Office of Science
U.S. DEPARTMENT OF ENERGY

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

LOUISIANA TECH UNIVERSITY

The University of Reading

Tennessee Tech UNIVERSITY

Active/Active Replication for Highly Available HPC System 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

Scientific High-End Computing (HEC)

- Large-scale HPC systems.
  - Tens-to-hundreds of thousands of processors.
  - Current systems: IBM Blue Gene/L and Cray XT3
  - Next-generation systems: IBM Blue Gene/P and Cray XT4
- Computationally and data intensive applications.
  - 10 TFLOP – 1PFLOP with 10 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.

Availability of Current HPC Systems

- Today's supercomputers typically need to reboot to recover from a single failure.
- Entire systems go down (regularly and unscheduled) for any maintenance or repair (MTBF = 40-50h).
- Compute nodes sit idle while their head node or one of their service nodes is down.
- Availability will get worse in the future as the MTBI decreases with growing system size.
- ➢ *Why do we accept such significant system outages due to failures, maintenance or repair?*

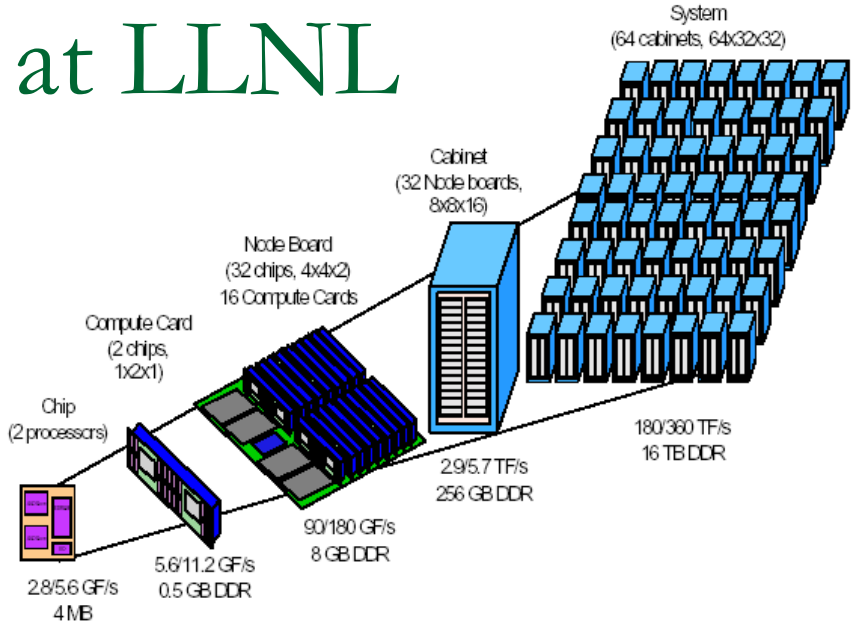
Availability Measured by the Nines

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

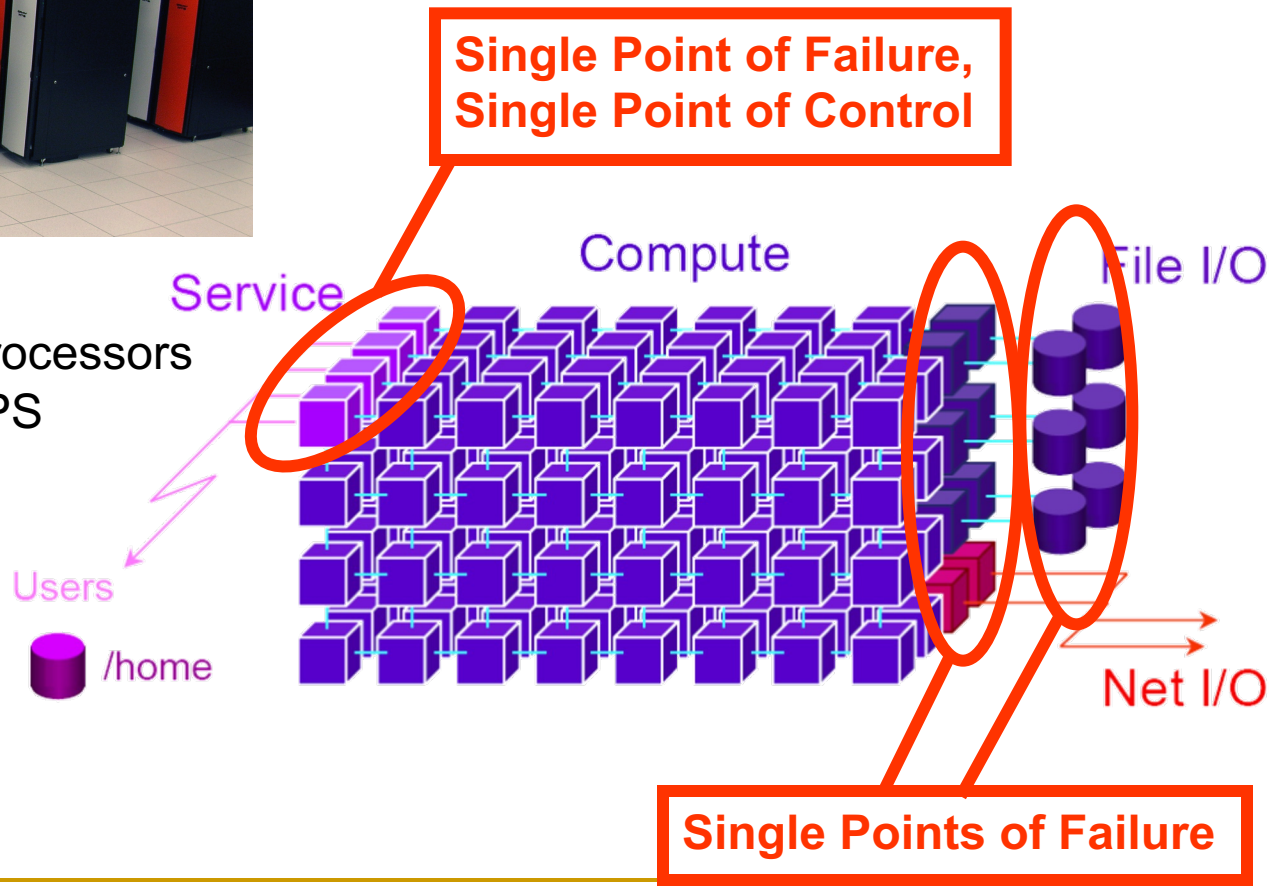
IBM Blue Gene/L at LLNL

- #1 in Top 500.
- 367 TFLOPS.
- 131072 (700MHz) Power PC processors.
- 32 TB RAM.
- Partition (512 nodes) outage on single failure.
- MTBF = 40-50 hours.
- Weak I/O system prohibits checkpointing.

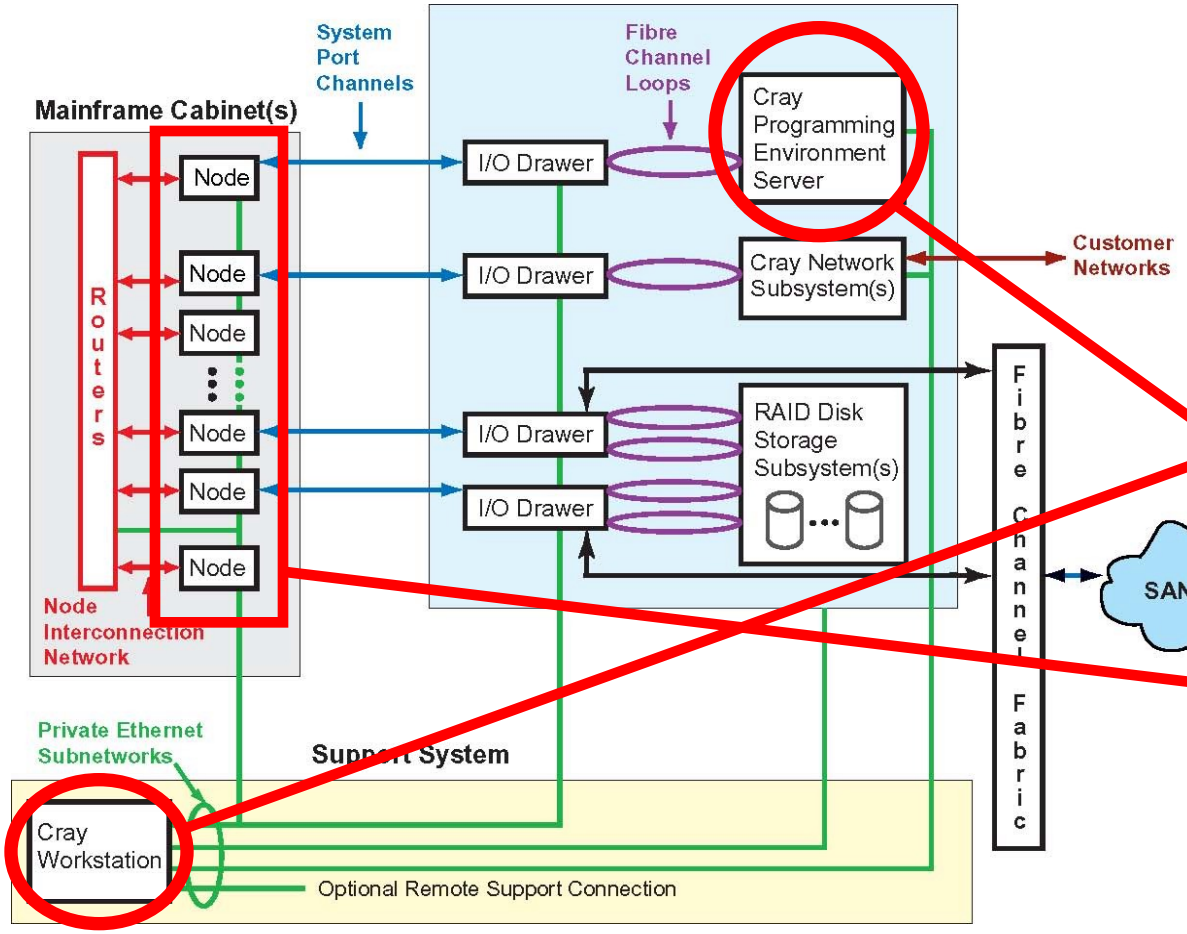


Clusters: Cray XT3 (Jaguar)

- Number 10 in Top 500
- 5212 AMD Opteron Processors
- 10TB RAM, 25 TFLOPS



Vector Machines: Cray X1 (Phoenix)



Single Points of Failure, Single Points of Control

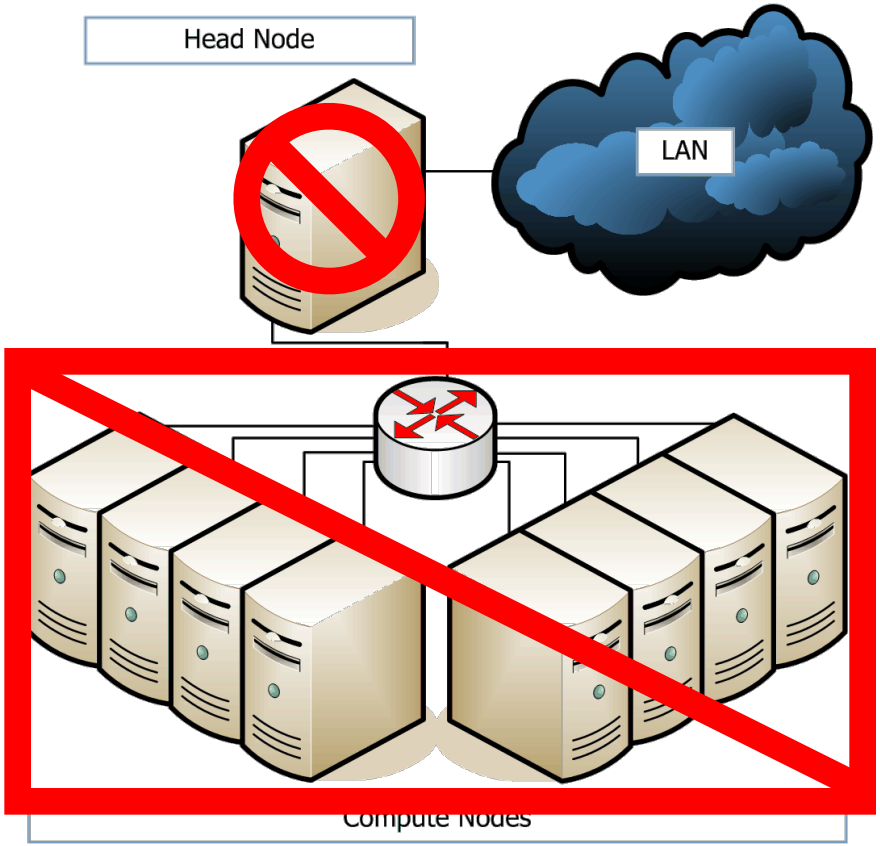
Single Points of Failure

- Number 17 in Top 500
- 1024 Vector Processors
- 4TB RAM, 18 TFLOPS

Research and Development Goals

- Provide high-level RAS capabilities for current tera-scale and next-generation petascale HEC systems.
- Eliminate many of the numerous single-points of failure and control in today's HEC systems.
- *Development of techniques to enable HEC systems to run computational jobs 24x7.*
- *Development of proof-of-concept prototypes and production-type RAS solutions.*

Single Head/Service Node Problem



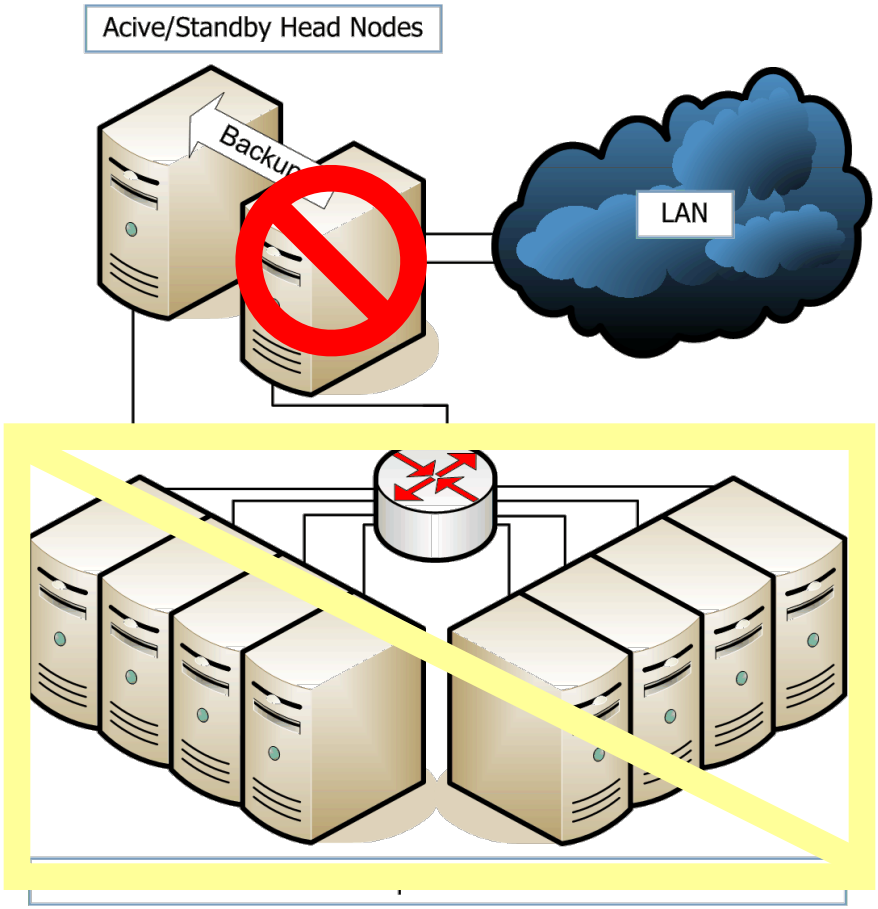
- Single point of failure.
- Compute nodes sit idle while head/service node is down.
- A = MTTF / (MTTR + MTTF)
- MTTF depends on head node hardware/software quality.
- MTTR depends on the time it takes to repair/replace node.

➢ MTTR = 0 ➔ A = 1.0 (100%) **continuous availability**.

High Availability though Redundancy

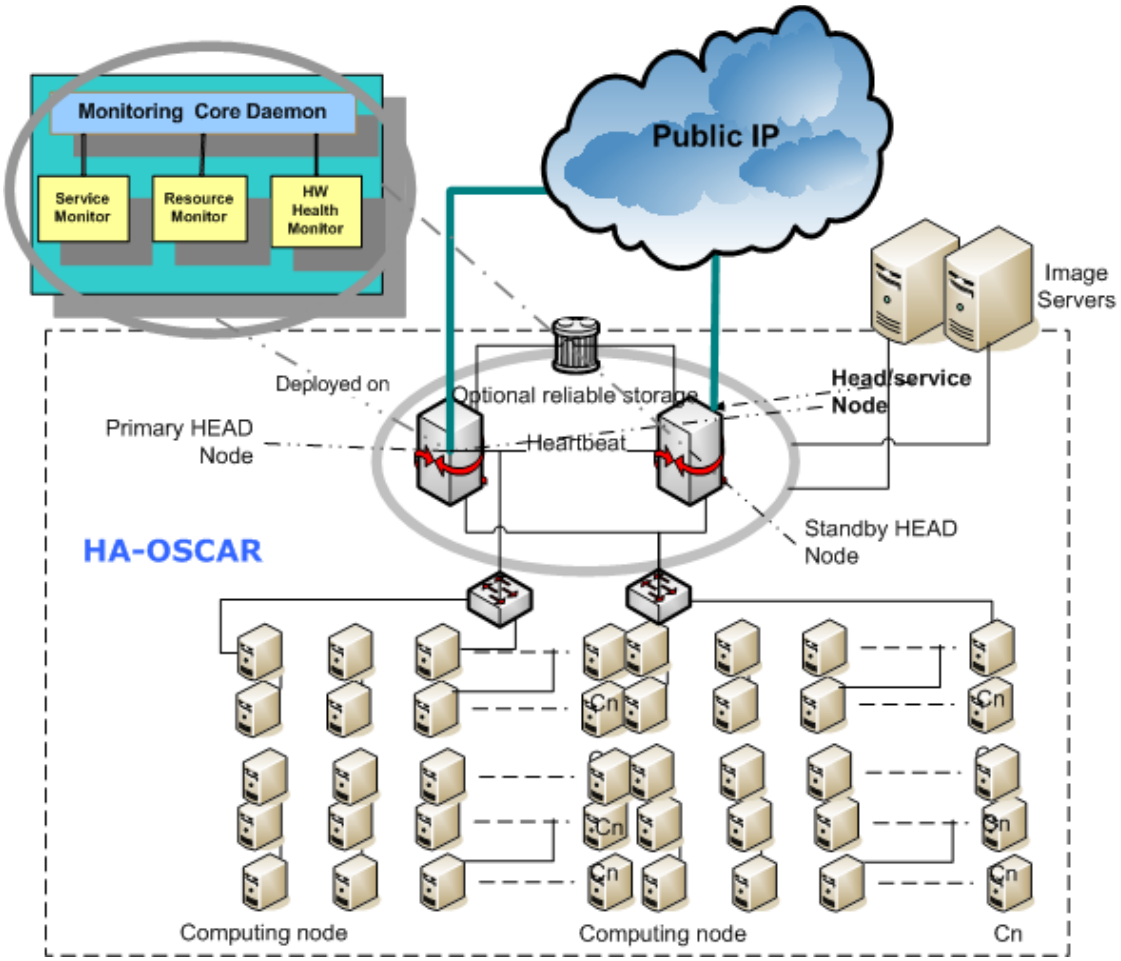
- High availability solutions are based on system component redundancy.
- If a component fails, the system is able to continue to operate using a redundant component.
- The level of availability depends on high availability model and replication strategy.
- MTTR of a system can be significantly decreased.
- Loss of state can be considerably reduced.
- SPoF and SPoC can be completely eliminated.

Active/Standby Head/Service Nodes



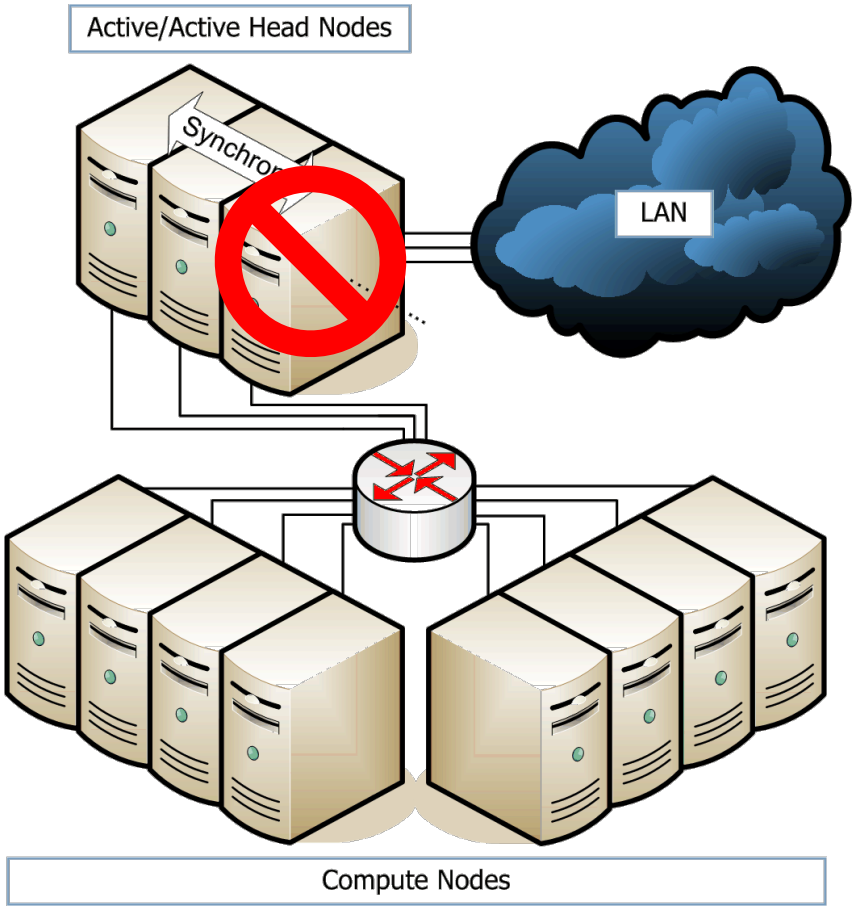
- Single active head node.
- Backup to shared storage.
- Simple checkpoint/restart.
- Fail-over to standby node.
- Idle standby head node.
- Rollback to backup.
- Service interruption for the time of the fail-over.
- Service interruption for the time of restore-over.

Active/Standby PBS with HA-OSCAR



LOUISIANA TECH UNIVERSITY

S-Active/Active Head/Service Nodes

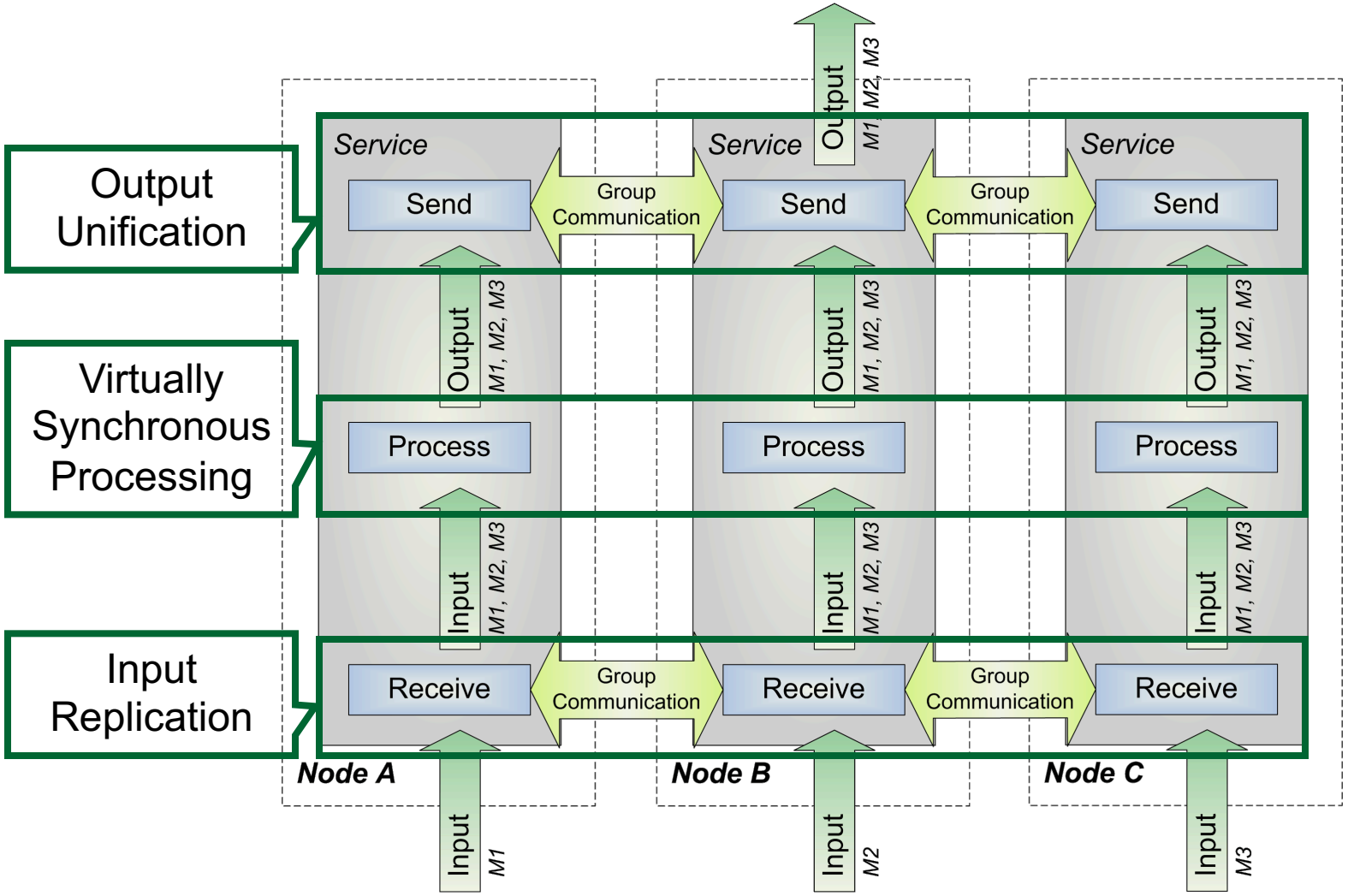


- Many active head nodes.
- Work load distribution.
- Symmetric replication between head nodes.
- Continuous service.
- Always up-to-date.
- No fail-over necessary.
- No restore-over necessary.
- Virtual synchrony model.
- Complex algorithms.

Group Communication

- Process group communication layers can provide symmetric active/active replication by:
  - Capturing the input of a service from the user.
  - Feeding multiple redundant services with the same input.
  - Receiving and unifying the output of these services.
  - Notifying the administrator upon single service failure.
  - Dynamically managing the service group (join/leave).
- However, the virtual synchrony model is complex.
- Also, event-based programming is required.

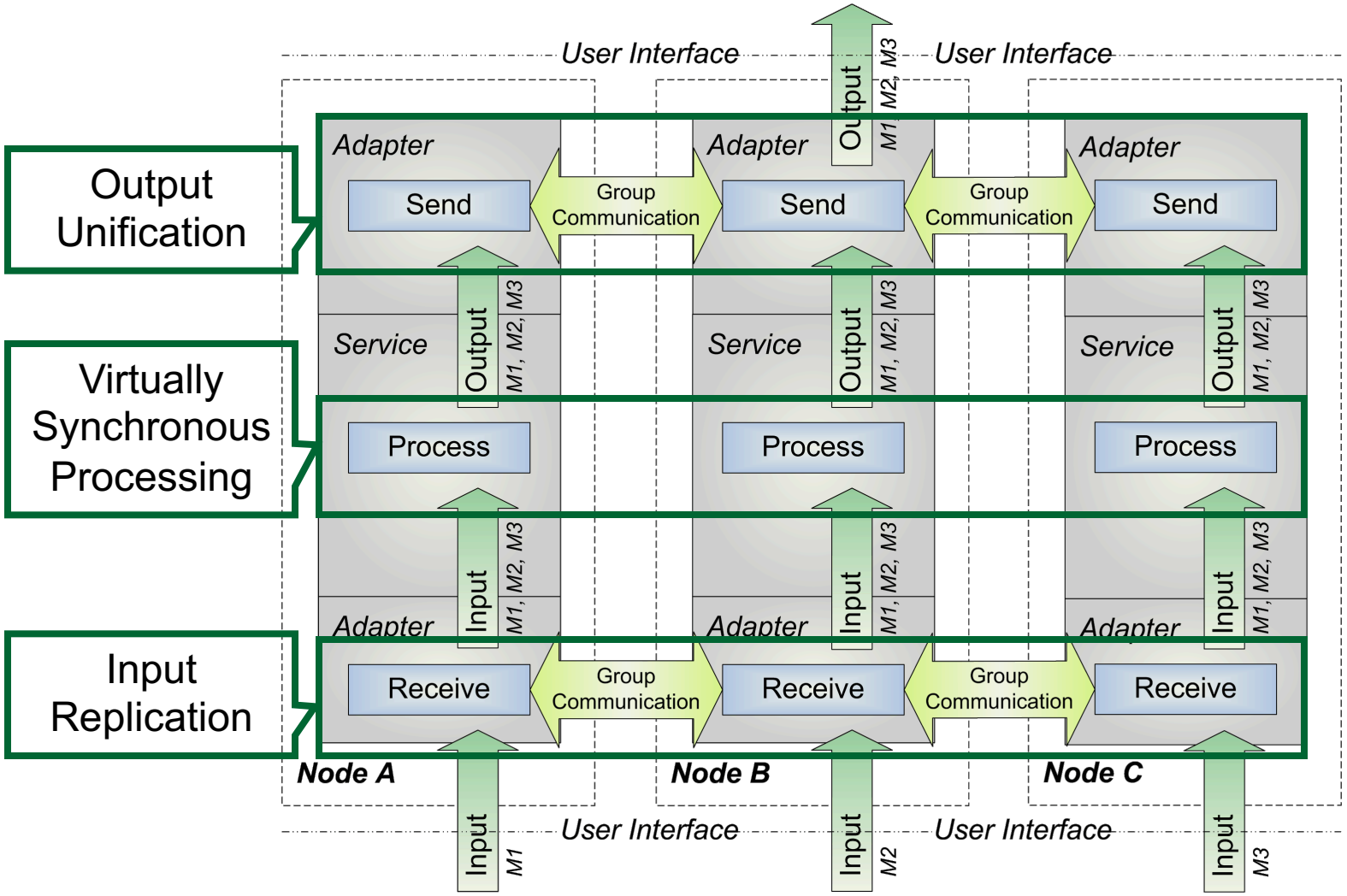
Symmetric Active/Active Replication



Symmetric A/A with Existing Services

- Two distinct replication methods: internal / external.
- Internal replication, i.e. inside the service:
  - Adapter catches input and output.
  - Service follows the event-based programming model.
  - Additional hooks may perform fine-grain state transitions.
  - Performance gain by interleaving non-concurrent fine-grain state transitions (similar to processor pipelines).
  - ↑ Higher performance.
  - ↓ Modification of existing code.

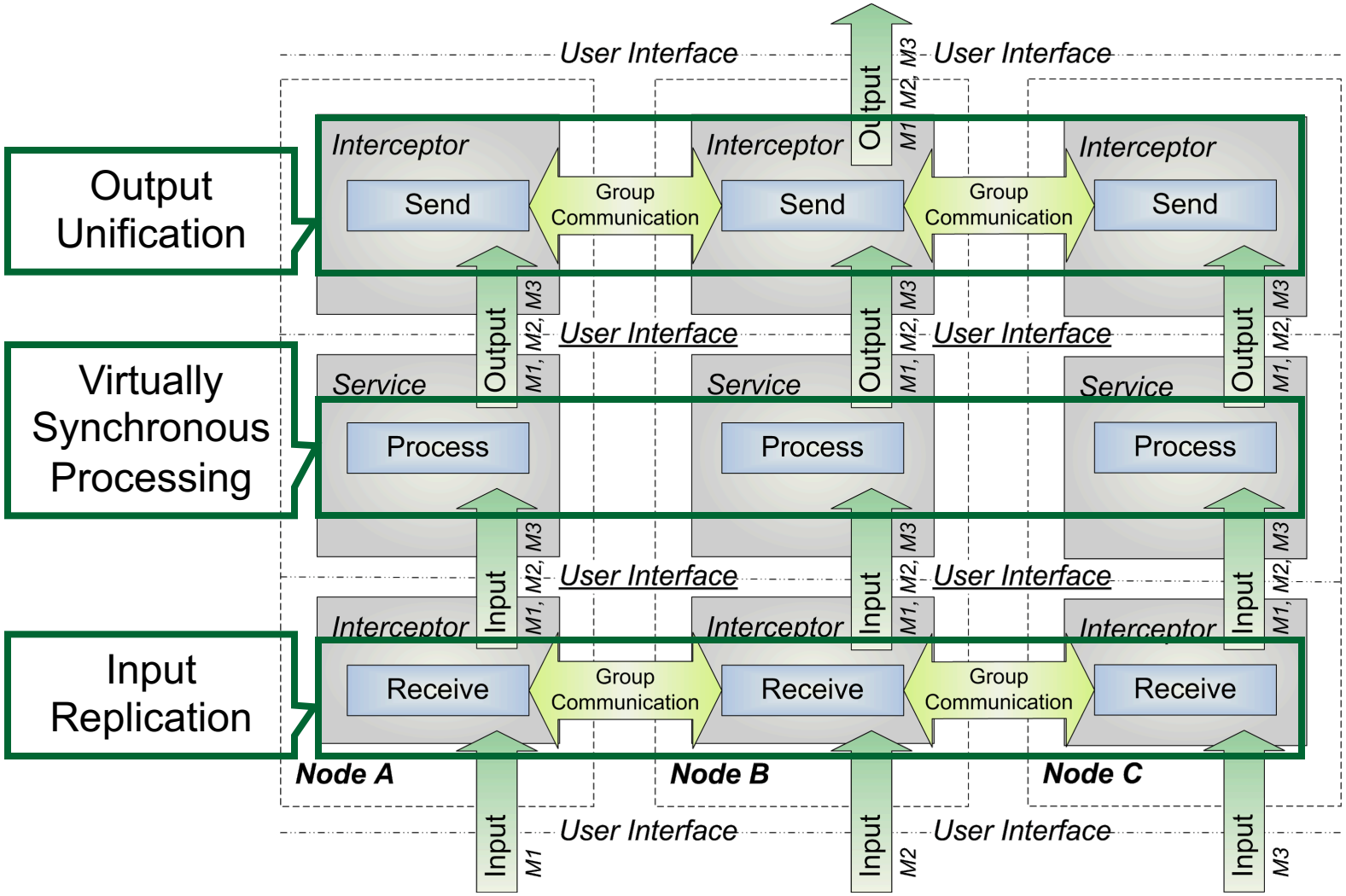
Internal Symmetric A/A Replication



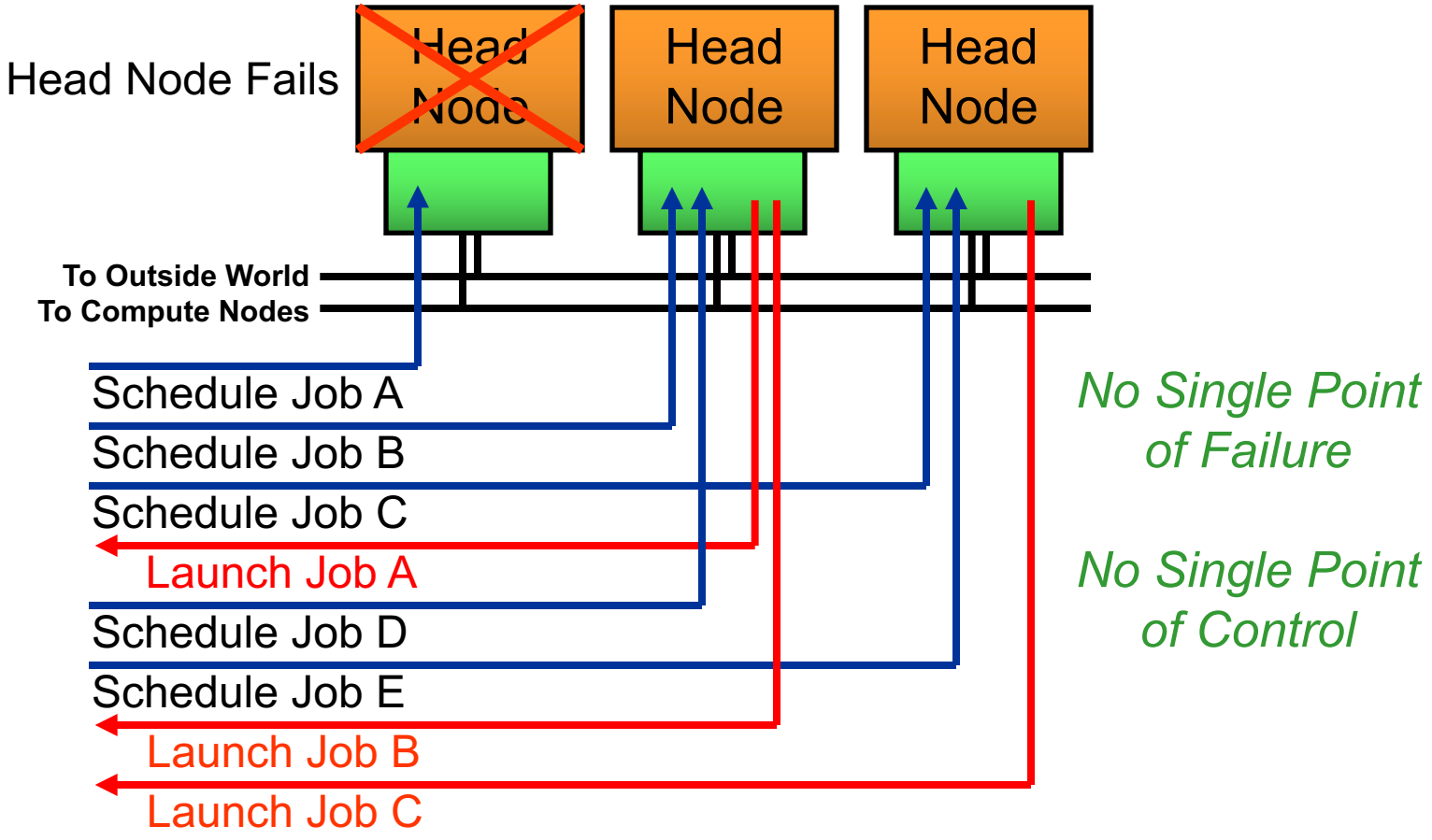
Symmetric A/A with Existing Services

- External replication, i.e. outside the service.
  - Service enclosed in a virtually synchronous environment.
  - All state transitions are based on the user interface.
  - Additional interceptor catches input and output.
  - Service follows its own programming model.
  - Coarse-grain mutual exclusive state transitions.
  - ↑ No modification of existing code.
  - ↓ Lower performance.

External Symmetric A/A Replication



Symmetric A/A Job Scheduler Example



Active/Active Redundancy for Nines

$A_{component} = MTBF / (MTBF + MTTR)$

$A = 1 - (1 - A_{component})^n$

$T_{down} = 8760 * (1 - A)$

No. HN	Availability	Downtime
1	98.580441640%	5d 4h 21m

Based on: MTBF 5000-hours, MTTR 72-hours

Active/Active Redundancy for Nines

$A_{component} = MTBF / (MTBF + MTTR)$

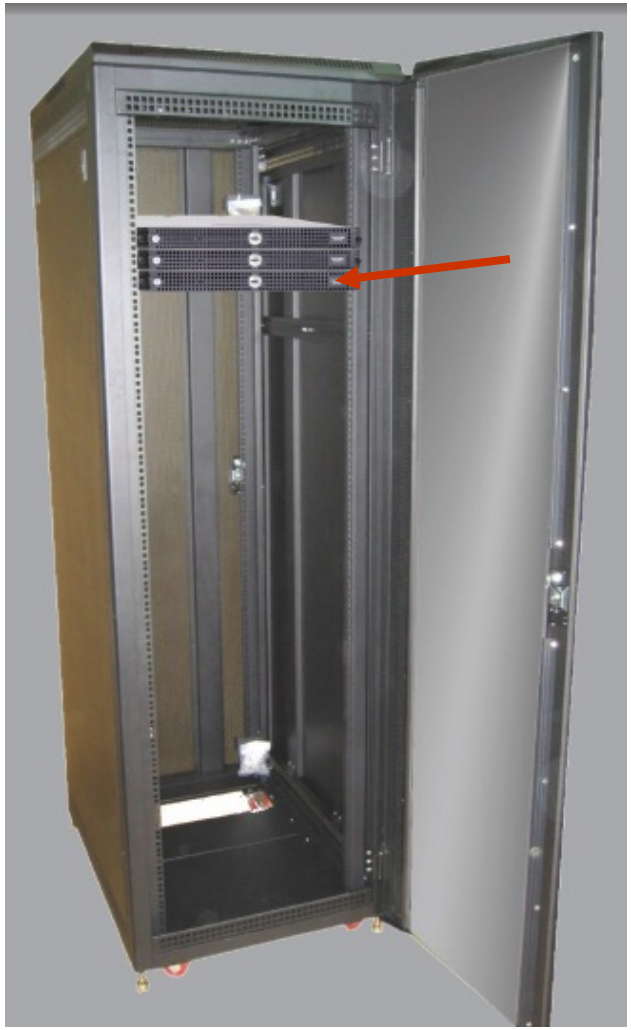
$A = 1 - (1 - A_{component})^n$

$T_{down} = 8760 * (1 - A)$

No. HN	Availability	Downtime
1	98.580441640%	5d 4h 21m
2	99.979848540%	1h 45m

Based on: MTBF 5000-hours, MTTR 72-hours

Active/Active Redundancy for Nines



$A_{component} = MTBF / (MTBF + MTTR)$

$A = 1 - (1 - A_{component})^n$

$T_{down} = 8760 * (1 - A)$

No. HN	Availability	Downtime
1	98.580441640%	5d 4h 21m
2	99.979848540%	1h 45m
3	99.999713938%	1m 30s

Based on: MTBF 5000-hours, MTTR 72-hours

Active/Active Redundancy for Nines

$A_{component} = MTBF / (MTBF + MTTR)$

$A = 1 - (1 - A_{component})^n$

$T_{down} = 8760 * (1 - A)$

No. HN	Availability	Downtime
1	98.580441640%	5d 4h 21m
2	99.979848540%	1h 45m
3	99.999713938%	1m 30s
4	99.999995939%	1s

Based on: MTBF 5000-hours, MTTR 72-hours

Active/Active Redundancy for Nines

$A_{component} = MTBF / (MTBF + MTTR)$

$A = 1 - (1 - A_{component})^n$

$T_{down} = 8760 * (1 - A)$

No. HN	Availability	Downtime
1	98.580441640%	5d 4h 21m
2	99.979848540%	1h 45m
3	99.999713938%	1m 30s
4	99.999995939%	1s
5	99.999999942%	18ms

Based on: MTBF 5000-hours, MTTR 72-hours

Ongoing Work

- Symmetric active/active solutions for:
  - Job and resource management service (Torque).
  - Parallel file system metadata service (PVFS2).
- More HPC system services targeted in the future.
- More Information:
  - HA-OSCAR: xcr.cenit.latech.edu/ha-oscar
  - MOLAR: www.fastos.org/molar