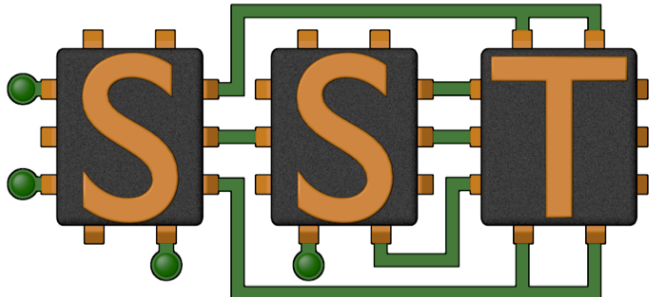
SAND2017-4129C

Exceptional service in the national interest

Sandia National Laboratories



SST Update – April 2017

Scott Hemmert

Scalable Computer Architectures Department

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

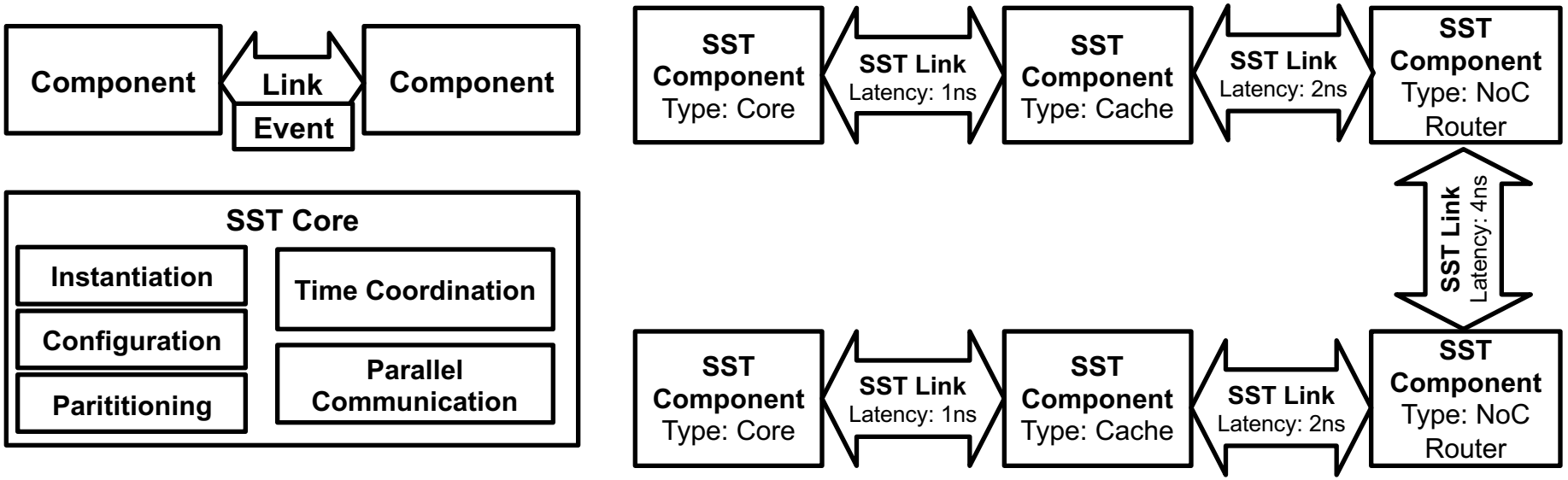
Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

SST in a Nutshell

Sandia National Laboratories

- Parallel Discrete-Event Simulator Framework
  - Flexible framework allows multitude of custom “simulators”
  - Demonstrated scaling to over 512 processors
- Open API
  - Open-source core and models
    - Available at https://github.com/sstsimulator
  - Easily extensible with new models
  - Modular framework
- Comes with many built-in simulation models
  - Processors, Memory, Network
- Time-scale independent core
  - Handles Micro-, Meso-, Macro-scale simulations

SST's discrete-event algorithm



- Simulations are comprised of **components** connected by **links**
- **Components** interact by sending events over **links**
- Each **link** has a *minimum* latency
- **Components** can load **subComponents** and **modules** for additional functionality

SST 6.0 Highlights

Sandia National Laboratories

- Released June 2016
- Move to github from svn
  - Adopted Master/Devel development model using github pull requests
  - Can accept pull requests from outside Sandia
- Split Core / Elements
  - Multiple repositories: core, elements, sst-macro, sqe
  - Configuration Changes (multiple core and element installs)
  - Introduced backward compatibly guarantees for core
- On node threading (using c++11 threads)
- Integrated statistics engine

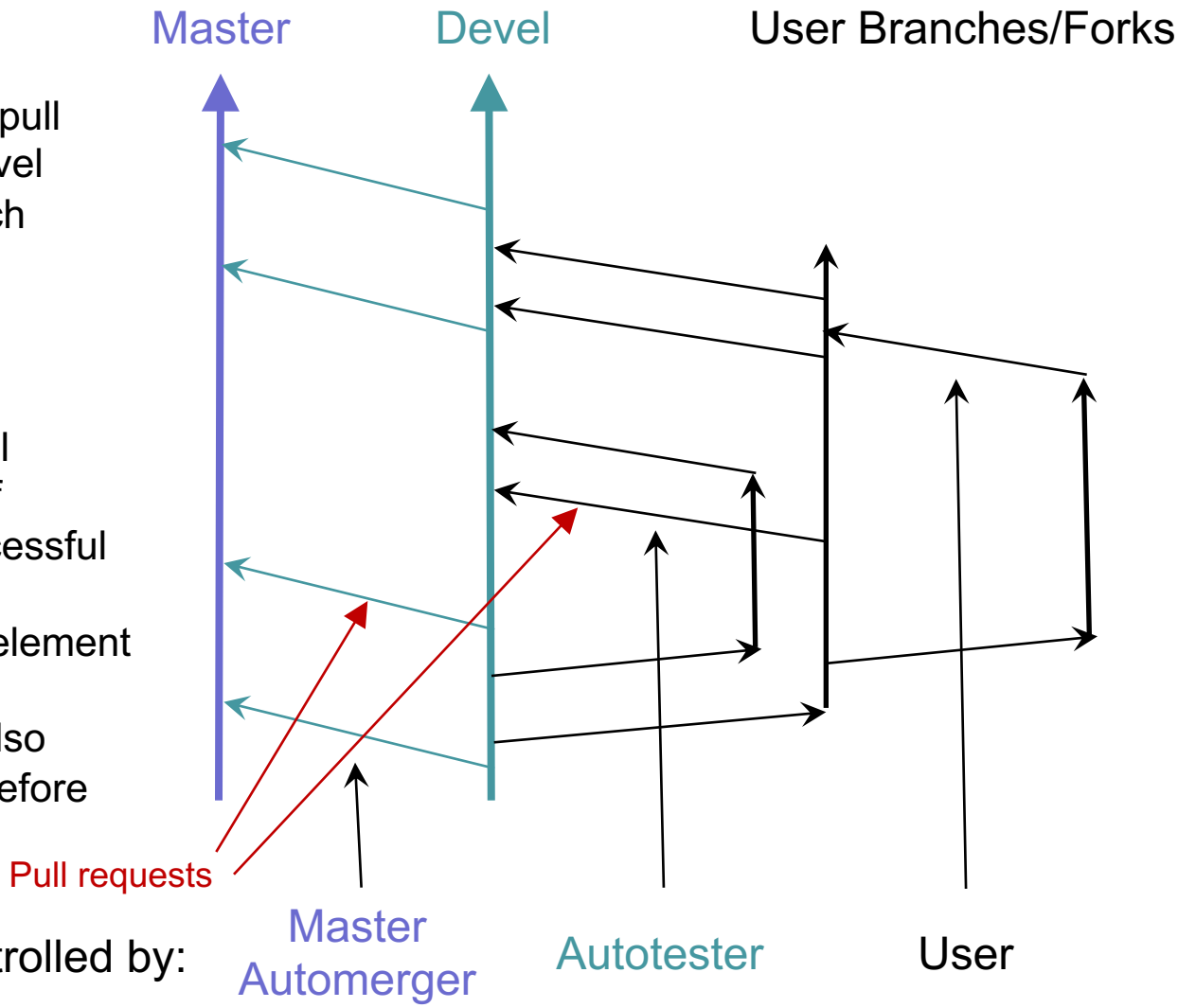
- In use at multiple vendors:
  - AMD, Cray, Intel, IBM, Nvidia

SST Git Structure

Sandia National Laboratories

Master Automerger:
Automatically generates pull requests and merges devel branch into master branch when devel passes all overnight tests.

Autotester:
When user submits a pull request, runs a subset of overnight tests. On successful pass, branch/fork can be merged (automerged in element repository and manually merged in core). Core also requires a code review before merge.



SST 7.0 Highlights

Sandia National Laboratories

- Scheduled for release early May 2017

- New core features:
  - Removes Boost dependency
  - SubComponent enhancements
    - SubComponents can own ports
    - Named SubComponentSlots
  - Embedded ElementLibraryInfo
    - Element information now specified in element definition
    - Simplifies library-level python modules
  - Early HDF5 support for statistics (serial HDF5 only for now)
  - Hybrid parallel execution (stable)

SST 7.0 Highlights, cont

Sandia National Laboratories

- New Elements features
  - Non-volatile memory models
  - Updated timings for DRAM (supports DDR4 up to 3200 MT/s)
  - Scratchpad memory support in memHierarchy
  - TLB modeling (page table walking)
  - Beta support for dynamically changing link bandwidths in network models
  - Early support for memory congestion modeling in network motifs
    - Ember (motif) simulation where one node models application memory traffic in addition to network traffic (currently uses miranda to model on-node memory traffic)

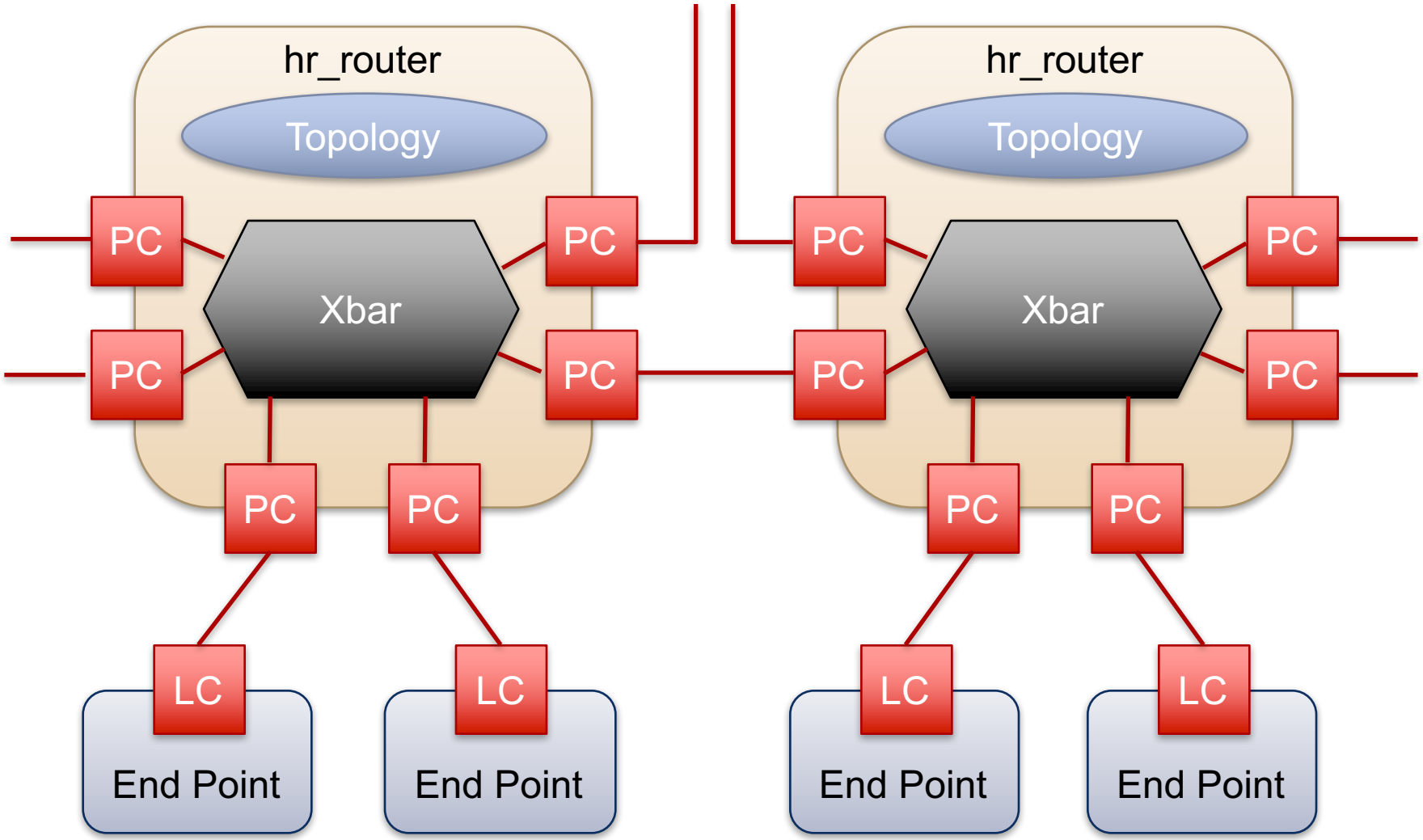
Sandia National Laboratories

EXAMPLES OF NEW SST FEATURES

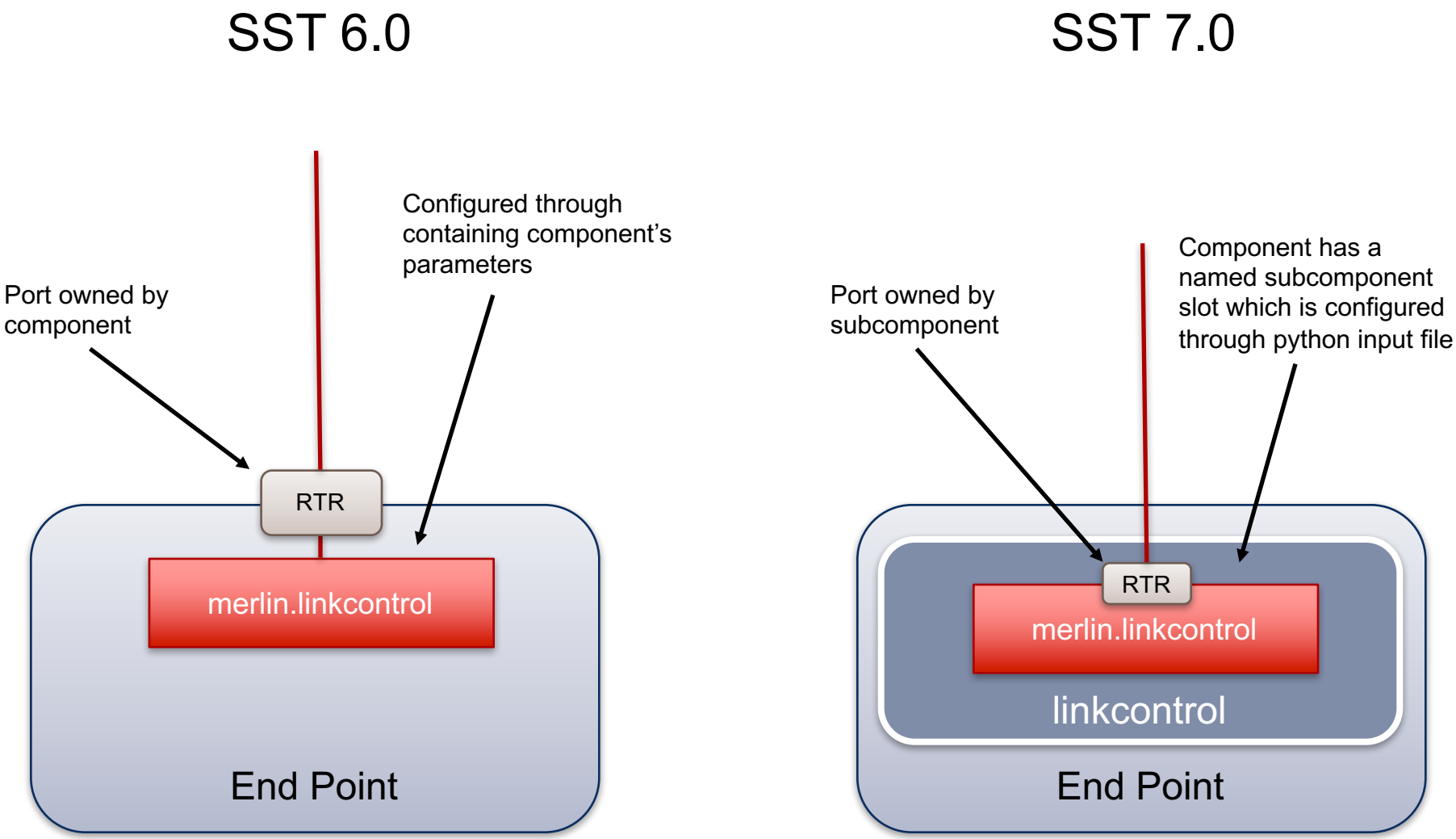
Sandia National Laboratories

Named SubComponent Slots

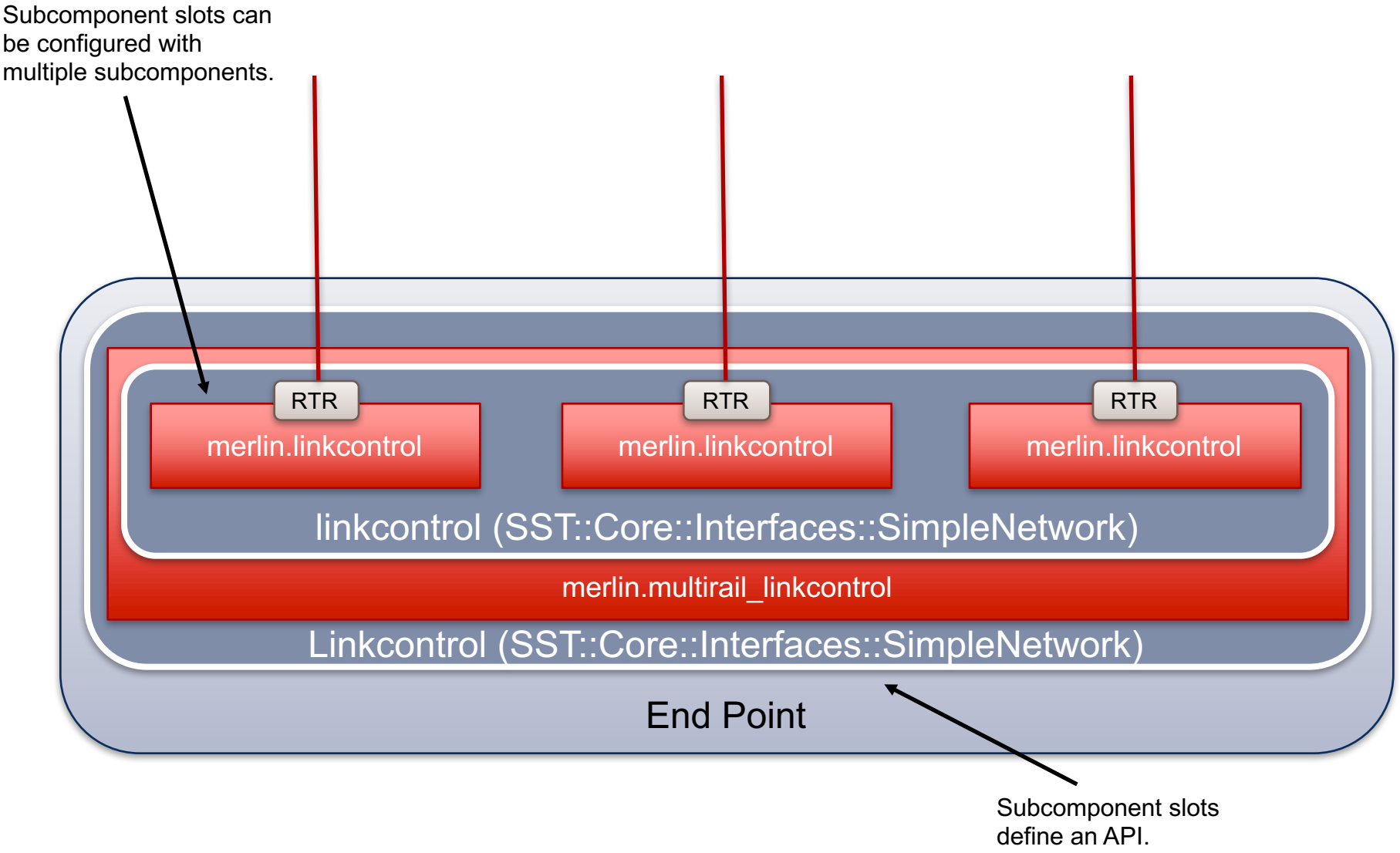
SubComponent Use Example



Named SubComponent Slots



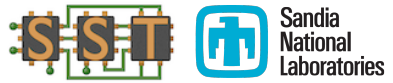
Enhanced Flexibility



Messier:

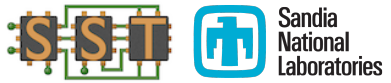
Non-Volatile Memory Model

Why is it important to have NVM model?



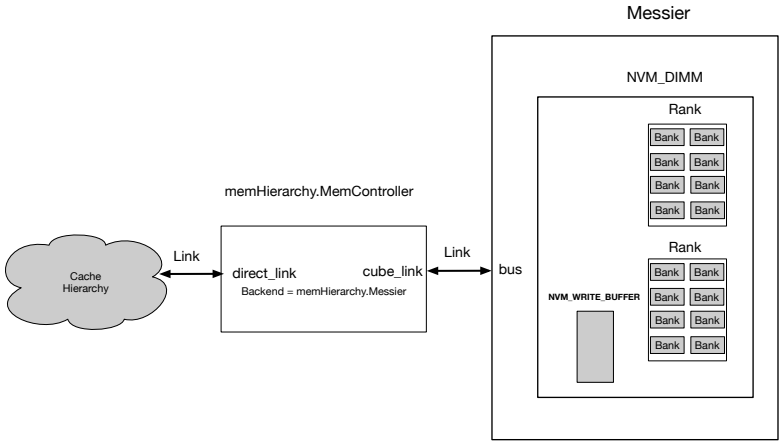
- Evaluate the performance impact of using NVMs as main memory in future HPC system.
- Finding interesting design points and key parameters that affect performance.
- Enables comparisons between different NVM products from different vendors.
- Enables better understanding of bottlenecks for various designs of NVMs.

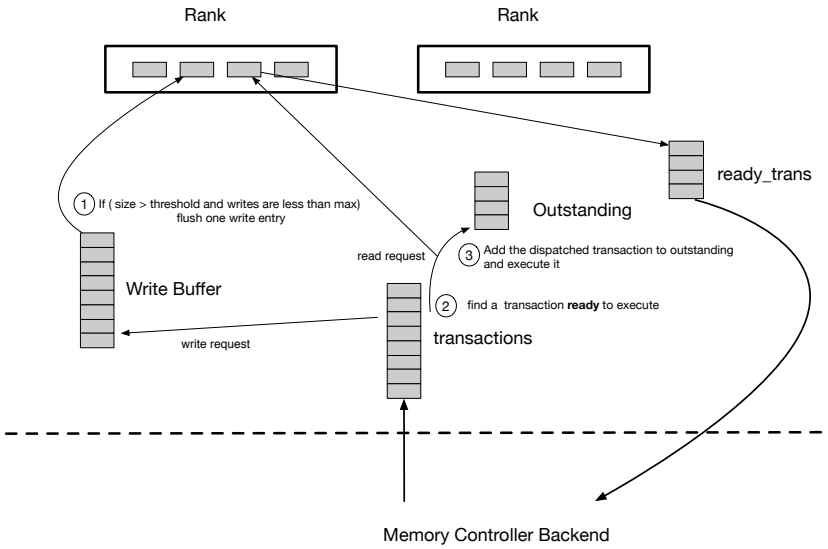
Messier: A detailed NVM-based DIMM Memory Model for SST



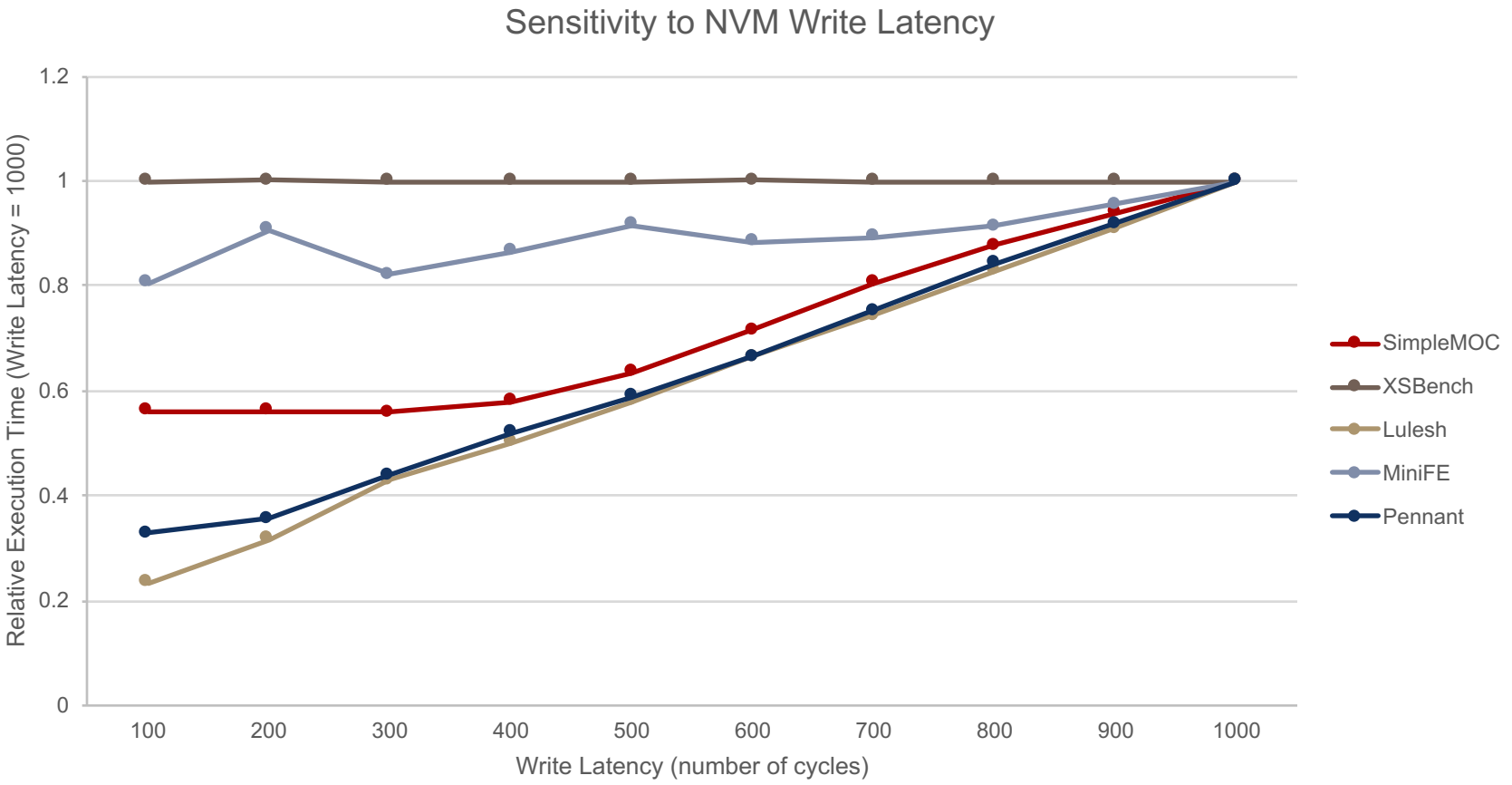
- Highly configurable modular element for SST.
- Models NVM devices and internal high-end DIMM controller.
- Detailed models for contentions, bank conflicts, row buffers, channel contention, read latencies, write latencies, internal buffering, internal caching, scheduling, etc.
- More than 20 parameters that enables detailed modeling of different NVM devices.
- The NVM component can be configured differently for each instance, which enables modeling systems with different NVM devices.

Software Architecture and Model

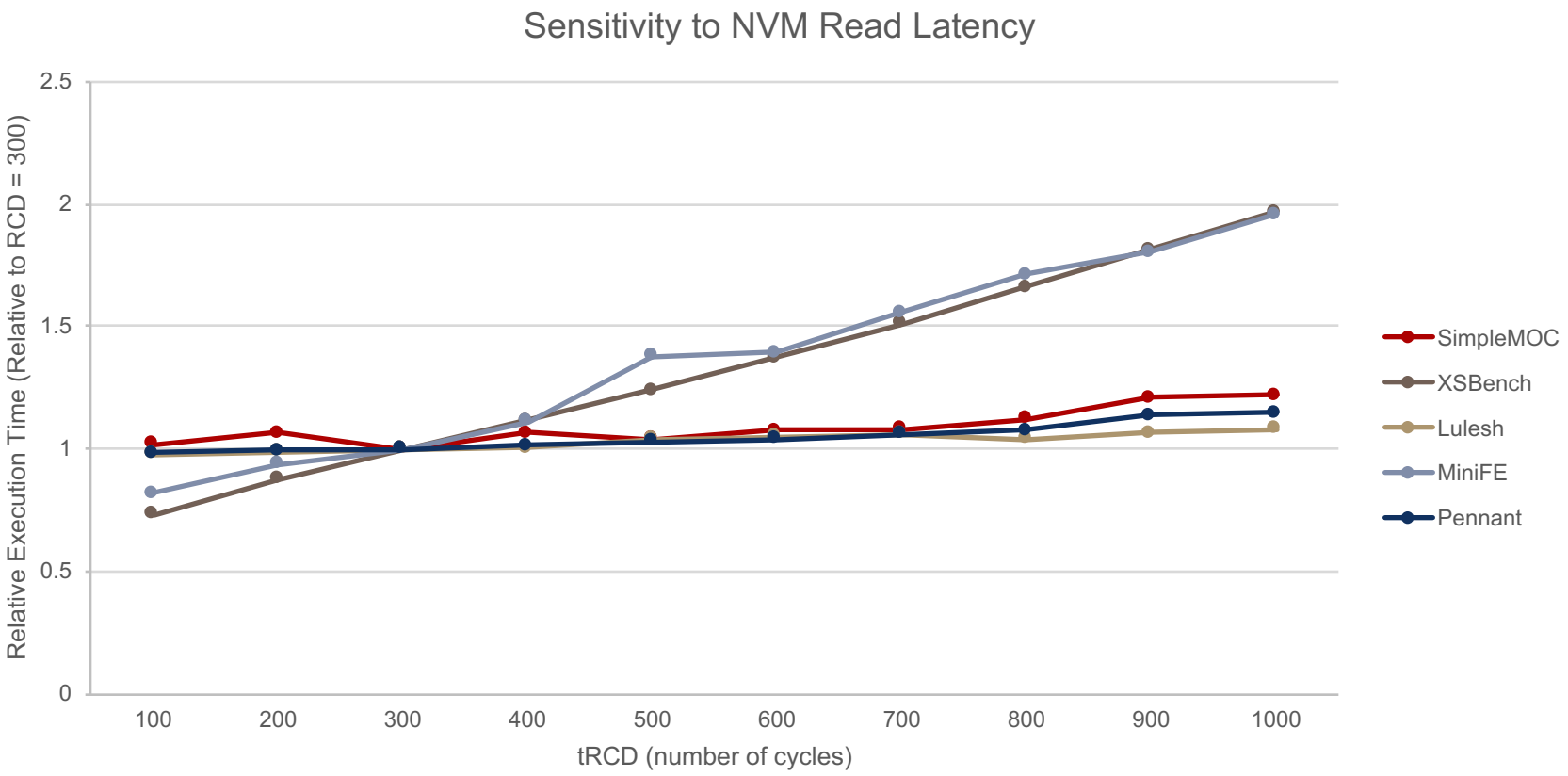




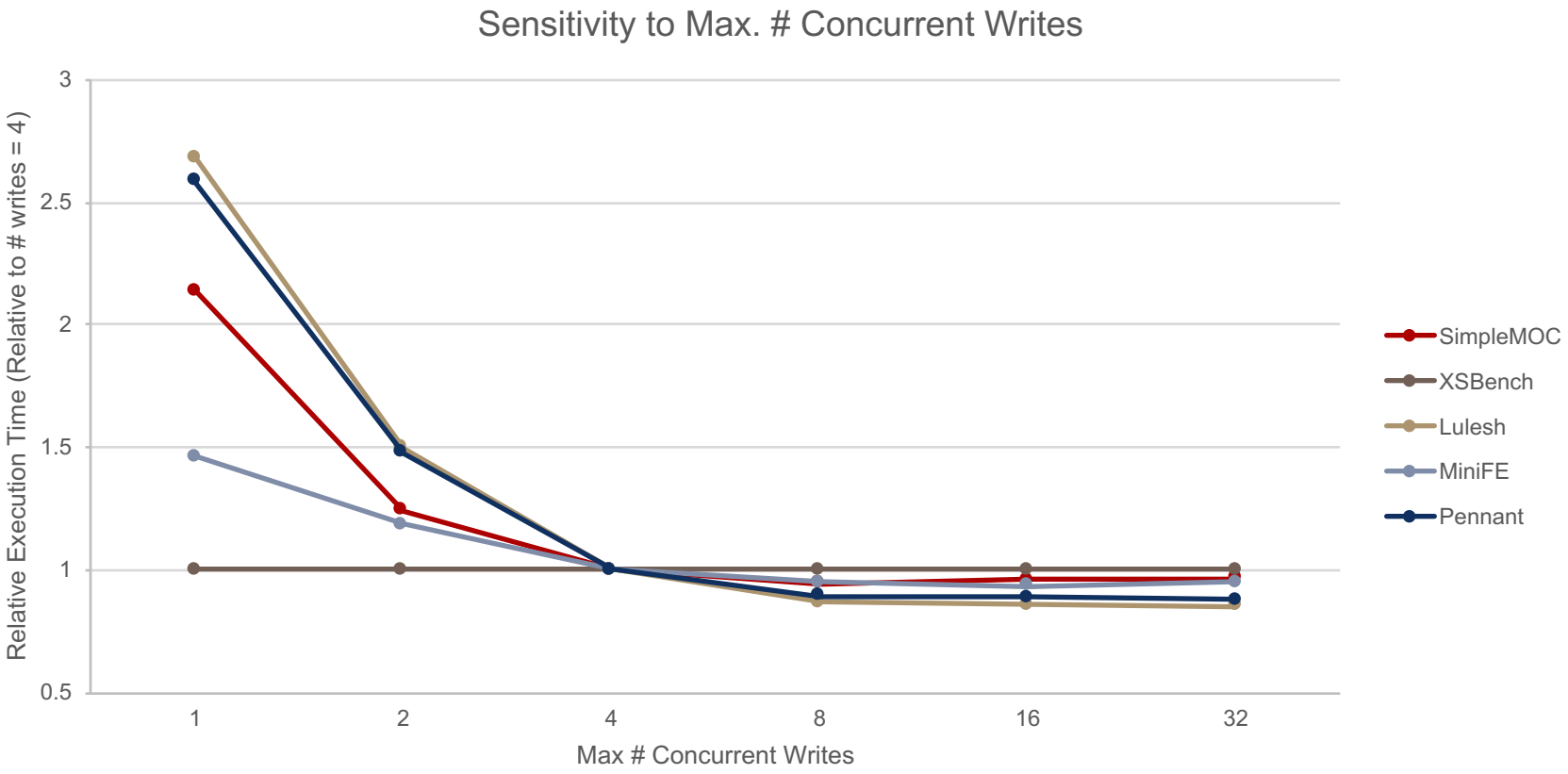
Early Results: Sensitivity to Write Latency



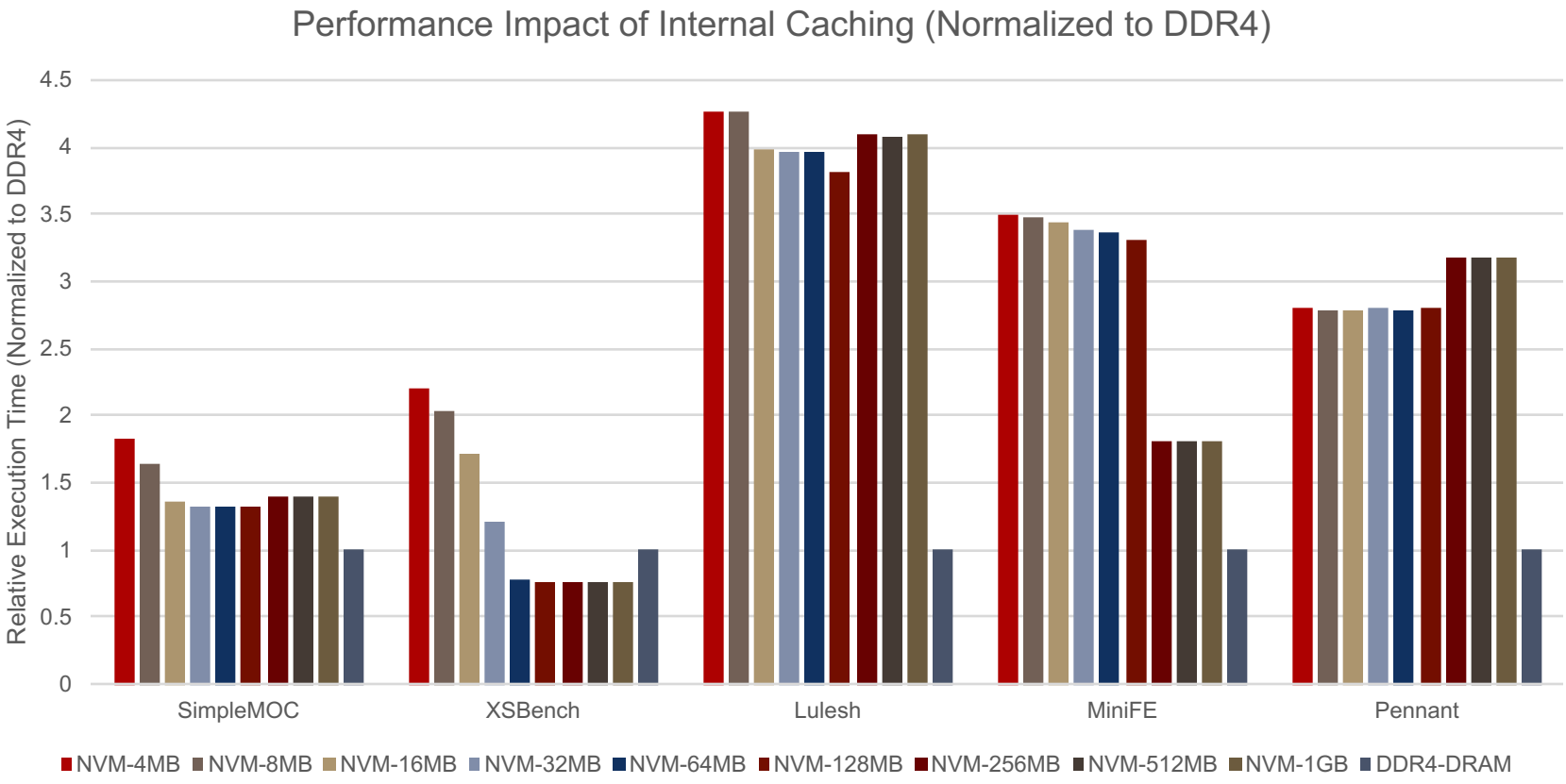
Early Results: Sensitivity to Read Latency



Early Results: Sensitivity to Max Concurrent Writes



Early Results: Impact of Caching



Questions

Sandia National Laboratories