Symmetric Active/Active Replication for Dependent Services

Christian Engelmann[1,2], Stephen L. Scott[1],
Chokchai (Box) Leangsuksun[3], Xubin (Ben) He[4]

[1] Oak Ridge National Laboratory, Oak Ridge, USA
[2] The University of Reading, Reading, UK
[3] Louisiana Tech University, Ruston, USA
[4] Tennessee Tech University, Cookeville, USA

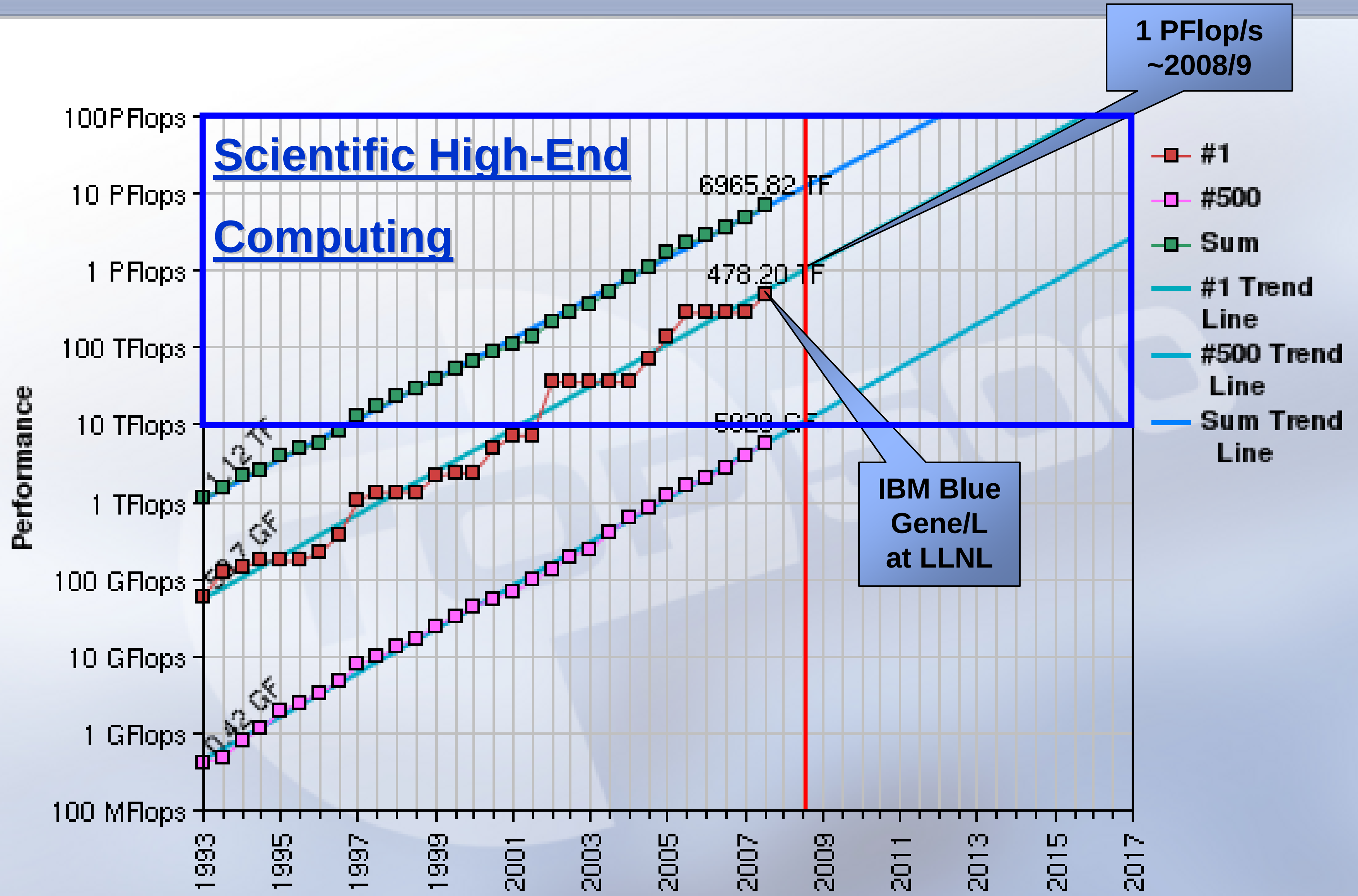
Overview

- Overall background
  - Scientific high-end computing
  - Availability issues in high-performance computing systems
  - High availability for head and service nodes
  - Symmetric active/active (state-machine or active) replication
  - Past accomplishments and limitations
- Motivation and approach
- High-level abstraction for symmetric active/active replication in:
  - Client/service scenarios
  - Dependent service scenarios

Scientific High-End Computing (HEC)

- Large-scale high-performance computing (HPC)
  - Tens-to-hundreds of thousands of processors
  - Current systems: IBM Blue Gene/L and Cray XT4
  - Next-generation: Petascale IBM Blue Gene/P and Cray XT
- Computationally and data intensive applications
  - 100 TFlops - 1 PFlops with 100 TB - 1 PB of data
  - Climate change, nuclear astrophysics, fusion energy, materials sciences, biology, nanotechnology, …
- Capability vs. capacity computing
  - Single jobs occupy large-scale high-performance computing systems for weeks and months at a time

Projected Performance Development



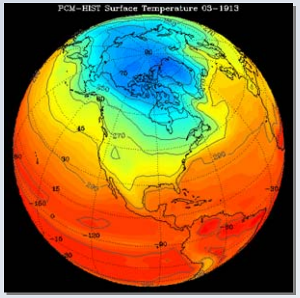
National Center for Computational Sciences

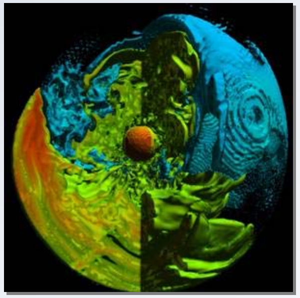
- 40,000 ft² (3700 m²) computer center:
  - 36-in (~1m) raised floor, 18 ft (5.5 m) deck-to-deck
  - 12 MW of power with 4,800 t of redundant cooling
  - High-ceiling area for visualization lab:
    - 35 MPixel PowerWall, Access Grid, etc.
- 3 systems in the Top 500 List of Supercomputer Sites:

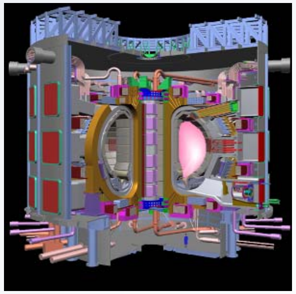
Jaguar:	7.	Cray XT3,	MPP	with	11508 dual-core Processors	⇨	119 TFlop
	41.	IBM Blue Gene/P,	MPP	with	2048 quad-core Processors	⇨	27 TFlop
Phoenix:	80.	Cray X1E,	Vector	with	1014 Processors	⇨	18 TFlop

At Forefront in Scientific Computing and Simulation

- Leading partnership in developing the National Leadership Computing Facility
  - Leadership-class scientific computing capability
  - 250 TFlop/s in 2008 (upgrade in progress)
  - 500 TFlop/s in 2008 (commitment made)
  - 1 PFlop/s in 2008/9 (commitment made)
- Attacking key computational challenges
  - Climate change
  - Nuclear astrophysics
  - Fusion energy
  - Materials sciences
  - Biology
- Providing access to computational resources through high-speed networking







Availability Measured by the Nines

see <http://www.nccs.gov/computing-resources/systems-status/> for current ORNL system status

9's	Availability	Downtime/Year	Examples
1	90.0%	36 days, 12 hours	Personal Computers
2	99.0%	87 hours, 36 min	Entry Level Business
3	99.9%	8 hours, 45.6 min	ISPs, Mainstream Business
4	99.99%	52 min, 33.6 sec	Data Centers
5	99.999%	5 min, 15.4 sec	Banking, Medical
6	99.9999%	31.5 seconds	Military Defense

- Enterprise-class hardware + Stable Linux kernel = 5+
- Substandard hardware + Good high availability package = 2-3
- Today's supercomputers = 1-2
- My desktop = 1-2

Typical Failure Causes in HPC Systems

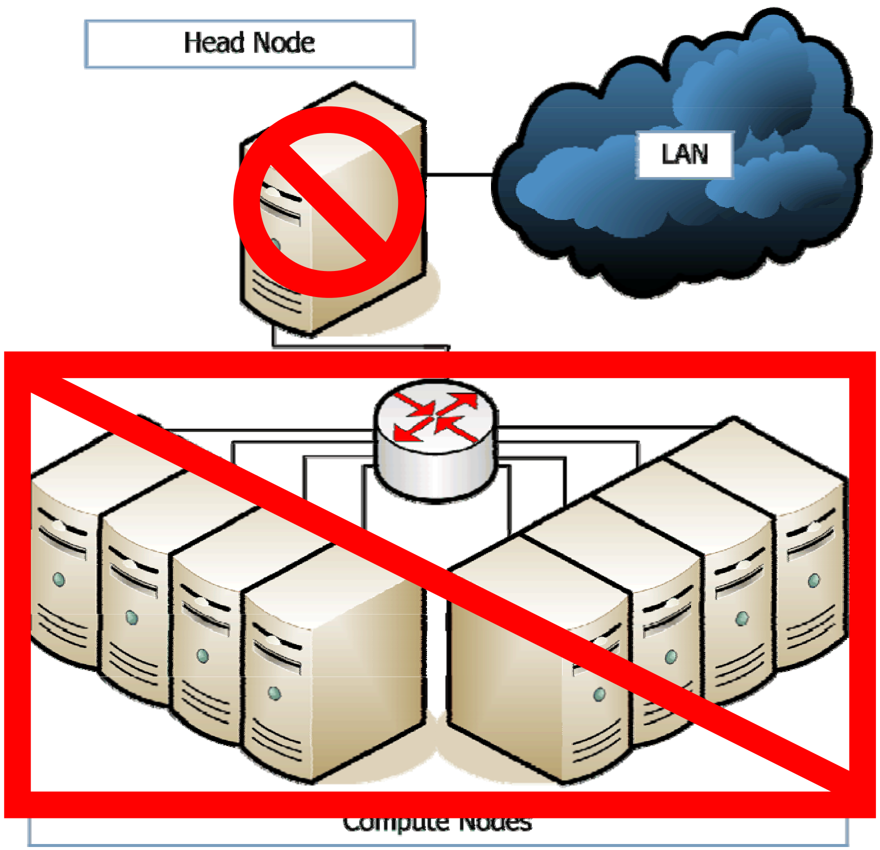
- Overheating (design errors - specification vs. usage)
- Memory and network errors (soft errors)
- Hardware failures due to wear/age of:
  - Hard drives, memory modules, network cards, processors
- Software failures due to bugs in:
  - Operating system, middleware, applications

➔ Different scale requires different solutions:

  ➔ Compute nodes (up to ~200,000)

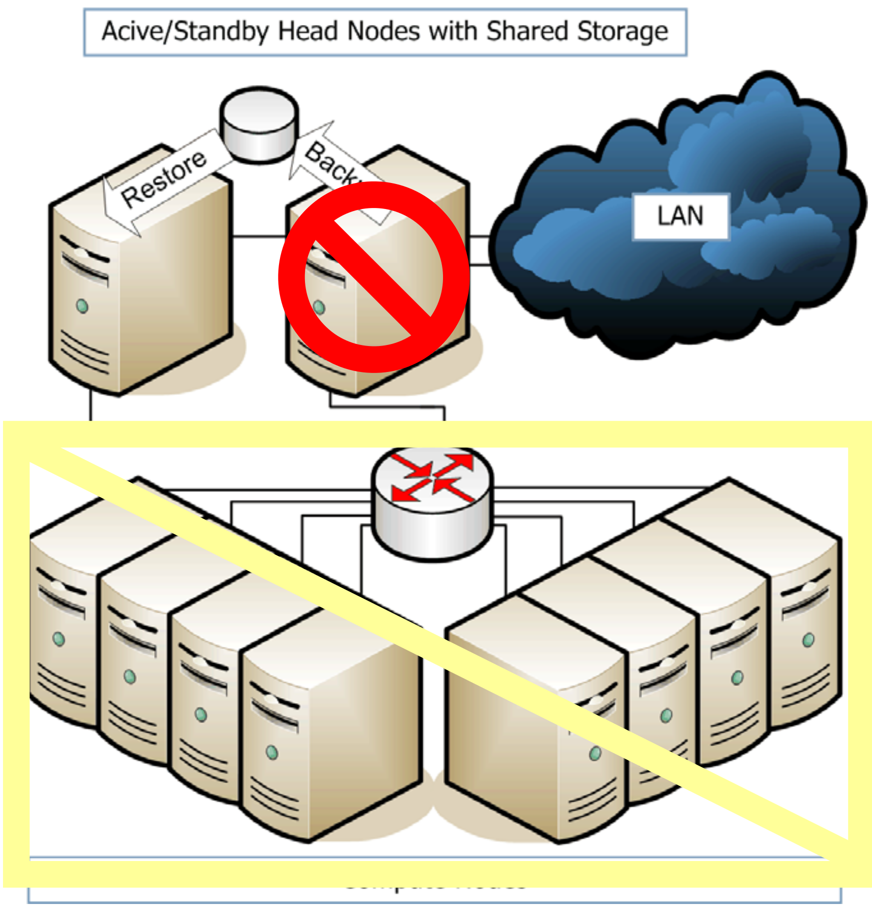
  ➔ <u>Front-end, service, and I/O nodes (1 to ~200)</u>

Single Head/Service Node Problem



- Single point of failure
- Compute nodes sit idle while head node is down
- A = MTTF / (MTTF + MTTR)
- MTTF depends on head node hardware/software quality
- MTTR depends on the time it takes to repair/replace node
- ➢ MTTR = 0 ➔ A = 1.00 (100%) continuous availability
- ➢ Fail-stop model

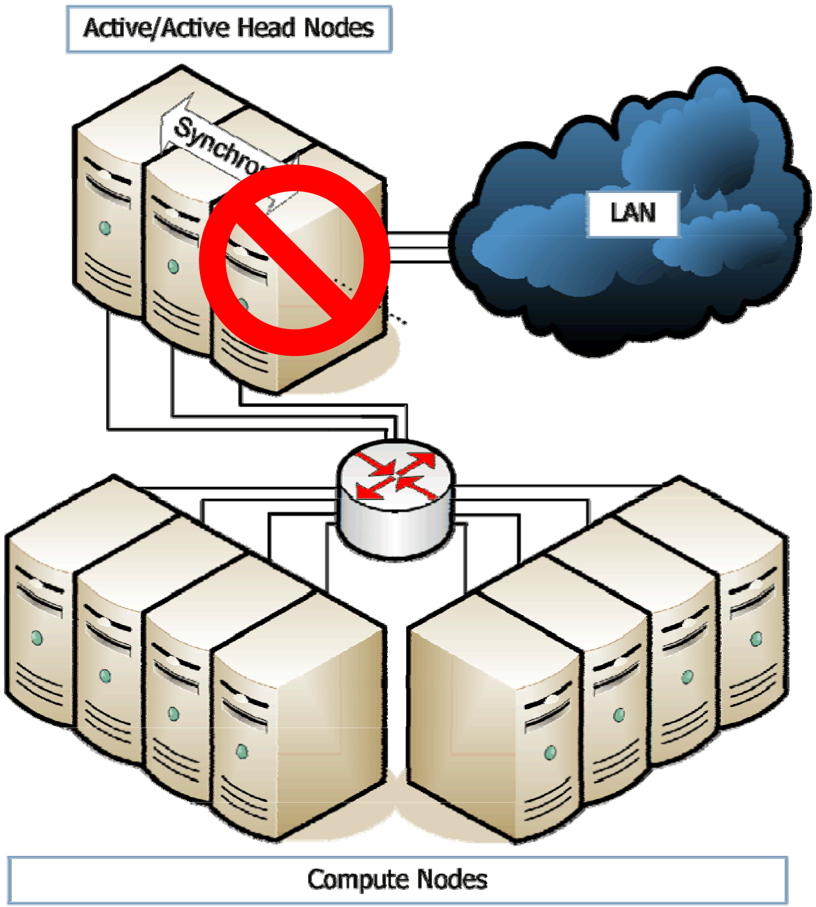
Active/Standby with Shared Storage



- Single active head node
- Simple checkpoint/restart
- Fail-over to standby node
- Interruption of service
- Possible corruption of backup
- New single point of failure
- Correctness and availability NOT ALWAYS guaranteed

➔ Existing solutions:
  - SLURM batch job manager
  - PVFS/Lustre metadata server

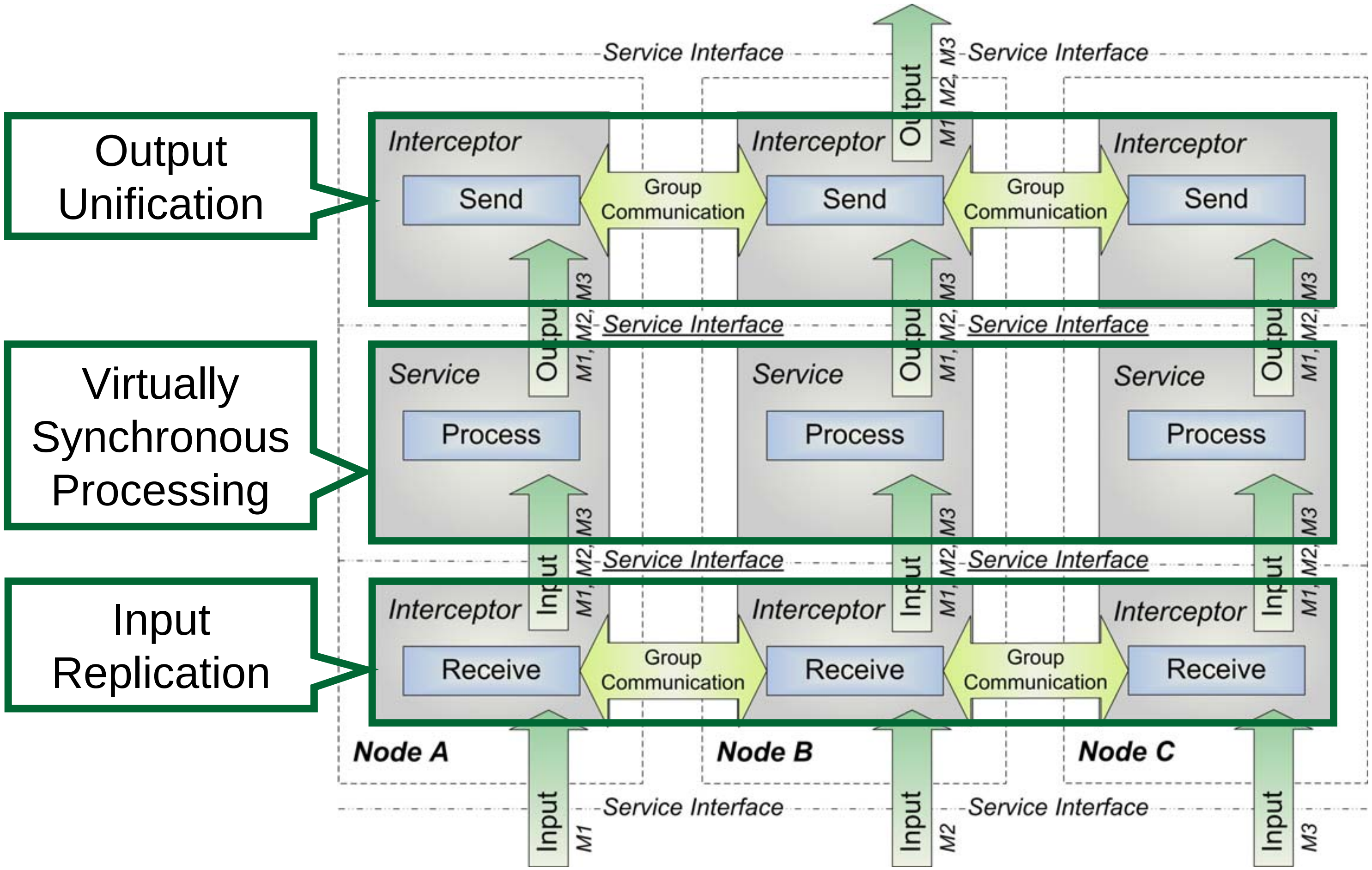
Symmetric Active/Active Redundancy



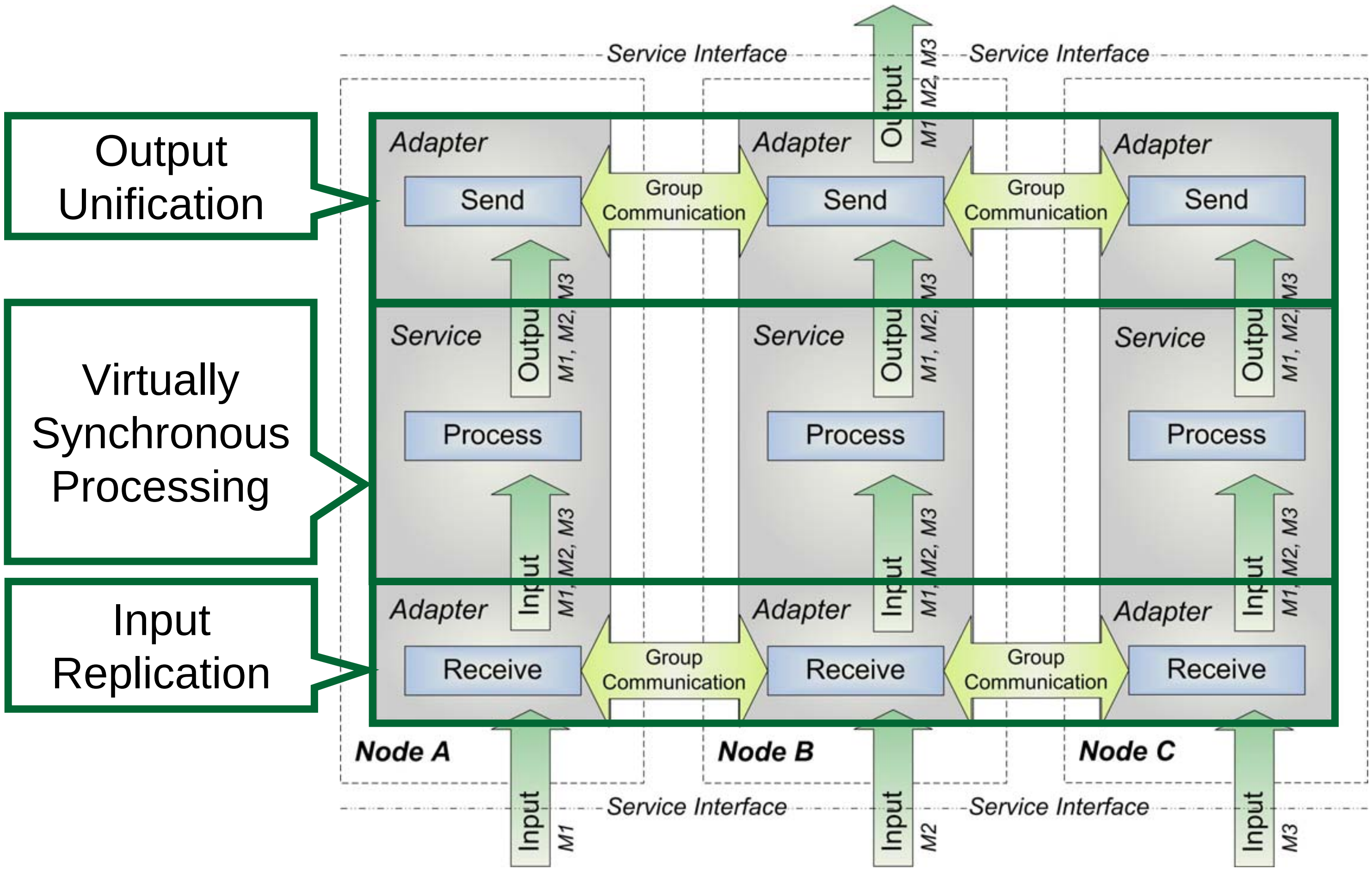
- Many active head nodes
- State-machine replication
- Virtual synchrony model
- Continuous service
- Always up-to-date
- No fail-over, no restore-over
- Work load distribution
- Complex algorithms

➔ Developed prototypes:
  - PBS Torque
  - PVFS metadata server

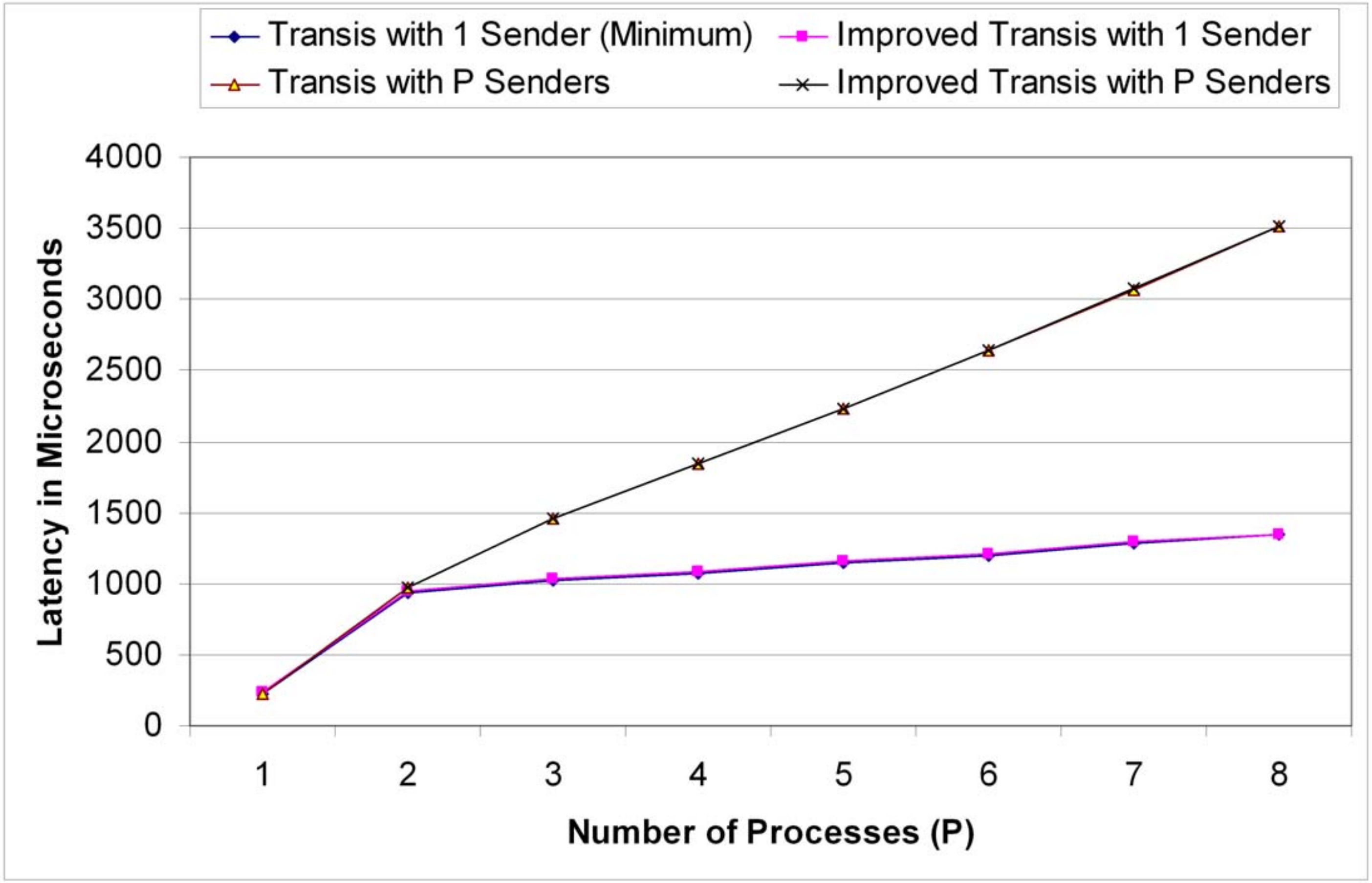
External Symmetric Active/Active Replication for Client/Service Scenarios



Internal Symmetric Active/Active Replication for Client/Service Scenarios



Total Message Order Latency of Enhanced Transis Process Group Communication Protocol



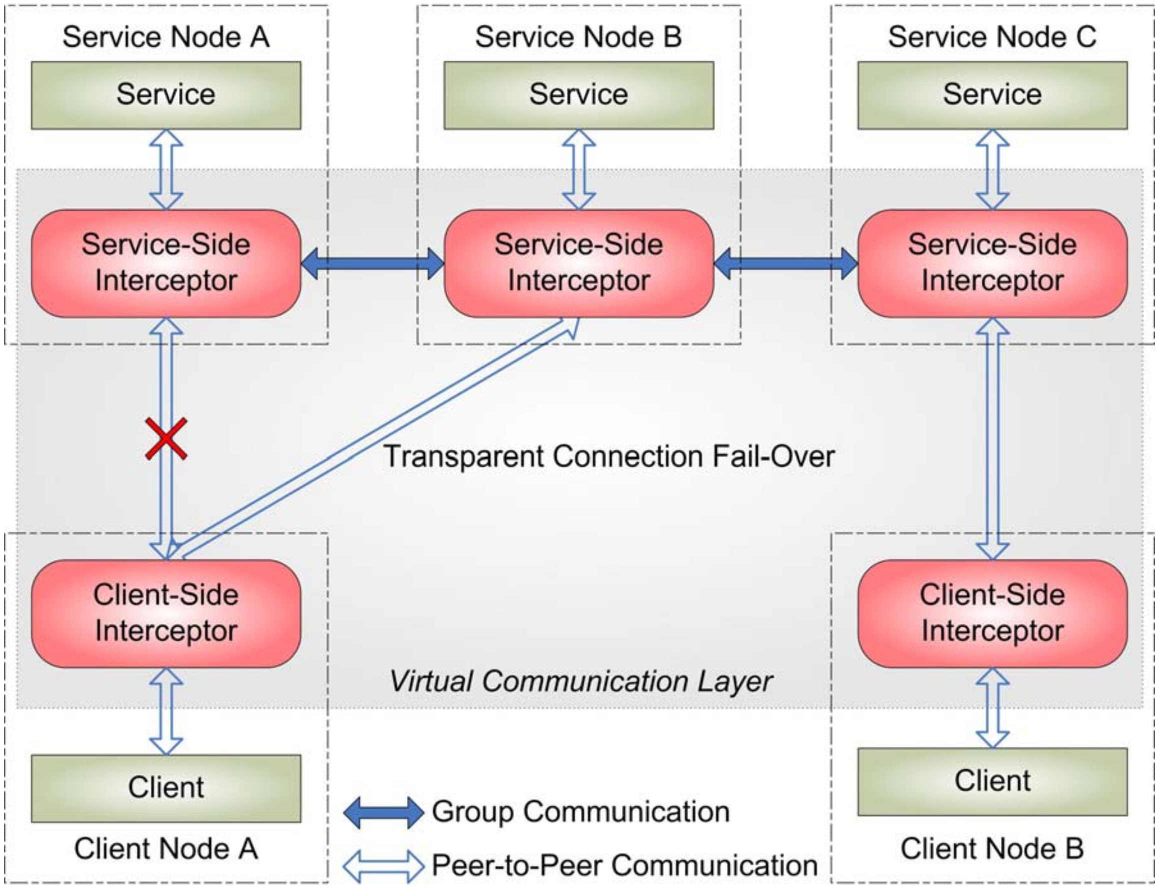
Past Accomplishments

- Symmetric active/active proof-of-concept prototypes
  - External: PBS Torque (demonstrated output unification)
  - Internal: PVFS metadata server (showed performance)
- Generalization of HA programming models
  - Active/standby replication (w/o shared disk)
  - Asymmetric active/active (HA clustering, w/o shared disk)
  - Symmetric active/active (state-machine replication)
- Enhancing the transparency of the HA infrastructure
  - Minimum adaptation to the actual service protocol
  - Virtualized communication layer (VCL) for abstraction

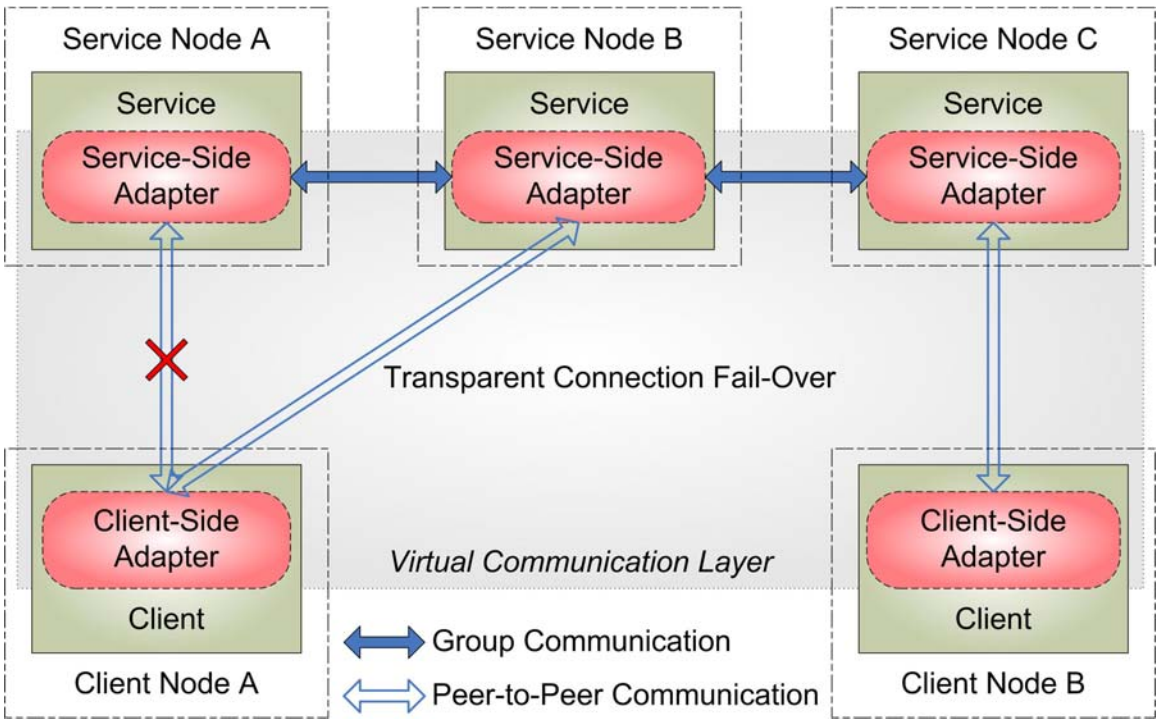
Motivation and Approach

- Inability to deal with complex dependent service scenarios, e.g., the Lustre cluster file system:
  - *n* compute nodes depend on *1* metadata service
  - *n* compute nodes depend on *m* object storage services
  - *1* metadata service depends on *m* object storage services
  - *m* object storage services depend on *1* metadata service
- Symmetric active/active replication concept and solution needed for dependent services
- If replicated services can be clients of each other, then existing replication mechanisms are sufficient

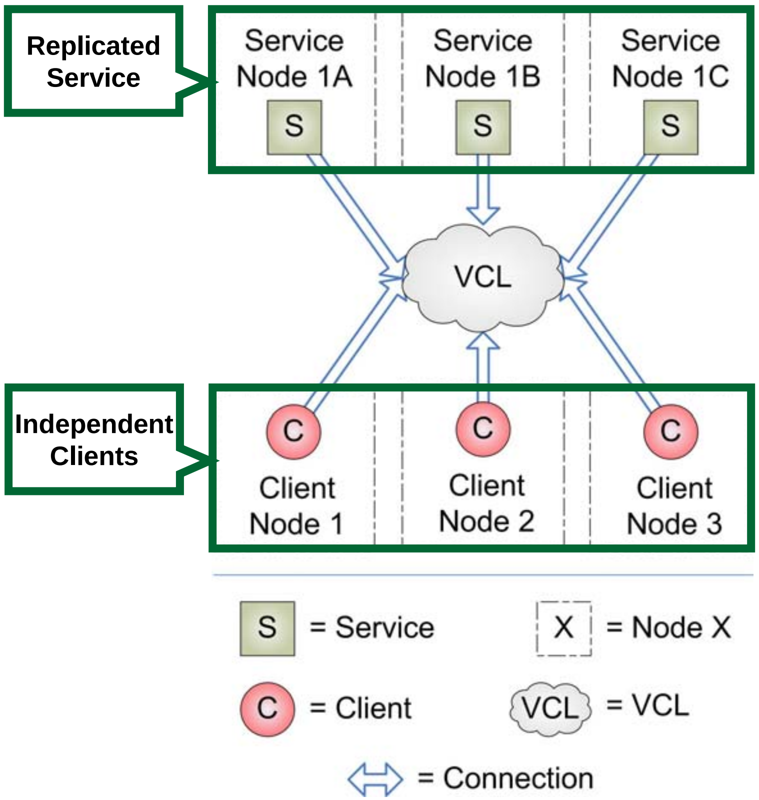
Transparent External Symmetric Active/Active Replication for Client/Service Scenarios



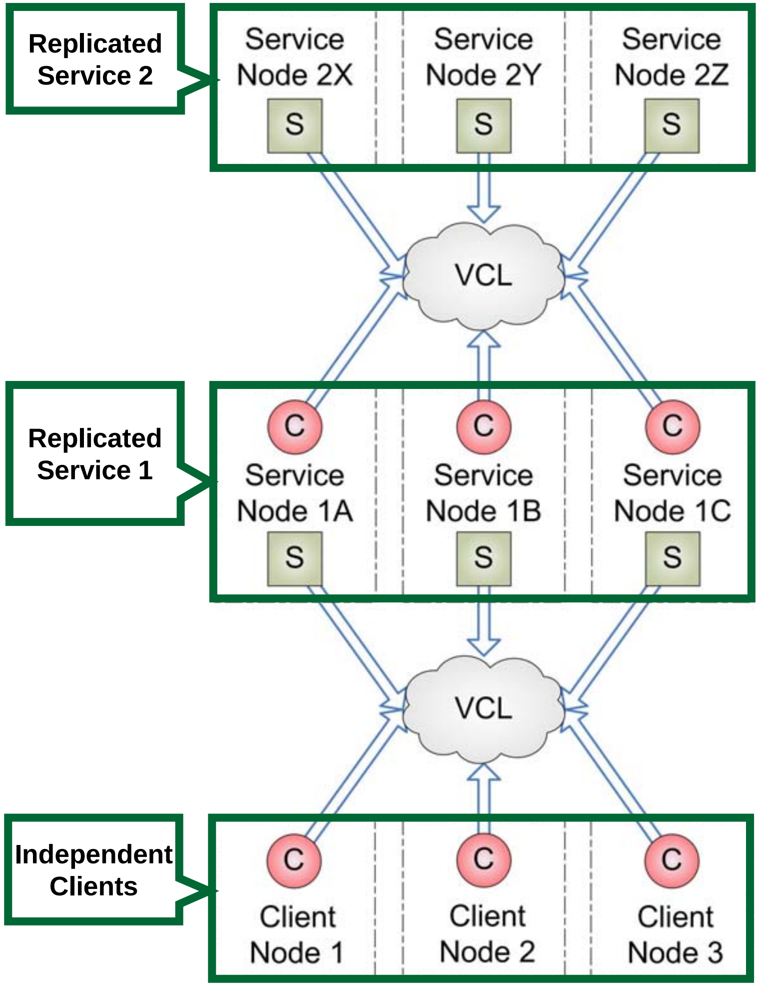
Transparent Internal Symmetric Active/Active Replication for Client/Service Scenarios



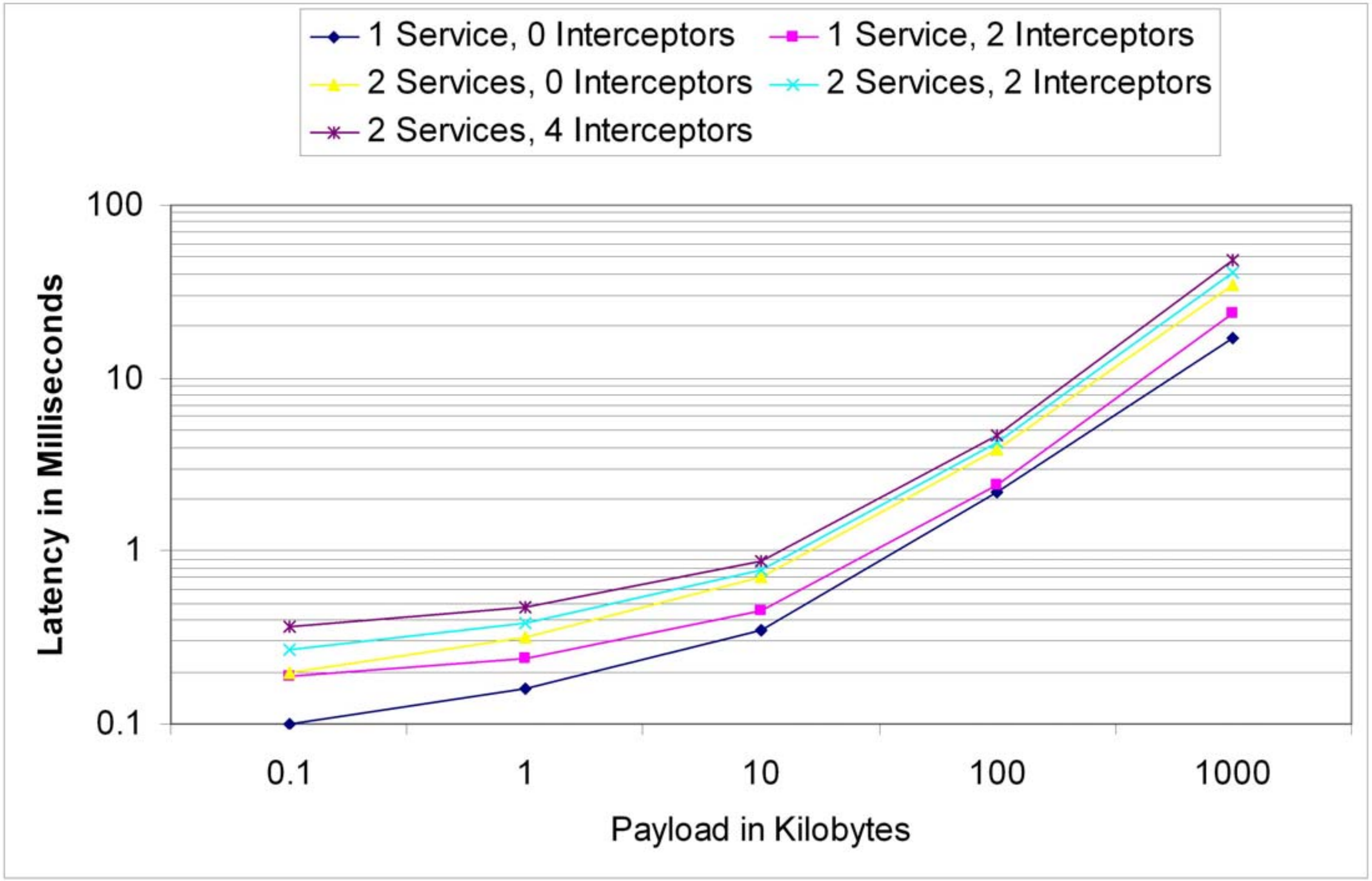
Transparent Symmetric Active/Active Replication for Client/Service Scenarios – High-Level Abstraction



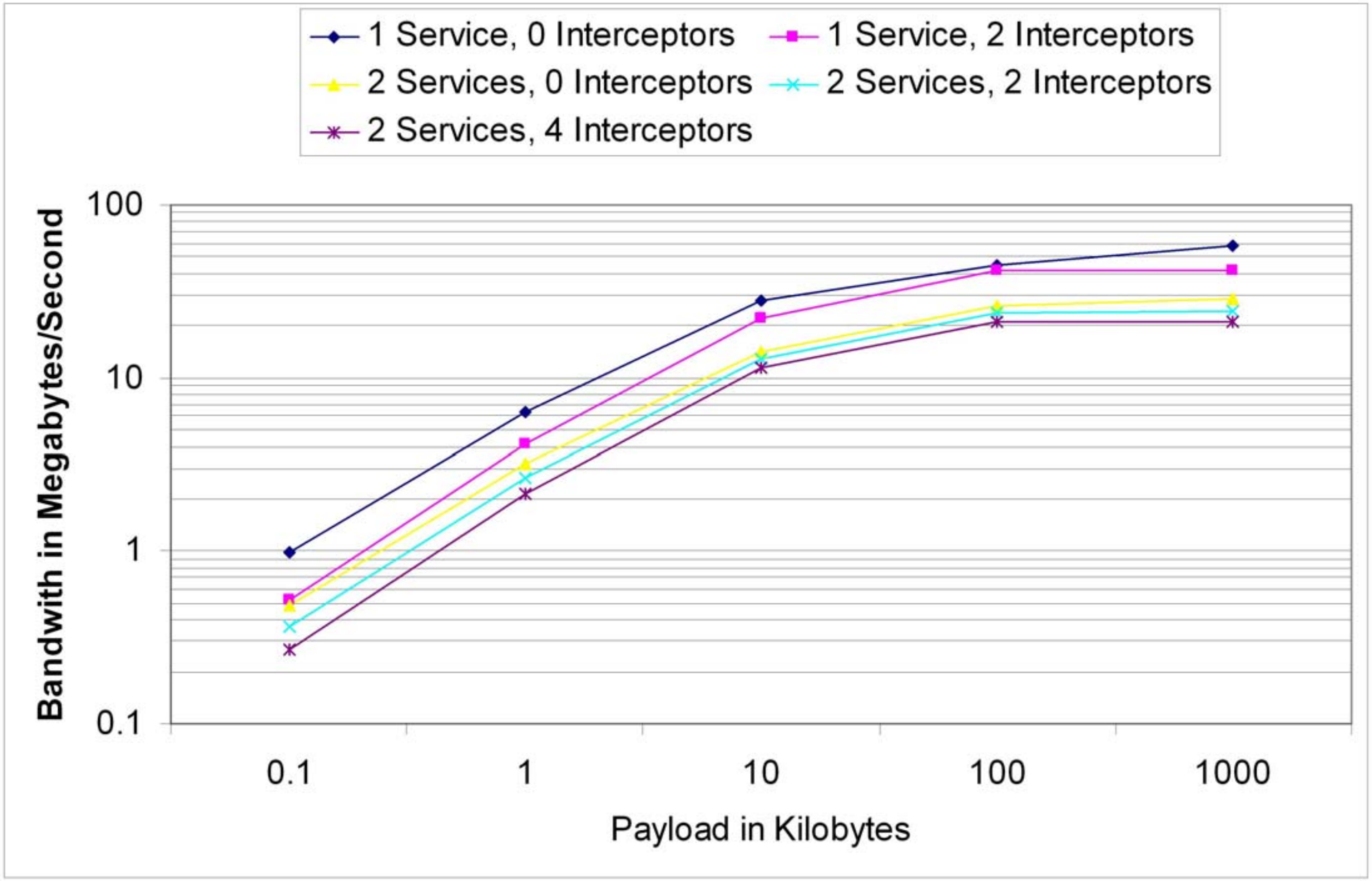
Transparent Symmetric Active/Active Replication for Client/Client+Service/Service Scenarios



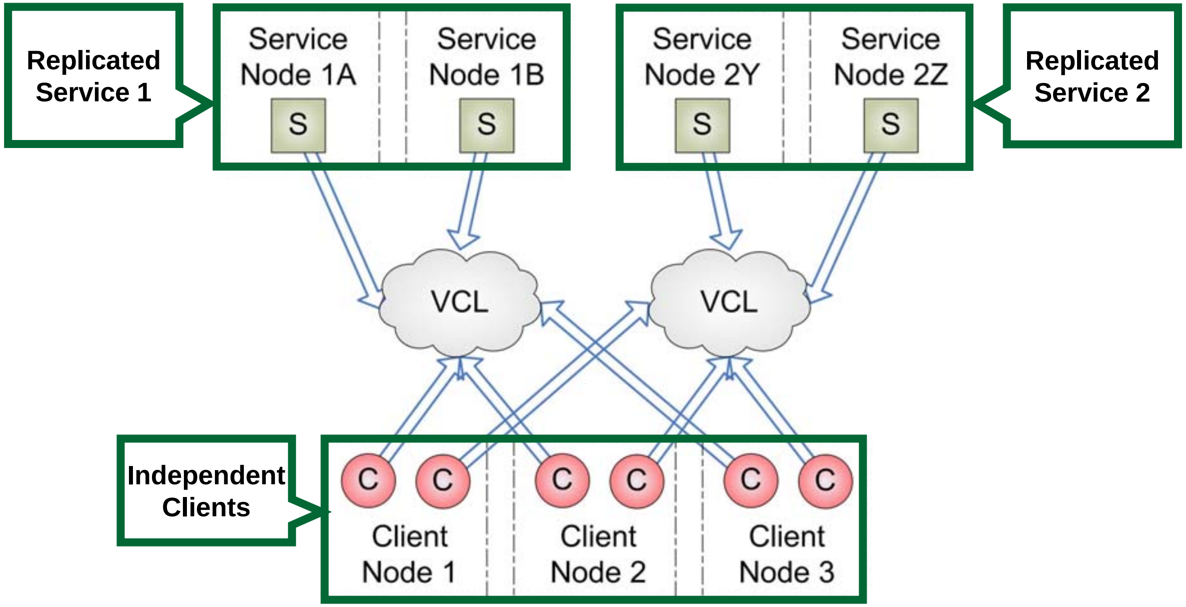
Transparent Symmetric Active/Active Replication for Client/Client+Service/Service Scenarios: Latency



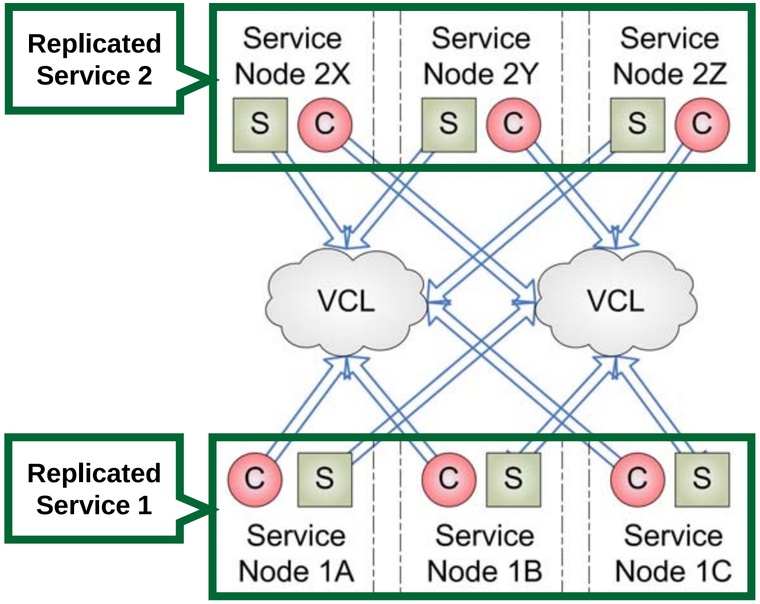
Transparent Symmetric Active/Active Replication for Client/Client+Service/Service Scenarios: Bandwidth



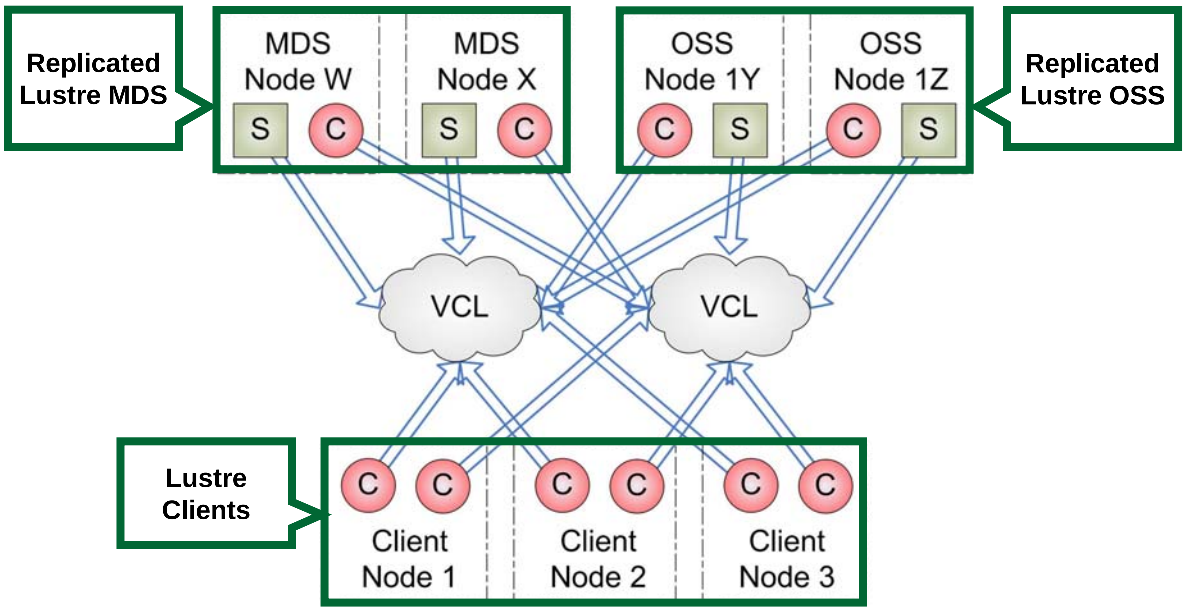
Transparent Symmetric Active/Active Replication for Client/2 Services Scenarios



Transparent Symmetric Active/Active Replication for Service/Service Scenarios



Example: Transparent Symmetric Active/Active Replication for the Lustre Cluster File System



Conclusion

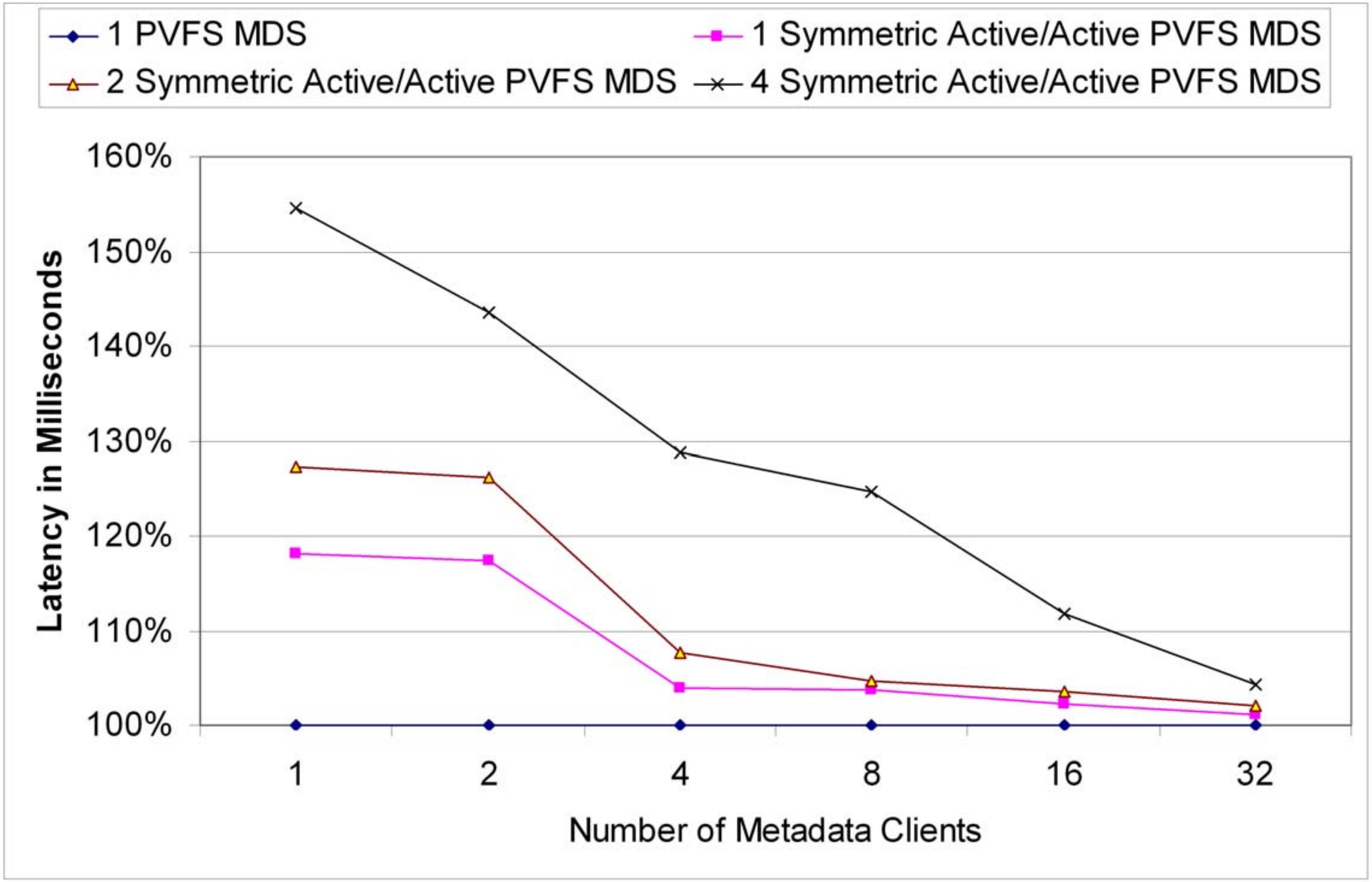
- Provided a concept for symmetric active/active replication in complex dependent service scenarios
- Since replicated services can be clients of each other, existing replication mechanisms can be used
- A high-level abstraction allows to decompose service interdependencies into client/service dependencies
- Future work focuses on implementing the presented concept with specific services in the field
- Possible adaptation for service-level HA with strong consistency semantics in critical SOA infrastructures

Symmetric Active/Active Replication for Dependent Services

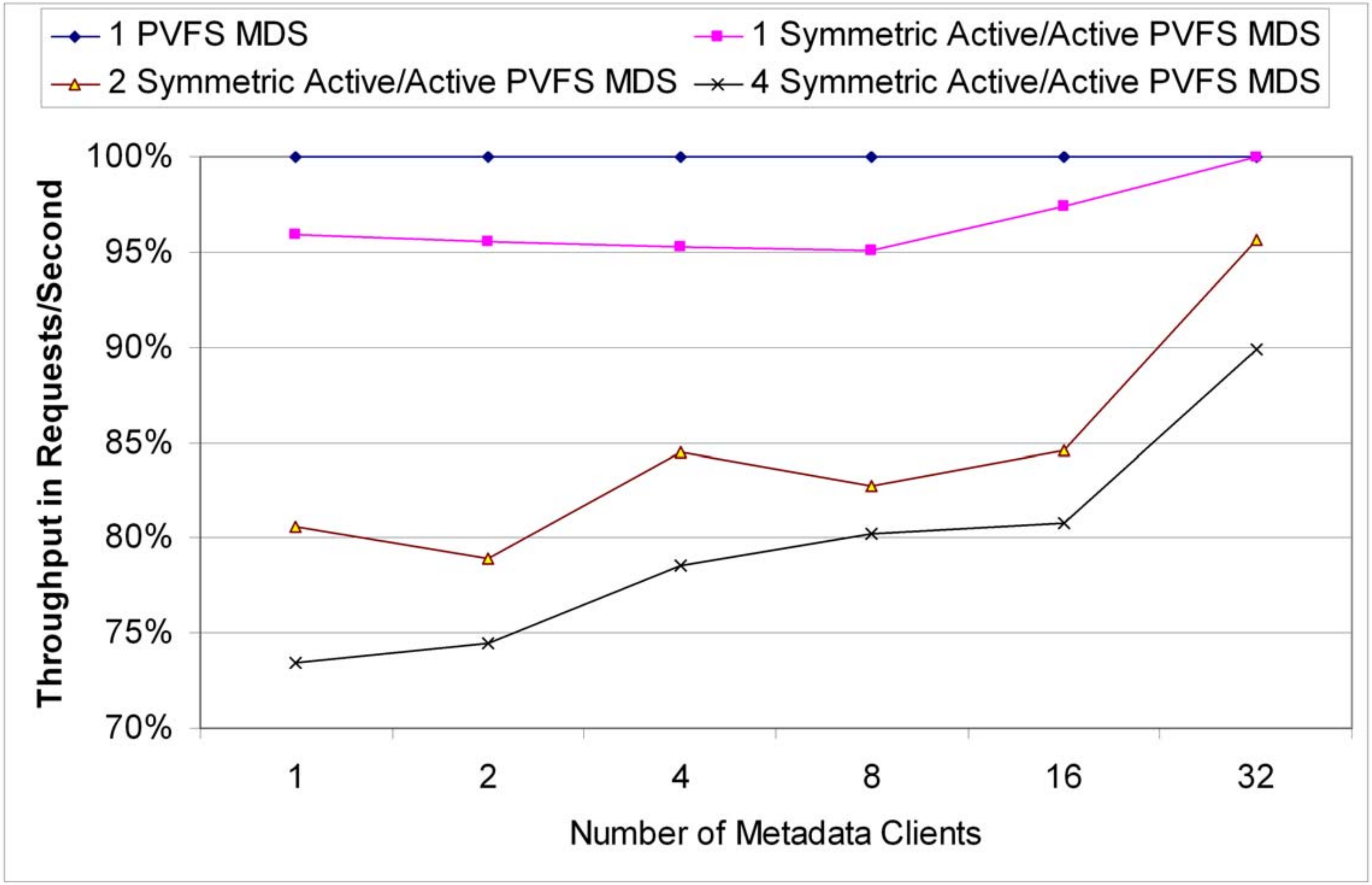
Christian Engelmann[1,2], Stephen L. Scott[1],
Chokchai (Box) Leangsuksun[3], Xubin (Ben) He[4]

[1] Oak Ridge National Laboratory, Oak Ridge, USA
[2] The University of Reading, Reading, UK
[3] Louisiana Tech University, Ruston, USA
[4] Tennessee Tech University, Cookeville, USA

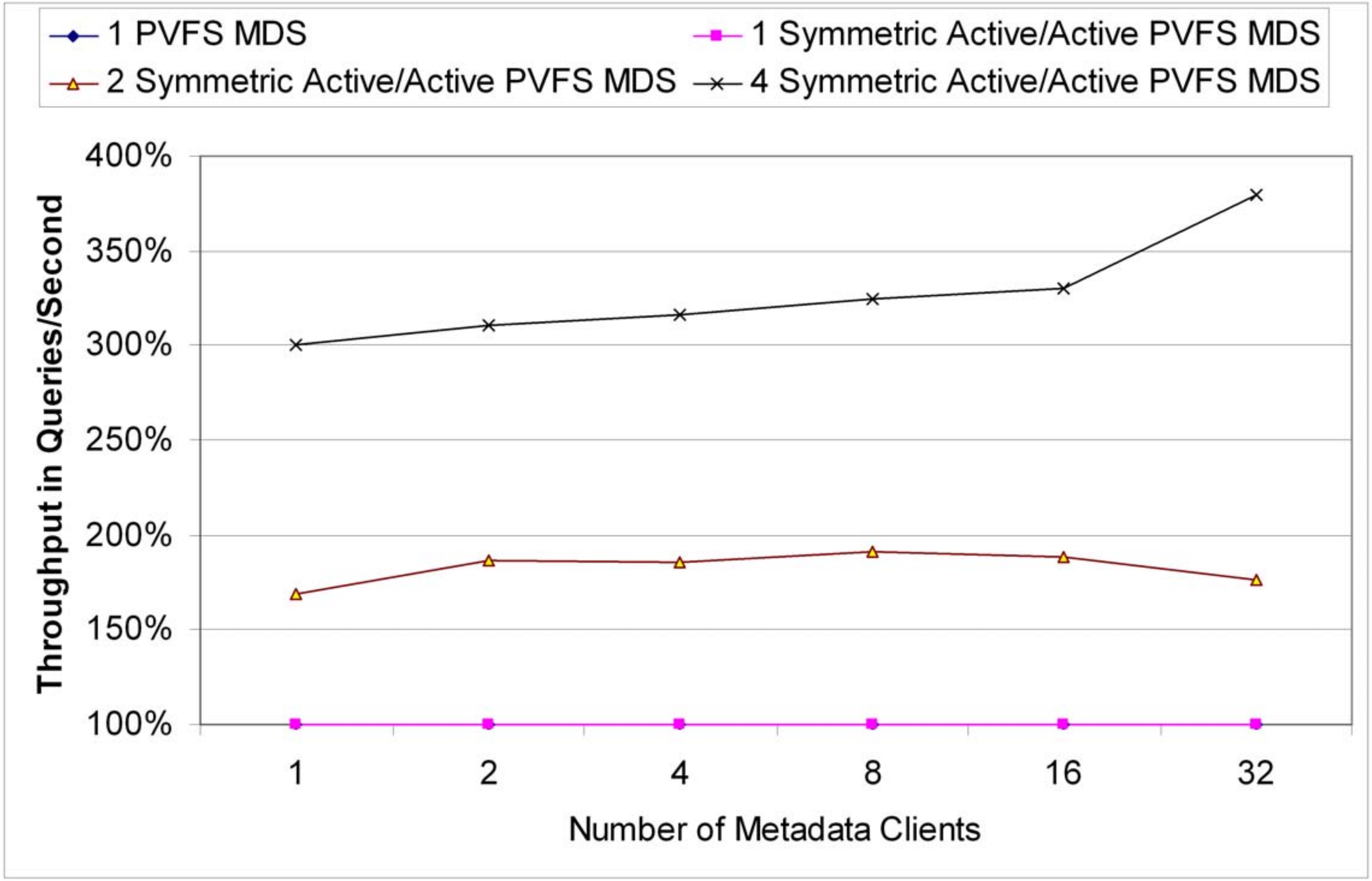
Replication & Performance: Symmetric Active/Active PVFS Metadata Service Latency



Replication & Performance: Symmetric Active/Active PVFS Metadata Service Write/Request Throughput



Replication & Performance: Symmetric Active/Active PVFS Metadata Service Read/Query Throughput



Replication & Availability: Symmetric Active/Active Availability Measured by the Nines

- $A_{component}$ = MTTF / (MTTF + MTTR)
- A_{system} = 1 - (1 - $A_{component}$) n
- T_{down} = 8760 hours * (1 – A)
- Single node MTTF: 5000 hours
- Single node MTTR: 72 hours

Nodes	Availability	Est. Annual Downtime
1	98.58%	5d 4h 21m
2	99.97%	1h 45m
3	99.9997%	1m 30s
4	99.999995%	1s

Single-site redundancy for 7 nines does not mask catastrophic events.