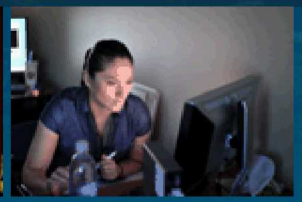
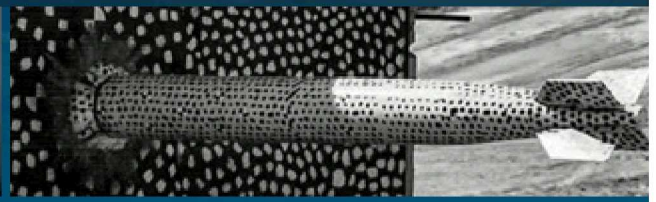
This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

SAND2018-12905C

Vanguard NNSA HQ Meeting

PRESENTED BY

Kenneth Alvin, James H. Laros III

mission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.


ASTRA
"Per aspera ad astra"
Sandia National Laboratories
NNSA
Advanced Simulation & Computing
Hewlett Packard Enterprise
ENERGY
VANGUARD

Vanguard Program: Advanced Technology Prototype Systems

- **Prove viability of advanced technologies for NNSA integrated codes, at scale**
- Expand the HPC-ecosystem by developing emerging yet-to-be proven technologies
  - Is technology viable for future ATS/CTS platforms supporting ASC mission?
  - Increase technology AND integrator choices
- Buy down risk and increase technology and vendor choices for future NNSA production platforms
  - Ability to accept higher risk allows for more/faster technology advancement
  - Lowers/eliminates mission risk and significantly reduces investment
- Jointly address hardware and software technologies
  - More from Kevin Pedretti on ATSE
- First Prototype platform targeting Arm Architecture

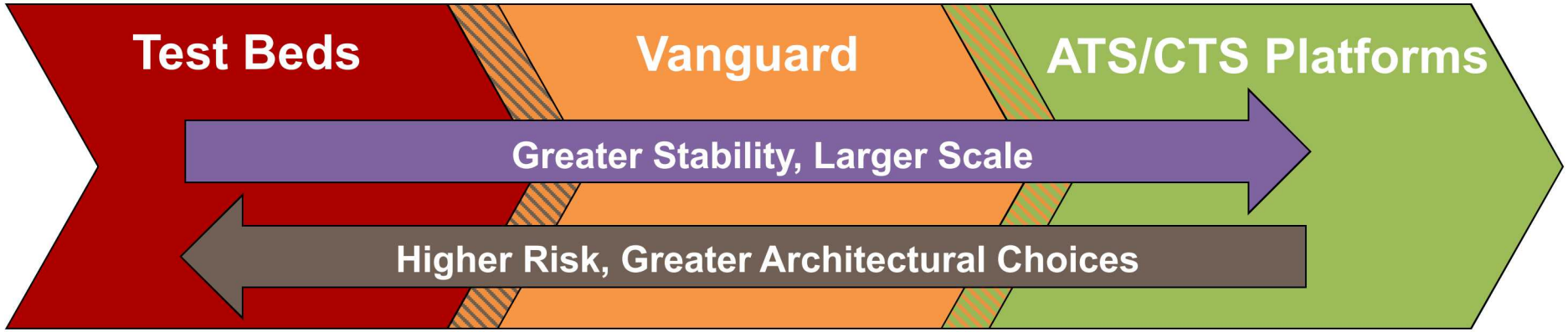
Success achieved through Tri-Lab involvement and collaboration

What is ATSE?

- Advanced Tri-lab Software Environment
  - Sandia leading development with input from Tri-lab Arm team
  - Will be the user programming environment for Vanguard-Astra
  - Version 1.0 targets including the software components needed for Milestone 1
- Lasting value
  - Documented specification of:
    - Software components needed for ASC production applications
    - How they are configured (i.e., what features and capabilities are enabled) and interact
    - User interfaces and conventions
  - Reference implementation:
    - Deployable on multiple ASC systems with common look and feel
    - Tested against real ASC workloads (open networks and classified)
    - Something to point vendors at in procurements

ATSE is an integrated software environment for ASC workloads

Where Vanguard Fits



Test Beds

- Small testbeds (~10-100 nodes)
- Breadth of architectures Key
- Brave users

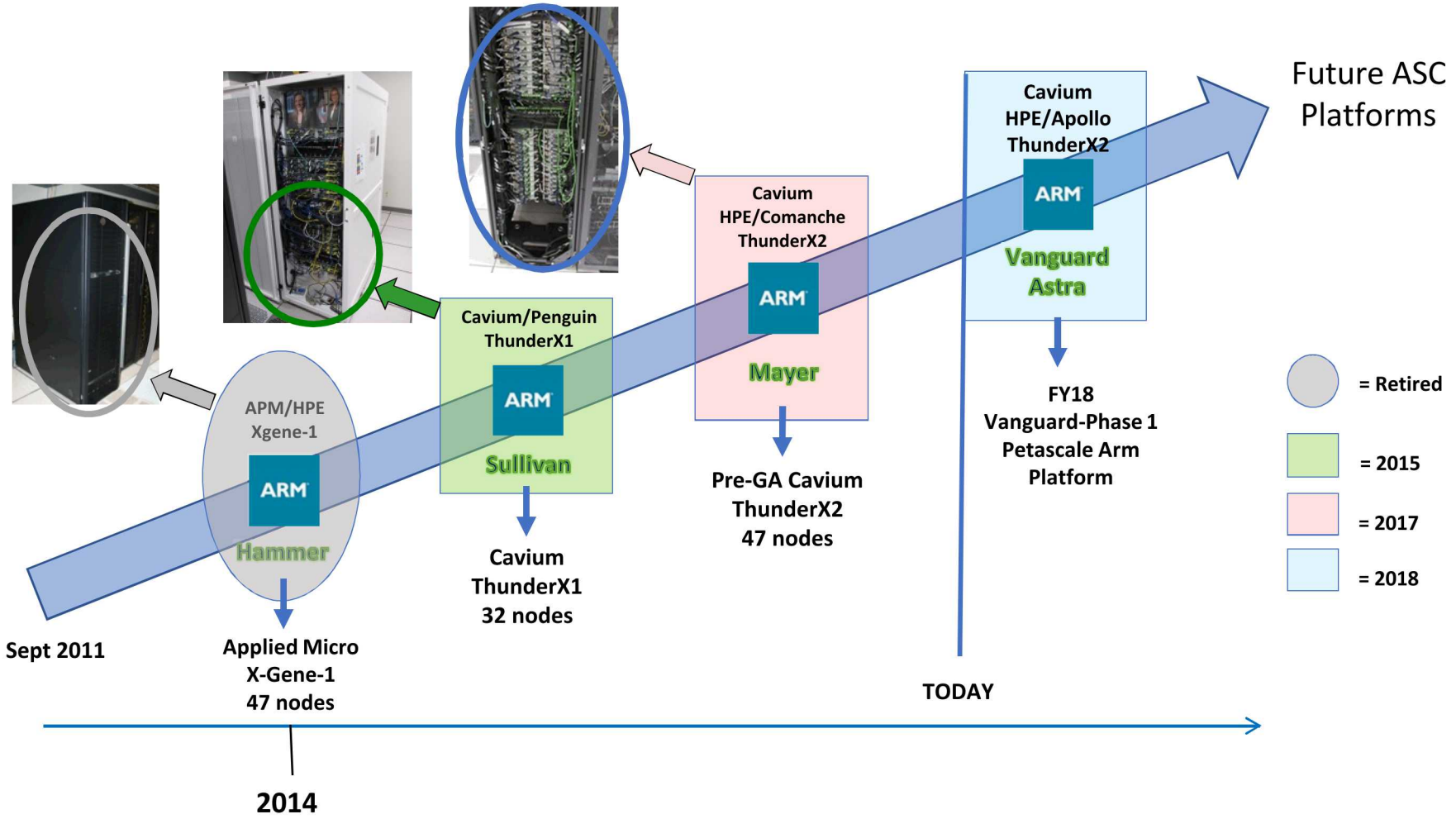
Vanguard

- Larger-scale experimental systems
- Focused efforts to mature new technologies
- Broader user-base
- Not Production
- **Tri-lab resource but not for ATCC runs**

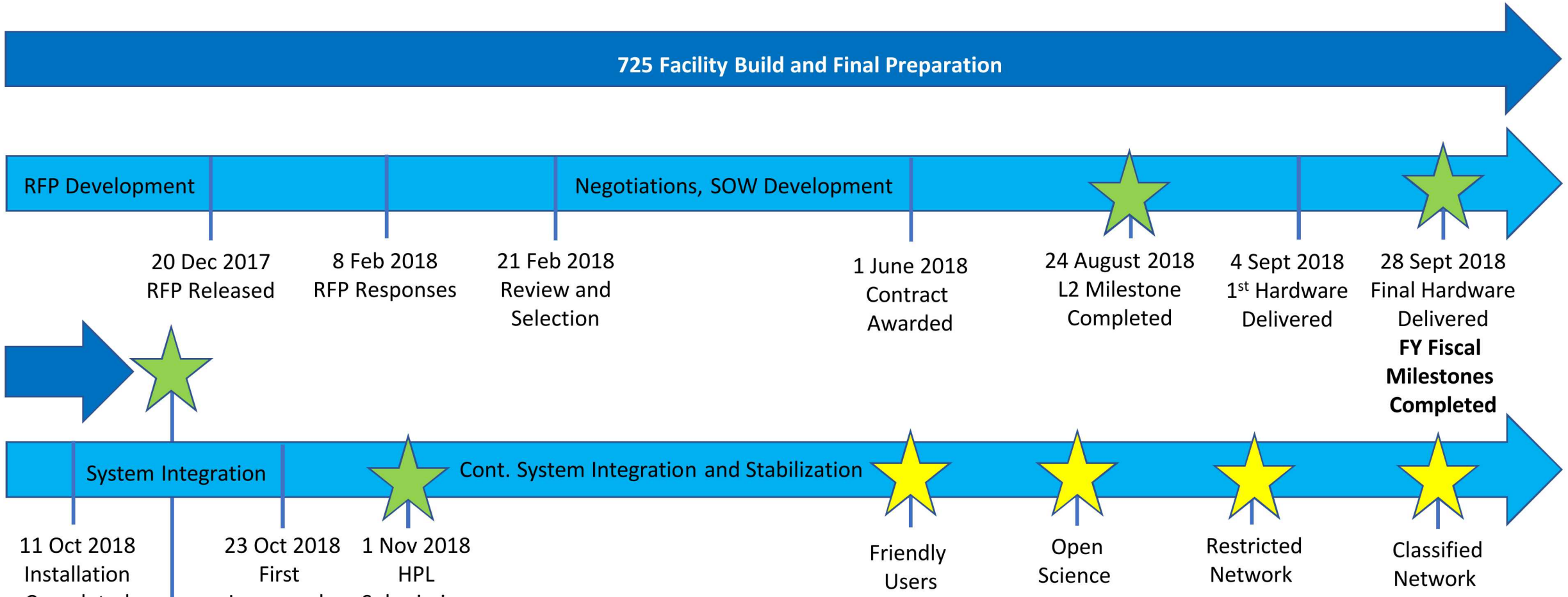
ATS/CTS Platforms

- Leadership-class systems (Petascale, Exascale, ...)
- Advanced technologies, sometimes first-of-kind
- Broad user-base
- PRODUCTION USE

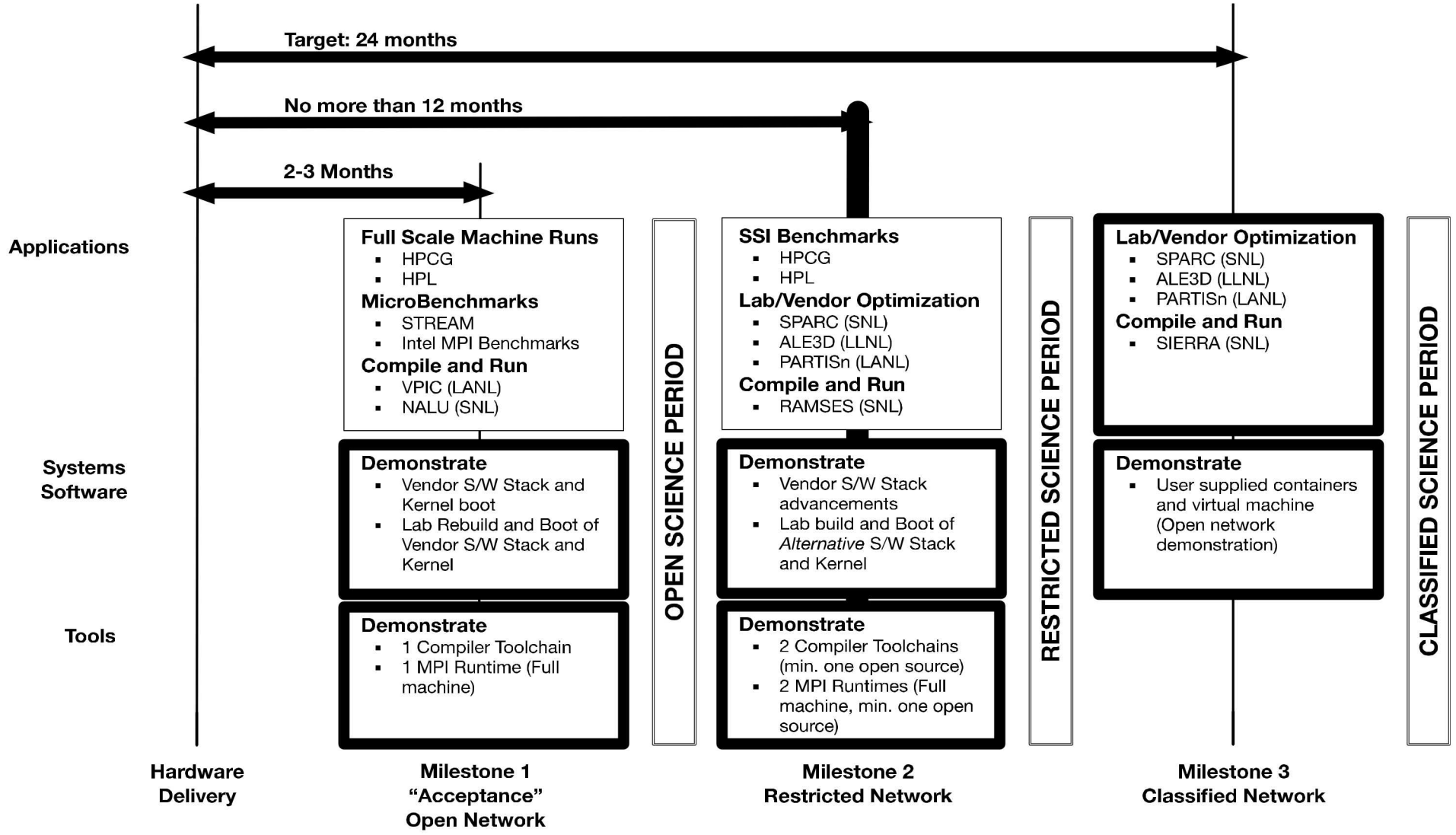
Sandia's NNSA/ASC ARM Platform Evolution



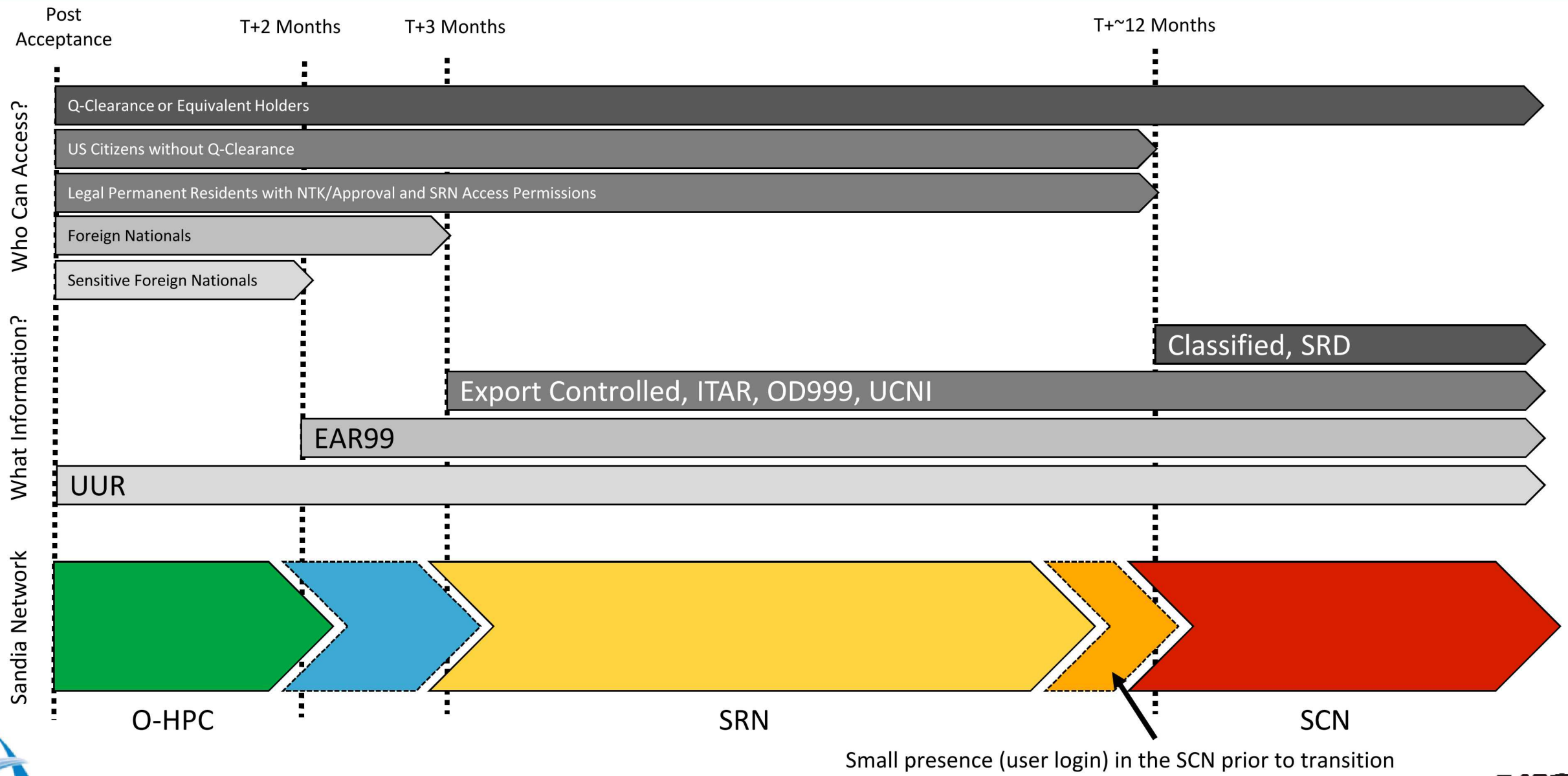
Vanguard-Astra: Timeline

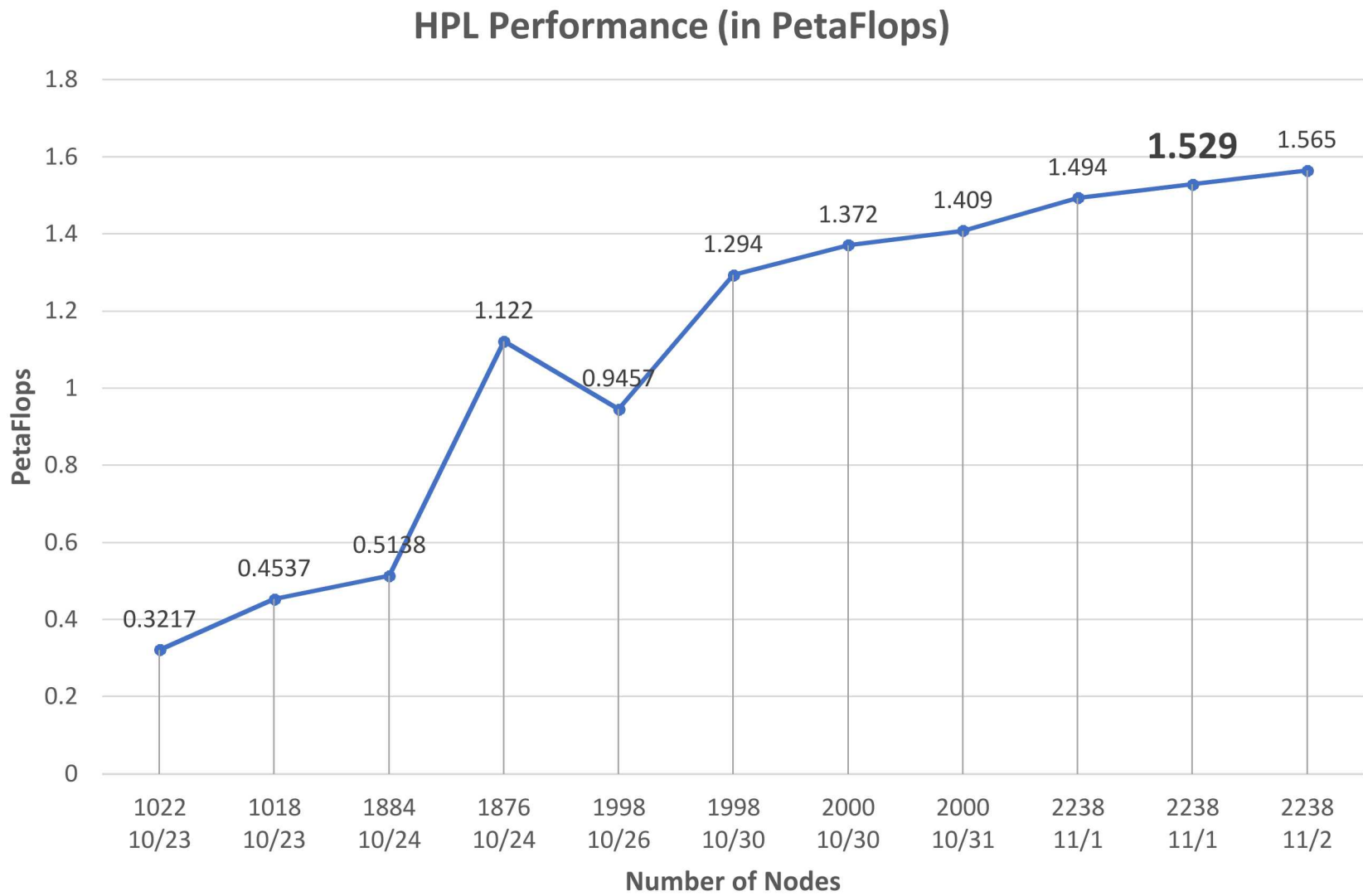


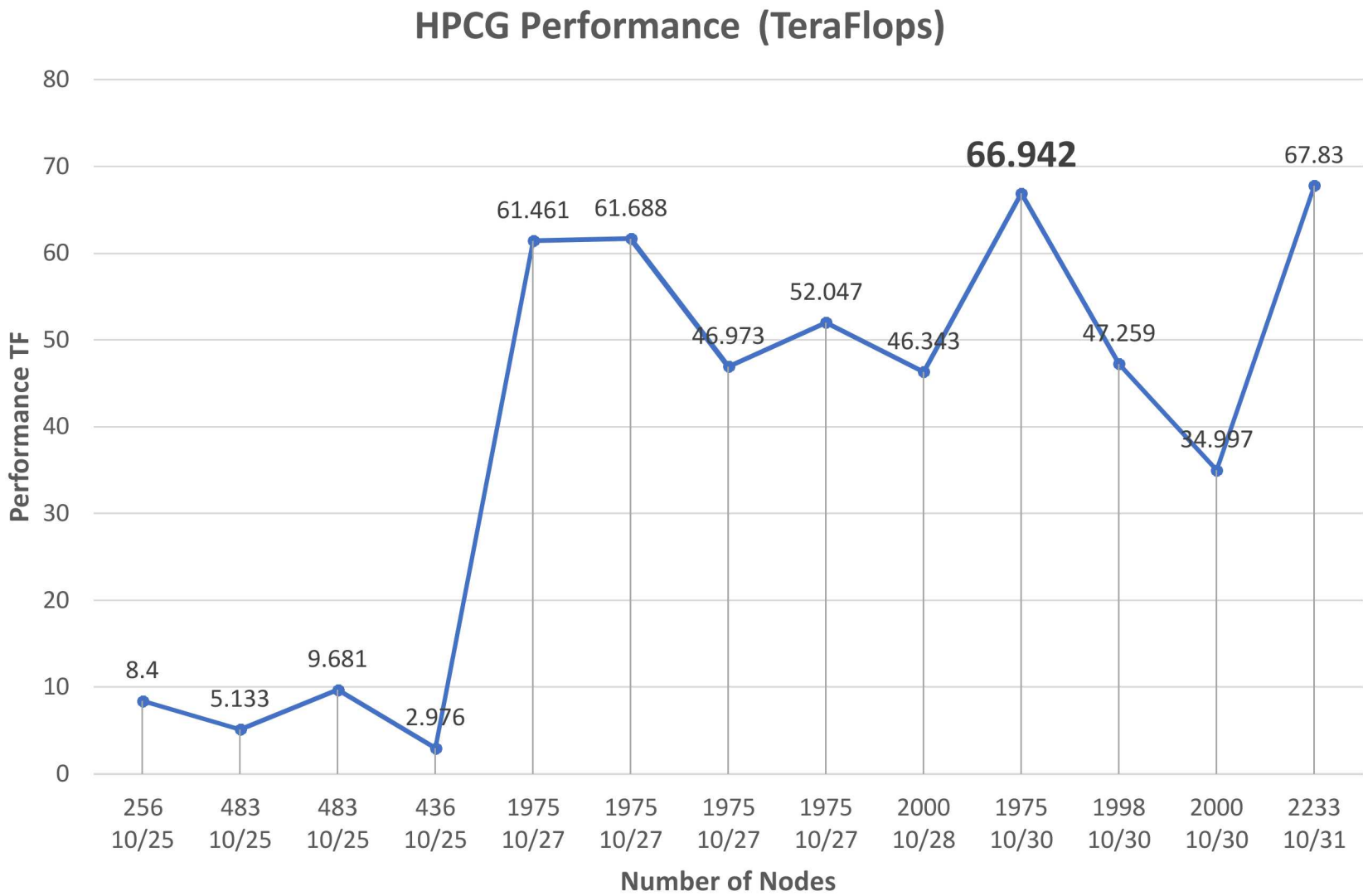
Vanguard-Astra: Project Milestones



User Access Program – Access and Information Handling




HPL Performance (in PetaFlops)
PetaFlops
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0.3217
0.4537
0.5138
1.122
0.9457
1.294
1.372
1.409
1.494
1.529
1.565
1022 10/23
1018 10/23
1884 10/24
1876 10/24
1998 10/26
1998 10/30
2000 10/30
2000 10/31
2238 11/1
2238 11/1
2238 11/2
Number of Nodes


HPCG Performance (TeraFlops)
Performance TF
80
70
60
50
40
30
20
10
0
8.4
5.133
9.681
2.976
61.461
61.688
46.973
52.047
46.343
66.942
47.259
34.997
67.83
256 10/25
483 10/25
483 10/25
436 10/25
1975 10/27
1975 10/27
1975 10/27
1975 10/27
2000 10/28
1975 10/30
1998 10/30
2000 10/30
2233 10/31
Number of Nodes

HPC wire

Since 1987 - Covering the Fastest Computers in the World and the People Who Run Them

- Home
- Technologies
- Sectors
- AI/ML/DL
- Exascale
- Specials
- Resource Library
- Events
- Job Bank
- About
- Solution Channels
- Subscribe

Sandia to Take Delivery of World's Largest Arm System

By Tiffany Trader

June 18, 2018

While the enterprise remains circumspect on prospects for Arm servers in the datacenter, the leadership HPC community is taking a bolder, brighter view of the x86 server CPU alternative. Amongst current and planned Arm HPC installations – i.e., the innovative Mont-Blanc project, led by Bull/Atos, the 'Isambard' Cray XC50 going into the University of Bristol, and commitments from both Japan and France among others — HPE is announcing that it will supply the United States National Nuclear Security Administration (NNSA) with a 2.3 petaflops peak Arm-based system, named Astra. On track for deployment at Sandia National Labs later this year as part of the NNSA's Arm-centric Vanguard project, Astra will be the world's largest Arm system ever built, according to HPE.


725
725

Exceptional Service in the National Interest

BACKUP

Vanguard-Astra: Procurement Schedule

- September through November 2017: multiple Draft RFI's released
- Week of September 25th 2017: Prime F2F presentations
- Target RFP release no later than January 12th 2018
  - **ACTUAL:RFP released December 20th**
- January 11th 2018: Vendor pre-proposal brief at Sandia NM
- February 8th 2018: RFP responses due
  - Distributed to technical team members same day!
- February 20th 2018: Opportunity for groups to meet Face to Face
- February 21st 2018: Technical review (SNL Albuquerque)
- February 21st 2018: Source Selection (with Tri-lab members)
- February/March 2018: Negotiations and SOW development
- April/May 2018: SOW development and contract placement
  - **ACTUAL: Award May 31st, Signed PO June 1st**
- July 2018: Facility (725-E) targeted completion
  - **ACTUAL: August**
- August 2018: Astra hardware delivery begins
  - **ACTUAL Aug 31st, postponed for labor day weekend 1st delivery September 4th**
- September 2018: Astra hardware delivery completed
  - **ACTUAL: Contractual September 14th, anticipated before September 26th**

Vanguard Program: Tri-Lab Software Effort

- Accelerate maturity of ARM ecosystem for ASC computing
  - Prove viability for NNSA integrated codes running at scale
  - Harden compilers, math libraries, tools, communication libraries
    - Heavily templated C++, Fortran 2003/2008, Gigabyte+ binaries, long compiles
  - Optimize performance, verify expected results
- Build integrated software stack
  - Programming env (compilers, math libs, tools, MPI, OMP, SHMEM, I/O, ...)
  - Low-level OS (HPC-optimized Linux, network stack, filesystems, containers/VMs, ...)
  - Job scheduling and management (WLM, scalable app launcher, user tools, ...)
  - System management (OS image management, boot, system monitoring, ...)
- Leverage prototype aspect of system for scalable system software R&D

Improve 0 to 60 time... Vanguard-Astra arrival to useful work done

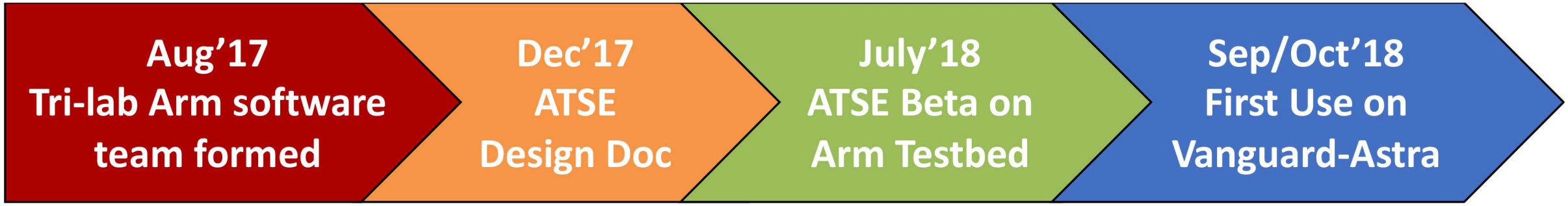
What is ATSE?

- Advanced Tri-lab Software Environment
  - Sandia leading development with input from Tri-lab Arm team
  - Will be the user programming environment for Vanguard-Astra
  - Version 1.0 targets including the software components needed for Milestone 1
- Lasting value
  - Documented specification of:
    - Software components needed for ASC production applications
    - How they are configured (i.e., what features and capabilities are enabled) and interact
    - User interfaces and conventions
  - Reference implementation:
    - Deployable on multiple ASC systems with common look and feel
    - Tested against real ASC workloads (open networks and classified)
    - Something to point vendors at in procurements

ATSE is an integrated software environment for ASC workloads

ATSE: High Level Goals

- Build an open, modular, extensible, community-engaged, and vendor-adaptable ecosystem
- Leverage existing efforts such as Tri-lab OS (TOSS), programing environments, and Exascale Computing Project software technologies
- Prototype new technologies that may improve the DOE ASC computing environment (e.g., ML frameworks, containers, VMs, OS optimizations)

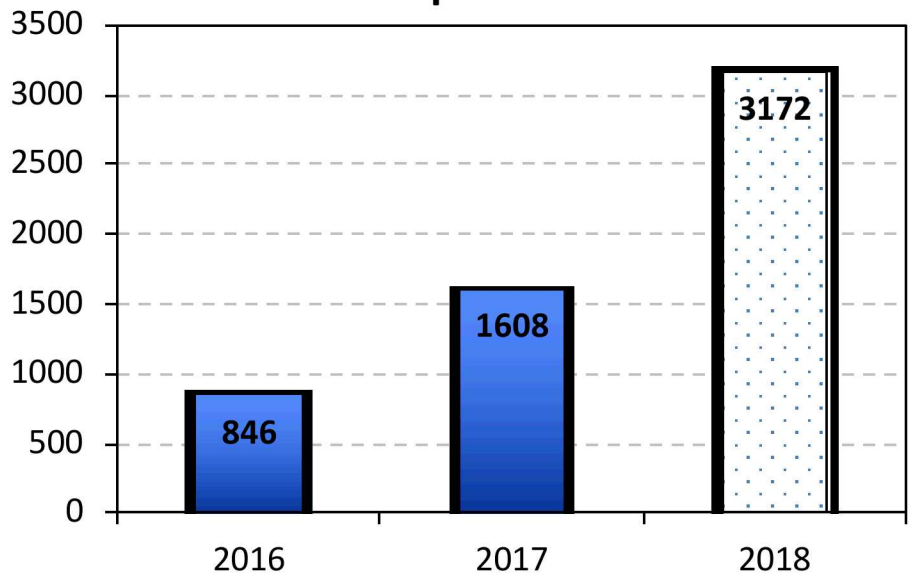


ATSE: OpenHPC Overview

- Targets HPC Linux clusters
  - Community effort, part of Linux Foundation
  - Common ingredients needed to deploy and manage an HPC cluster
  - Goal to enhance modularity and interchangeability of key components
  - Current release 1.3.5, builds on Centos 7.5 or SLES12, arm64 + x86_64
- Arm actively participating to make OpenHPC work on Arm, add Arm compilers
- Linaro actively participating, setting up CI functionality and performance testing

Number of Packages in OpenHPC

Number of Unique Visitors Per Month



ATSE: OpenHPC Evaluation

- Deployed OpenHPC on Mayer Arm-based testbed at Sandia
  - 47 compute nodes, dual-socket Cavium ThunderX2 28-core @ 2.0 GHZ
- Identified several gaps:
  - Focused on providing latest version of a given package
    - E.g., OpenHPC 1.3.5 moved to gcc7.3.0
      SNL validating with gcc7.2.0, need to use that version
  - Difficult to install multiple versions of a given package
  - Lacks architecture-optimized builds
    - HPL is 4.8x faster when compiled with an OpenBLAS targeting CaviumTX2 vs. OpenHPC OpenBLAS
  - Doesn't support static linking (build recipes actively remove static libraries)
    - Many users like to build static binaries, ship single binary to classified
  - Hard to rebuild packages due to reliance on Open Build Service

Engaging with OpenHPC community to address gaps

ATSE: Tri-lab Related Discussion Topics

- Sharing info and experiences working with Arm testbeds at each lab
- Vanguard-Astra status, governance, and user-access
- Milestone 1 and 2 application requirements, packages to include in ATSE 1.0
- System Management
  - HP Cluster Manager vs. Sandia GMI/oneSIS => **Try HPCM, be ready with Sandia Tools**
- Base Operating System
  - RedHat, CentOS, TOSS => **Use TOSS for Astra base OS, CentOS for Open-ATSE release**
- Network Stack
  - RedHat Inbox vs. Mellanox-OFED => **Use MOFED initially, push Mellanox to upstream**
- Lustre on Arm – Who is testing what and is it working?
- Container and VM support – CharlieCloud, Singularity, Docker, KVM
- ATSE Build and Packaging – TOSS Koji build farm, Open Build Service, Spack
- Cross-lab IT collaboration challenges

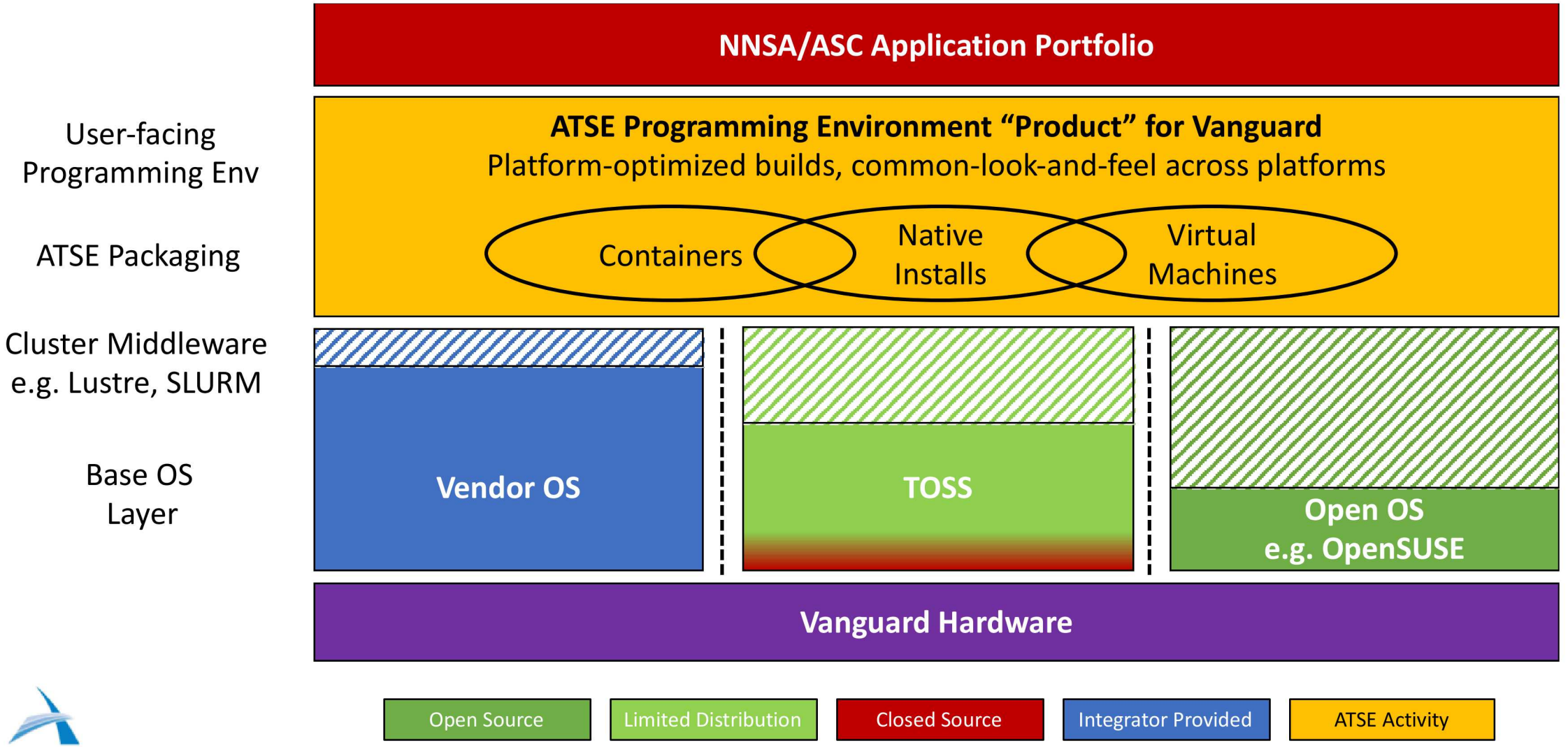
ATSE: Guiding Tenets

- Test with real ASC mission applications and problems
- Work in context of broader Arm community
- Upstream as much as possible
- Measure performance over time, quantify improvement
- Distribute something openly (important for external collaboration)
- Explore new technologies and approaches

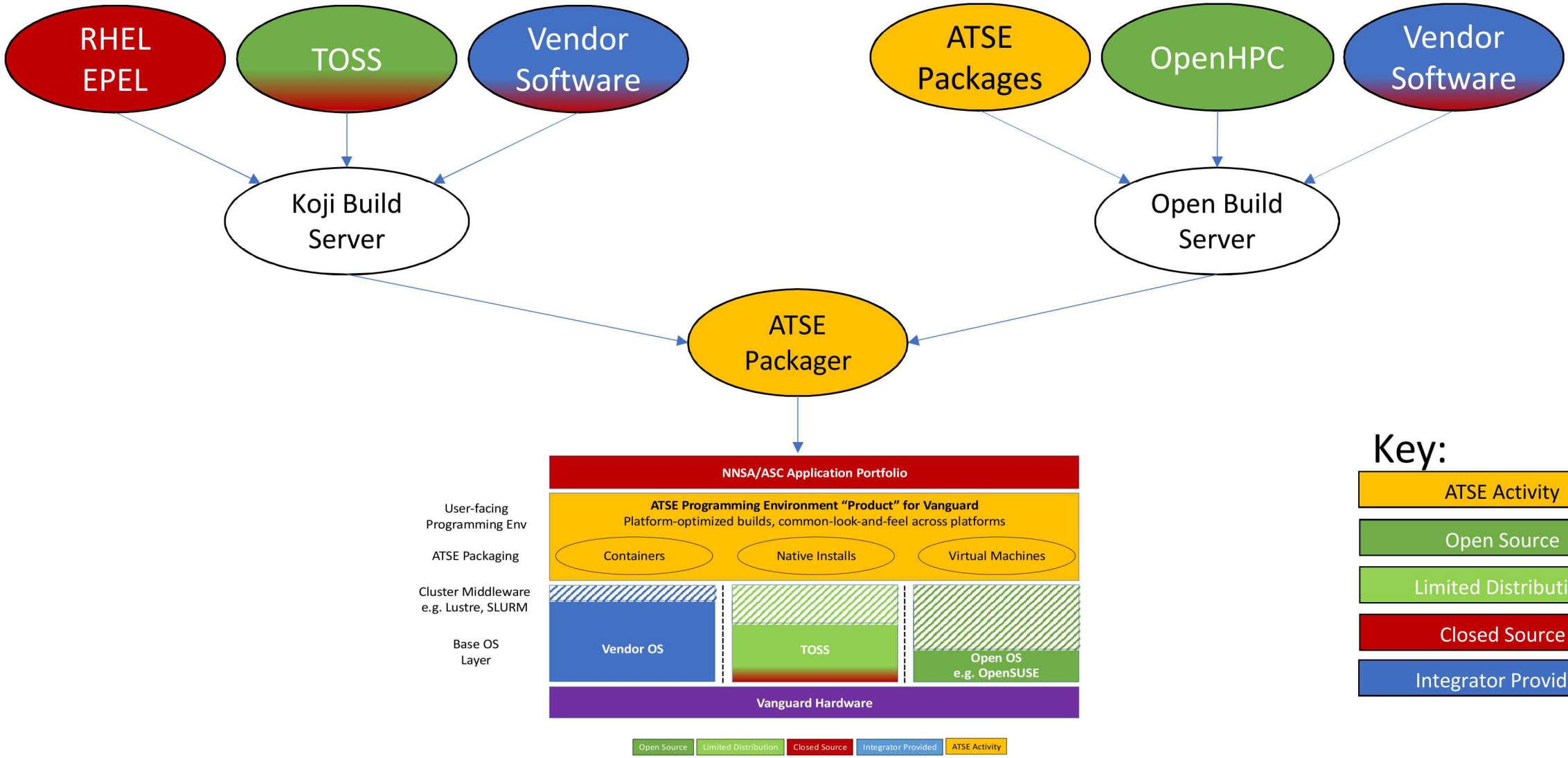
ATSE: Requirements

- Support NNSA mission applications, test against realistic input problems
- Be able to operate disconnected from the public internet
- Operate stand-alone or integrated with existing vendor infrastructure
- Support multiple processor architectures (arm64, x86_64, ppc64le, ...)
- Support multiple system architectures (Linux clusters, Cray, IBM, ...)
- Support multiple compiler toolchains (gcc, arm, intel, xlc, ...)
- Support multiple versions+configs of software packages with user selection
- Support both static and dynamic linking of applications
- Build static libraries with –fPIC
- Provide architecture-optimized packages
- Distribute container and virtual machine images of each ATSE release
- Source included, easy to rebuild and replace any non-proprietary package
- Provide open version of ATSE that is distributed publicly

ATSE: Integration with Multiple Base Operating Systems



ATSE: Pulling Components from Many Sources



Key:

- ATSE Activity
- Open Source
- Limited Distribution
- Closed Source
- Integrator Provided

ATSE: Milestone 1 Package List

	A	B	C	D	E	F	G	H	I	J
1	**Package Name**	**Priority**	**In Toss?**		**In TCE3?**		**In OpenHPC?**		**OpenHPC Path**	**ATSE Notes**
2	lmod	High	6.5.1		7.4.17		6.5.11, 7.7.14		/opt/ohpc/admin/lmod	
3	git	High	1.8.3		1.8.3, 2.8.3					Require version >= 2.9.5
4	autoconf	High	2.6.9				2.69		/opt/ohpc/pub/utils/autotools	
5	automake	High	1.13.4				1.15		/opt/ohpc/pub/utils/autotools	
6	libtool	High	2.4.2				2.4.6		/opt/ohpc/pub/utils/autotools	
7	make	High	3.82		4.2.1					
8	cmake	High	2.8.12		3.5.2, 3.8.2, 3.9.2		3.10.2		/opt/ohpc/pub/utils/cmake	
9	numactl	High	2.0.9							
10	hwloc	High	1.11.8		1.11.2		1.11.9		/opt/ohpc/pub/libs/hwloc	
11	openmpi	High	2.0, 2.1, 3.0		1.10.2, 2.0.0, 2.1.0, 3.0.1		1.10.6, 1.10.7, 3.0.0		/opt/ohpc/pub/mpi/openmpi3-{compiler}	Need UCX support
12	curl	High	7.29.0							
13	glibc	Low								Have patch to glibc libm for aarch64 performance. Should proba
14	binutils	High	2.27, 2.28		2.30					Need >= 2.27.x for aarch64 features
15	gcc	High	4.8.5, 7.1.0, 7.2.1, 7.3.1		6.1.0, 7.1.0, 7.3.0, 8.1.0		5.4.0, 7.3.0		/opt/ohpc/pub/compiler/gcc	To build gcc, need to be sure host has libstdc++-static package ins
16	gdb	High	7.6.1, 8.0.1							
17	openblas	High	0.2.20				0.2.19, 0.2.20		/opt/ohpc/pub/libs/{compiler}/openblas	
18	scalapack	High					2.0.2		/opt/ohpc/pub/libs/{compiler}/{mpi}/scalapck	
19	zlib	High	1.2.7							
20	hdf5-parallel with zlib	High	1.8.12		1.8.17		1.8.17, 1.10.1		/opt/ohpc/pub/libs/{compiler}/{mpi}/hdf5	
21	pnetcdf	High			1.9.0		1.8.1		/opt/ohpc/pub/libs/{compiler}/{mpi}/pnetcdf	
22	netcdf	High	4.3.3.1		4.4.1.1		4.2.1, 4.3.0, 4.4.1, 4.4.4, 4.5.0		/opt/ohpc/pub/libs/{compiler}/{mpi}/netcdf	
23	superlu	High					5.2.1		/opt/ohpc/pub/libs/{compiler}/superlu	
24	superlu_dist	High					4.2, 5.3.0		/opt/ohpc/pub/libs/{compiler}/{mpi}/superlu_dist	
25	parmetis	High								Not in OpenHPC due to restrictive licensing
26	metis	High					5.1.0		/opt/ohpc/pub/libs/{compiler}/metis	
27	boost	High	1.53		1.58, 1.62, 1.66		1.61.0, 1.63.0, 1.66.0		/opt/ohpc/pub/libs/{compiler}/{mpi}/boost	
28	yaml-cpp	High	0.5.1							
29	valgrind	High	3.13.0		3.11.0, 3.12.0, 3.13.0		3.11.0, 3.13.0		/opt/ohpc/pub/utils/valgrind	
30	trilinos	Low					12.6.4, 12.10.1, 12.12.1		/opt/ohpc/pub/libs/{compiler}/{mpi}/trilinos	Low priority because most apps build their own special version. S
31	netcdf-exo	High	? Exodus patch needed ?		? Exodus patch needed ?		? Exodus patch needed ?			SNL modified, requires patch
32	pnetcdf-exo	High	? Exodus patch needed ?		? Exodus patch needed ?		? Exodus patch needed ?			SNL modified, requires patch
33	mpihello	Low								
34	ARM compilers	High	18.1, 18.3				In testing			Proprietary
35	ARM performance libs	High	18.3.0							Proprietary
36	HPE MPI	Low	? LLNL and SNL have for x86							Proprietary. aarch64 port won't be ready until after ATSE initial r
37	SLURM	High	17.02.10				16.05.10, 17.11.5		/usr	Need build with hwloc and pmix support. OpenHPC recipe needs
38	singularity	High	2.4.4, 2.5.1				2.4.5		/opt/ohpc/pub/libs/singularity	
39	spack	High					0.8.17, 0.11.2		/opt/ohpc/admin/spack	Need to move to /pub, make user accesible
40	Alinea tools	High								Proprietary
41	gasnet	Low								Not needed for initial target benchmarks and apps
42	kokkos	Low								
43	kokkos kernels	Low								
44	powerapi	Low	? This used to be there ?							
45	fftw	High	2.1, 3.3.3		3.3.4, 3.3.7		3.3.4, 3.3.6, 3.3.7		/opt/ohpc/pub/libs/{compiler}/{mpi}/fftw	
46	yorick	?								Needed for PF3D
47	xpmem	Low	? Needs security audit							Linux kernel module and user level library. Not necessary until HI
48	IB stack, inbox or MOFED	High	RHEL Inbox Drivers		N/A		RHEL Inbox Drivers			Need socket direct support and SHARP support. Currently SNL te
49	Lustre Client	High	2.8, 2.10		N/A		2.8		? Not currently using	Need 2.10 with patches for aarch64, 2.12 needs patches as well?
50										

ATSE: Deployed Beta Stack

- Setup local Open Build Service (OBS) build farm at Sandia
- Built set of software packages needed for Astra milestone 1
  - When OpenHPC recipe was available, we tried to use it, modifying as necessary
  - Otherwise, we built a new build recipe in same style as OpenHPC
- Installed on Mayer testbed at Sandia, now using as the default user environment
- Tested with STREAM, HPL, HPCG, ASC mini-apps
- Compiler toolchain support
  - GNU compilers, 7.2.0
  - ARM HPC compilers

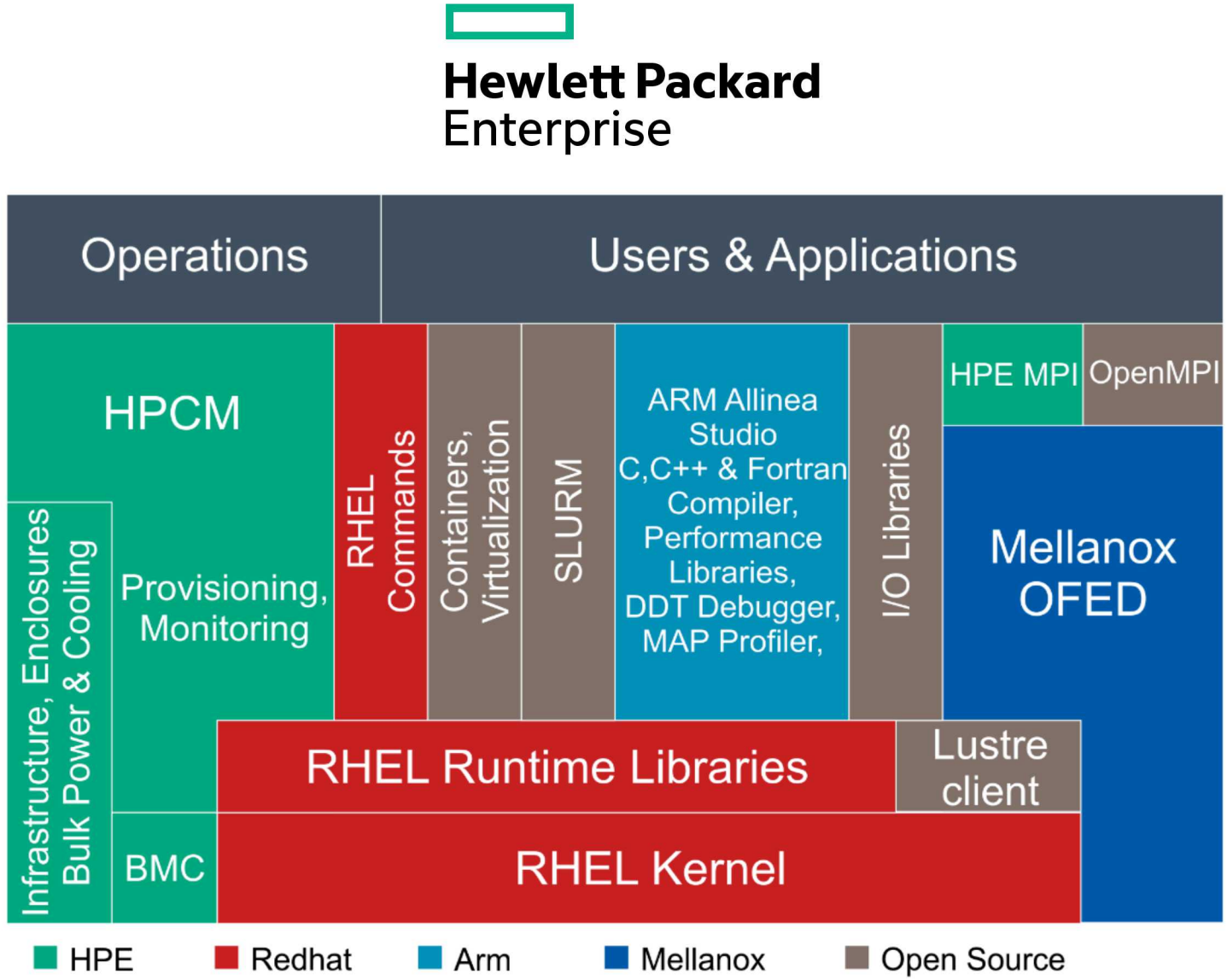
ATSE Modules Interface, Mirrors OpenHPC

```
[ktpedre@mayer2 ~]$ module avail

------------------------------ /opt/atse/pub/moduledeps/gnu7-openmpi3 -------------------------
   phdf5/1.10.2    pnetcdf/1.9.0

---------------------------------- /opt/atse/pub/moduledeps/gnu7 ------------------------------
   hdf5/1.10.2    openblas/0.2.20    openmpi3/3.1.1 (L)

-------------------------------------- /opt/atse/pub/modulefiles ------------------------------
   arm/18.3          binutils/2.30 (L)    gnu7/7.2.0     (L)    pmix/2.1.1          spack/0.11.2
   atse       (L)    cmake/3.11.1  (L)    hwloc/1.11.10         prun/1.2            zlib/1.2.11
   autotools  (L)    git/2.18.0    (L)    numactl/2.0.12        singularity/2.5.2

  Where:
   L:  Module is loaded

Use "module spider" to find all possible modules.
Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".
```

ATSE: Collaboration with HPE OLSS Effort

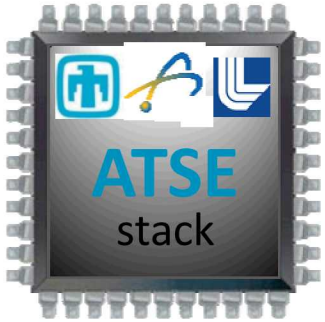
Open Leadership Software Stack (OLSS)

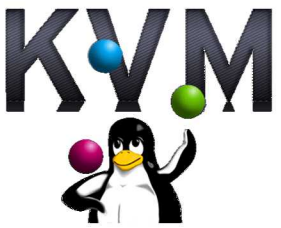
- HPE:
  - HPE MPI (+ XPMEM)
  - HPE Cluster Manager
- Arm:
  - Arm HPC Compilers
  - Arm Math Libraries
  - Allinea Tools
- Mellanox-OFED & HPC-X
- RedHat 7.x for aarch64



ATSE: R&D Efforts

- Workflows leveraging containers and virtual machines
  - Support for machine learning frameworks
  - ARMv8.1 includes new virtualization extensions, SR-IOV
- Evaluating parallel filesystems + I/O systems @ scale
  - GlusterFS, Ceph, BeeGFS, Sandia Data Warehouse, …
- Resilience studies over Astra lifetime
- Improved MPI thread support, matching acceleration
- OS optimizations for HPC @ scale
  - Exploring spectrum from stock distro Linux kernel to HPC-tuned Linux kernels to non-Linux lightweight kernels and multi-kernels
  - Arm-specific optimizations





ATSE: Next Steps

- Continue adding and optimizing packages and build test framework
- Package container and VM images
  - Lab-internal version, hosted on Sandia Gitlab Docker registry
  - Externally distributable versions, stripping out proprietary components
- Submit ”trial-run” patches filling gaps to OpenHPC community
- Explore Spack build and packaging
- Continue collaboration with HPE OLSS team, first external ATSE customer

- LANL: CharlieCloud/Bee support, LLVM compiler work, application readiness
- LLNL: TOSS, “Spack Stacks” ATSE build, application readiness

User Access Program and Applications

User Access Program - Prioritization

Access to the Astra platform and system allocations will initially be performed using the following prioritization:

1. Vanguard/Astra Trilab System Bring Up/ATSE Software Development Teams and Vanguard/Astra Trilab NNSA/ASC Priority Application Porting/Testing and Benchmarking Activities
2. NNSA/ASC Trilab Application Teams, including ATDM Apps
3. Limited Vendor/System Integrator testing (not expected once acceptance is complete)
4. Selected NNSA PSAAP-II Centers (Negotiated with ASC)
5. Selected ECP/ST and ECP/AD Projects (Negotiated with ASC)
6. Selected users for Collaborators/SSP Relationships
7. Selected other University Partnerships/Programs

All projects are eligible to request a Dedicated Application Time (DAT) through the Vanguard/Astra allocation request mechanism

User Access Program – Scheduling

In general the system will provide a fair-share system scheduling mechanism. Users are expected to submit jobs subject to the system queue maximum limits and to have these run via standard scheduling priorities.

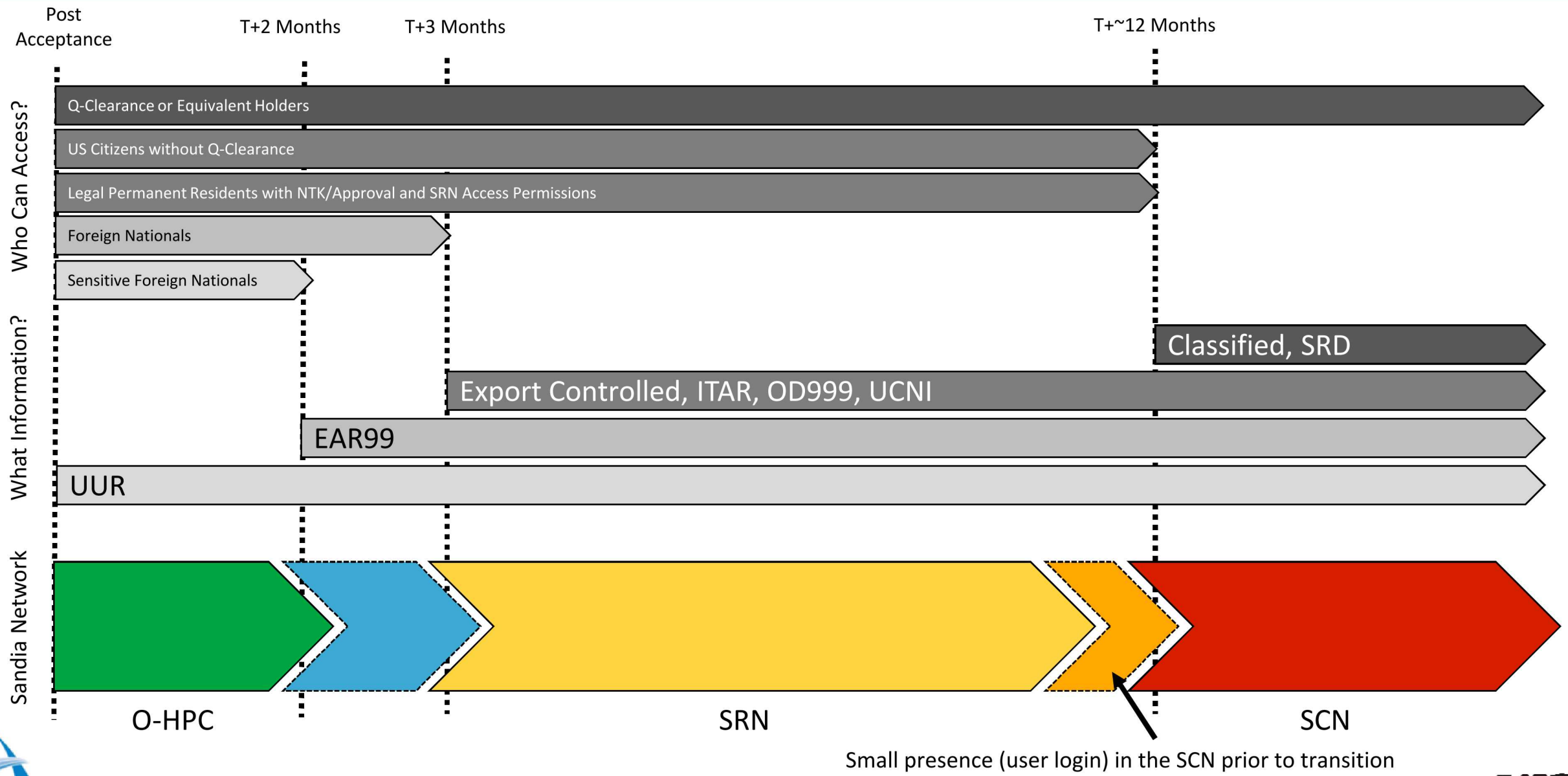
- The maximum queue limit will be set to 24 hours for all queues/users
- There will be a maximum job size applied to all queues
- There will be a debug queue with limited hardware resources and shorter maximum time duration (4 hours)
- There will be a maximum number of jobs permitted to be eligible to be run on a per user basis

Requests for Dedicated Access Time will be judged based on:

- User project's priority (see general Vanguard/Astra prioritization)
- System and software stability
- Project planning schedule
- Project management approval
- Security profile of the system (i.e. what network the machine is connected to etc.)

Dedicated (full) system time periods can be requested but are not expected to exceed 24 hours. Additionally, reservations for specific resources/node allocations can be requested

User Access Program – Access and Information Handling

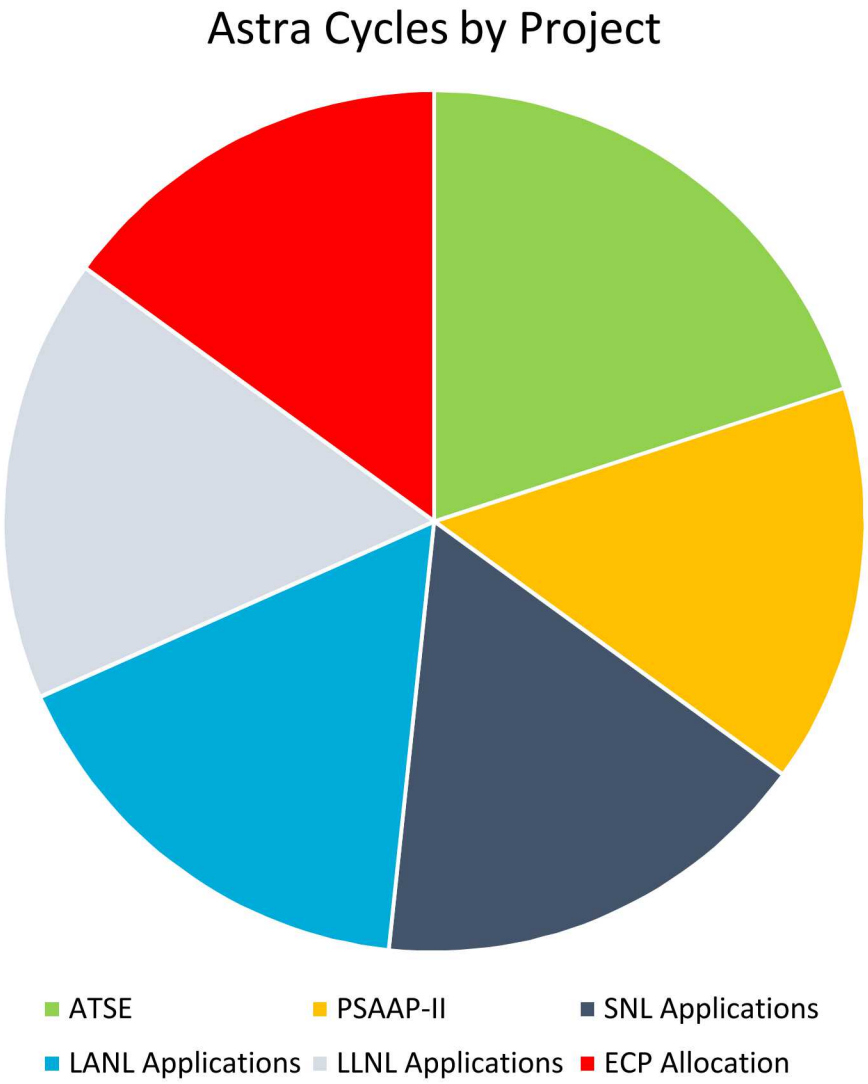


Scheduling Fair Share and Cycles – Open Period

Proposal for priority breakdown:

- ATSE Software Stack – approximately 20% of cycles
  - ATSE will be shared across all three Tri-lab users as testing is required
- Tri-Lab Application Bring up – approximately 50% of cycles
  - 1/3 of the 50% (=17%) for each Trilab, priority fair-shared by user within the bin
  - Including ATDM Apps
- PSAAP-II users – approximately 15% of cycles
- ECP-ST and AD – approximately 15% of cycles

Jobs will be fair-share priority within the banks

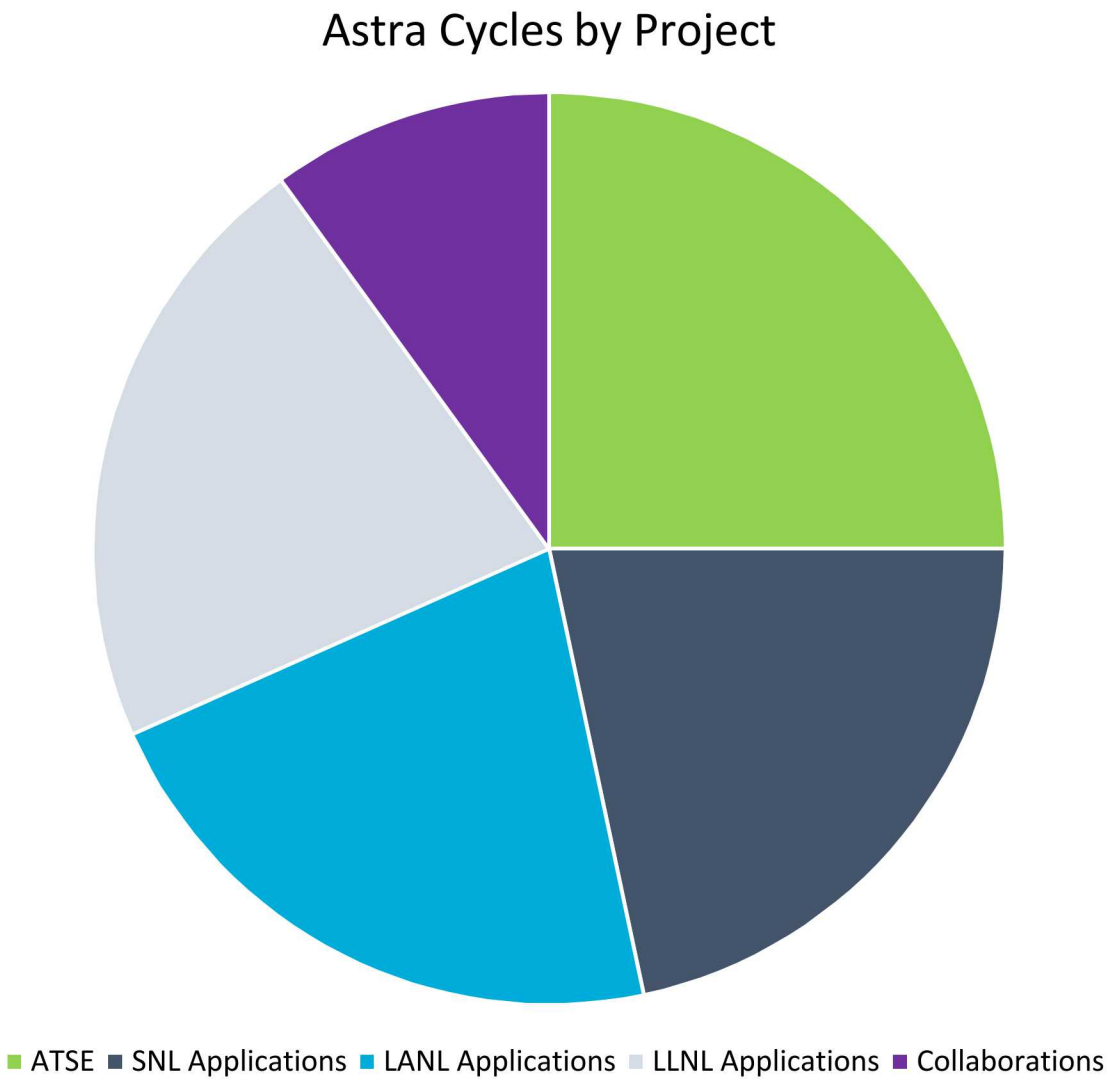


Scheduling Fair Share and Cycles – Restricted Period

Proposal for priority breakdown:

- ATSE Software Stack – approximately 25% of cycles
  - ATSE will be shared across all three Tri-lab users as testing is required
- Tri-Lab Application Bring up – approximately 65% of cycles
  - 1/3 of the 65% (=~21%) for each Trilab, priority fair-shared by user within the bin
  - Including ATDM Apps
- Small allocation for collaborations (SSP/WFO, selected University projects, limited PSAAP-II) – approximately 10% of cycles

Jobs will be fair-share priority within the banks

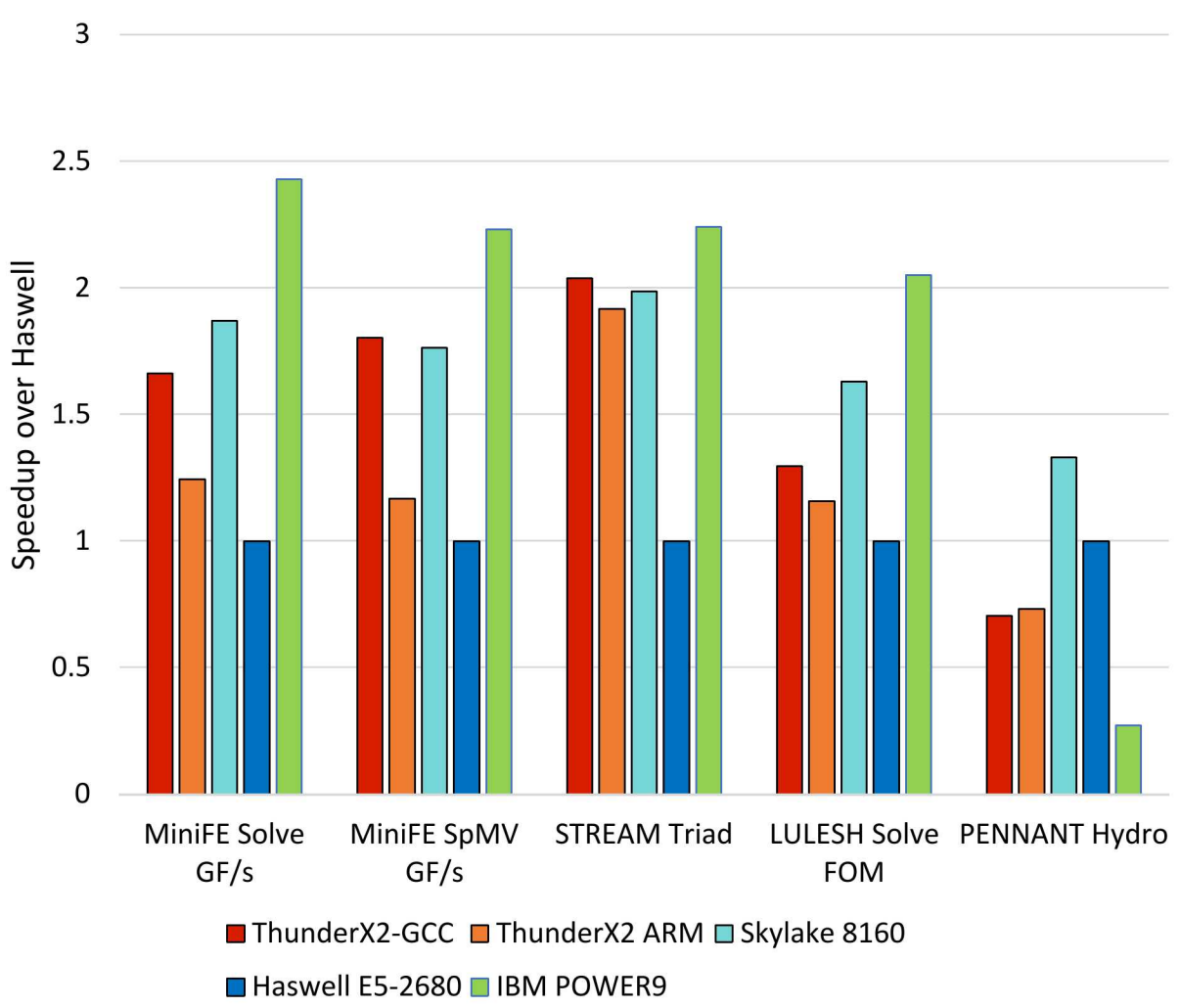


Early Performance Results from ThunderX2

Early results show strong memory subsystem performance

- A0 silicon has significant number of performance issues in coherency handler
- Difficult to take early performance results because of limited B0 silicon availability
  - B1 silicon + firmware will improve performance further
- Mixed results from ARM compiler
  - Still forming early picture
  - Outperform GCC in number of benchmarks

- Local results showing around 775GF/s (HPL) single node and around 24GF/s on HPCG for ARM variant
- Expecting this to be boosted by B1 silicon (HPL), updated firmware and HPE implementation of HPCG

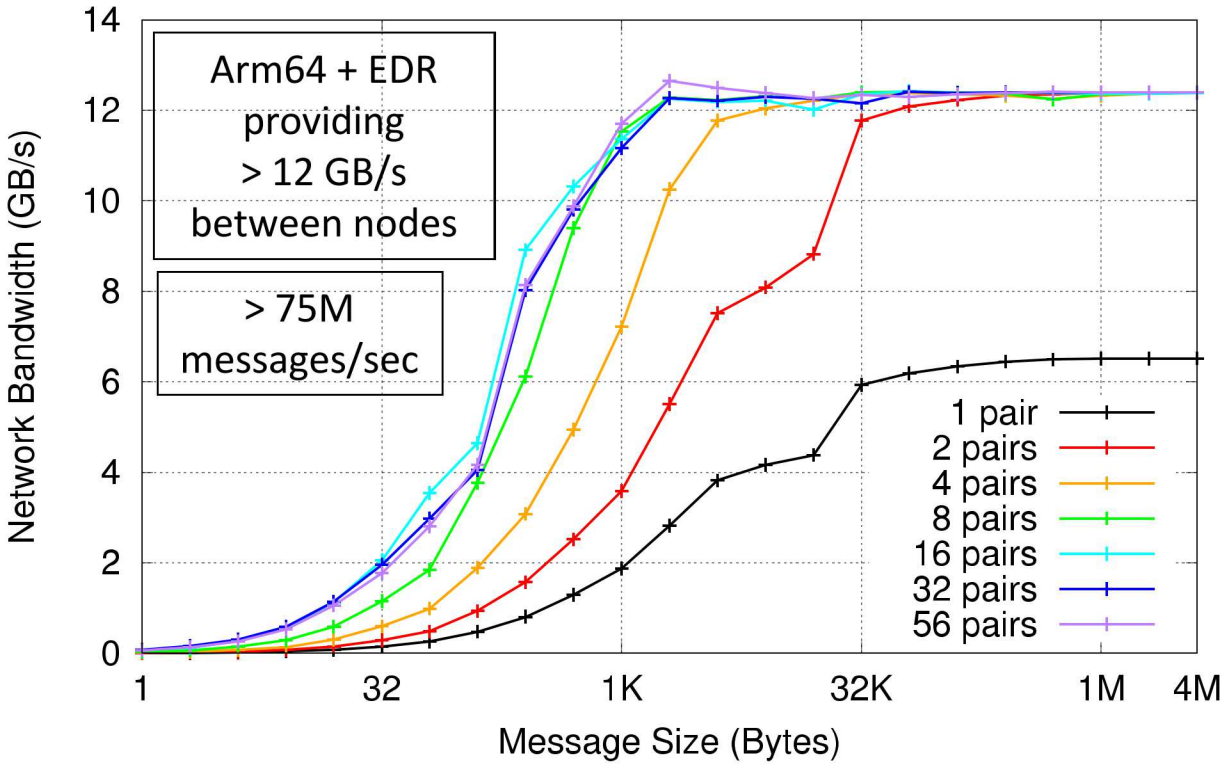


Early Performance Results from Mayer InfiniBand

Socket Direct feature enables a single NIC to be shared by multiple host processor sockets

- Share a single physical link to reduce cabling complexity and costs
- NIC arbitrates between host processors to ensure a fair level of service
- Required some complex O/S patches early on in test systems

OSU MPI Multi-Network Bandwidth



State of Application Porting

Functional Application Porting activities have been underway for a short time

- Mainly focused on trying to get “framework” packages built (e.g. Trilinos) and then a range of codes
- Mainly focused on Sandia codes as a quick assessment of toolchain issues etc
- Just intended to guide our initial functional checks

- Lots of discussions underway with LLNL (thank you – Dave Richards) about suite of codes we can use for benchmarking and tools assessment
- Similar discussion with Randy Baker at LANL on using PARTISn

Application	Ported?
SPARTA	
SPARC	
CTH	
Xyce-UUR	
NALU	
VPIC (LANL)	
LAMMPS	
SSPARKS	
PENNANT	
Sweep3D	
MiniFE	
LULESH	

Applications – Next Steps and Milestones

Milestone 1
Open Science
3-4 months

Full Scale Machine Runs
- HPCG
- HPL

Micro-benchmarks
- STREAM
- Intel MPI Benchmarks

Compile and Run
- **NALU (SNL)**
- **VPIC (LANL)**
- **PF3D (LLNL)**

Milestone 2
Restricted Science
12-15 months

SSI Benchmarks
- HPCG
- HPL

Lab/Vendor Optimization
- **SPARC (SNL)**
- **PARTISn (LANL)**
- **ALE3D (LLNL)**

Compile and Run
- **RAMSES (SNL)**

Milestone 3
Classified Science
Remainder of Life

Lab/Vendor Optimization
- **SPARC (SNL)**
- **PARTISn (LANL)**
- **ALE3D (LLNL)**

Compile and Run
- **SIERRA (SNL)**

Demonstrate
- **User-specified containers and virtual machines**

Exceptional Service in the National Interest