

Parallelware Training Series

Motif-guided Parallelization of ZPIC with OpenMP and OpenACC

Welcome and introductions

Manuel Arenaz | Oct-Nov 2020

©Appentra Solutions S.L.



Why are we organizing this course at NERSC?

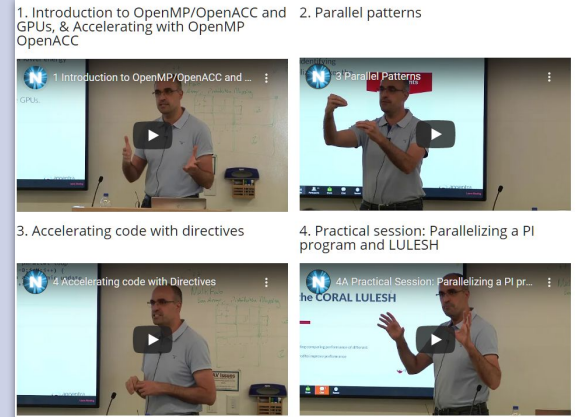
- NERSC mission is to accelerate scientific discovery at the DOE Office of Science through high performance computing and data analysis.
- Computing is a tool as vital as experimentation and theory in solving the scientific challenges.
- [Cori supercomputer](#): Cray XC40 comprised of 2,388 Intel CPUs running a Cray Linux environment.
- Upcoming [Perlmutter supercomputer](#): Cray pre-exascale system with AMD EYPC CPUs and NVIDIA GPUs.
- NERSC is organizing activities to facilitate the transition to Perlmutter, including helping NERSC users to port their codes to GPUs so that they benefit from the huge computational power of these accelerators.
- Appentra and NERSC have been organizing innovative training activities since 2019.
 - Parallelware Tool Workshop, June 6, 2019
www.nersc.gov/users/training/events/parallelware-tool-workshop-june-6-2019/
 - Parallelware Tool Workshop, October 17, 2019
<https://www.nersc.gov/users/training/events/parallelware-tool-workshop-october-17/>

Parallelware Workshop (Jun 2019) - Introductory

www.nersc.gov/users/training/events/parallelware-tool-workshop-june-6-2019/

8:45 - 9:00	Welcome and introductions
9:00 - 9:30	Lecture 1: An introduction to OpenMP/OpenACC and GPUs
9:30 - 10:15	Lecture 2: Accelerating with OpenMP/OpenACC
10:15 - 10:30	Break
10:30 - 11:00	Lecture 3: Parallel Patterns
11:00 - 11:30	Lecture 4: Accelerating code: Directives
11:30 - 12:00	Practical 4A: Parallelizing the calculation of Pi
12:00 - 13:00	Working lunch (hands-on activities)
13:00 - 17:00	Hands-on time with your code
17:00	Close

Watch official videos of the training session in YouTube



[Click to watch](#)

Parallelware Workshop (Oct 2019) - Intermediate

www.nersc.gov/users/training/events/parallelware-tool-workshop-october-17/

8:15 - 8:45	<i>Morning refreshment and coffee</i>
8:45 - 9:00	<i>Welcome and introductions</i>
9:00 - 9:30	Lecture 1: An introduction to OpenMP/OpenACC optimizations for CPUs/GPUs
9:30 - 10:15	Lecture 2: A wider set of code patterns: compute patterns, memory patterns and flow patterns
10:15 - 10:30	<i>Break</i>
10:30 - 11:00	Lecture 3: Minimizing data transfers
11:00 - 11:30	Lecture 4: Optimizing memory usage
11:30 - 12:00	Lecture 5: Exploiting massive parallelism
12:00 - 13:00	<i>Working lunch (hands-on activities)</i>
13:00 - 14:00	Practical 6A-6B: Parallelizing the calculation of HEAT and MATMUL
14:00 - 17:00	Hands-on time with your code and LULESHmk , inspired in the CORAL benchmark LULESH
17:00 pm	<i>Close</i>

Parallelware Training Series 2020: Learning outcomes

- **Leverage the learning outcomes of the Parallelware training series 2019**

- Learn a practical step-by-step approach to parallelization based on code patterns.
- First, pattern-based analysis of the source code to decide how to parallelize it.
 - Understanding of the code as a whole, not just the hotspots isolatedly.
 - Set of patterns: computation patterns, memory patterns and flow patterns.
- Second, pattern-guided implementation of parallel versions using OpenMP/OpenACC for CPU/GPU.
 - Learn how to minimize data transfers, optimize memory usage and exploit massive parallelism.
- Overall, learn best practices for parallel programming using OpenMP/OpenACC for CPUs and GPUs.

- **Understand the complexity of porting C/C++/Fortran applications to GPUs**

- The importance of writing the source code with parallelism in mind from the start.
- Realize the benefits of investing effort in the preparation of the code (e.g. avoid Array-of-Structs)

- **Consult the public catalog of defects and recommendations for parallelism for C/C++/Fortran**

- Learn common parallel errors and how to fix them, including data races and data movement issues
- Website: www.appentra.com/knowledge/checks/

- **Ensure parallel programming best practices in your code using Parallelware Analyzer and Trainer**

- Automate source code checking to reduce software development effort and increase code quality.
- Use case: ZPIC educational code based on the Particle-in-Cell method.

Agenda

www.nersc.gov/users/training/events/parallelware-training-series-oct-nov-2020/

Part 1, Tuesday, October 27

8:30 am - 12:00 pm PDT

Introduction to Parallelware tools: Ensuring parallel programming best practices

- 10' - Introduction by NERSC
- 10' - Welcome
- 20' - Introduction
- 5' break
- 20' - NESAP Applications & Motifs
- 30' - Catalog of defects and best practice recommendations
- 5' break
- 60' - Parallelware tools: Trainer & Analyzer
- 5' break
- 20' - Homework
- 20' - Q&A

Format: Remote lectures, and demos, and homework exercises

Part 2, Thursday, October 29

9:00 am - 12:00 am PDT

Office hours [Optional]

- 60' - Homework exercises demo
- 5' break
- 110' - Support, Questions, FAQs for using Parallelware tools

Format: Remote office hours

Part 3, Wednesday, November 4

8:00 am - 1:00 pm PST

Guided parallelization of ZPIC: Ensuring best practices with Parallelware tools

- 30' - Case study: Guided parallelization of ZPIC with Parallelware tools
- 20' - Performance evaluation of ZPIC
- 10' break
- 220' - Bring your own applications
- 20' - Q&A

Format: Remote demos and hands-on

Parallelware Training Series

Motif-guided Parallelization of
ZPIC with OpenMP and
OpenACC



Welcome and introductions