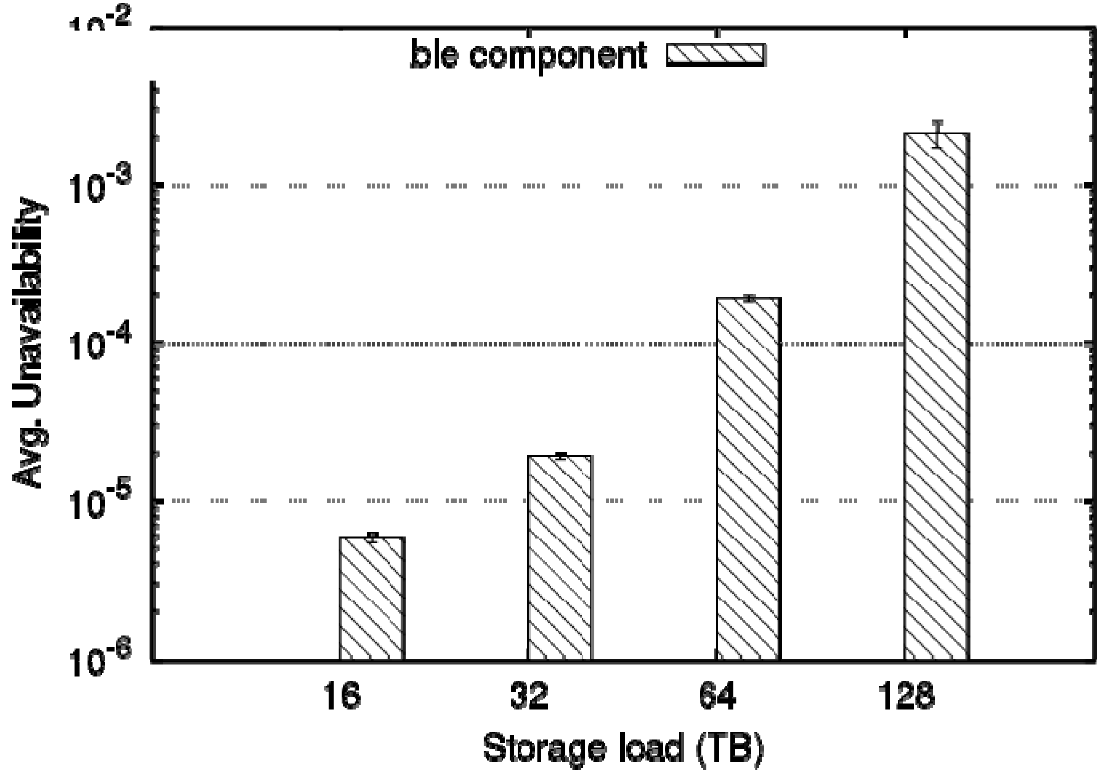


	Hybrid (Mbps)	Symmetric (Mbps)
Mean	133	343
Median	122	280
90^{th} per.	214	560
Maximum	892	6,472

Configuration: ***Symmetric Architecture***: **n = 8 replicas**, b = 8Mbps
Hybrid Architecture: **n = 4 replicas**, b = 2Mbps, B = 1Mbps

Availability of Hybrid Architecture

The hybrid system is able to support acceptable availability



Configuration: ***n*** = 4 replicas, ***b*** = 2Mbps, ***B*** = 1Mbps

Outline

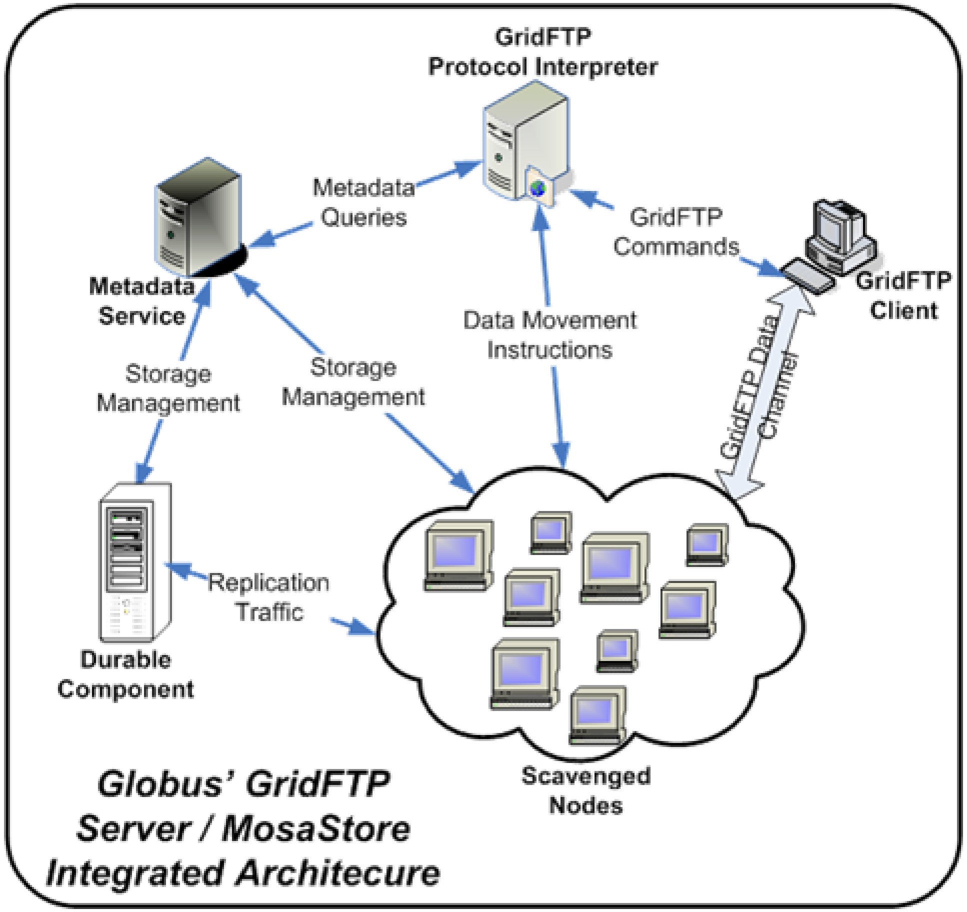
- Availability Study
- Performance Evaluation: GridFTP Server

A Scavenged GridFTP Server

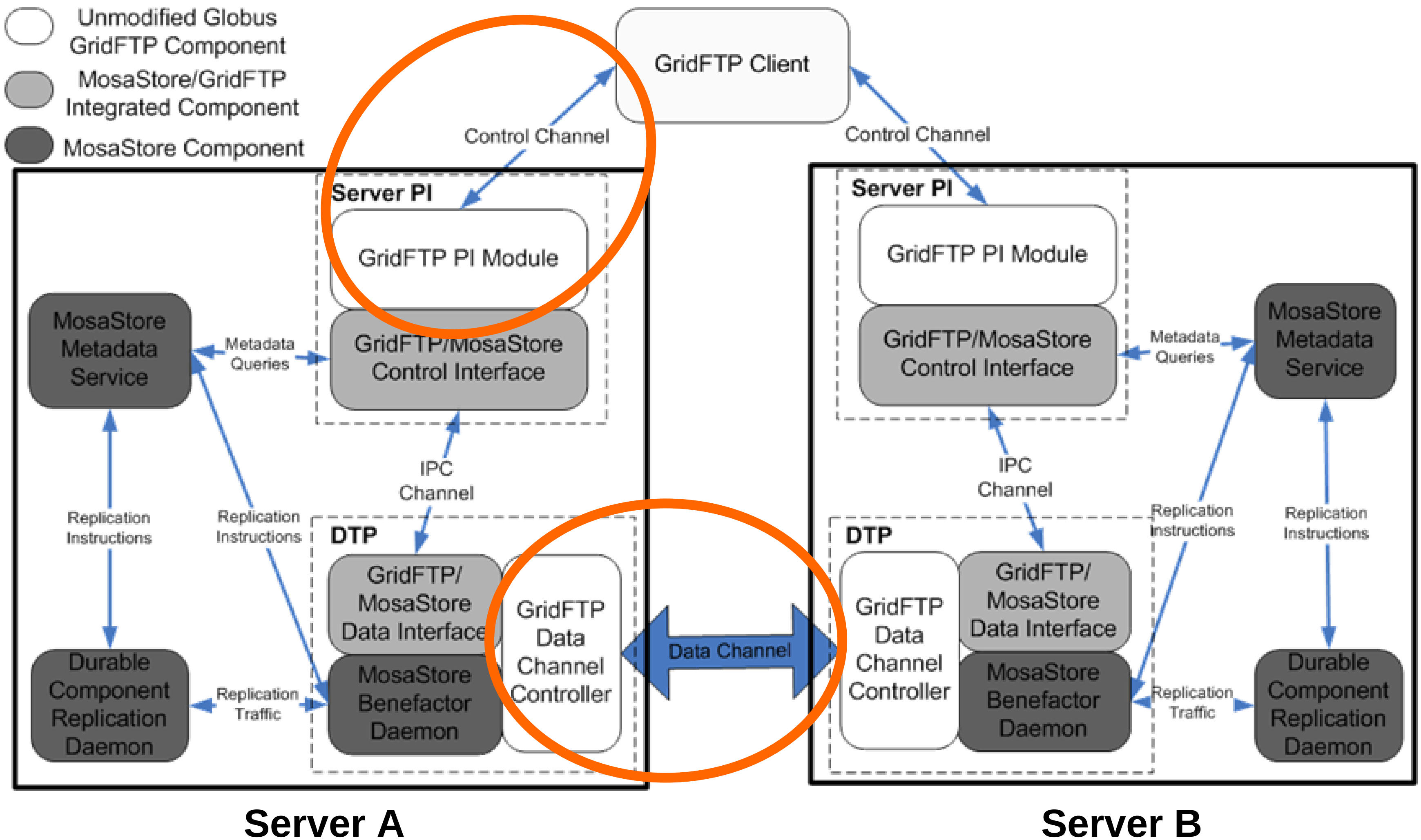
Prototype Components

- Globus' GridFTP Server
- MosaStore scavenged sotrage system

Main challenge:
transparent integration of legacy components

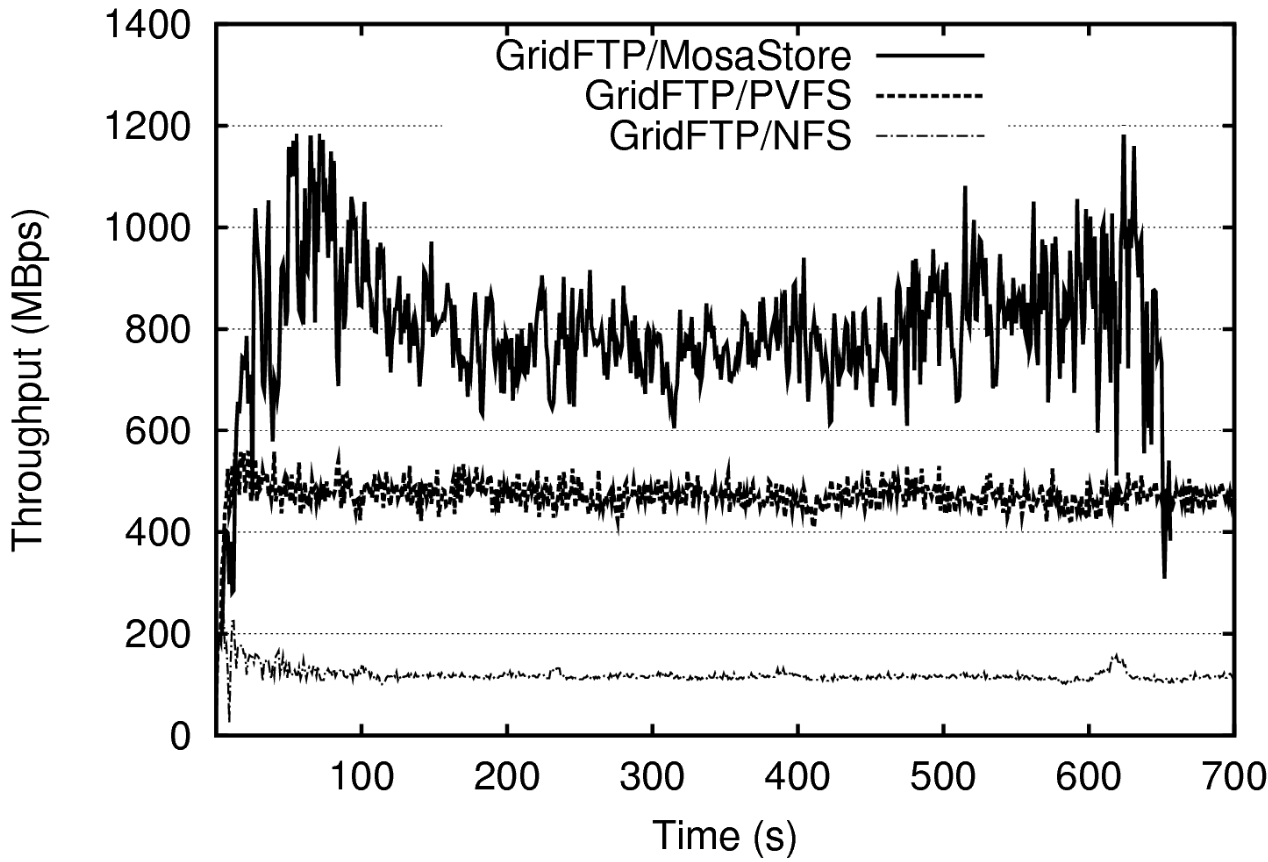


Scavenged GridFTP Software Components



Evaluation -- Throughput

Ability to support an intense workload:
=> 60% increase in aggregate throughput



Throughput for 40 clients reading 100 files of 100MB each. The GridFTP server is supported by 10 storage nodes each connected at 1Gbps.

Summary and Contributions

This study demonstrates a hybrid storage architecture that combines scavenged and durable storage

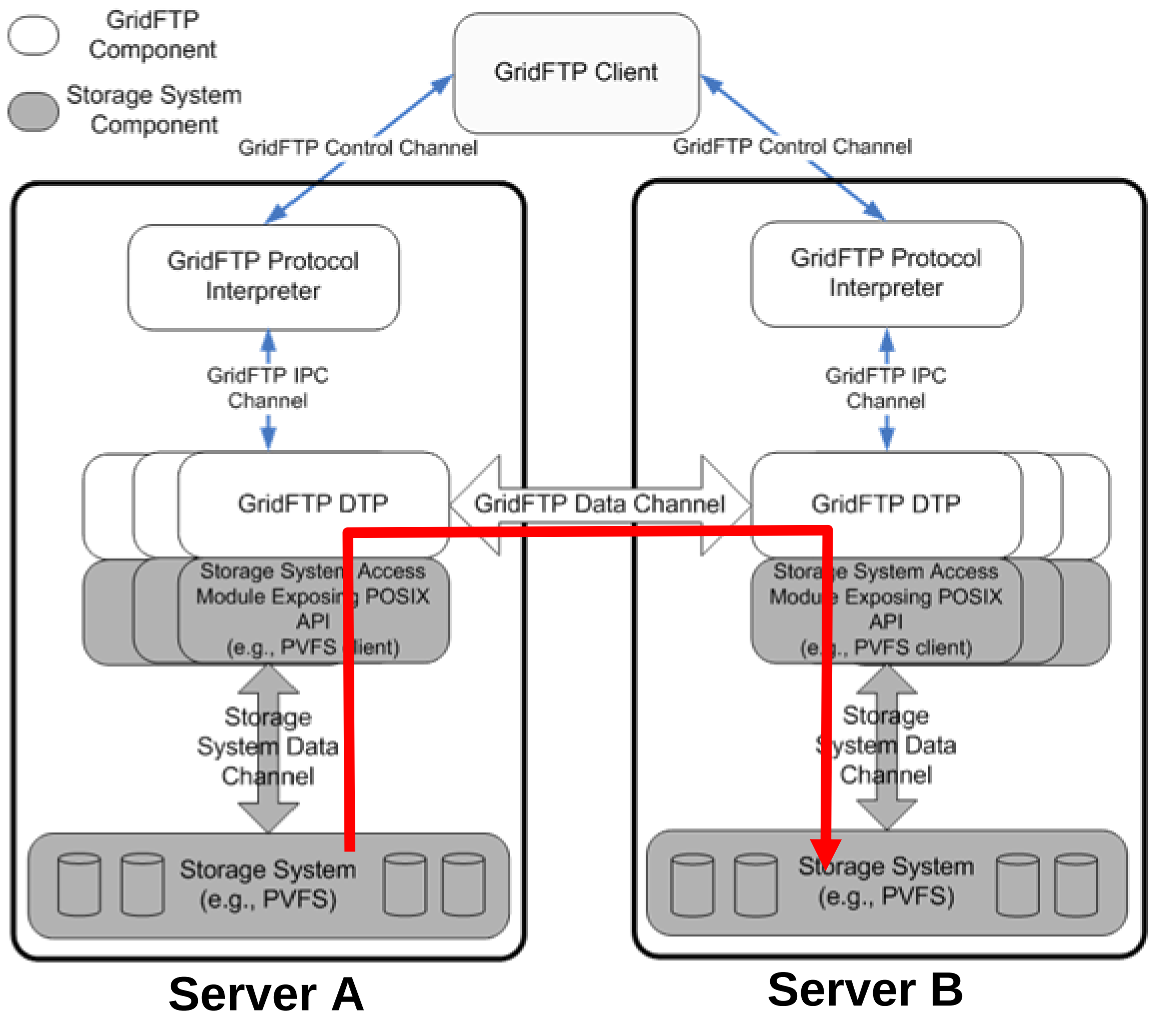
Features:

- Reliable – full durability, configurable availability
- Low-cost - built atop scavenged resources
- Offers high-performance throughput

Contributions:

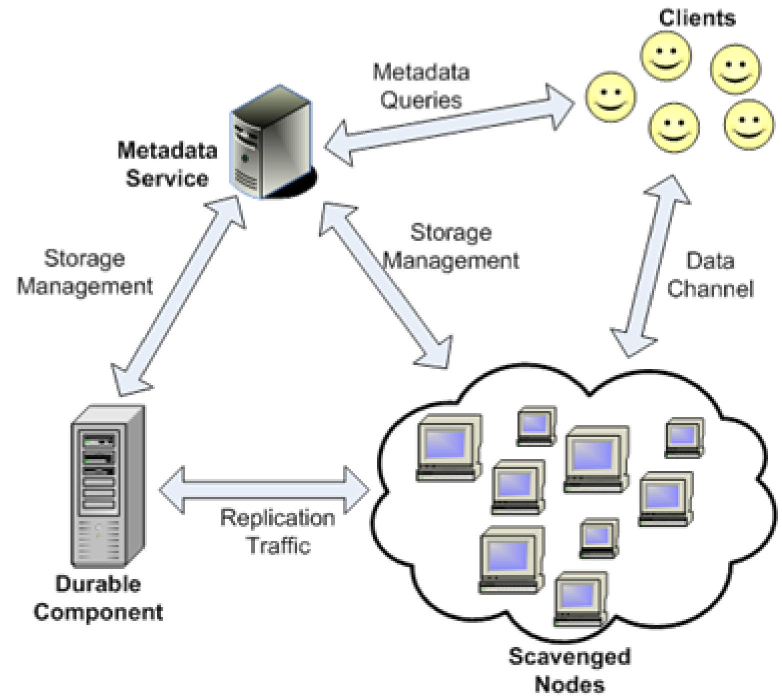
- Integrating scavenged with low-bandwidth durable storage
- Tools to provision the system:
  - Analytical model => course grained prediction
  - Low-level simulator => detailed predictions
- A prototype implementation => demonstrates high-performance

Standard Deployments: Data Locality Limitation Explained



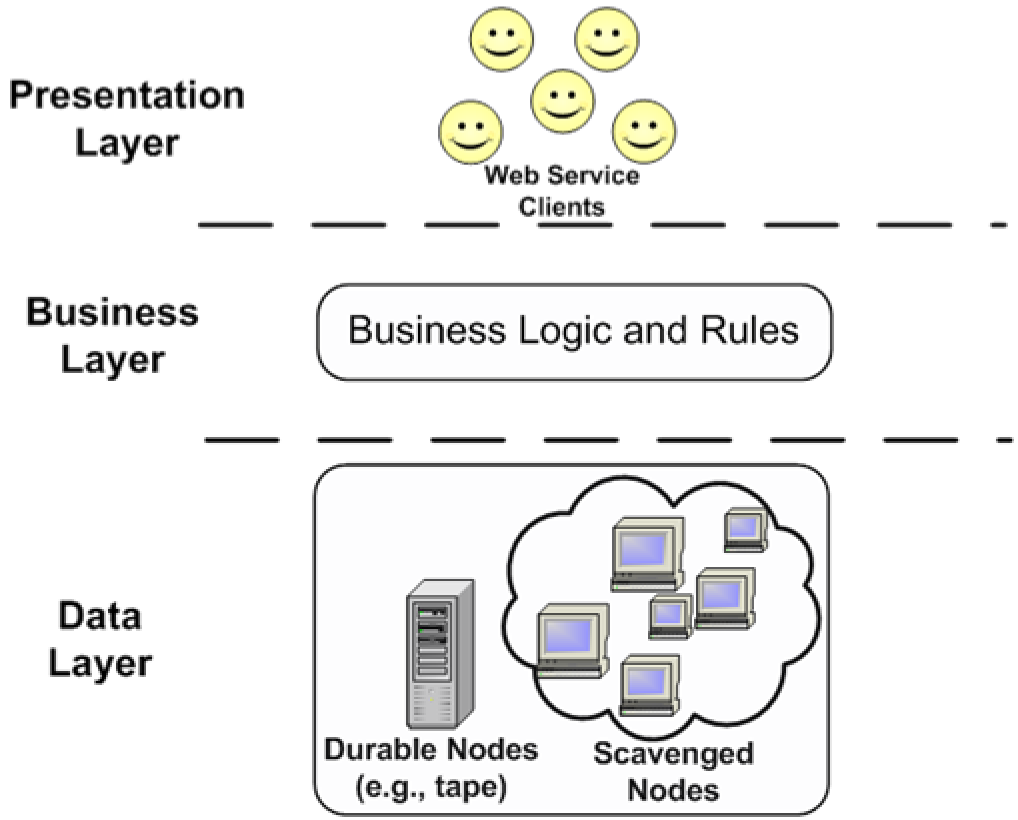
The Solution: Limitations

- *Lower availability:* trade-off availability for stronger durability and lower maintenance overhead
- *Asymmetric system*: the hybrid nature of the system may increase its complexity
- *The system mostly benefit read-dominant workloads*: due to the limited bandwidth of the durable node



Another Usage Scenario

A data-store geared towards read-mostly workload: photo-sharing web services (e.g., Flickr, Facebook)

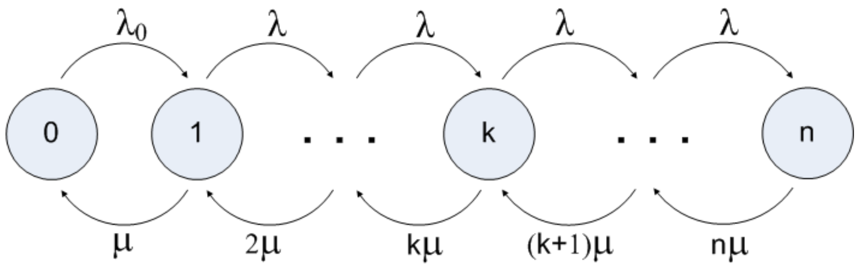


Hybrid Architecture

Analytical Modeling (1)

the number of replicas is modeled using a Markov chain model, assume exponentially distributed μ and λ.

=> Can be analyzed analytically as an ***M/M/K/K*** queue.



Each state represents the number of available replicas at the ***volatile nodes***. The rate λ_0 depends on the durable node's bandwidth.

$$Availability = 1 - p_0$$

$$p_0 = \frac{1}{1 + \gamma \sum_{k=1}^{n} \frac{\rho^{k-1}}{k!}}$$

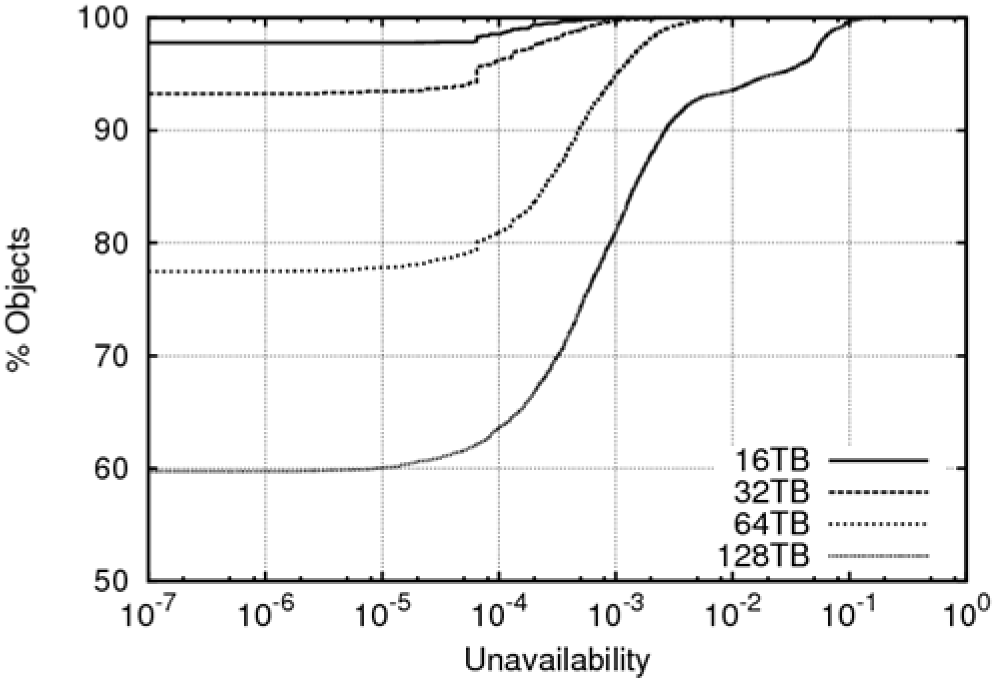
Where $\rho = \lambda/\mu$, $\gamma = \lambda_0/\mu$

Analytical Modeling (2)

- Limitations:
  - The model does not capture transient failures
  - The model assumes exponentially distributed replica repair and life times
  - The model analyzes the state of a single object
- Advantages:
  - unveils the key relationships between system characteristics
  - offers a good approximation for availability which enables validating the simulator

Distribution of Availability

What is the effect of having one replica stored on a medium with low access rate on the resulting maintenance overhead* and *availability?



Storage load (TB)	16	32	64	128
Mean	$5.8*10^{-6}$	$1.9*10^{-5}$	$1.8*10^{-4}$	$2.0*10^{-3}$
90^{th} percentile	0	0	$4.7*10^{-4}$	$2.6*10^{-3}$
Maximum (worst)	$1.1*10^{-3}$	$4.9*10^{-3}$	$9.8*10^{-3}$	$2.2*10^{-1}$

Configuration: n = 4 replicas, b = 2Mbps, B = 1Mbps

Impact of Durable Node Replication Bandwidth

Statistics of Unavailability

Durable node's bandwidth (*B*)	1 Mbps	2 Mbps	4 Mbps	8 Mbps
Mean	$1.90*10^{-5}$	$9.78*10^{-6}$	$7.09*10^{-6}$	$4.54*10^{-6}$
99^{th} percentile	$5.17*10^{-4}$	$4.52*10^{-4}$	$3.69*10^{-4}$	$3.44*10^{-4}$
Maximum	$4.93*10^{-3}$	$2.95*10^{-3}$	$1.15*10^{-3}$	$1.07*10^{-3}$

Statistics of Aggregate Replication Bandwidth

Durable node's bandwidth (*B*)	1 Mbps	2 Mbps	4 Mbps	8 Mbps
Mean	41	41	41	41
99^{th} percentile	196	194	198	202
Maximum	892	906	906	920

Impact of Scavenged Nodes Replication Bandwidth

Statistics of Unavailability

Volatile nodes' bandwidth (*b*)	1 Mbps	2 Mbps	4 Mbps	8 Mbps
Mean	$7.07*10^{-5}$	$1.97*10^{-5}$	$7.05*10^{-6}$	$3.44*10^{-6}$
99^{th} percentile	$1.18*10^{-3}$	$5.17*10^{-4}$	$2.86*10^{-4}$	$7.93*10^{-5}$
Maximum	$6.07*10^{-3}$	$4.93*10^{-3}$	$4.03*10^{-3}$	$4.01*10^{-3}$

Statistics of Aggregate Replication Bandwidth

Volatile nodes' bandwidth (*b*)	1 Mbps	2 Mbps	4 Mbps	8 Mbps
Mean	38	40	41	42
99^{th} percentile	120	196	292	424
Maximum	438	892	1,864	3,616

Impact of Replication Level

Statistics of Unavailability

Replication level (*n*)	3	4	5	6
Mean	$1.97*10^{-5}$	$1.49*10^{-6}$	$1.39*10^{-7}$	$2.46*10^{-8}$
99^{th} percentile	$5.17*10^{-4}$	$5.70*10^{-6}$	0	0
Maximum	$4.93*10^{-3}$	$3.99*10^{-3}$	$3.23*10^{-4}$	$2.42*10^{-4}$

Statistics of Aggregate Replication Bandwidth

Replication level (*n*)	3	4	5	6
Mean	40	50	60	70
99^{th} percentile	196	244	286	336
Maximum	892	1152	1322	1458