An Exascale Operating System and Runtime Software Research & Development Project

Developing vendor neutral, open-source OS/R software

ANL: **Pete Beckman** (*PI*), Marc Snir (*Chief Scientist*), Pavan Balaji, Rinku Gupta, Kamil Iskra, Franck Cappello, Rajeev Thakur, Kazutomo Yoshii
LLNL: Maya Gokhale, Edgar Leon, Barry Rountree, Martin Schulz, Brian Van Essen
PNNL: Sriram Krishnamoorthy, Roberto Gioiosa
UC: Henry Hoffmann
UIUC: Laxmikant Kale, Eric Bohm, Ramprasad Venkataraman
UO: Allen Malony, Sameer Shende, Kevin Huck
UTK: Jack Dongarra, George Bosilca, Thomas Herault

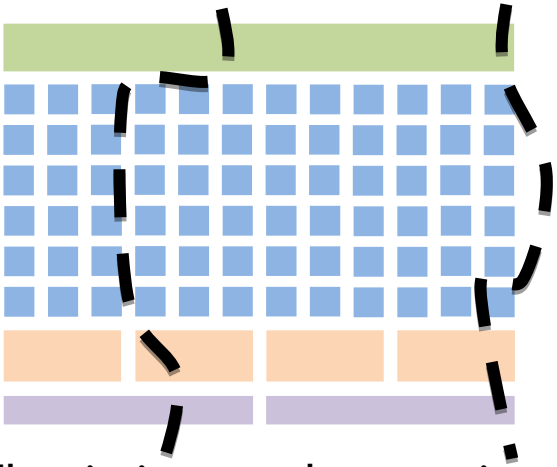
See http://www.argo-osr.org/ for more information

What did Argo build?
New System Software Components:

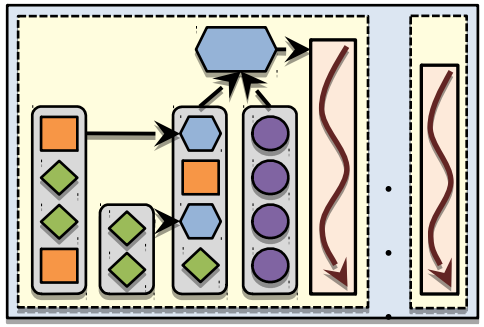
- **Improve application performance**
  - **Argobots** lightweight thread/task layer
    - Improve performance of MPI+OpenMP, math libraries: PLASMA, etc.
    - Support new, more dynamic / load-balanced programming models
  - **Argo Backplane** hierarchical pub/sub backplane
    - Provide APIs to build application resilience with out-of-band events
- **Support new application modes**
  - **Argo Containers** manage cores, memory, and power within a node
    - Improve resource mgmt in support of in-situ analysis & burst buffers, etc.
  - **Argo Backplane:** in-situ data reduction, analysis, and introspection
- **Provide new capabilities to applications**
  - **DIMMAP**: provides new programmer interfaces for NVRAM
  - **Argo Power**: provides APIs; enables machine-learning & adaptation
  - **Argo Global OS/R**: support for new workflows, coupled apps, etc.

Argo Innovations to Address Exascale Gaps

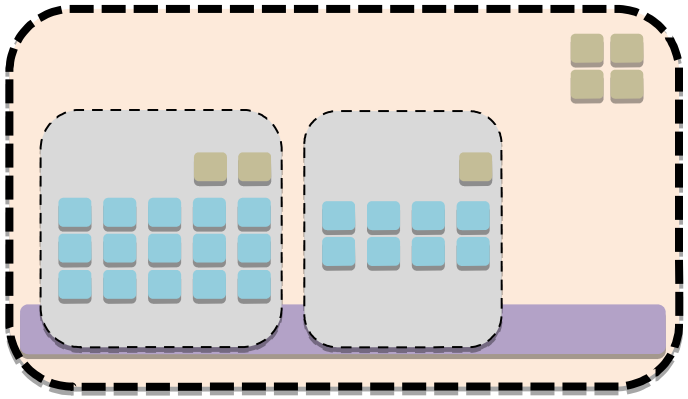
(starting with the key abstractions)



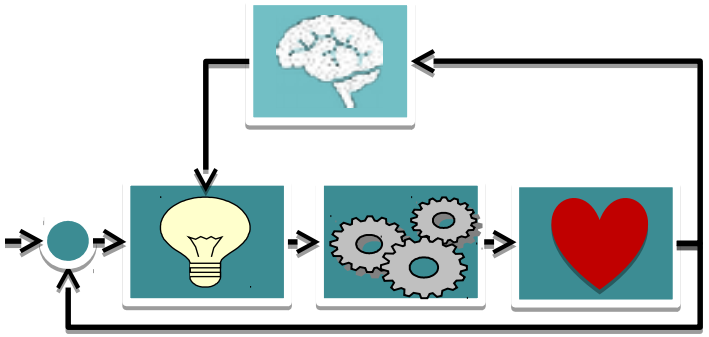
Elastic intranode containers with resource knobs



Lightweight thread/tasks designed for containers, messaging, and memory hierarchy



Hierarchy of *Enclaves* connected via a *Backplane*



Adaptive, learning, integrated control system

Argo Node OS/R Adoption and Deployment

- All code is Open Source
  - Argo Containers: kernel patches, user-space tools (to be officially released)
  - Argo Power: kernel modules, user-space tools, libraries
    `https://github.com/scalability-llnl/`,
    `https://github.com/coolr-hpc/`
  - DI-MMAP: kernel module, user-space tools
    `https://bitbucket.org/vanessen/di-mmap`
  - HPC-Sched: kernel patches (to be officially released)
- Deployment
  - Success working with CESAR to improve Node OS/R performance
  - Some components already in use by applications (DI-MMAP)
  - Argo Power components from LLNL (libmsr) deployed in production
  - We expect to test all components on CORAL systems
- Vendor Collaboration
  - Some Argo Power components to be included in RedHat. PAPI too (NDA Partner)
  - Discussions with IBM on ways to provide Argo tech in CORAL
  - Working with Intel on Aurora mem mgmt improvements (NDA)
  - Working with Cray on Aurora and Argo partitioning and power

Argo Node OS/R Roadmap

- 2016:
  - Centralized Node Resource Manager co-scheduling CPU, memory, network, and power resources
- 2017:
  - Optimized support for communication libraries (optimal memory mappings for put/get, fast thread wakeup)
  - Containers with dynamic power budgets
  - Callbacks to Fault Manager in user space on system fault events
- 2018
  - Integrated, hierarchical memory management including on-package and off-package DRAM, memory on GPU, NVRAM
    - including partitioning, software-based caching and prefetching, callbacks into runtime for latency hiding
- 2019:
  - System call forwarding and optimized support for on-node storage, including draining policies of burst buffers
  - Ensure optimal execution of workflows
    - caching/prefetching of executables, inputs, outputs; result coalescence, etc.

Argo Publications

[1] Leonardo Bautista-Gomez, Ana Gainaru, Swann Perarn… Cappello, Christian Engelmann, and Marc Snir. Reduci… irrespective analysis. In *IEEE International Parallel and …* 2016.

[2] Ming Jiang Brian Van Essen and Maya Gokhale.

[3] Chongxiao Cao, George Bosilca, Thomas Herault, and Ja… dynamic task-based runtime. In *29th IEEE International P… (IPDPS)*, Hyderabad, India, 05-2015 2015. IEEE, IEEE.

[4] Anthony Danalis, George Bosilca, Aurelien Bouteiller, T… abstraction for unhindered parallelism. In *International W… High-Level Frameworks for High Performance Computi…* 2014. IEEE Press, IEEE Press.

[5] Daniel Ellsworth, Tapasya Patki, Swann Perarnau, Sang… nero, Rinku Gupta, Kazutomo Yoshii, Henry Hoffmann… management with argo. In *Twelfth Workshop on High-… PAC)*. IEEE, 2016.

[6] Daniel A Ellsworth, Allen D Malony, Barry Rountree, a… for higher job throughput. In *Proceedings of the Inter… Computing, Networking, Storage and Analysis*, page 80. …

[7] Daniel A Ellsworth, Allen D Malony, Barry Rountree, an… reallocation of limited power in hpc. In *Proceedings of … Performance Parallel and Distributed Computing*, pages …

[8] Anne Farrell and Henry Hoffmann. Meantime: Achievi… approximate computing. In *USENIX ATC*. 2016.

[9] Antonio Filieri, Henry Hoffmann, and Martina Maggio. … with control-theoretical formal guarantees. In *ICSE*, 201…

[10] Roberto Gioiosa, Robert W. Wisniewski, Ravi Murty, a… multi-os hierarchical environments. In *Proceedings of the … Operating Systems for Supercomputers*, ROSS '15, pages …

[11] Blake Haugen, Stephen Richmond, Jakub Kurzak, Chad… execution traces with task dependencies. In *2nd Workshop … Austin*, TX, 11-2015 2015. ACM, ACM.

[12] Anthony Danalis Jack Dongarra Heike McCraw, James … treme scale architectures and dataflow-based programmin… *On Cluster Computing (CLUSTER)*, pages 385–391, 201…

[13] Henry Hoffmann. Jouleguard: energy guarantees for app… *25th Symposium on Operating Systems Principles*, pages …

[14] Henry Hoffmann and Martina Maggio. PCP: A generalized approach to optimizing performance under power constraints through resource management. In *ICAC*, 2014.

[15] Connor Imes and Henry Hoffmann. Minimizing energy under performance constraints on embedded platforms: resource allocation heuristics for homogeneous and single-isa heterogeneous multi-cores. *SIGBED Review*, 11(4).

[16] Connor Imes and Henry Hoffmann. Bard: A unified framework for managing soft timing and power constraints. In *SAMOS*, 2016.

[17] Connor Imes, David H. K. Kim, Martina Maggio, and Henry Hoffmann. Poet: A portable approach to minimizing energy under soft real-time constraints. In *RTAS*, 2015.

[18] Connor Imes, David H. K. Kim, Martina Maggio, and Henry Hoffmann. Portable multicore resource management for applications with performance constraints. In *MCSoC*, 2016.

[19] David H. K. Kim, Ivana Marincic, and Henry Hoffmann. Algorithmic analysis of maximizing performance under a power cap. In *In Submission*, 2016.

[20] Huiwei Lu, Sangmin Seo, and Pavan Balaji. MPI+ULT: Overlapping communication and computation with user-level threads. In *Proceedings of the 2015 IEEE 17th International Conference on High Performance Computing and Communications*, HPCC '15, pages 444–454, August 2015.

[21] Nikita Mishra, Huazhe Zhang, John D. Lafferty, and Henry Hoffmann. A probabilistic graphical model-based approach to minimizing energy under performance constraints. In *ASPLOS*, 2015.

[22] T. Patki, D. Lowenthal, A. Sasidharan, M. Maiterth, B. Rountree, M. Schulz, and B. de Supinski. Practical resource management in power-constrained, high performance computing. In *24th International ACM Symposium on High-Performance Distributed Computing (HPDC 2015)*, 2015.

[23] Swann Perarnau, Rajeev Thakur, Kamil Iskra, Ken Raffenetti, Franck Cappello, Rinku Gupta, Pete Beckman, Marc Snir, Henry Hoffmann, Martin Schulz, and Barry Rountree. Distributed Monitoring and Management of Exascale Systems in the Argo Project. In *15th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS 2015)*, June 2015. (Work-in-progress paper).

[24] Sangmin Seo, Abdelhalim Amer, Pavan Balaji, Cyril Bo… Genet, Thomas Herault, George Bosilca, Pratnek Jindal, … namoorthy, Jonathan Lifflander, Esteban Meneses, Marc … ots: A lightweight threading/tasking framework. Repo… Laboratory, 2016.

[25] ChunYi Su, Edgar A. Leon, Gabriel Loh, David Roberts, … and Bronis R. de Supinski. HpMC: An energy-aware … architectures. In *International Symposium on Memory S…*

[26] Brian Van Essen, Ming Jiang, and Maya Gokhale. Deve… ment within a memory management runtime for data-in… *Workshop*, San Diego, CA, March 2015.

[27] Wei Wu, Aurelien Bouteiller, George Bosilca, Mathie… dag scheduling for hybrid distributed systems. In 29… *Processing Symposium (IPDPS)*, Hyderabad, India, 05…

[28] Huazhe Zhang and Henry Hoffmann. A quantitative evaluation of the RAPL power control system. In *Feedback Computing*, 2015.

[29] Huazhe Zhang and Henry Hoffmann. Maximizing performance under a power cap: A comparison of hardware, software, and hybrid techniques. In *ASPLOS*, 2016.

[30] Yanqi Zhou, Henry Hoffmann, and David Wentzlaff. Cash: Supporting iaas customers with a sub-core configurable architecture. In *ISCA*, 2016.

[31] Judicael A. Zounmevo, Kamil Iskra, Kazutomo Yoshii, Roberto Gioiosa, Brian C. Van Essen, Maya B. Gokhale, and Edgar A. Leon. A single-kernel approach to OS specialization and code resource partitioning for exascale computing. 11th USENIX Symposium on Operating Systems Design and Implementation (OSDI '14), October 2014. (Poster).

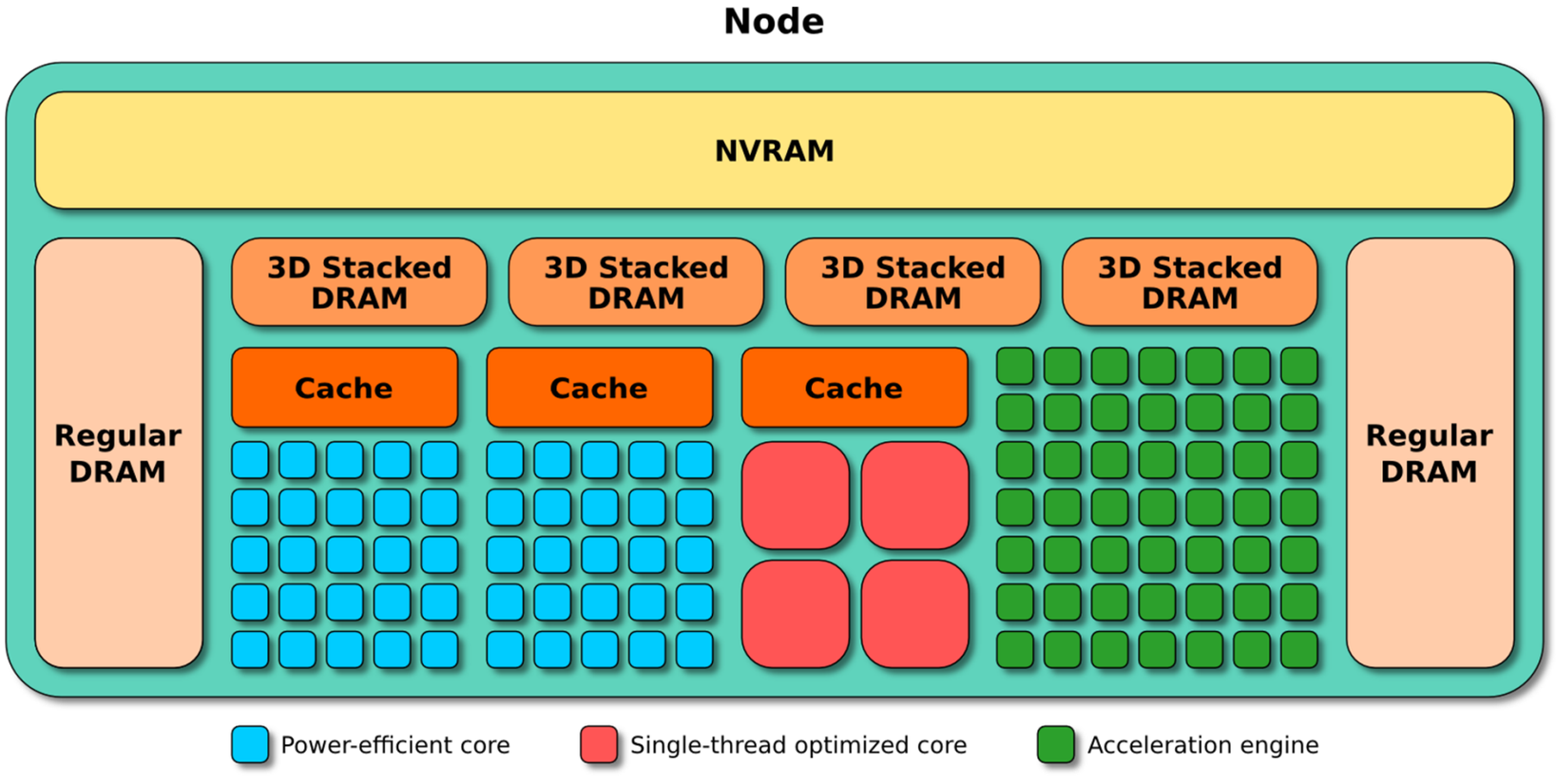
[32] Judicael A. Zounmevo, Swann Perarnau, Kamil Iskra, Kazutomo Yoshii, Roberto Gioiosa, Brian C. Van Essen, Maya B. Gokhale, and Edgar A. Leon. A container-based approach to OS specialization for exascale computing. In *1st International Workshop on Container Technologies and Container Clouds (WoC '15), held in conjunction with IEEE International Conference on Cloud Engineering (IC2E '15)*, Tempe, AZ, March 2015.


Node OS/R

Node OS/R Team

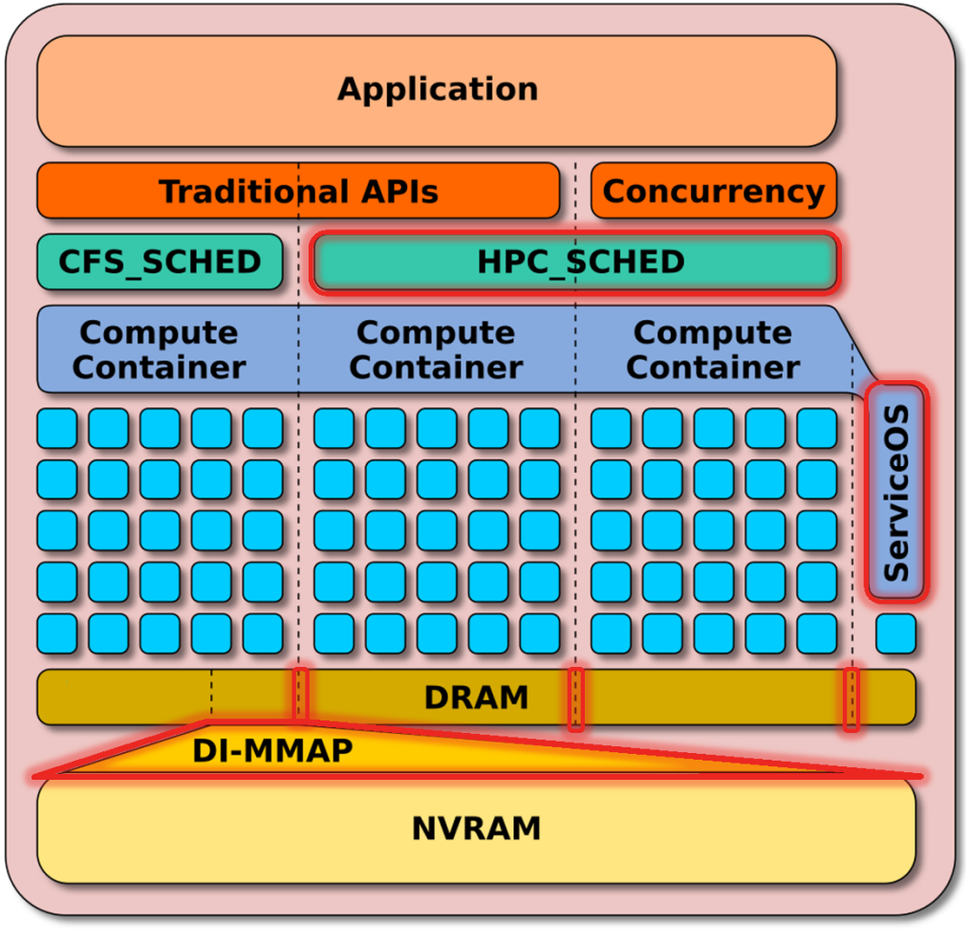
- Argonne: Kamil Iskra, Swann Perarnau, Kazutomo Yoshii, Judicael Zounmevo
- Livermore: Maya Gokhale, Brian Van Essen, Edgar Leon
- Pacific Northwest: Roberto Gioiosa

Exascale Node Architecture



- Heterogeneous compute resources
- Deep memory hierarchy
- Power management
- Fault management

Node OS/R Architecture



- Single kernel image, Linux-based
- Partition hardware resources using containers
  - ease of management, potential scalability improvements
- HPC-specific improvements
  - NVRAM management
  - DRAM management
  - task scheduler

Our key contributions

Highlights

Containers

- Performance isolation of system services, application components
- Co-scheduling of system resources
- Dynamic

FGMN

- Manage physical memory at sub-NUMA granularity
- Greater flexibility
- Reduces memory fragmentation

DI-MMAP

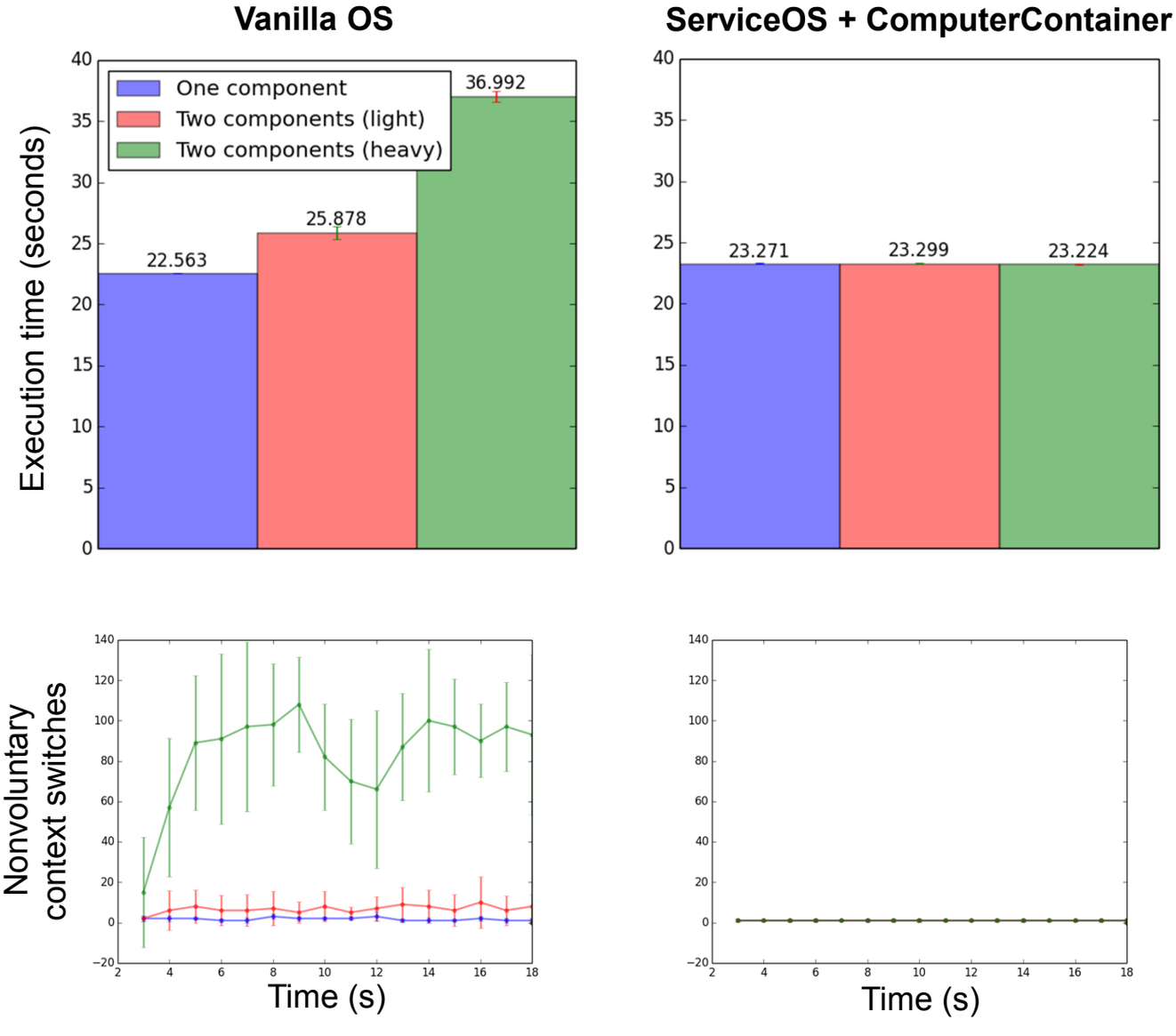
- Integrate NVRAM into HPC node memory hierarchy
- DRAM page cache optimized for HPC applications
- Enable scalable out-of-core data-intensive workloads

HPC_SCHED

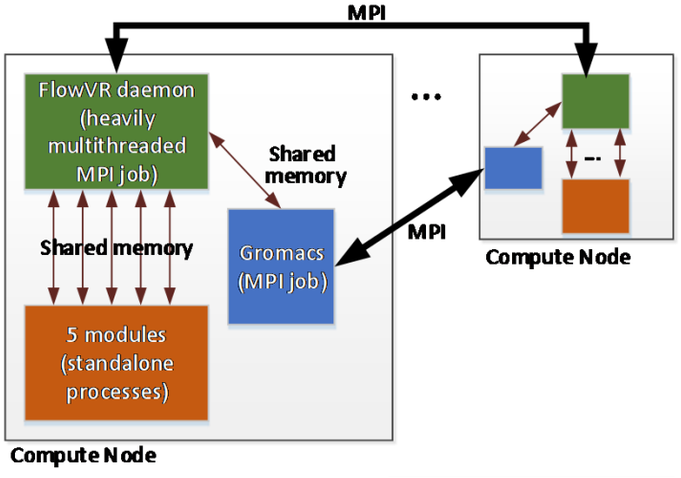
- Reduce task preemption and migration
- Reduce OS jitter, increase process responsiveness

Isolation with Containers

- First component:
  - HSBench (46 threads)
- Second component:
  - CPU load
  - light: 4 threads
  - heavy: 40 threads
- Hardware
  - dual 12-core Haswell (48 hw threads)



Isolation with Containers



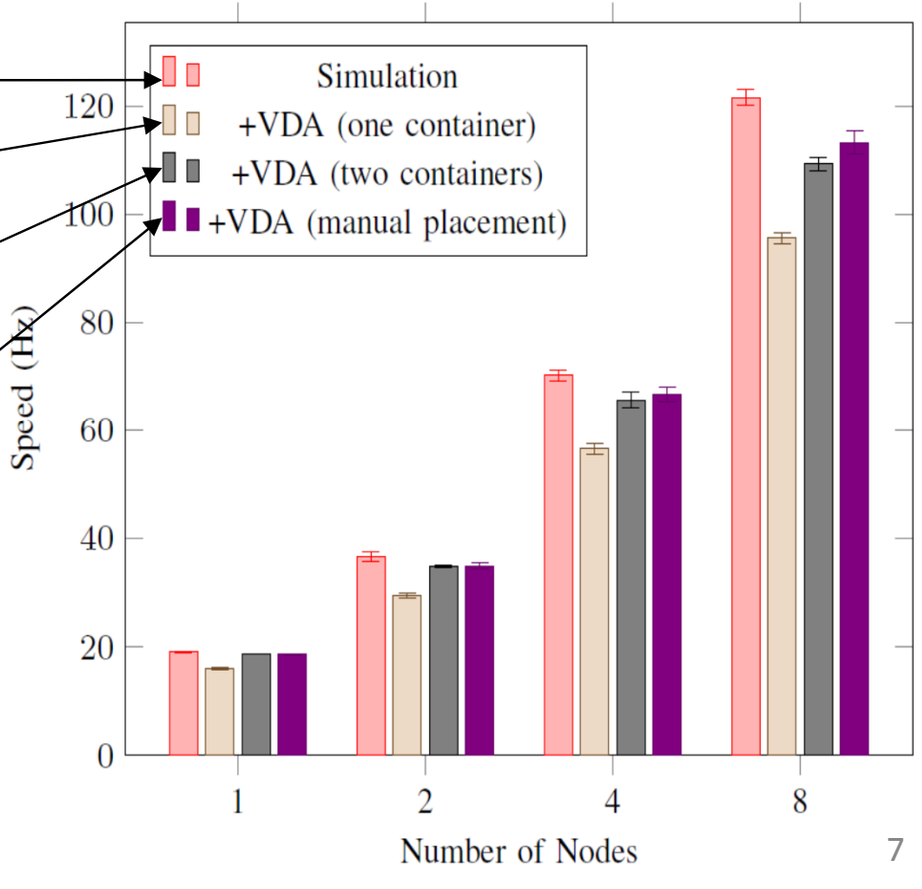
- Simulation + 5-stage VDA pipeline
- Containers improve performance isolation without hindering communication between components

Gromacs only

Gromacs + VDA in 1 container
Separate ServiceOS

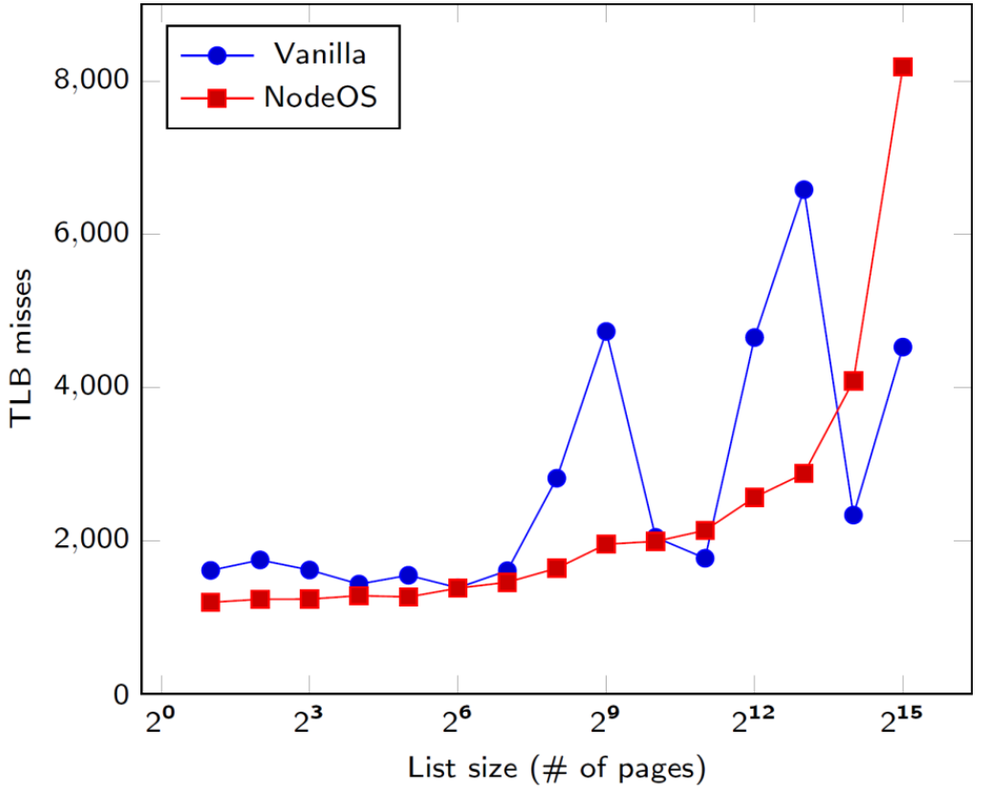
Gromacs in 1 container
Modules in 1 container
FlowVR daemon roams free
Separate ServiceOS

Gromacs + VDA
No NodeOS config.
Manual process placement



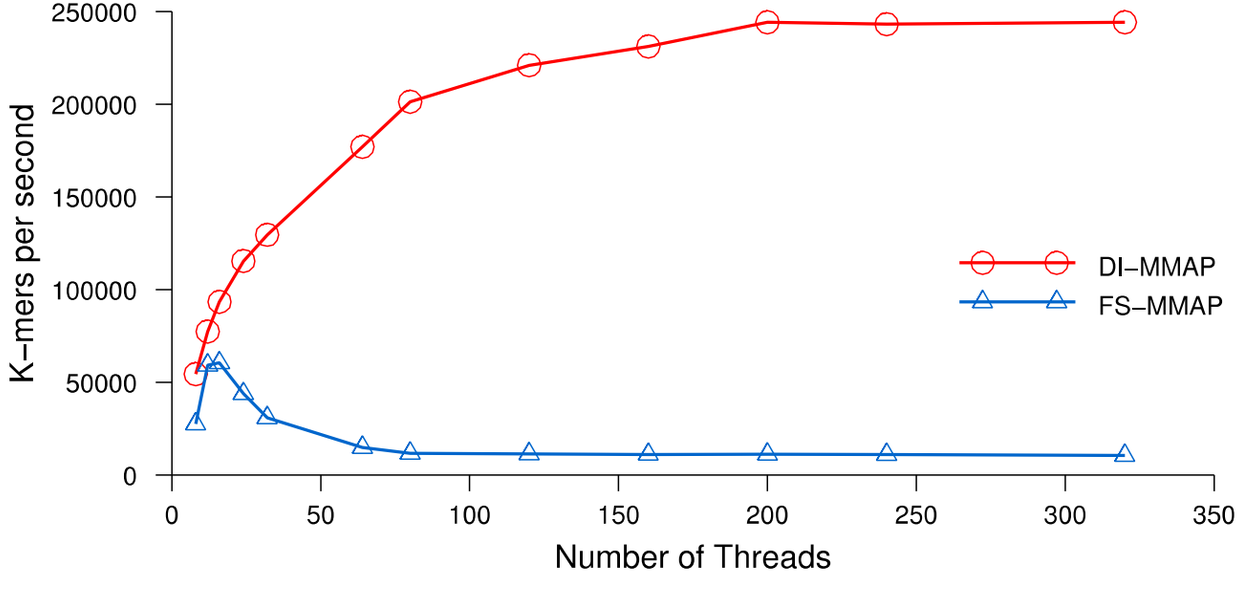
Performance Benefits of FGMN

- Partitioning memory improves isolation, reduces physical memory fragmentation
  - Graph shows how a memory stress workload can affect the performance of latency-sensitive workload (random walk) by preventing the vanilla Linux kernel from using Transparent Huge Pages due to memory fragmentation



DI-MMAP – Impact

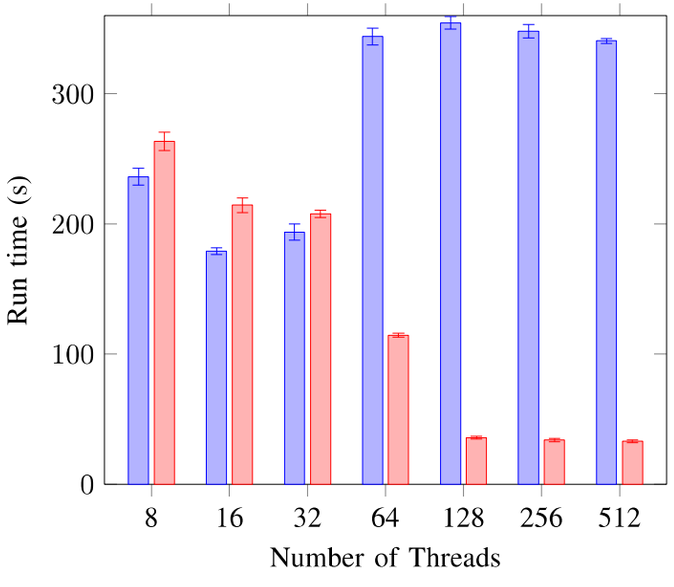
- Significant performance improvements over Linux mmap with out-of-core data intensive workloads
  - 3–4x on Livermore Metagenomics Analysis Toolkit
  - 2.4x on Graph500 Scale 40
- Transparent support eases portability for many existing HPC applications
- **Future:**
  - Multiple caching policies customized to HPC applications
  - Application-tailored prefetch



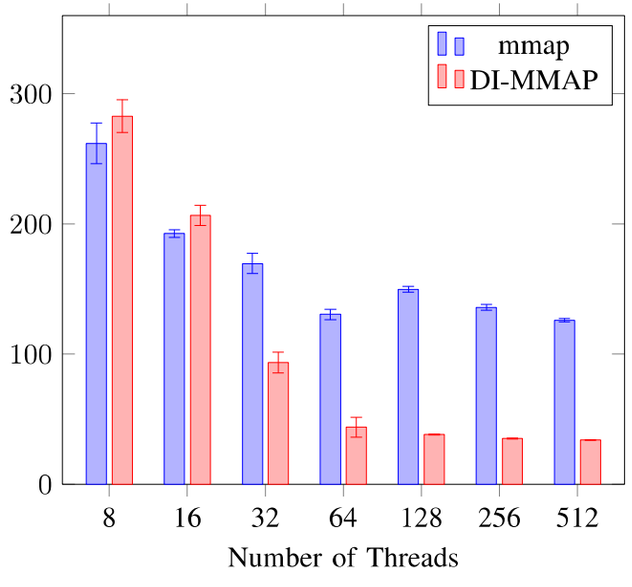
Bioinformatics database query: DI-MMAP vs linux mmap

Integration tests: Affinity vs Containers

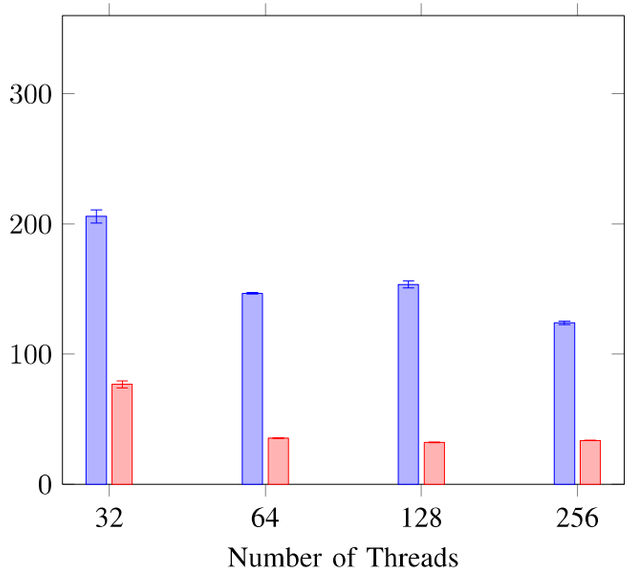
LRIOT run times



(a) Application and buffers unrestricted.



(b) Applications limited to a single socket with `taskset`.



(c) Application and buffers limited to a single socket with containers.

- Containers provide convenient isolation mechanism
  - Improves DI-MMAP performance by up to 20% over uncontained DI-MMAP
- For highly concurrent, threaded applications with read-heavy I/O DI-MMAP is substantially faster than regular Linux mmap

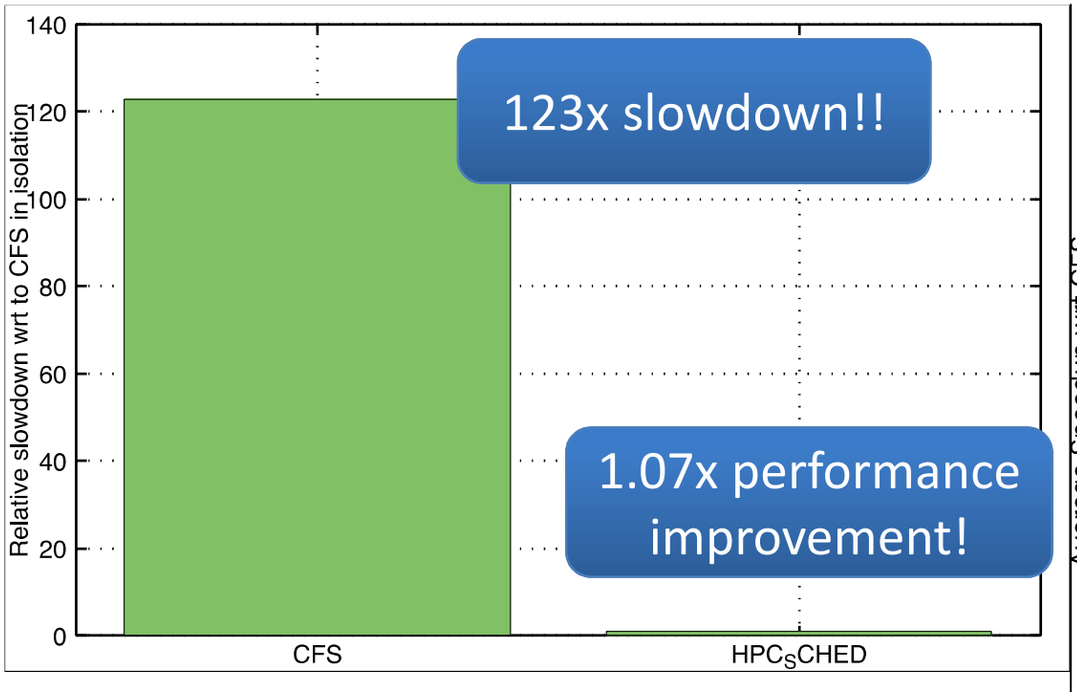
Argo NodeOS Scheduler: Impact

- HPC_SCHED and the noise-canceling w
  within a conta
  - Reduce system noise within a container
  - Allow Argobots runtime to hand-off cores to thr
  - Reduce task preemption and migration
  - Increase responsiveness to events (e.g., NVRAM read requests)

No benefits from CPU affinity

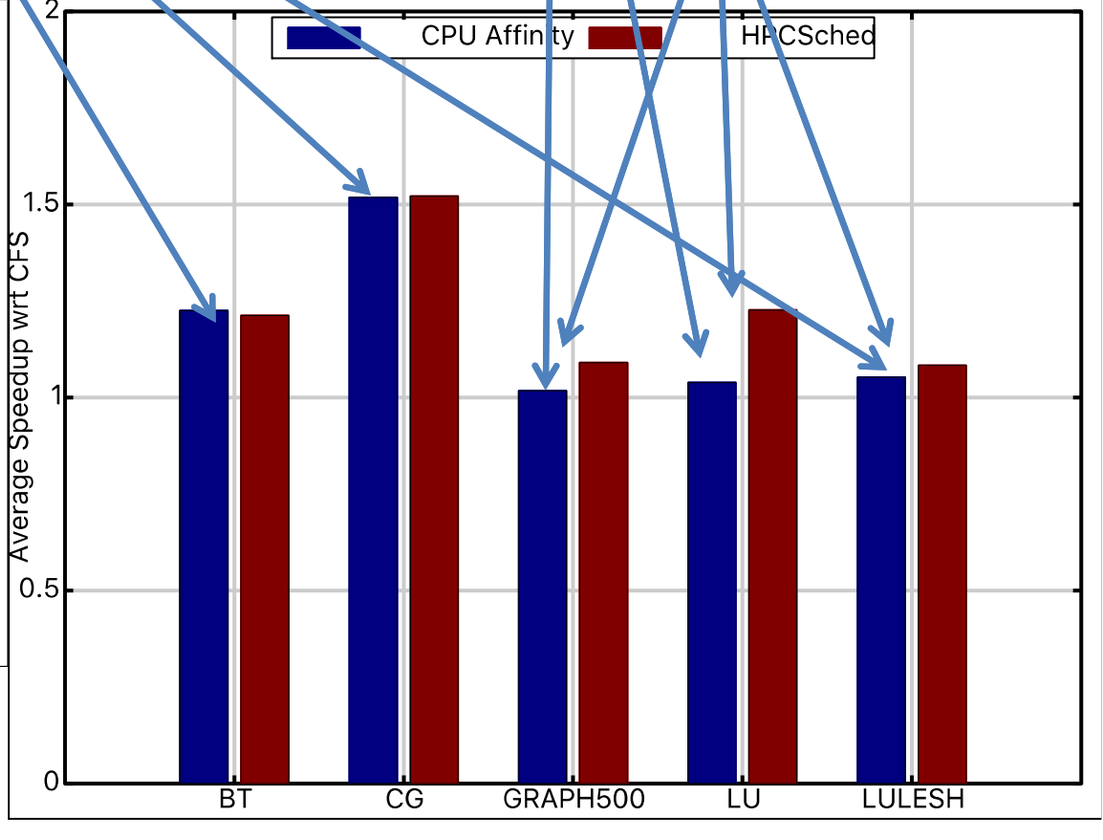
Benefits from CPU affinity

Benefits from CPU affinity **and** responsiveness

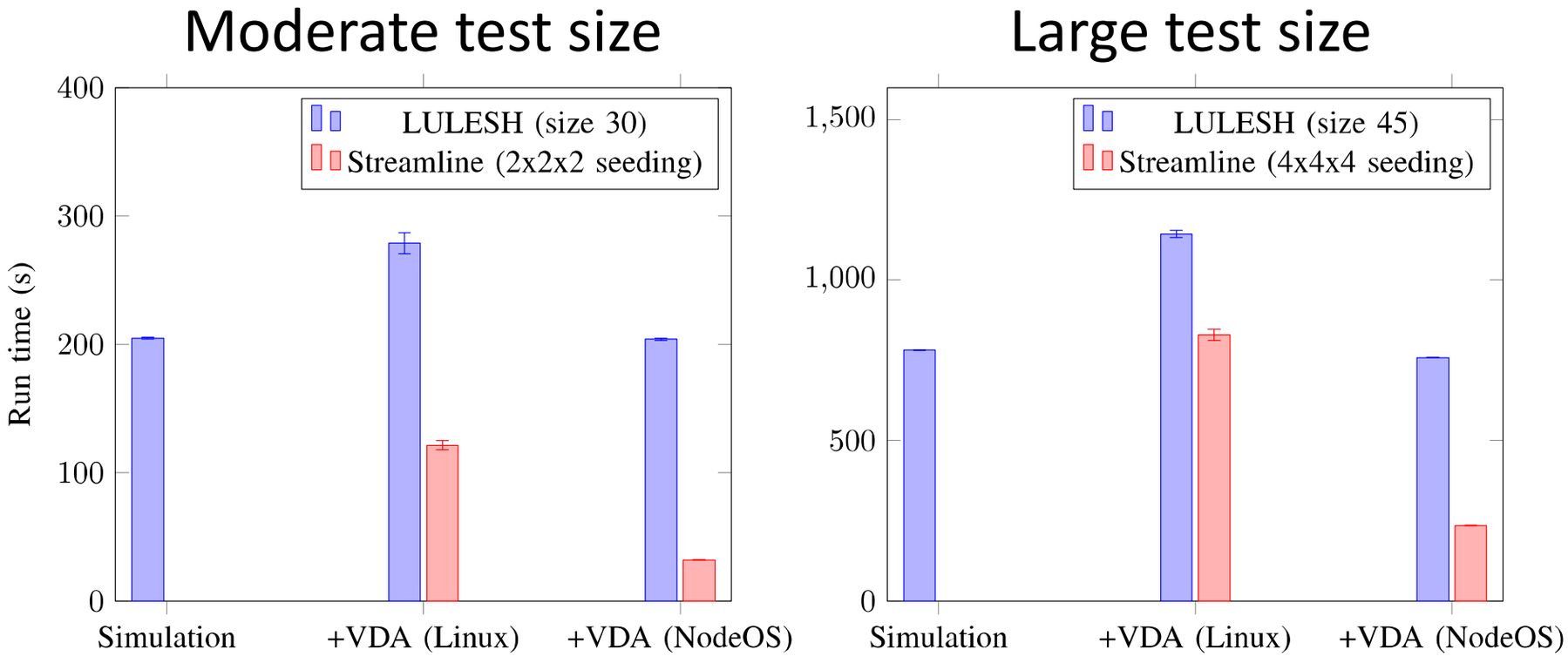


123x slowdown!!

1.07x performance improvement!



Simulation + VDA in-situ experiment



- Containers + DI-MMAP improves overall performance for in-situ experiment
  - 36% to 46% degradation in performance when VDA interferes with LULESH
  - Containers effectively isolate VDA from simulation and improve VDA performance by ~3.5x

Key Results

Containers

- Effective management of node resources in complex scenarios
- Transparent to both applications and system services
- Performance isolation comparable to manual placement

DI-MMAP

- Memory-mapped file I/O to Flash work well with over-decomposed, asynchronous concurrency
- Introspection enables application adaptation and tuning
- Non-native page size support

FGMN

- Effective management of physical memory
- Performance improvements through reduction of memory fragmentation

HPC_SCHED

- Improvements extend beyond process affinity
- Massive improvements over regular scheduler in some cases of oversubscription

Next Steps

- Integrated Node Resource Manager
- Integration with Power management
- Partitioning of NIC
- Support for holding data structures of tasking libraries in NVRAM
- Migration of custom DI-MMAP functionality to user level
- Deep memory management