



**Hewlett Packard  
Enterprise**



# **HPE Cray Programming Environment Training**

Barbara Chapman  
Soumitra Chatterjee

May 7, 2023

# **Training update**

- Chris Marsh is taking a sabbatical
- He'll return next June
- New trainer is coming up to speed





**Hewlett Packard**  
Enterprise

# **H8PG3S.A**

# **HPE Cray EX Programming & Optimization**

---

Compilers in the HPE Cray Programming Environment

Revision 22.11

# LEARNING OBJECTIVES

---

- After completing this module, you should be able to:
  - List the components of the HPE Cray Compiling Environment (CCE) and their current versions
  - Describe general compiler optimization techniques including:
    - Unrolling loops
    - Vectorization
    - Inlining
    - Whole Program Compilation
    - Floating Point Optimization
  - Use Clang compiler options to optimize a C or C++ program
  - Use HPE Cray compiler options to optimize a Fortran program
  - Compare basic compiler options on available compilers in the CCE



# CPEaaS

- Virtualized interface to the HPE/Cray Development Environment
  - Virtualized access to physical system(s)
  - Managed credentials
- Persistent workspace management
- Execution environment via PaaS

- Simplified selection of build tools
  - Compiler & version, platform
  - Target platform, capabilities, etc.
- Simplified build parameters
  - Debug, optimization, language features, sanitizers, etc.

The screenshot displays the CPEaaS interface with the following components:

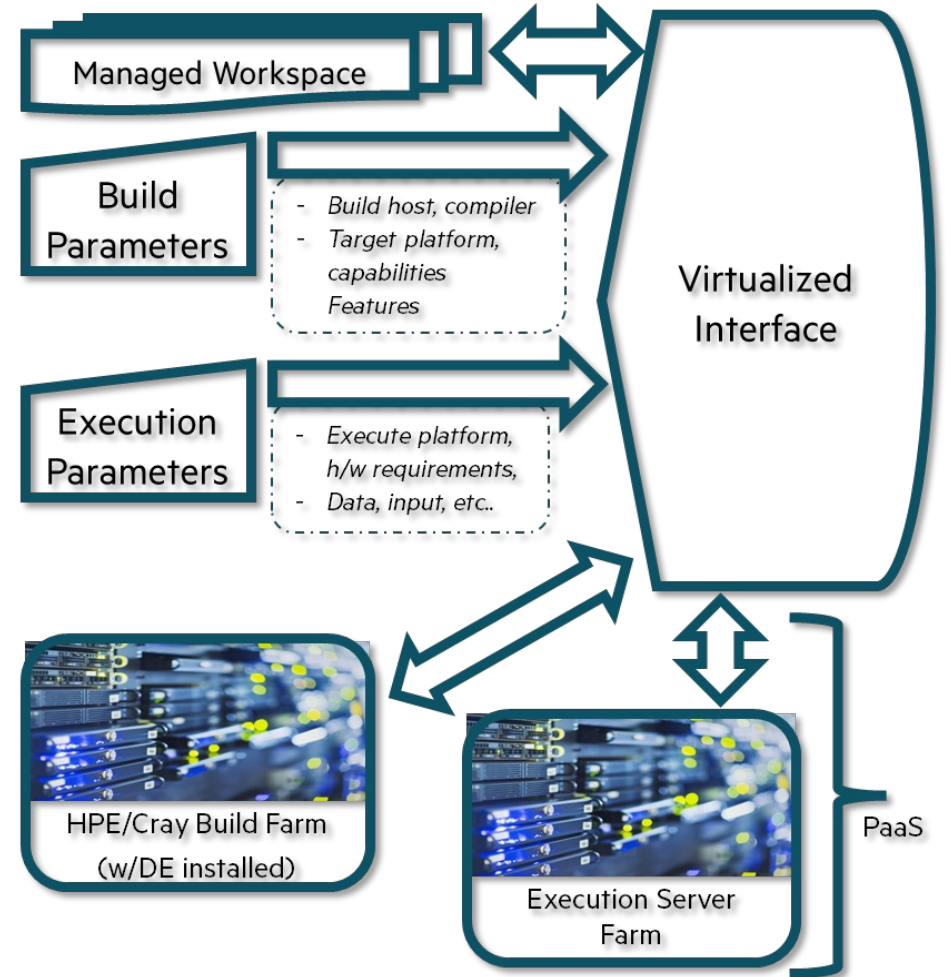
- Workspace:** A file tree showing two folders, 'folderA' and 'folderB'. 'folderA' contains files 'fileA1.c', 'fileA1.h', 'fileA2.c', and 'fileA2.h'. 'folderB' contains 'fileB1.c' and 'fileB1.h'. A 'makefile' icon is also present. Below the tree is a button 'Add source file Browse...' and the text 'No file selected.'
- Target Definition:** Three dropdown menus for 'Select CPU...', 'Select GPU...', and 'Select Operating System...'.
- Host Definition:** Two dropdown menus for 'Select platform...' and 'Select compiler...'.
- Compiler Options:** A large empty text input field.
- Features:** A list of radio buttons for 'OpenMP', 'AddressSanitizer', 'ThreadSanitizer', 'Debug', and 'Optimization Level' (with a dropdown arrow).
- Buildarea:** A terminal window showing a 'make' command execution log. Below it is a file tree with 'myProgram.elf' and 'folderA' containing 'fileA1.o', 'fileA1.d', 'fileA2.o', and 'fileA2.d'.
- Execution Definition:** A text area containing the command: `./myProgram.elf --arch=`uname -m` \ --name=`uname -n` \ -d `date` \ > run_output.log`
- Buttons:** A large blue button 'Build workspace »' at the bottom center and a smaller blue button 'Execute »' at the bottom right.

# DEaaS

## Features

- Academic introduction to development environment
  - Build systems managed in the backend
- Wide selection of build hosts, tools & target platforms
  - Experiential learning for academics
  - Evaluation of migration & portability across platforms and compilers
  - Seamless triage & compiler capability verification
- Managed workspace
  - Integration with SCM such as git, etc.
  - Persistent across sessions & build configurations
- Seamless availability of bleeding edge DE software components

## Architecture



**Questions?**



# Thank you

---

[barbara.chapman@hpe.com](mailto:barbara.chapman@hpe.com)





# OLCF Training Overview

CUG '23 PEAD

Chris Fuson

National Center for Computational Sciences

Oak Ridge Leadership Computing Facility

May 07, 2023

ORNL is managed by UT-Battelle LLC for the US Department of Energy

# OLCF Training Overview

- Educate user community on topics needed to fully leverage HPC resources
- Targets existing, new, future users
- Resources in both moderate and open security enclaves
- Workshops, webinars, tutorials, seminars, and hackathons
- Average ~2 events per month
  - [https://docs.olcf.ornl.gov/training/training\\_archive.html](https://docs.olcf.ornl.gov/training/training_archive.html)
- Work with vendors and other labs

# Training Examples

## OLCF user calls

- HPSS overview
- New User training
- Machine Learning for HPC Simulations
- MYOLCF User Porthole
- Automating Science with Workflows
- Frontier Announcement
- Remote Visualization with VNC
- Andes Overview
- Introduction to Julia

## Collaborative Training

- Coding for GPUs Using Standard C++
- Coding for GPUs Using Standard Fortran
- User R on HPC Clusters
- Nsight Profiling workshop

## Preparing for Frontier Series

- Introduction to the Frontier Supercomputer
- Introduction to HIP Programming
- HIP for CUDA Programmers
- OpenMP Offload Basics
- OpenMP Offload Data movement and optimization

## Data Visualization and Analytics Series

- Vampir and Score-P
- Jupyter
- Paraview
- Visit

## Community Development

- HPC Crash Course for UTK Students
- Winter Classic Hands-on Summit
- ECP OpenMP SOLLVE hackathon
- Summer Virtual HPC Crash Course
- HPC Crash course for MSI Faculty
- HPC Crash course at Tapia
- HPC Crash course at SIAM

# Training Lessons Learned

- Training can offer a good collaboration opportunity
  - Shared knowledge, effort, user bases
  - Hackathon mentors
- Modules on git
  - Ability to work outside of training, access after resource access ends
  - For trainings with a hands-on component
- Virtual / In-person
  - Virtual allows more to access / increased attendance; hybrid difficult
  - Recording event helpful for those who can not attend
    - <https://vimeo.com/channels/olcftraining>
    - Shorter better, more views than long video
    - Series better than long single training

# PEAD BoF – User Training CUG 2023

Dr Marco De La Pierre

Pawsey Supercomputing Research Centre, Australia



# User Training at Pawsey: Main Pillars

- Focus on new vendor and hardware architecture
  - AMD silicon
  - Heterogeneous system with significant GPU component
- Upskilling both researchers and HPC centre staff
  - The latter enables long-term advanced user support
- Evidence based approach
  - Informed by repeated user surveys to identify gaps
  - Both prior to migration and ongoing (at production)



# Phases of User/Staff Training

- Before and during migration

- Targets skills building
- Staff-focused\*: Cyber-security, Spack, ADIOS, Cray PE, AMD profiling
- User-focused\*: Slurm, Object Storage, Containers
- Dev-focused\*: Hipification, Codee, ADIOS, Cray PE, AMD profiling, Codee

- At production

- Repeats of basic training to onboard new users
- “Driving tests” for new users
  - Check on best practices and policies
- Hands-on mentored sprints
  - Tackle real life researchers’ challenges
  - Very popular amongst user community
- PaCER Conference (P’Con)
  - Targeting top tier research projects with dev focus
  - Knowledge sharing
  - Vendor talks
  - Mini-hackathons

\* not an exhaustive list

# Additional Aspects of Interest

- Accessibility of training
  - In person, virtual-only, hybrid training events
  - Recordings made available publicly in most cases
  - Modularity (chunking for “digestibility”)
- Smart sourcing of training materials
  - Pawsey staff
  - Pawsey internship students (Carpentry-style)
  - 3<sup>rd</sup> Party (vendors, consulting companies)
- Bespoke training sessions
  - E.g. domain specific editions of standard training (containers as a representative example)





# Made possible by ...

- Training Team
  - Ann Backhaus
  - Fathima Haseen
  - Kathryn Southall
- Events and Marketing Team
  - Karina Nunez
  - Aditi Subramanya
- The wider Services Team
- and indeed the whole Team Pawsey!



Who is Pawsey?

# Ookami Testbed - a Cray Apollo80 system



Node / System	
Processor	<b>Fujitsu A64FX</b>
#Nodes	176
Memory	32GB@1TB/s
Comms	IB HDR-100
Scheduler	Slurm
#Modules	> 300

1.5 mil node hours per year

# Ookami - User Training



- A lot of user do not read documentation
- Multi-modal approach:
  - Slack channel
  - Office hours
  - Webinars
    - Recordings and materials online
  - Documentation and FAQ - [www.stonybrook.edu/ookami](http://www.stonybrook.edu/ookami)