SAND2023-05478C

Sandia National Laboratories

Advanced Simulation and Computing DevOps Unified Environment at Sandia National Laboratories

Presented by Erik Strack, ASC Integrated Codes Sub-program Manager

Content developed by Scott Warnock and the DevOps Technical Leadership and Management Teams

28 June 2023, 9:50 –10:30

U.S. DEPARTMENT OF ENERGY NNSA

Sandia National Laboratories is a multimission 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.

Sandia National Laboratories is a multimission 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.

DevOps for National Security High-Performance Computing

The Advanced Simulation and Computing (ASC) Program provides high-fidelity computational simulation capabilities for safety and security of our nuclear deterrent

National Nuclear Security Administration

Advanced Simulation & Computing™

The focus of this presentation is on the Sandia National Laboratories internal ASC DevOps initiative, which is being designed to fit into the national ASC program

Sub-program Elements

- Integrated Codes (IC)
- Physics and Engineering Models (PEM)
- Verification and Validation (V&V)
- Computational Systems and Software Environment (CSSE)
- Facility Operations and User Support (FOUS)

Participating Laboratories

- Sandia National Laboratories (Design Agency - DA)
- Los Alamos National Laboratory (DA)
- Lawrence Livermore National Laboratory (DA)
- Production Agencies (PA) Integration
  - Kansas City Plant + Y-12 + Pantex

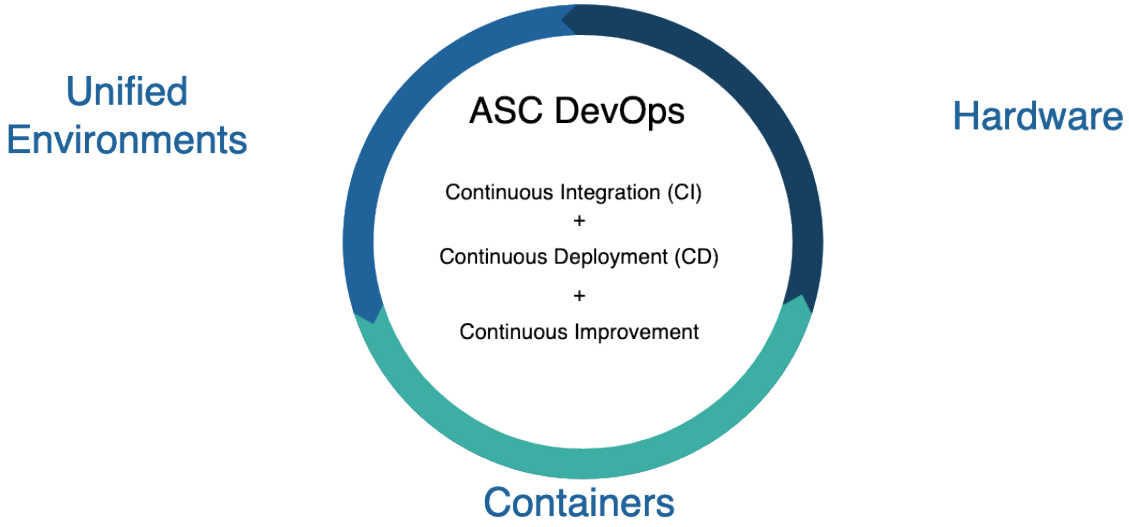
Program Platforms Top 500 Rankings

- Sierra #6
- Trinity #29
- Upcoming Platforms
  - Crossroads & El Capitan

Advanced Simulation and Computing (ASC) DevOps Initiative

ASC DevOps Vision

ASC codes are developed, tested, deployed, and released quickly, efficiently, and robustly with established credibility on all key ASC platforms. Codes are efficiently coupled and integrated in the ASC DevOps ecosystem, which is governed by well-defined processes, practices, and standards. Ultimately, end-users will be able to seamlessly run recently released credible ASC code(s) on key ASC platforms.



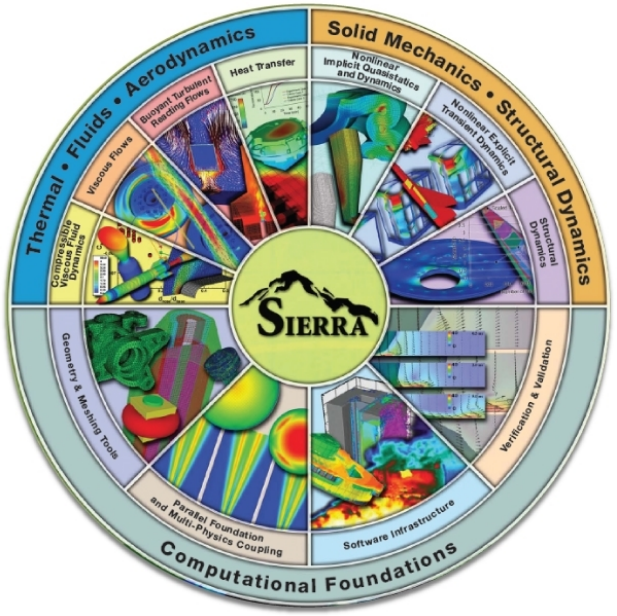
Complicated Effort

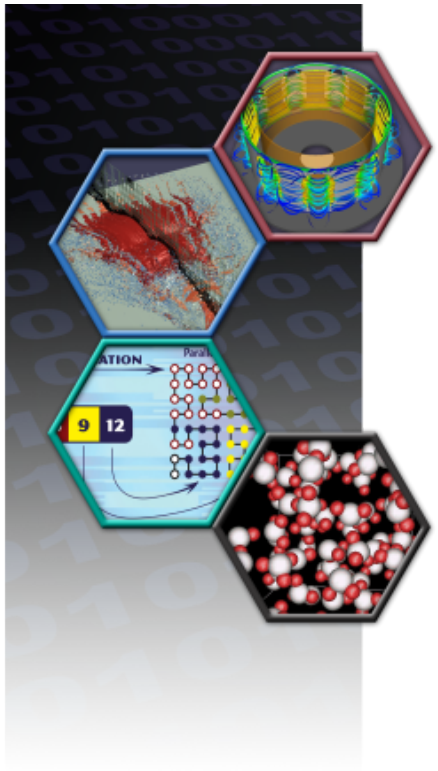
- Multiple prior attempts
- Required a new, different approach

Essentials to success

- Communication & Governance Structure
  - Broad consensus, from technical staff up to senior management
- Team Building & Managing Change
  - See Thorson's presentation Friday
- Managing complexity & Effort
  - Software ecosystem and library dependencies

ASC and a Unified Environment at Sandia





RAMSES

Charon CHEETAH EIGER EMPIRE Gemma
ITS NuGET Q SCEPTRE Xyce

Sub-program Integration

- Codes + Models + Credibility
- + HPC Systems, Operations, & Software Stack

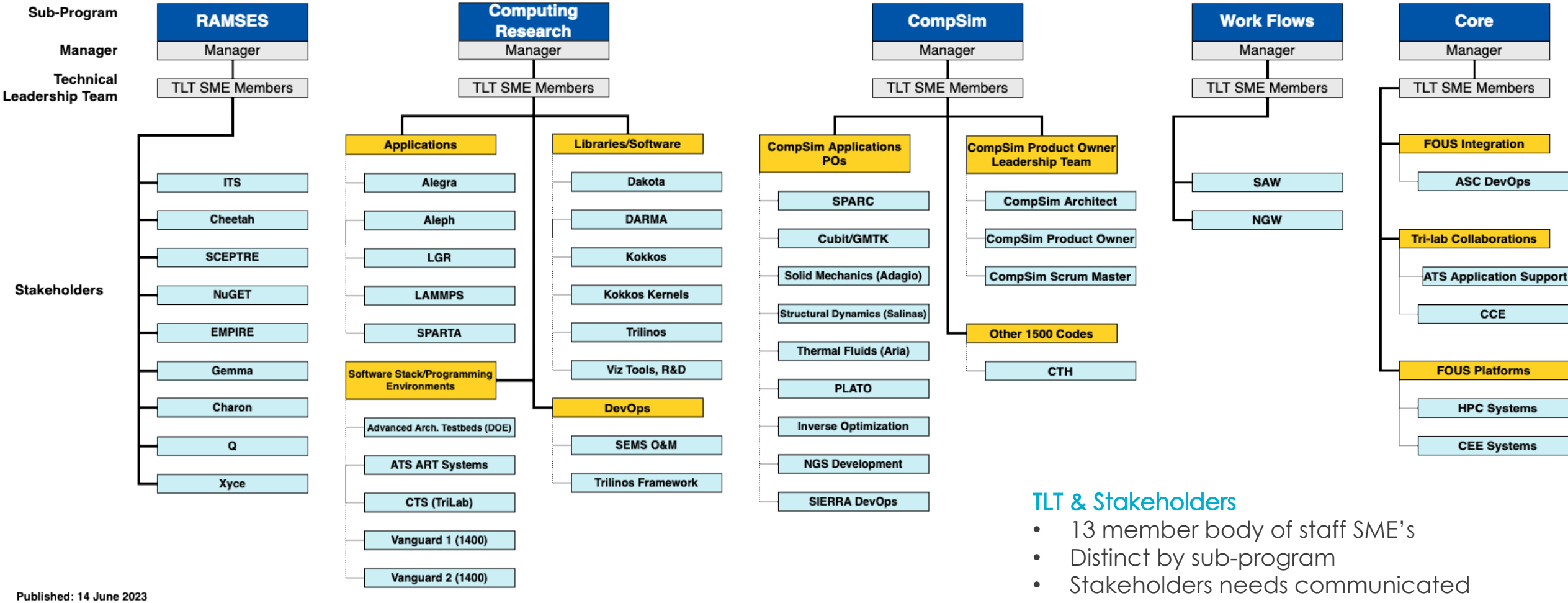
Scope

- 31 Scientific & Engineering Code Teams and multiple systems admin. teams
  - HPC: Prototype, Advanced, and Commodity
  - Desktop: Windows, Linux, Mac
- 200+ Developers
- Partner daily with users (designers & analysts)

History

- Some teams on build systems pre-date1995
- Multiple attempts at unified ASC DevOps

ASC Technical Leadership Team (TLT) and Stakeholders

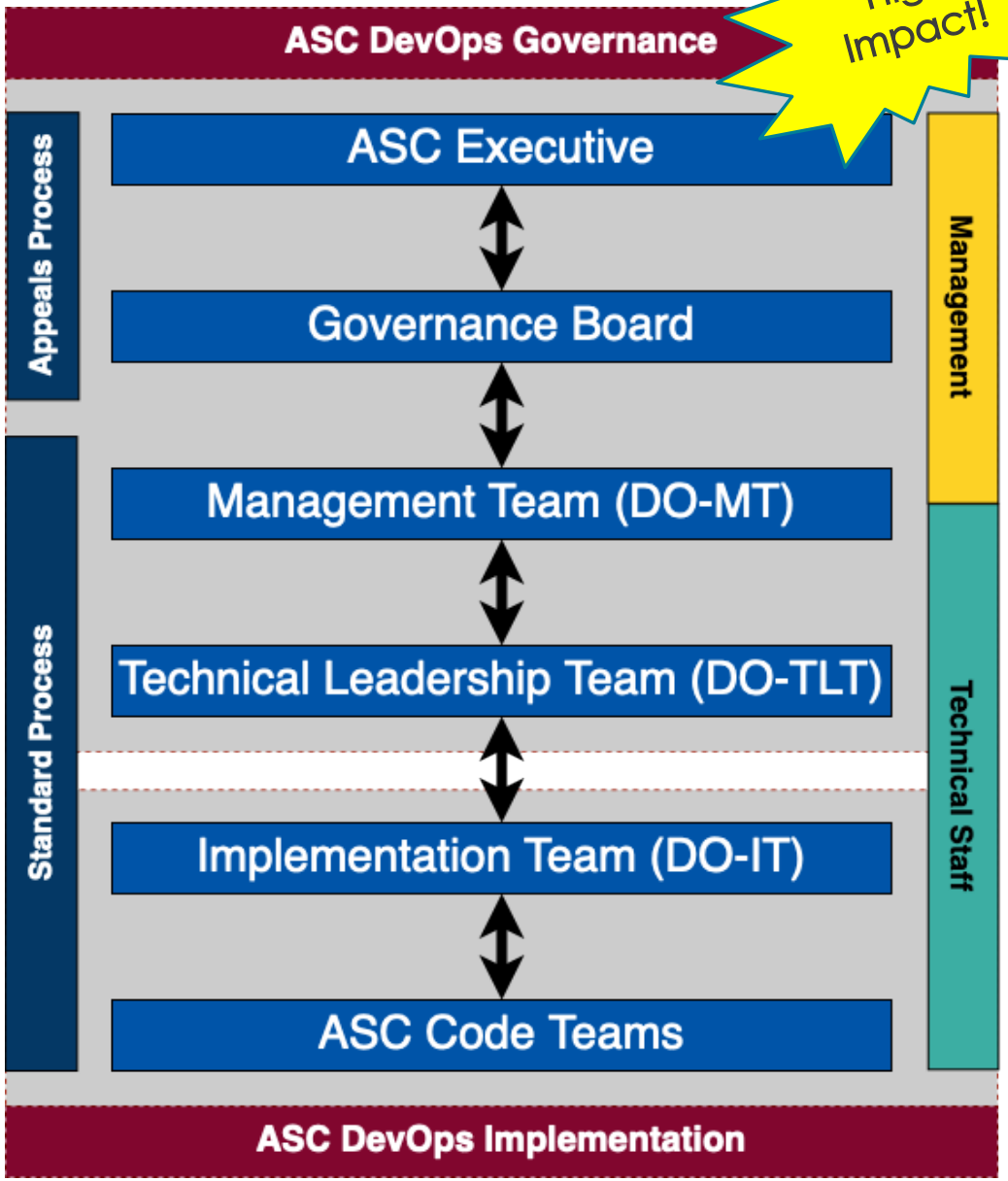


Published: 14 June 2023

TLT & Stakeholders

- 13 member body of staff SME's
- Distinct by sub-program
- Stakeholders needs communicated through TLT representatives

Governance Structure



High Impact!

ASC Stakeholders

Stakeholders:

- **Analysts**
- **Applications developers**
- **DevOps developers**
- **Platform systems designers/administrators**
- **Platform cyber security staff**

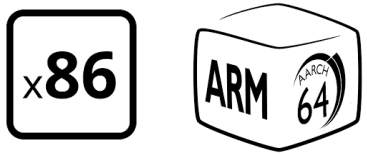
Programmatic & line management (L1s, L2s, Directors)

- Technical Staff part of strategic and tactical direction
- Established adjudication process
- DO-ITs
  - Auto-RAFTS
  - Containers
  - Solver Common Configuration

ASC Unified Environment

Continuous Integration

Heterogenous Architectures



Network Hierarchy

OPEN SECURE CLASSIFIED

The ASC Unified Environment Stack

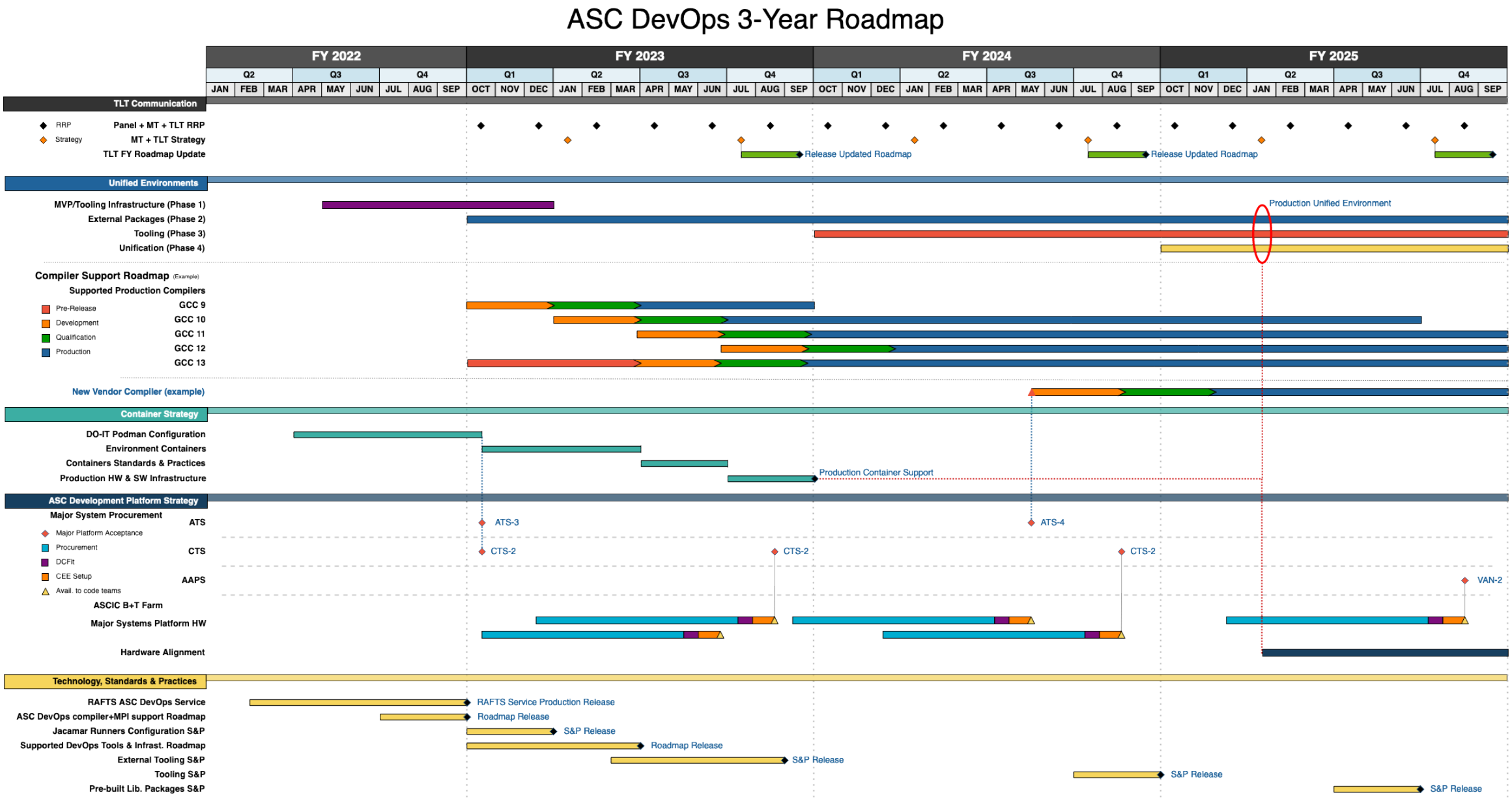
Tooling, Profilers, & Debuggers
- 9 tooling products
- 5 profilers & debuggers

Compilers & MPI
- 5 compilers
- 2 flavors of MPI

TPLs
- 13 TPLs

34 products, multiple architectures, multiple networks, many combinatorics

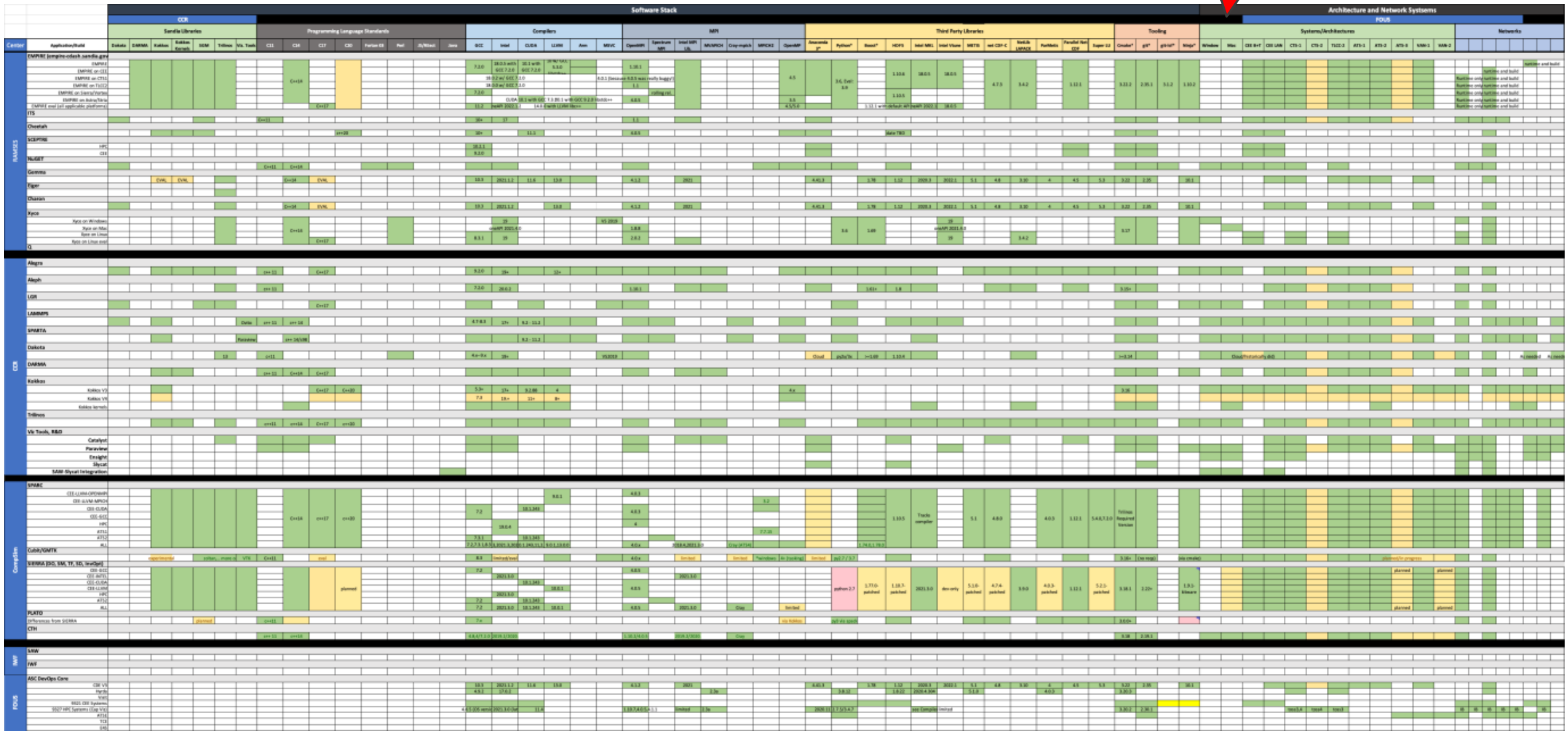
3-Year Roadmap



Managing Complexity

ASC DO Landscape – Motivating Down-selection of Tools and Utilities

High Impact!

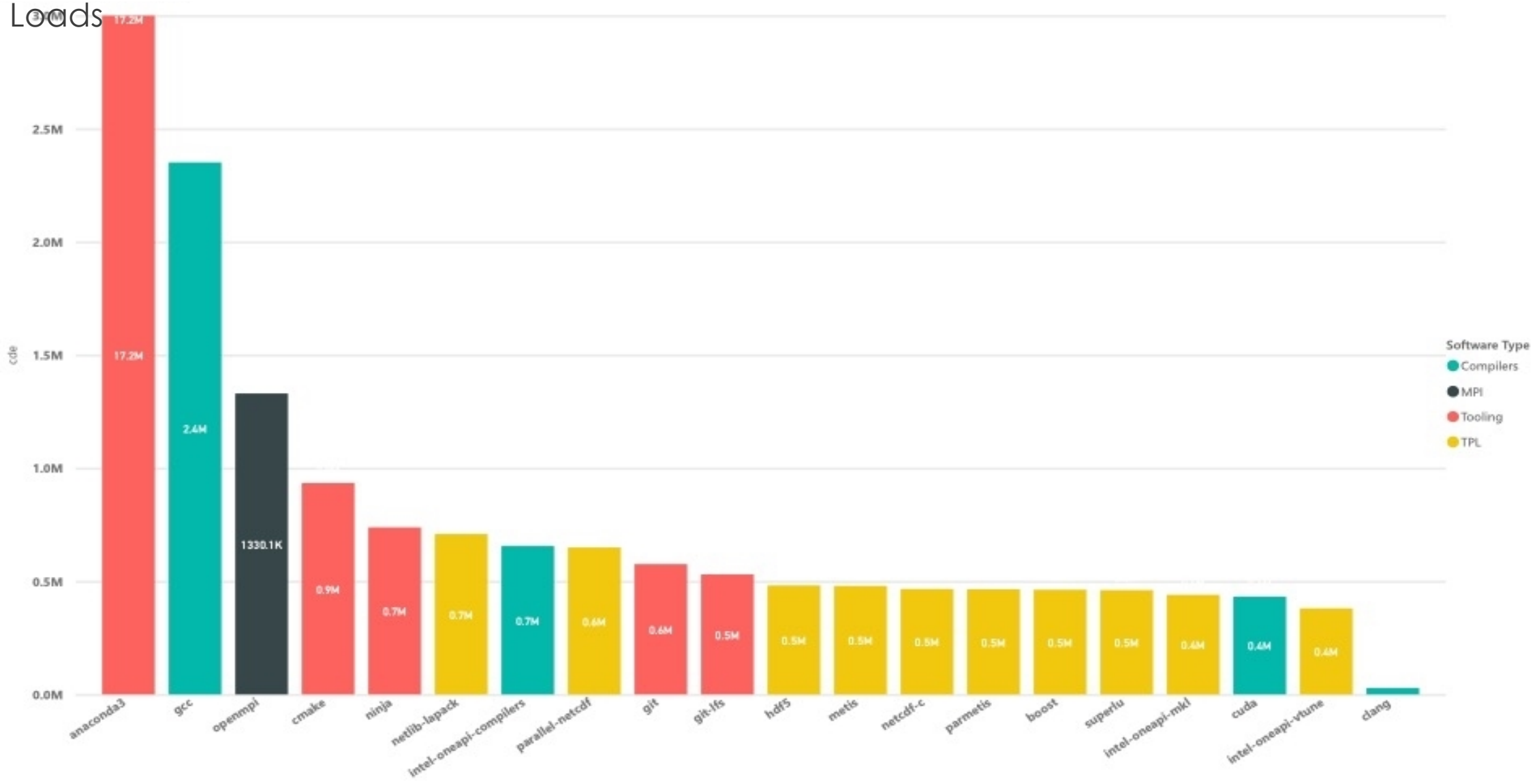


Minimizing Cost & Effort

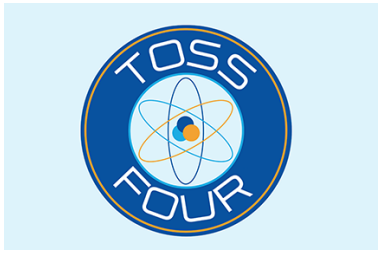
Solver Library (Trilinos) Builds: Motivating down-select of configurations

Solver Package	Configuration Options (27 out of 326)	alegra	aleph	charon	cubit_gui	dakota	empire	gemma	plato	sceptre	sierra	sparc	xyce	Conflict	Num. Of Code Teams Impacted
		Code Teams *(none of which have the same configuration)*													
Amesos2	Amesos2_ENABLE_Epetra	ON					ON					OFF		C	3
EpetraExt	EpetraExt_BUILD_BTF	OFF					ON						ON	C	3
Galeri	Trilinos_ENABLE_Galeri						ON				OFF			C	2
Intrepid2	Trilinos_ENABLE_Intrepid2						ON				ON	OFF		C	3
Isorropia	Trilinos_ENABLE_Isorropia	ON					ON				OFF		ON	C	4
Kokkos	KokkosKernels_ENABLE_SUPERNODAL_SPTRSV						ON				OFF			C	2
Kokkos	Kokkos_ENABLE_CUDA_UVM						ON					OFF		C	2
Kokkos	Kokkos_ENABLE_DEPRECATED_CODE_3						ON					OFF		C	2
Kokkos	Kokkos_ENABLE_OPENMP			OFF			ON				OFF			C	3
Kokkos	Trilinos_ENABLE_Kokkos				OFF		ON			ON	ON		ON	C	5
MueLu	Trilinos_ENABLE_MueLu	ON	OFF				ON			ON	ON	ON		C	6
NOX	NOX_ENABLE_LOCA	ON	OFF				ON						ON	C	4
Pamgen	Trilinos_ENABLE_Pamgen	ON					ON				ON	OFF		C	4
Panzer	Panzer_ENABLE_TESTS			OFF			ON							C	2
Panzer	Trilinos_ENABLE_Panzer						ON					OFF		C	2
SEACAS	Trilinos_ENABLE_SEACAS		ON				ON	ON		ON	OFF			C	5
STK	Trilinos_ENABLE_STK		ON				ON				OFF			C	3
STK	Trilinos_ENABLE_STKCoupling	ON					OFF					ON		C	3
STK	Trilinos_ENABLE_STKSimd						OFF					ON		C	2
Stokhos	Trilinos_ENABLE_Stokhos						ON				OFF	OFF	ON	C	4
Teuchos	Trilinos_ENABLE_TeuchosKokkosComm					OFF	ON							C	2
Teuchos	Trilinos_ENABLE_TeuchosKokkosCompat					OFF	ON							C	2
Tpetra	Tpetra_ENABLE_DEPRECATED_CODE						ON					OFF		C	2
Tpetra	Trilinos_ENABLE_Tpetra	ON	OFF				ON	ON	ON	ON	ON	ON		C	8
TrilinosCouplings	Trilinos_ENABLE_TrilinosCouplings	ON					ON				OFF		ON	C	4
Zoltan2	Trilinos_ENABLE_Zoltan2		OFF							ON	ON	ON		C	4

Building Towards Code Team Adoption



Effective DevOps Requires Collaboration Across NNSA



Programming Environments

- TOSS 4
- Spack
- Flux
- CSSE/FOUS Programming Environments

Remote Computing Enablement (RCE)

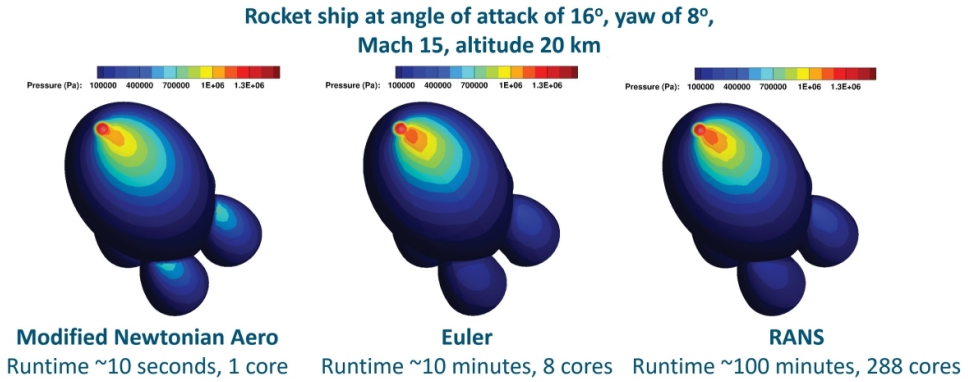
Increasing Collaboration

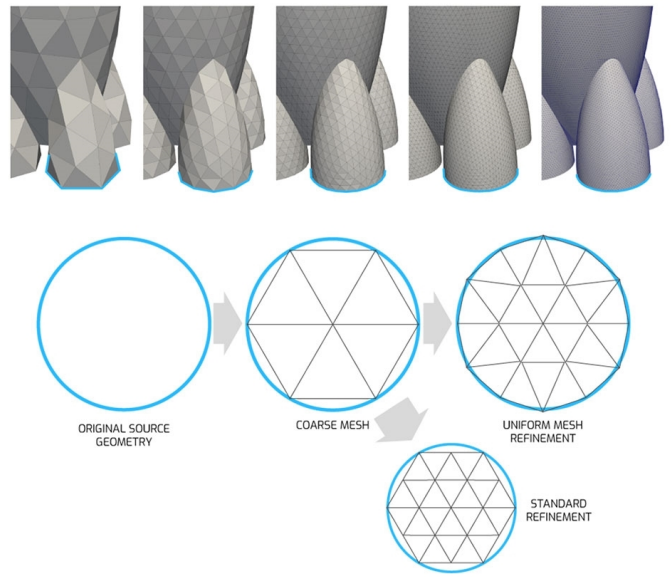
- Configuration Exchanges (sharing CDash and OpenShift configurations)
- Development working groups (Kitware GitLab Runners, CMake/CDash)
- ASC S3C + NLIT with special thanks to:
  - Mike Lang (NNSA)
  - Ben Santos (LANL)
  - Nicholas Jones (LANL)
  - Todd Heer (LLNL)
  - Cyrus Harrison (LLNL)
  - Todd Gamblin (LLNL).

What is Next: Raising the Bar

Evolving from a unified environment…

- Provide centralized DevOps personnel and services to help teams adopt and fully leverage common ecosystem
  - CMake
  - Complier upgrades
  - New services
  - Unified Environment Containers
- Common infrastructure across all code teams
  - Build system
  - Common test harness
  - Dashboards and metrics support
  - Data management plan
- Reduce build & test times
- Find economies of scale across code teams
- Standards, Best Practices, Policies

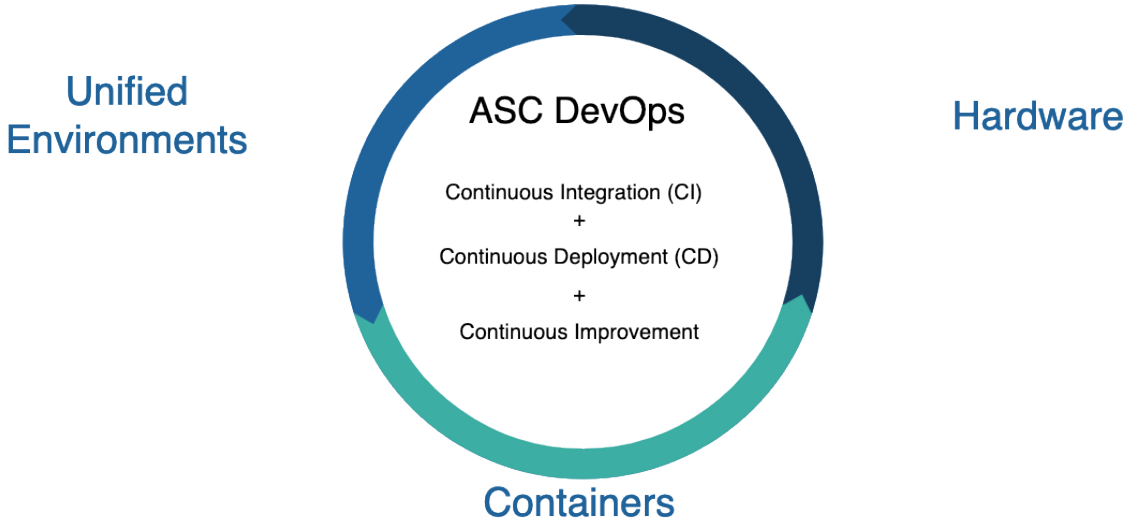




Advanced Simulation and Computing (ASC) DevOps Initiative

ASC DevOps Vision

ASC codes are developed, tested, deployed, and released quickly, efficiently, and robustly with established credibility on all key ASC platforms. Codes are efficiently coupled and integrated in the ASC DevOps ecosystem, which is governed by well-defined processes, practices, and standards. Ultimately, end-users will be able to seamlessly run recently released credible ASC code(s) on key ASC platforms.



Complicated Effort

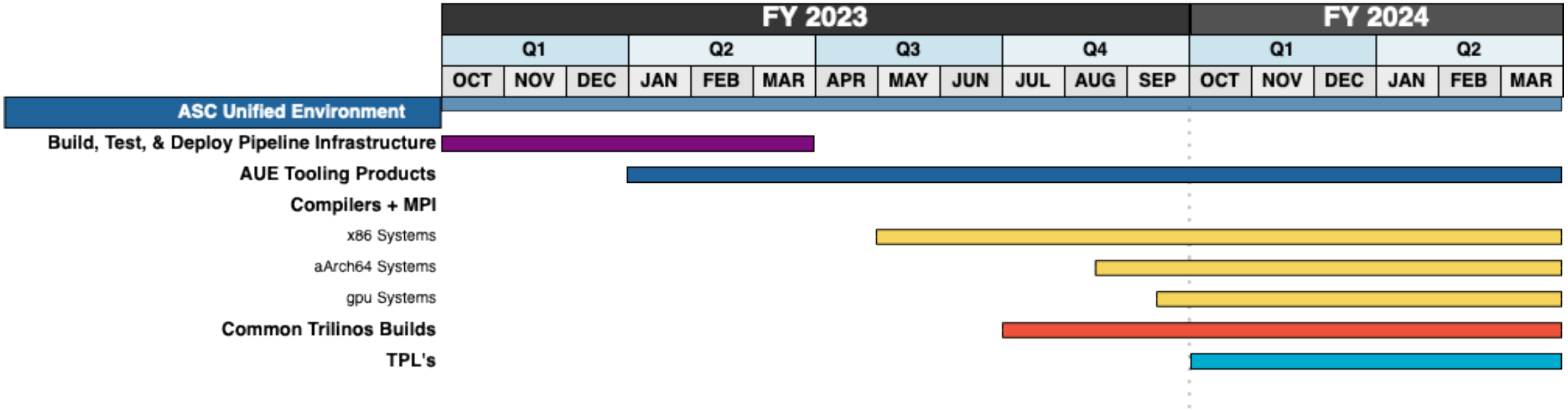
- Multiple prior attempts
- Required a new, different approach

Essentials to success

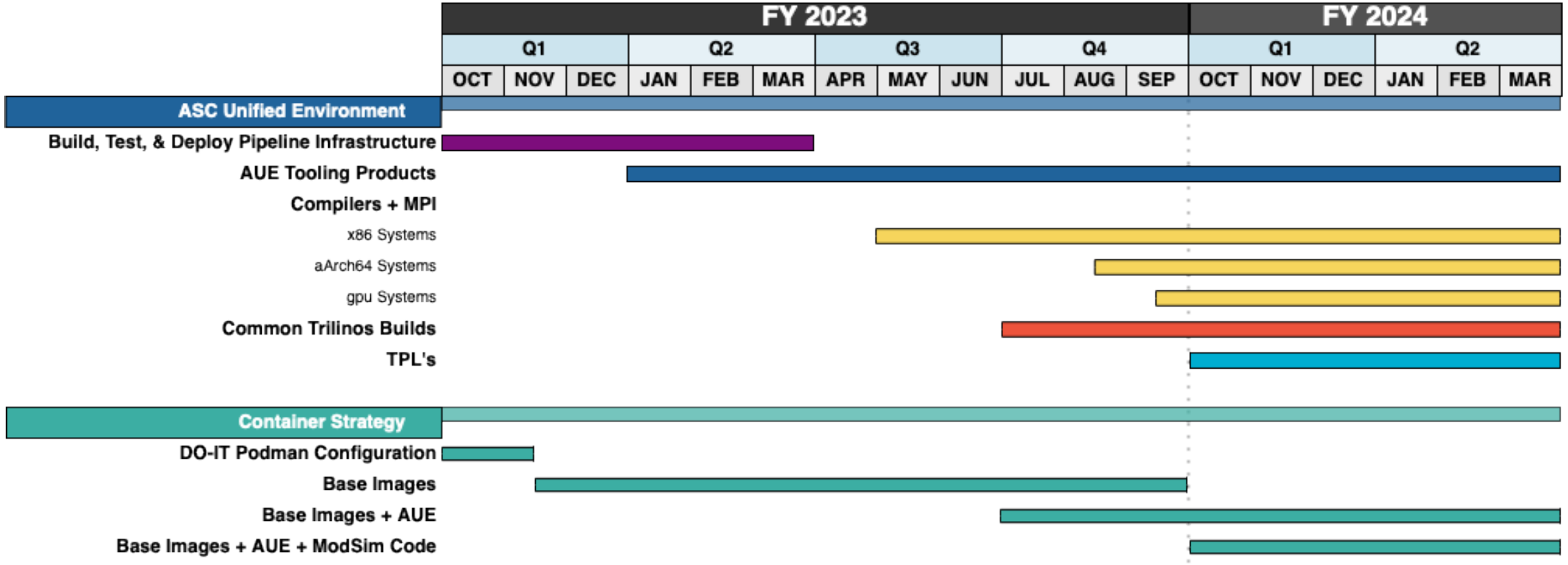
- Communication & Governance Structure
  - Broad consensus, from technical staff up to senior management
- Team Building & Managing Change
  - See Thorson's presentation Friday
- Managing complexity & Effort
  - Software ecosystem and library dependencies

Backup

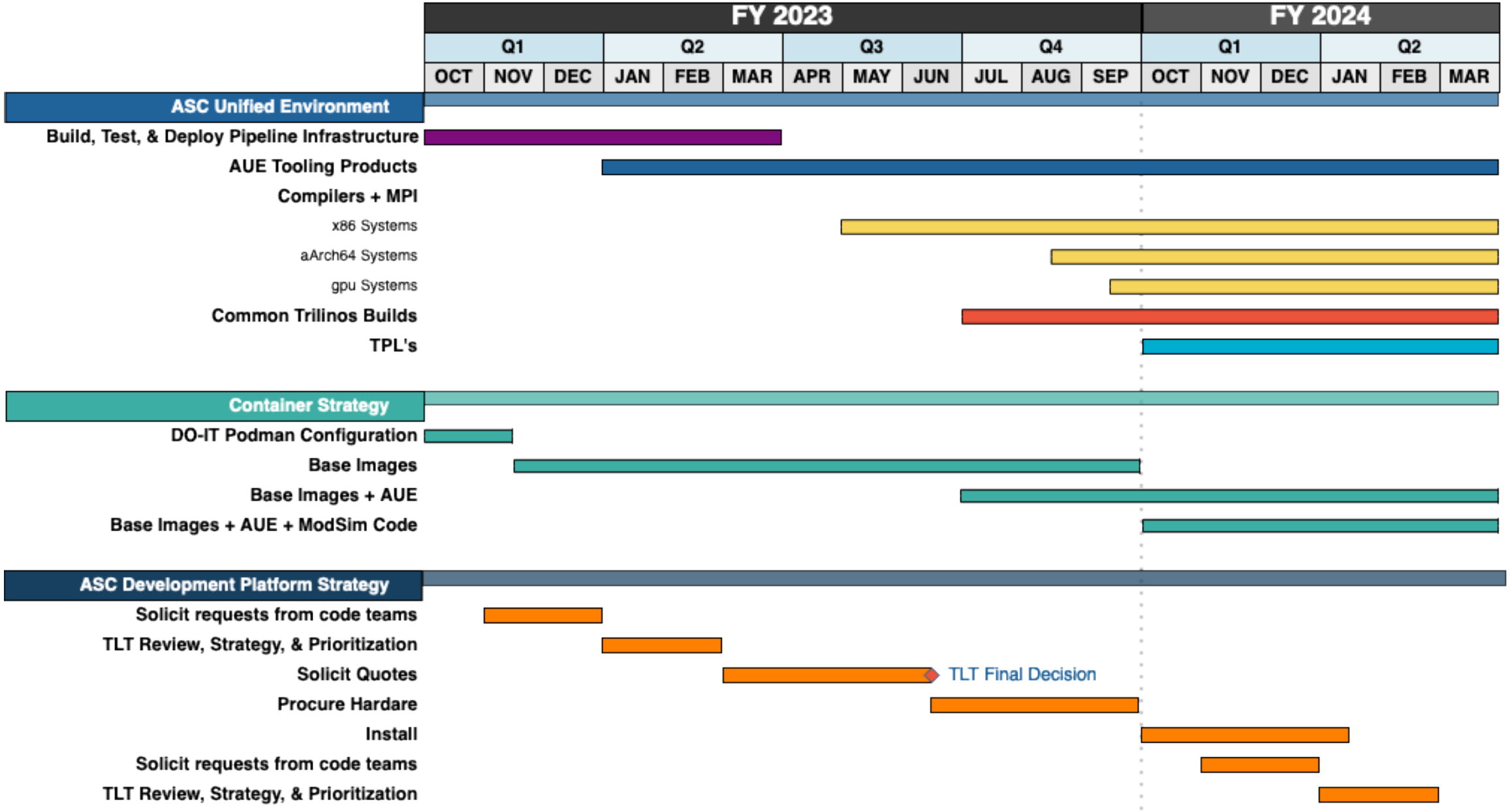
Unified Environment



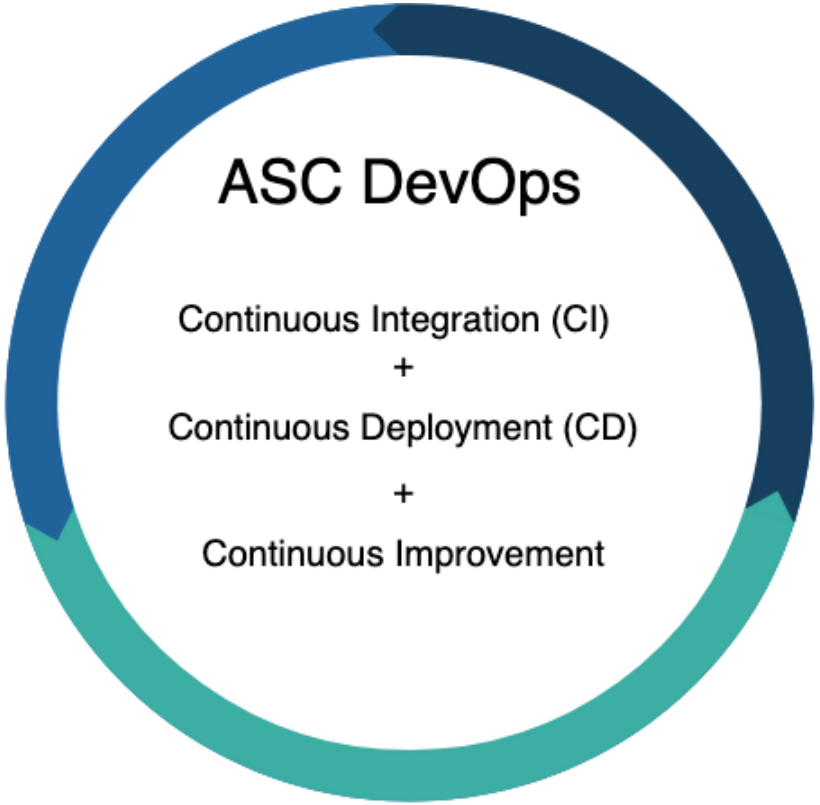
Containers


FY 2023
FY 2024
Q1
Q2
Q3
Q4
Q1
Q2
OCT
NOV
DEC
JAN
FEB
MAR
APR
MAY
JUN
JUL
AUG
SEP
OCT
NOV
DEC
JAN
FEB
MAR
ASC Unified Environment
Build, Test, & Deploy Pipeline Infrastructure
AUE Tooling Products
Compilers + MPI
x86 Systems
aArch64 Systems
gpu Systems
Common Trilinos Builds
TPL's
Container Strategy
DO-IT Podman Configuration
Base Images
Base Images + AUE
Base Images + AUE + ModSim Code

Hardware


FY 2023
FY 2024
Q1
Q2
Q3
Q4
Q1
Q2
OCT
NOV
DEC
JAN
FEB
MAR
APR
MAY
JUN
JUL
AUG
SEP
OCT
NOV
DEC
JAN
FEB
MAR
ASC Unified Environment
Build, Test, & Deploy Pipeline Infrastructure
AUE Tooling Products
Compilers + MPI
x86 Systems
aArch64 Systems
gpu Systems
Common Trilinos Builds
TPL's
Container Strategy
DO-IT Podman Configuration
Base Images
Base Images + AUE
Base Images + AUE + ModSim Code
ASC Development Platform Strategy
Solicit requests from code teams
TLT Review, Strategy, & Prioritization
Solicit Quotes
TLT Final Decision
Procure Hardare
Install
Solicit requests from code teams
TLT Review, Strategy, & Prioritization

DevOps for National Security High-Performance Computing



Unified Environments

Coordinated Development Teams

Long-Term, Inter-ModSim-Team Collaboration/Communication

Hardware

Unified Environments + Containers + System Hardware

Long-Term, Collaboration/Communication w/System Hardware Teams (e.g. CEE, HPC Systems, HAPPs)

Containers

Unified Environments + Software Systems

Long-Term, Collaboration/Communication w/System Software Teams (e.g. TOSS, ATSE, ATS)