Reliability, Availability, and Serviceability (RAS) for High-Performance Computing

Presented by

Stephen L. Scott
Christian Engelmann

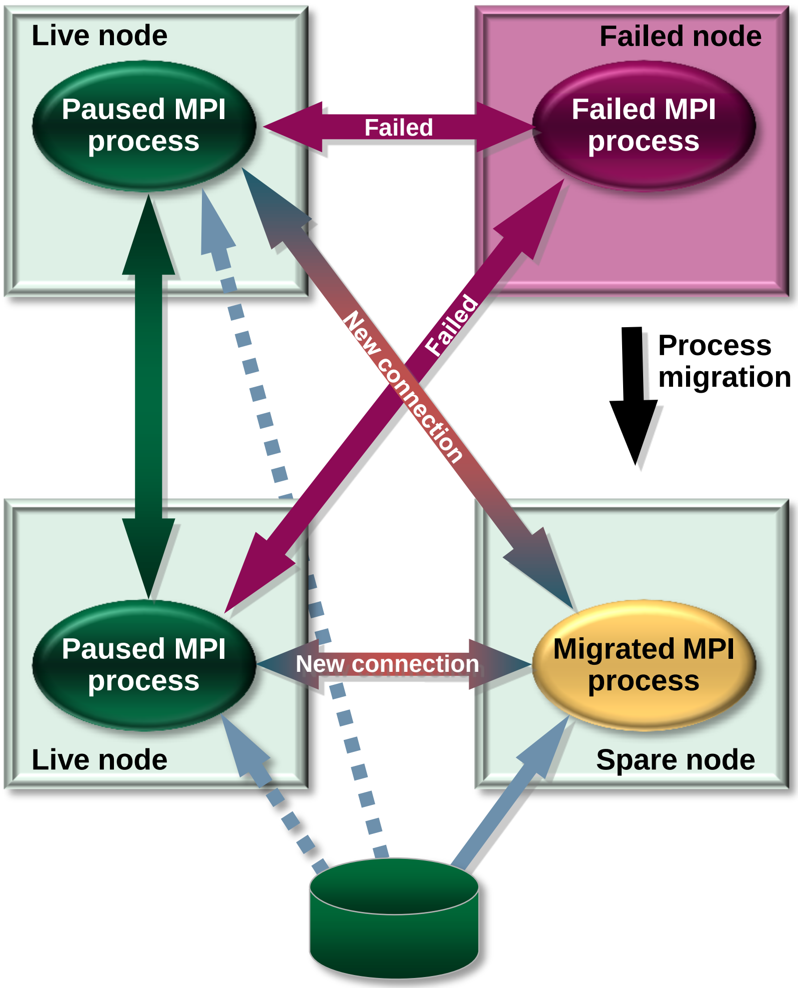
Computer Science Research Group
Computer Science and Mathematics Division

Research and development activities

- **Reactive fault tolerance** for HPC compute nodes utilizing the job-pause approach and checkpoint placement adaptation
- **Proactive fault tolerance** using migration of computation away from compute nodes that are about to fail
- **Reliability analysis** for identifying pre-fault indicators, predicting failures, and modeling and monitoring reliability
- **Holistic fault tolerance** through combination of adaptive proactive and reactive fault tolerance mechanisms

U.S. DEPARTMENT OF ENERGY | Office of Science
OAK RIDGE National Laboratory
NC STATE UNIVERSITY
LOUISIANA TECH UNIVERSITY®

Reactive fault tolerance for HPC with LAM/MPI+BLCR job-pause mechanism



- Operational nodes: Pause
  - BLCR reuses existing processes
  - LAM/MPI reuses existing connections
  - Restore partial process state from checkpoint
- Failed nodes: Migrate
  - Restart process on new node from checkpoint
  - Reconnect with paused processes
- Scalable MPI membership management for low overhead
- Efficient, transparent, and automatic failure recovery

Incremental checkpointing with BLCR

- Recent enhancement for Berkeley Lab Checkpoint/ Restart (BLCR)
- Track differences with dirty bit at PTE
- Hybrid: 1 full and k incremental checkpoints
- Available through BLCR distribution

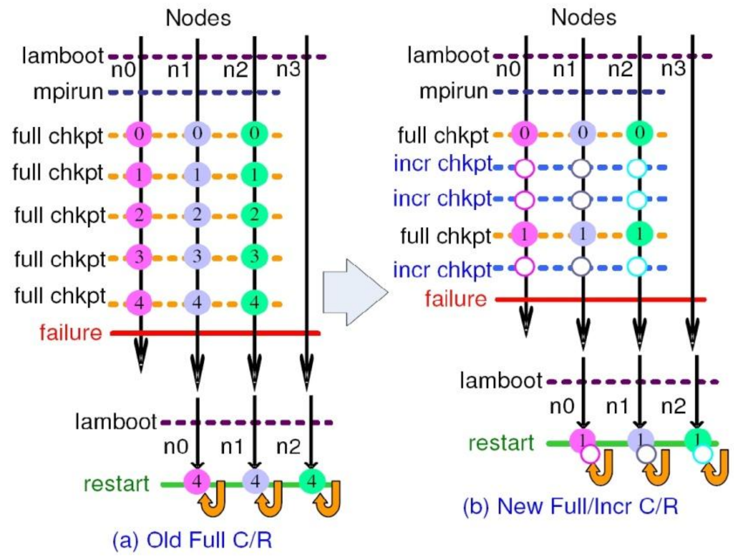
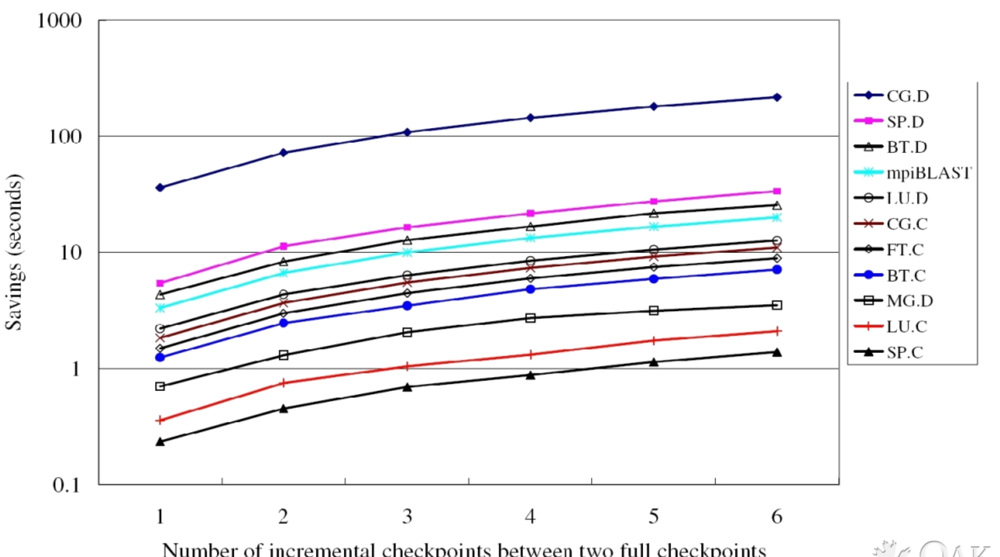
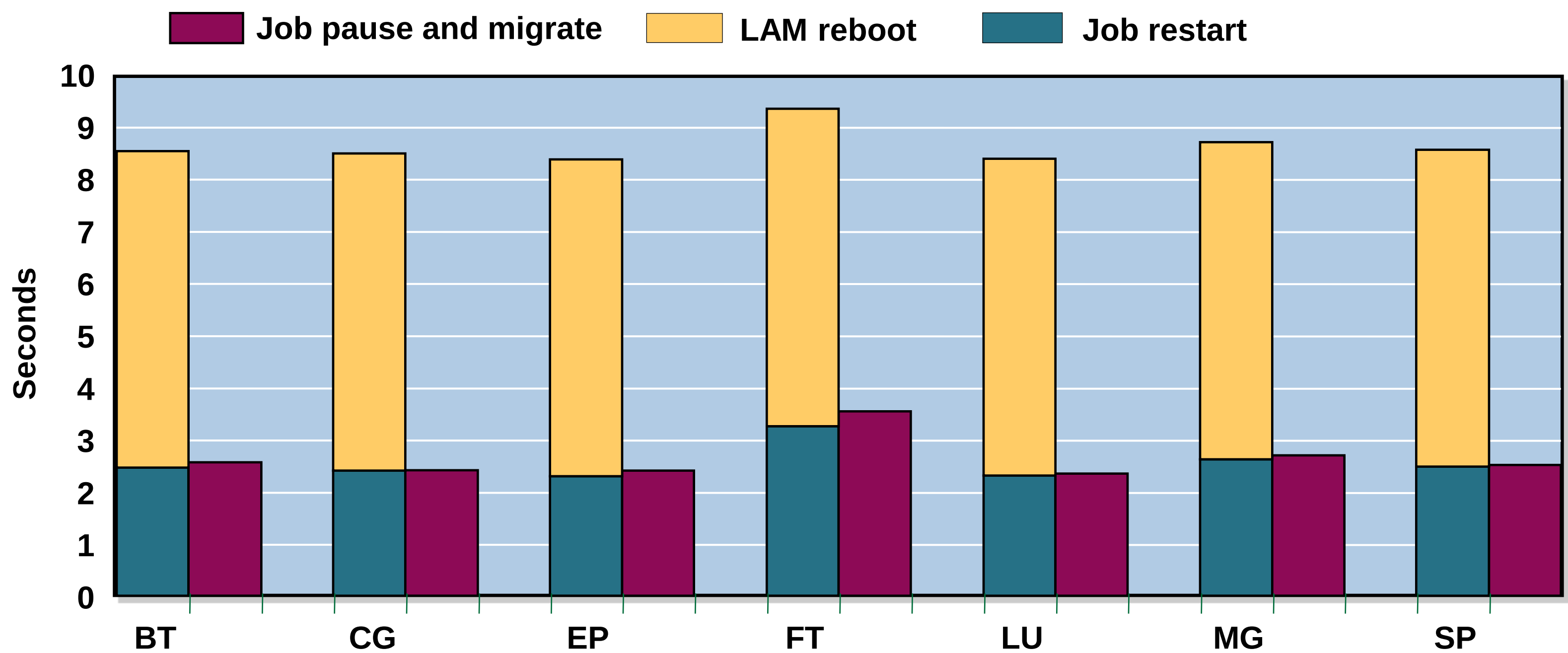


Fig. 1: Hybrid Full/Incremental C/R Mechanism vs. Full C/R



Managed by UT-Battelle
for the U.S. Department of Energy

LAM/MPI+BLCR job-pause performance



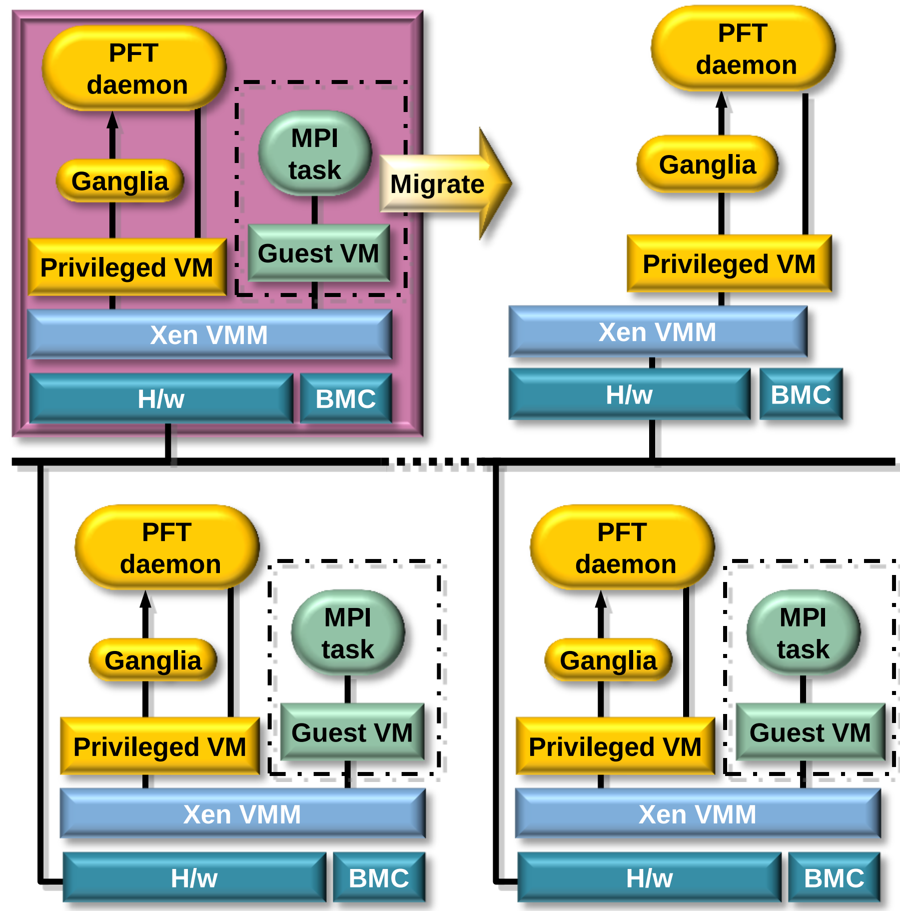
- 3.4% overhead over job restart, but
  - No LAM reboot overhead
  - Transparent continuation of execution
- No requeue penalty
- Less staging overhead

Proactive fault tolerance with migration

- Relies on a feedback-loop control mechanism
  - Application health is constantly monitored and analyzed
  - Application is reallocated to improve its health and avoid failures
- Real-time control problem
  - Need to act in time to avoid imminent failures
- No 100% coverage
  - Not all failures can be anticipated

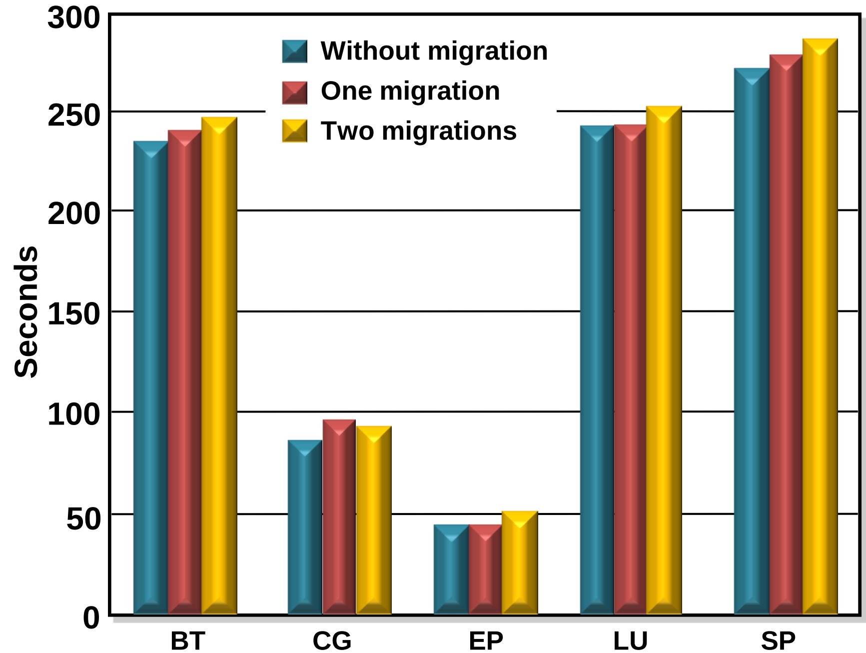
VM-level migration using Xen

- System setup
  - Xen VMM on entire system
  - Host OS for management
  - Guest OS for computation
  - Spare nodes without Guest OS
  - System monitoring in Host OS
  - Decentralized scheduler/load balancer using Ganglia
- Deteriorating node health
  - Ganglia threshold trigger
  - Migrate guest OS to spare
- Utilize Xen's migration facility



VM-level migration performance

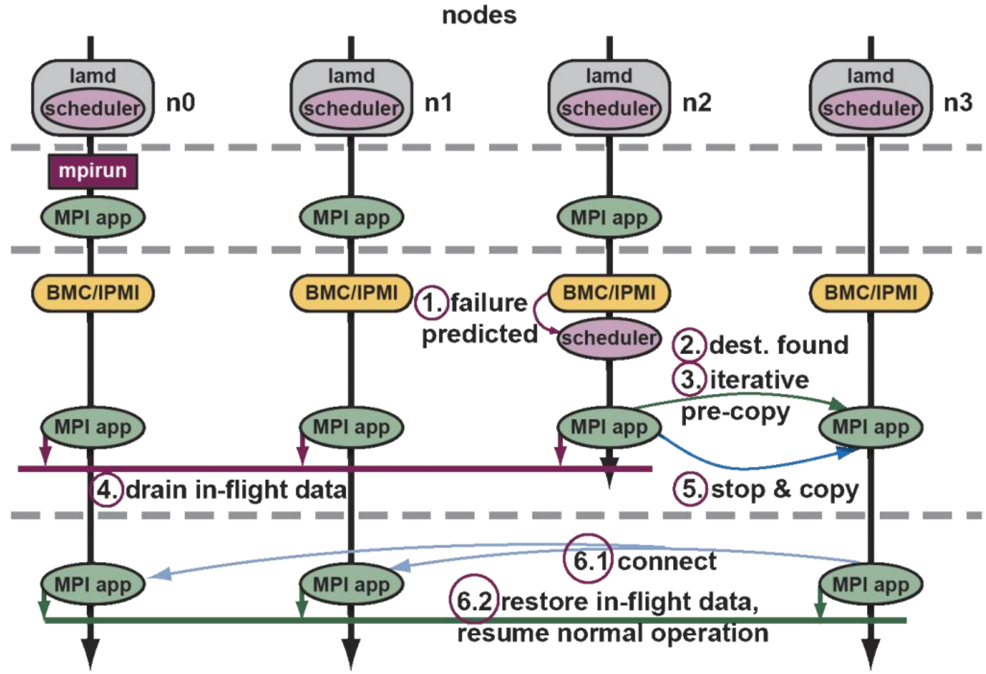
- Single node migration
  - 0.5–5% longer run time
- Double node migration
  - 2–8% longer run time
- Migration duration
  - Stop & copy : 13–14 s
  - Live : 14–24 s
- Application downtime
  - Stop & copy > live



16-node Linux cluster at NCSU with dual core, dual-processor AMD Opteron and Gigabit Ethernet

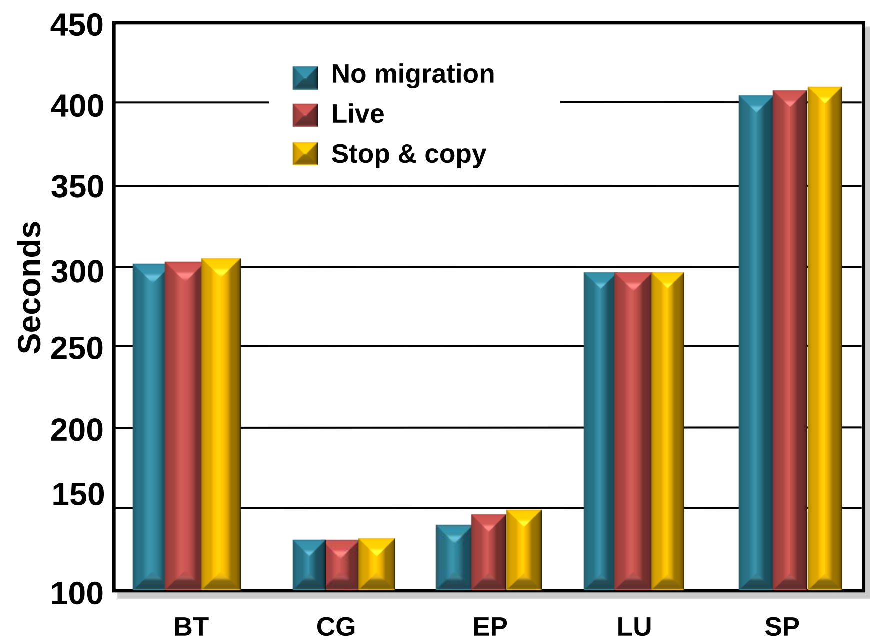
Process-level migration with BLCR

- LAM/MPI with Berkeley Lab Checkpoint/Restart (BLCR)
- Per-node health monitoring
- New decentralized scheduler/ load balancer in LAM
- New process migration facility in BLCR (stop & copy and live)
- Deteriorating node health
  - Simple threshold trigger
  - Migrate process to spare
- Available through BLCR distribution



Process-level migration performance

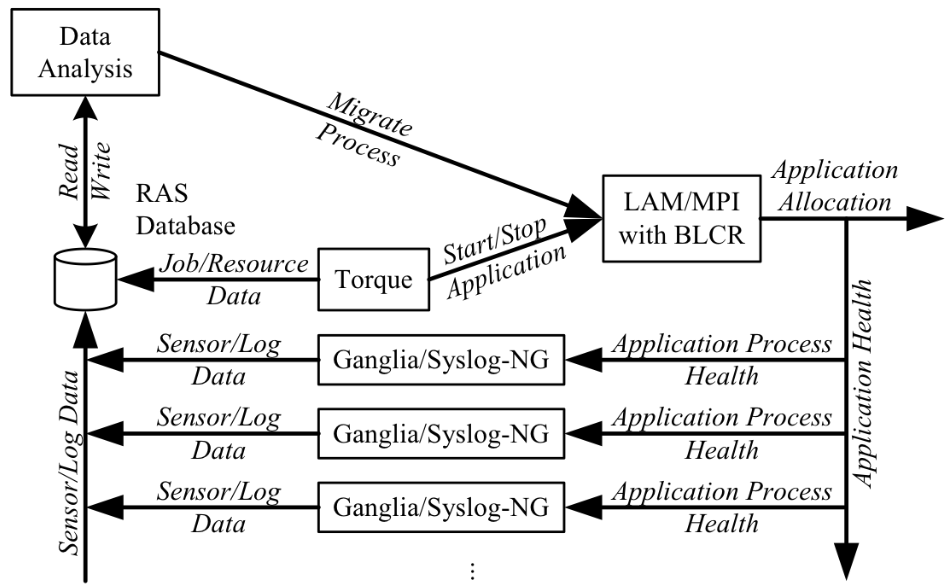
- Single node migration overhead
  - Stop & copy : 0.09–6%
  - Live : 0.08–2.98%
- Single node migration duration
  - Stop & copy : 1.0–1.9 s
  - Live : 2.6–6.5 s
- Application downtime
  - Stop & copy > live
- Node eviction time
  - Stop & copy < live



16-node Linux cluster at NCSU with dual core, dual-processor AMD Opteron and Gigabit Ethernet

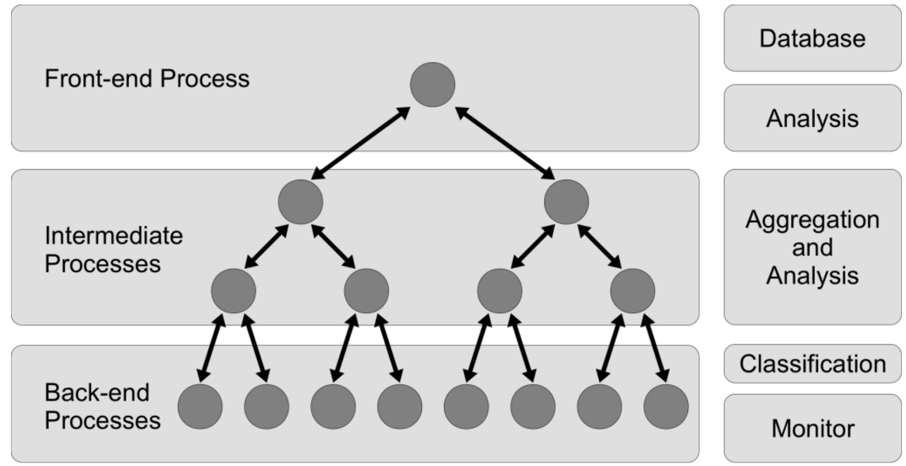
Proactive fault tolerance framework

- Central MySQL database
- Environmental monitoring
  - OpenIPMI and Ganglia
- Event logging and analysis
  - Syslog forwarding
- Job and resource monitoring
  - Torque (epilogue/prologue)
- Migration mechanism
  - Process-level with BLCR



MRNet-based system monitoring

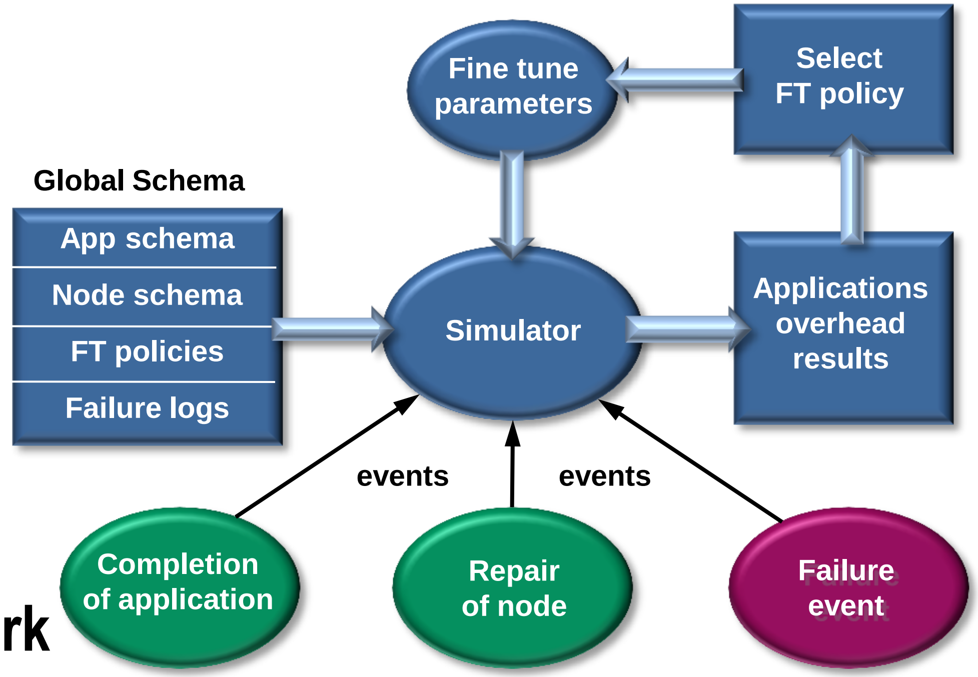
- Aggregation of metrics
- Tree-based overlay network
- Fan-in for metric data
- Fan-out for management
- Classification of data on back-end nodes
- In-flight processing on intermediate nodes
- Collection and storing on front-end node



- 1 MB of data in 4 hours
- ≈250 kB/hour
- ≈2 kb/interval
- ≈56x less than Ganglia

Simulation of fault tolerance policies

- Evaluation of fault tolerance policies
  - Reactive only
  - Proactive only
  - Reactive/proactive combination
- Evaluation of fault tolerance parameters
  - Checkpoint interval
  - Prediction accuracy
- Event-based simulation framework using actual HPC system logs
- Customizable simulated environment
  - Number of active and spare nodes
  - Checkpoint and migration overheads



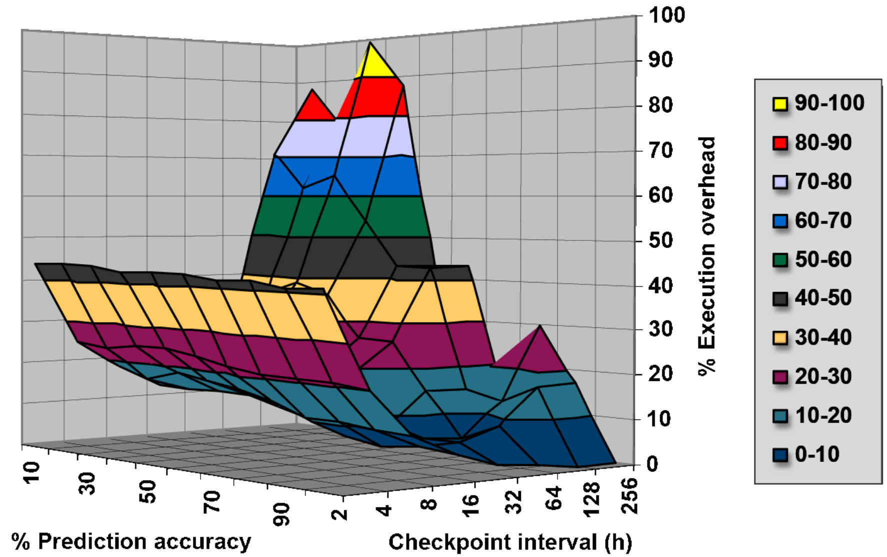
Combining proactive and reactive FT

- Best: Prediction accuracy >60% and checkpoint interval 16−32 h
- Better than only proactive or only reactive
- Results for higher accuracies and very low intervals are worse than only proactive or only reactive

Number of processes	125
Active/Spare nodes	125/12
Checkpoint overhead	50 min
Migration overhead	1 min

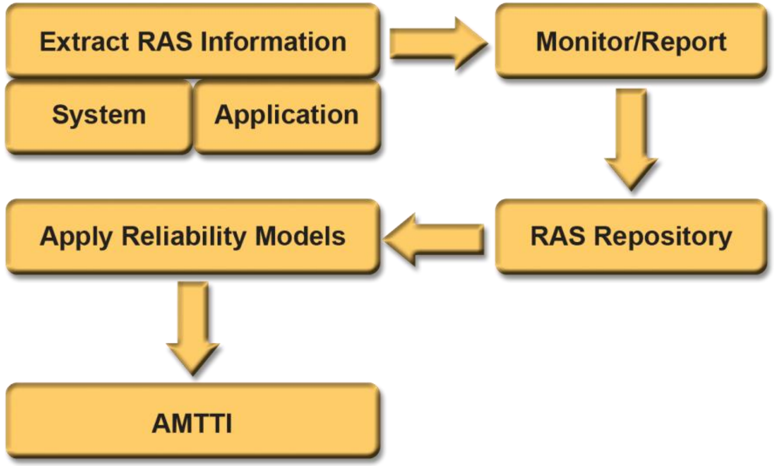
Simulation based on ASCI White system logs (nodes 1−125 and 500−512)

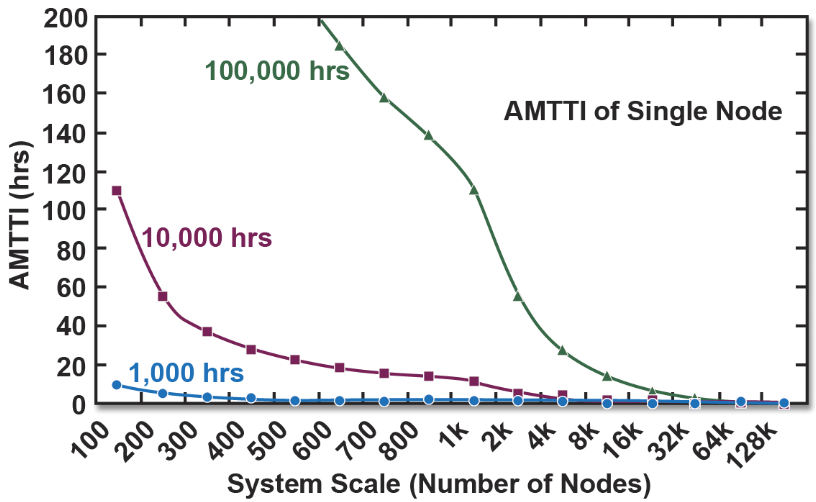
Execution overhead for various checkpoint intervals and different prediction accuracy



Research in reliability modeling

- Type 3 system setup
  - Monitoring of application and system health
  - Recording of application and system health monitoring data
  - Reliability analysis on recorded data
  - Application mean time to interrupt (AMTTI) estimation
- Type 4 system setup
  - Additional recording of application interrupts
  - Reliability analysis on recent and historical data





Acknowledgments

- Investigators at Oak Ridge National Laboratory
  - *Stephen L. Scott (Lead PI)*, Christian Engelmann, Geoffroy Vallée, Thomas Naughton, Anand Tikotekar, George Ostrouchov
- Investigators at Louisiana Tech University
  - *Chokchai (Box) Leangsuksun (Lead PI)*, Nichamon Naksinehaboon, Raja Nassar, Mihaela Paun
- Investigators at North Carolina State University
  - *Frank Mueller (Lead PI)*, Chao Wang, Arun Nagarajan, Jyothish Varma
- Funding sources
  - U.S. Department of Energy, Office of Science, FASTOS 2

Contacts regarding HPC RAS research

Stephen L. Scott

Computer Science Research Group
Computer Science and Mathematics Division
(865) 574-3144
scottsl@ornl.ornl

Christian Engelmann

Computer Science Research Group
Computer Science and Mathematics Division
(865) 574-3132
engelmannc@ornl.ornl

www.fastos.org/ras