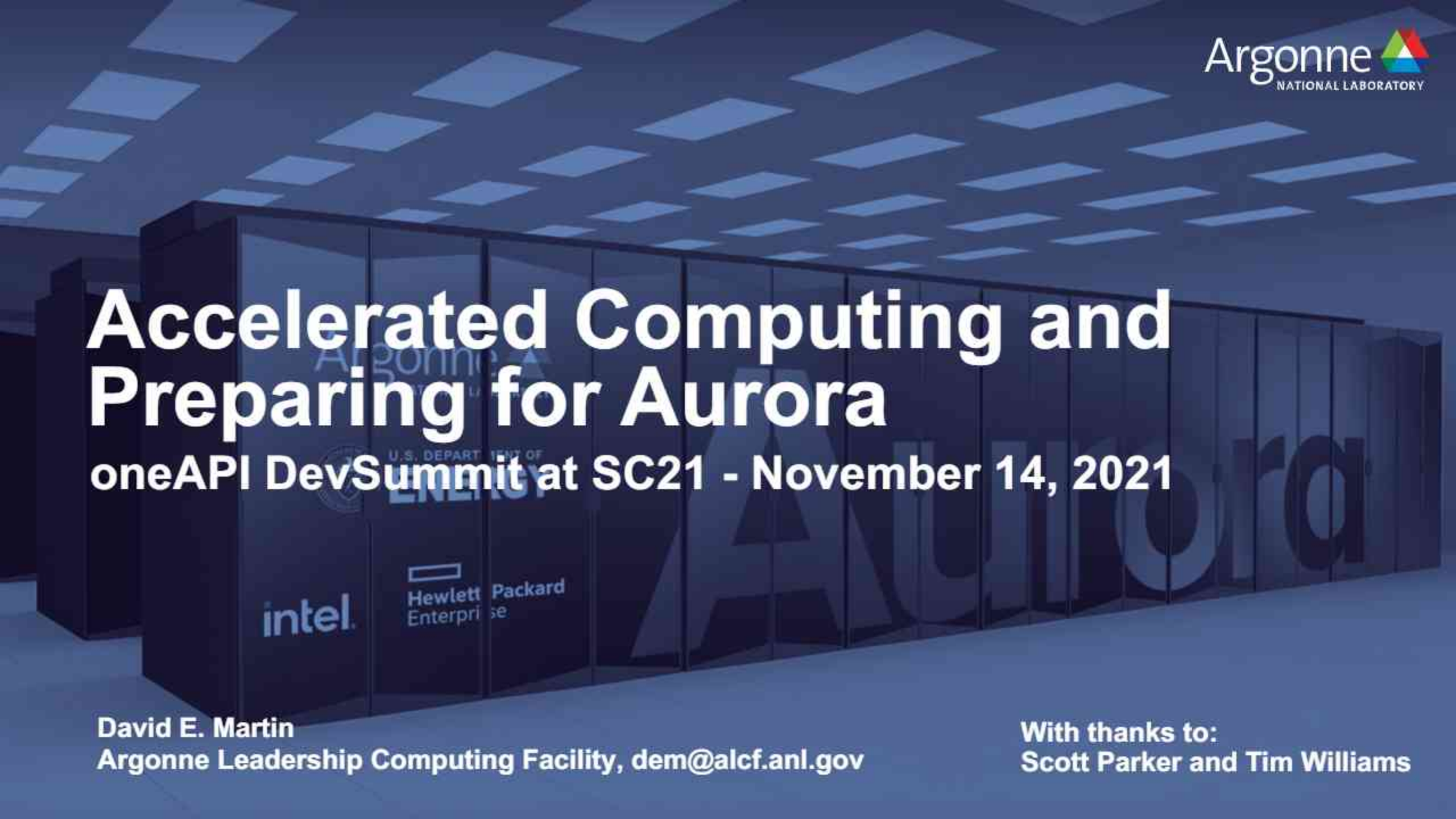


Accelerated Computing and Preparing for Aurora

oneAPI DevSummit at SC21 - November 14, 2021

David E. Martin
Argonne Leadership Computing Facility, dem@alcf.anl.gov

With thanks to:
Scott Parker and Tim Williams

About Argonne

Argonne is a multidisciplinary science and engineering research center located outside Chicago.

- Born out of the University of Chicago's work on the Manhattan Project in the 1940s.
- Managed by UChicago Argonne, LLC, for the U.S. Department of Energy's Office of Science.
- Works with universities, industry, and other national labs on questions and experiments too large for any one institution to do by itself.

Our one-of-a-kind facilities enable science from the nanoscale to the exascale

Argonne's five flagship facilities support one of the largest user communities in the U.S. Department of Energy complex.

Advanced Photon Source

Argonne Tandem Linear Accelerator System

Argonne Leadership Computing Facility

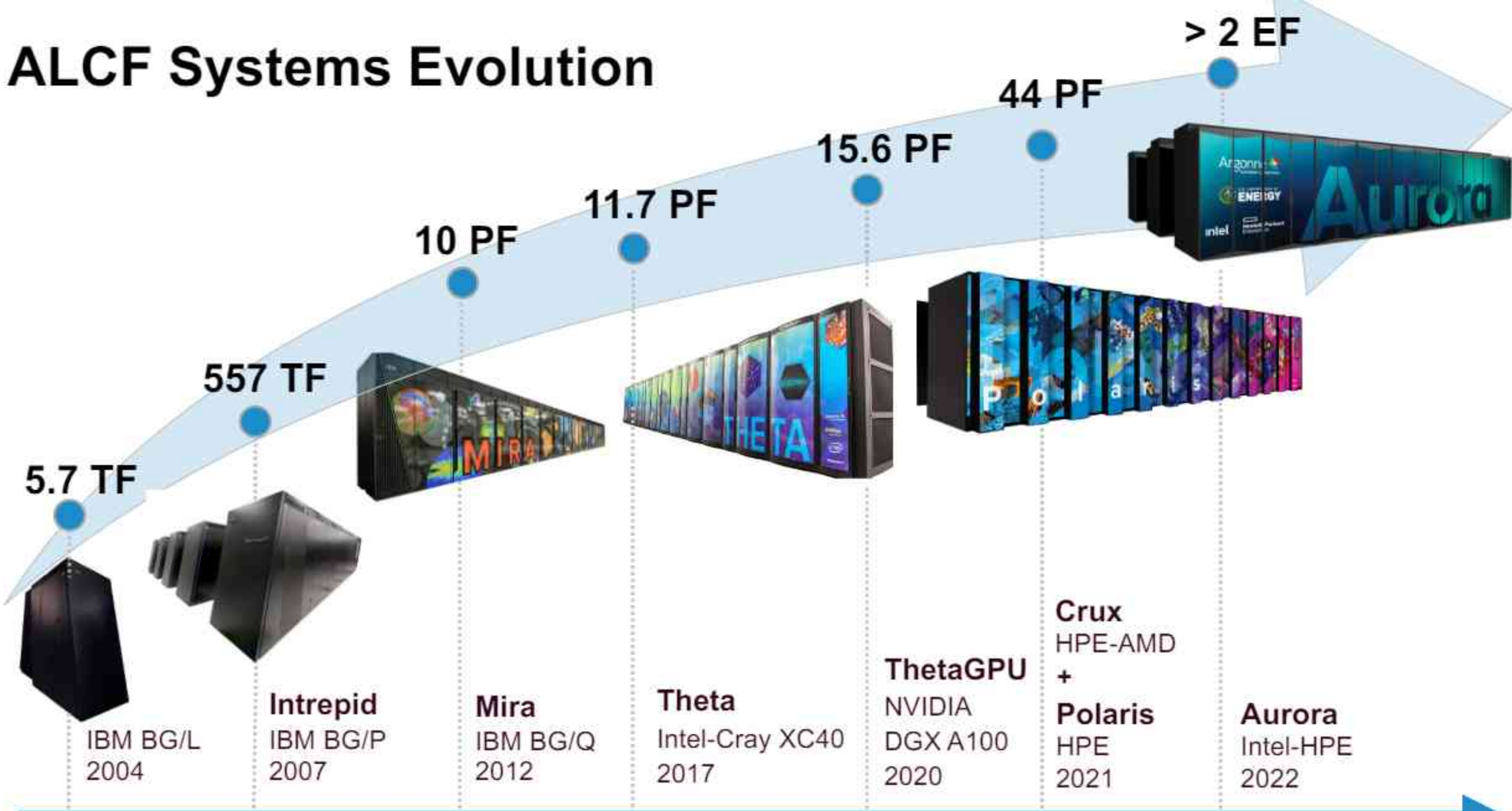
Center for Nanoscale Materials

Atmospheric Radiation Measurement – The Southern Great Plains

DOE Leadership Computing Facility

- Established in 2004 as a collaborative, multi-lab initiative funded by DOE's ***Advanced Scientific Computing Research*** program
- Operates as **one facility** with two centers, at Argonne and at Oak Ridge National Laboratory
- Deploys and operates at least two advanced architectures that are **10-100 times more powerful** than systems typically available for open scientific research
- **Fully dedicated** to open science to address the ever-growing needs of the scientific community

ALCF Systems Evolution



DOE HPC Road to Exascale Systems

FY 2012	FY 2016	FY 2018	FY 2021	FY 2022	FY 2023

decommissioned

ORNL
Cray/AMD/NVIDIA

Argonne
IBM BG/Q

ANL
Cray/Intel KNL

LBNL
Cray/Intel Xeon/KNL

LLNL
IBM BG/Q

LANL/SNL
Cray/Intel Xeon/KNL

ORNL
IBM/NVIDIA

LLNL
IBM/NVIDIA

ORNL
HPE/AMD

Exascale Systems

ANL
HPE/AMD/NVIDIA

ANL
Intel/HPE

LBNL
HPE/AMD/NVIDIA

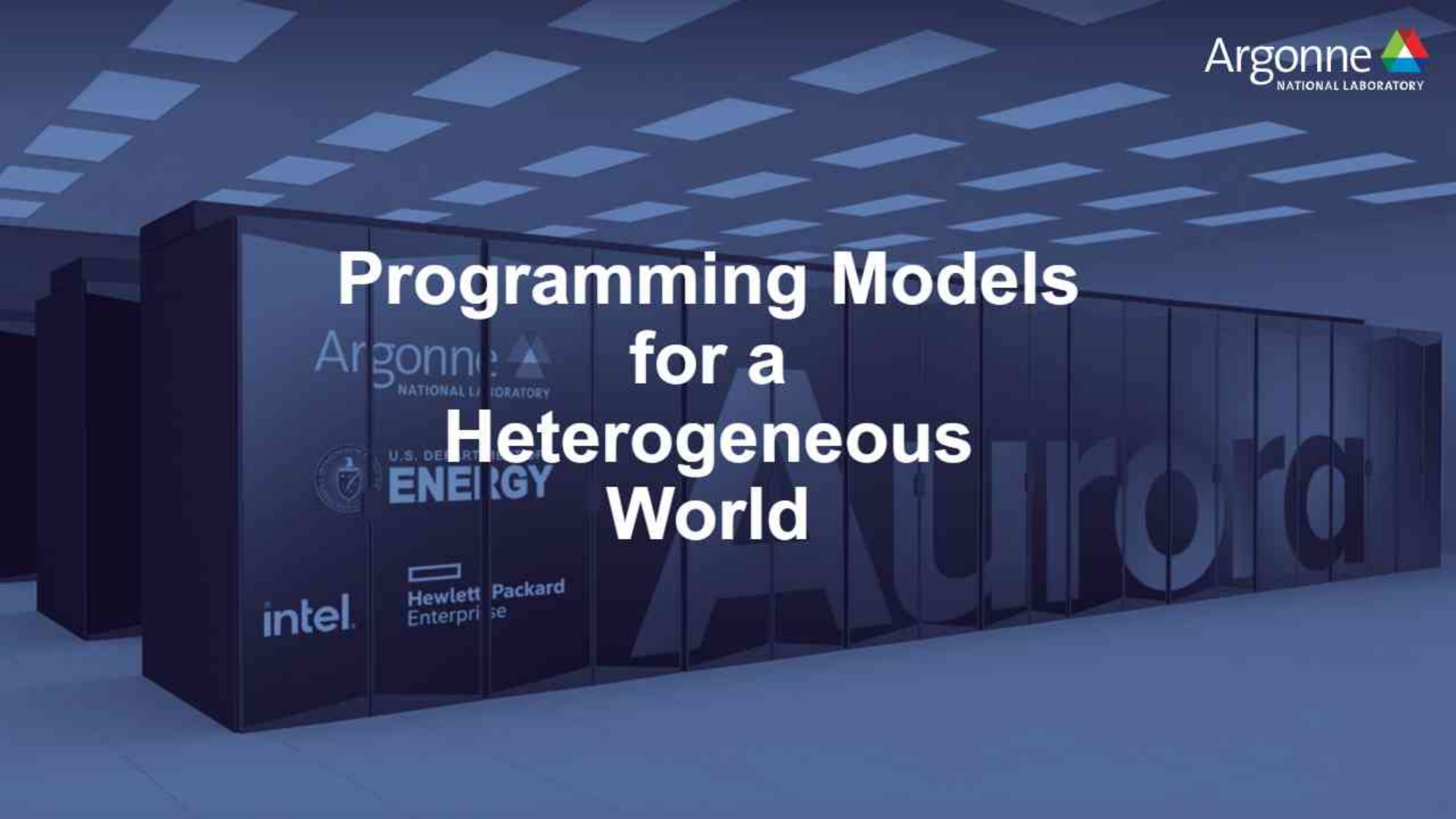
LANL/SNL
HPE/Intel

LLNL
HPE/AMD

Pre-Exascale and Exascale Landscape

System	Delivery	CPU + Accelerator Vendor
Summit	2018	IBM + NVIDIA
Polaris	2021	AMD + NVIDIA
Perlmutter	2021	AMD + NVIDIA
Frontier	2021	AMD + AMD
Aurora	**2022**	**Intel + Intel**
El Capitan	2023	AMD + AMD

- Heterogenous Computing (CPU + Accelerator)
- Varying vendors


Argonne
NATIONAL LABORATORY
Programming Models
for a
Heterogeneous
World
Aurora
intel
Hewlett Packard
Enterprise

Heterogenous System Programming Models

- Applications will be using a variety of programming models for Exascale:
  - —CUDA
  - —OpenCL
  - —HIP
  - —OpenACC
  - —OpenMP
  - —DPC++/SYCL
  - —Kokkos, Raja
  - —TensorFlow, PyTorch, Horovod
- Not all systems will support all models.
- Libraries may help you abstract some programming models.

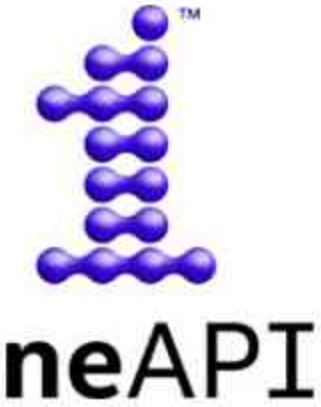
Paths to Portability across Heterogeneous Systems

- Optimized, platform-specific implementations of key kernels
  - —Conditional compilation
- Standard language-level portability
  - —C++17 parallel algorithms
  - —Fortran do concurrent
  - —OpenMP 4.5+
- Lower-level portable programming models
  - —HIP
  - —SYCL/DPC++
- Higher-level portable programming models
  - —Kokkos
  - —TensorFlow, PyTorch, Horovod
- Libraries
  - —PETSc
  - —AMREx
  - —TAMM
  - —gtensor
  - —….

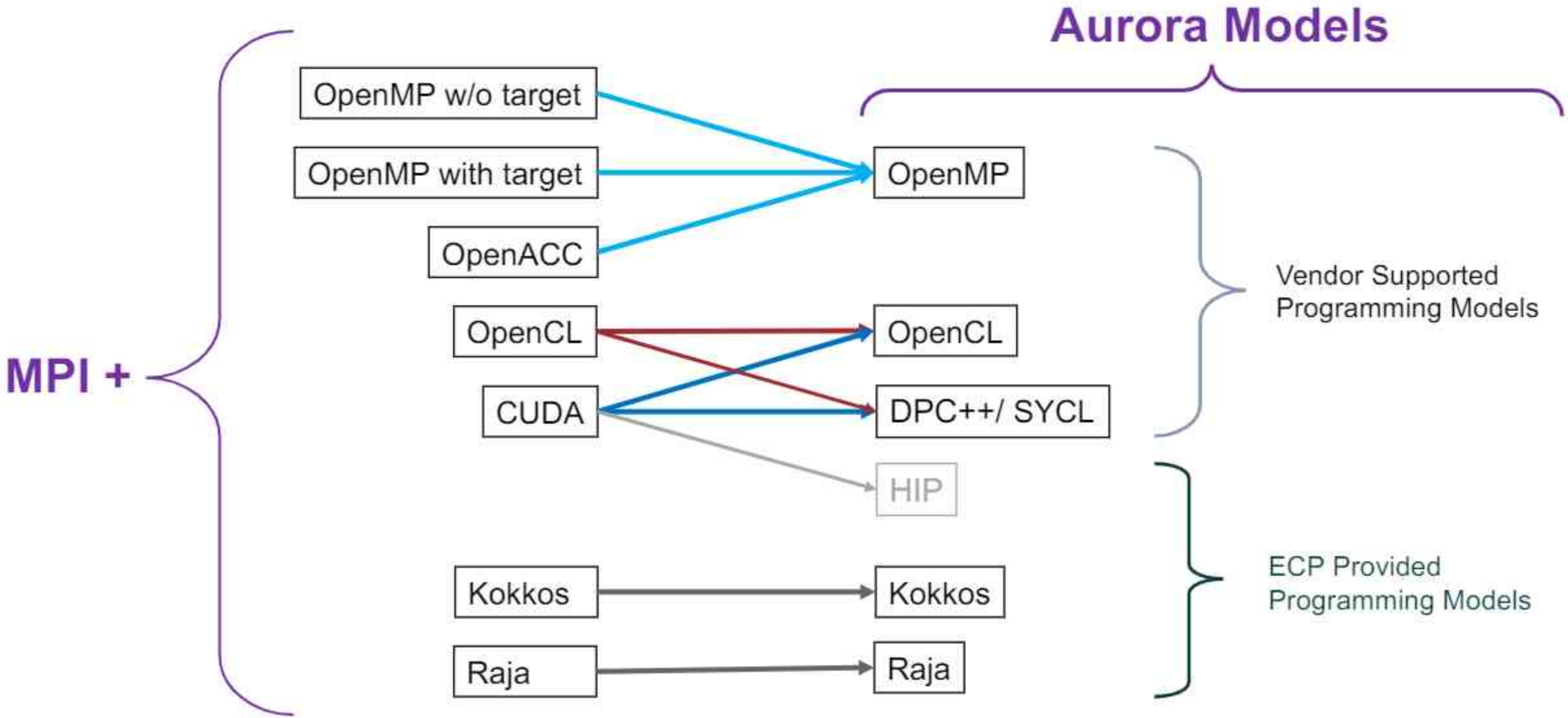
Programming Models and Frameworks

Aurora applications may use:

- ~~CUDA~~
- OpenCL
- HIP
- ~~OpenACC~~
- OpenMP
- DPC++/SYCL
- Kokkos
- Raja



Mapping of Existing Programming Models to Aurora



Aurora Software Stack

Languages

- Fortran (w/ OpenMP 5)
- C/C++ (w/ OpenMP 5)
- DPC++
- Python

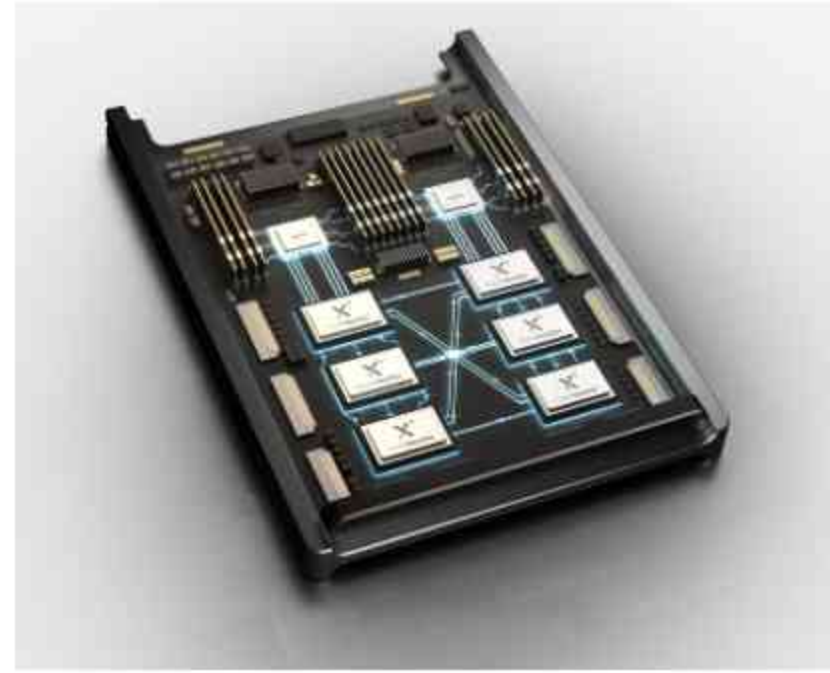
Libraries

- oneAPI MKL (oneMKL)
- oneAPI Deep Neural Network Library (oneDNN)
- oneAPI Data Analytics Library (oneDAL)
- MPI

Tools

- Intel Advisor
- Intel VTune
- Intel Inspector

oneAPI



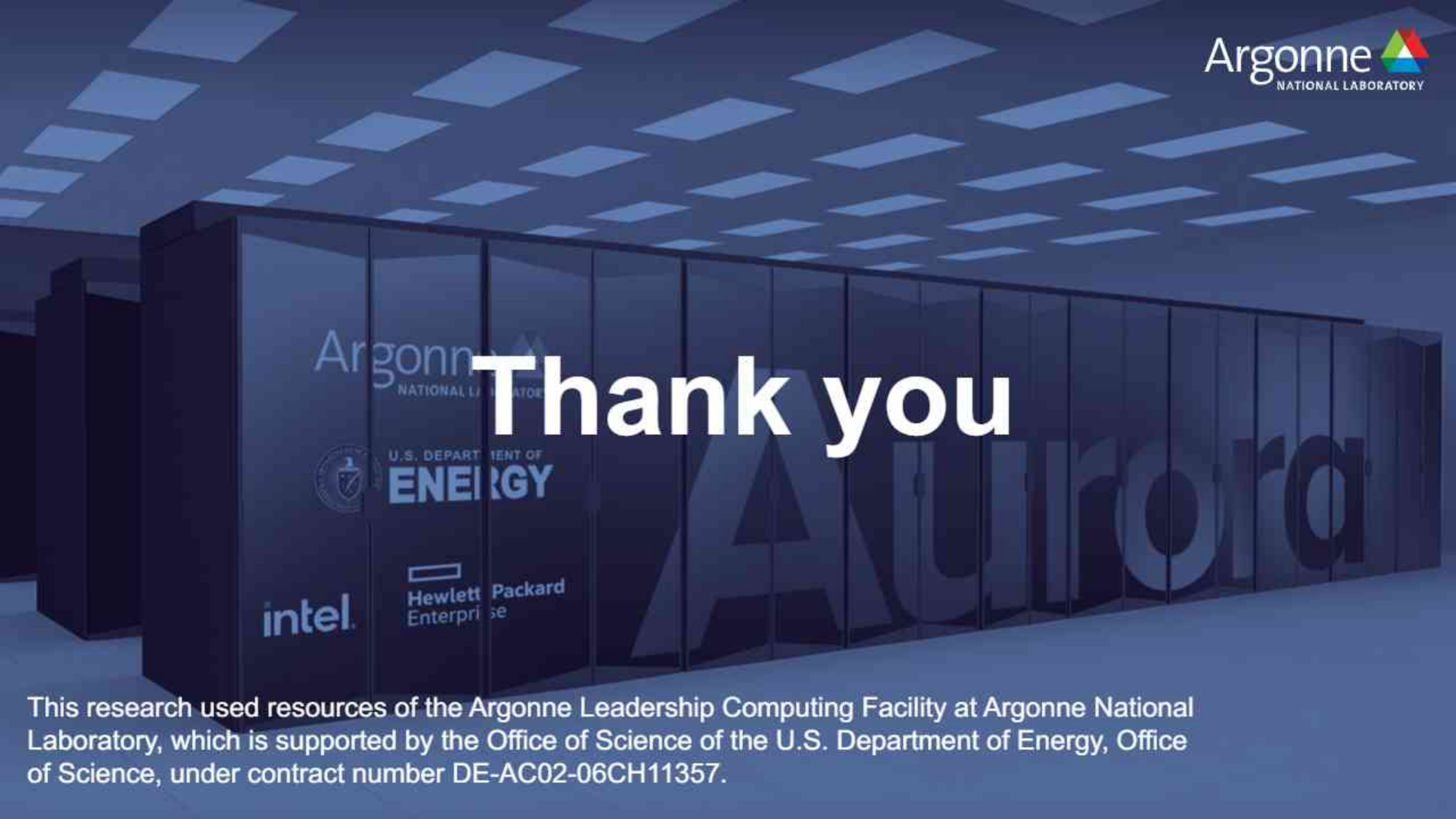
Best Practices

- Consider GPU acceleration generically
  - —May need code refactoring, different algorithms
  - —Do this work on present-day GPU systems
- Consider high-level portability layers
  - —Canned (Kokkos, e.g.)
  - —Roll your own (DSL, e.g.)
- For future systems such as Aurora
  - —Work with closest previous-generation GPU accelerators
  - —Work with beta developer software
- Work with the portability-layer developers
- Do not optimize prematurely
  - —Beta software and previous-generation hardware may optimize differently than final targets
  - —When you do start profiling, use a physically realistic problem

It is easy to add platform-specific code, such as memory allocators or warp sizes. Build and test on other platforms to identify these sooner rather than later.

Ways to Experiment

- Many recent PCs have Intel GPU hardware that supports oneAPI
  - —Integrated GEN9 or higher GPUs including latest Intel® Iris® Xe MAX graphics
  - —Download oneAPI Base Toolkit
- Intel DevCloud for oneAPI
  - —Cloud-based sandbox for prototyping and experimenting
  - —Integrated GPU or Gen9 GPU, Xeon CPU
  - —Full oneAPI software stack
- Argonne Joint Laboratory for System Evaluation
  - —Early releases of Aurora hardware
  - —Early releases of oneAPI software
  - —Restricted to Argonne, early science and ECP projects



This research used resources of the Argonne Leadership Computing Facility at Argonne National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy, Office of Science, under contract number DE-AC02-06CH11357.