Introduction to High-Performance Parallel Distributed Computing using Chapel, UPC++, and Coarray Fortran

Dr. Michelle Mills Strout (HPE), Dr. Damian Rouson (LBNL)

The International Conference for High Performance Computing, Networking, Storage, and Analysis 2023 Tutorial

go.lbl.gov/sc23
Introduction to High-Performance Parallel Distributed Computing using Chapel, UPC++, and Coarray Fortran

The International Conference for High Performance Computing, Networking, Storage, and Analysis 2023 Tutorial

Intro session
go.lbl.gov/sc23
Introduction to High-Performance Parallel Distributed Computing using Chapel, UPC++ and Coarray Fortran

Other Contributors:

Dan Bonachea, Jeremiah Corrado, Paul H. Hargrove, Amir Kamil, Katherine Rasmussen, Sameer Shende, Daniel Waters

Dr. Michelle Mills Strout

Dr. Damian Rouson
Acknowledgements

This work was supported in part by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two U.S. Department of Energy organizations (Office of Science and the National Nuclear Security Administration) responsible for the planning and preparation of a capable exascale ecosystem, including software, applications, hardware, advanced system engineering and early testbed platforms, in support of the nation’s exascale computing imperative.

This work used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility operated under Contract No. DE-AC02-05CH11231, as well as This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.
Schedule for Chapel, UPC++ and Fortran Tutorial

Sun Nov 12th, 8:30am - noon (all times MST)

● 8:30 - 8:40:  Tutorial Overview
● 8:40 - 9:40:  Parallel programming in Chapel
● 9:40 - 10:00: Parallel programming with Fortran Coarrays, Part 1

● 10:00-10:30: Coffee Break

● 10:30 - 11:05: Parallel programming with Fortran Coarrays, Part 2
● 11:05 - noon: Parallel programming with UPC++
Motivation

● You have …
  ○ A lot of data to process and analyze
  ○ A big simulation to run
  ○ Or both of the above

● Resources are available
  ○ Your laptop has multiple cores that can process in parallel
  ○ Your lab/institution has a cluster
  ○ Or your lab/institution has a supercomputer

● Writing a parallel program enables you to analyze data and/or perform simulations significantly faster.
Which programming language(s) do you use the most? (you can respond to this question 3 times)

<table>
<thead>
<tr>
<th>Language</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>C/C++</td>
<td>0%</td>
</tr>
<tr>
<td>Fortran</td>
<td>0%</td>
</tr>
<tr>
<td>Chapel</td>
<td>0%</td>
</tr>
<tr>
<td>Python</td>
<td>0%</td>
</tr>
<tr>
<td>Java</td>
<td>0%</td>
</tr>
<tr>
<td>R</td>
<td>0%</td>
</tr>
<tr>
<td>Perl</td>
<td>0%</td>
</tr>
<tr>
<td>Haskell, Scala, ...</td>
<td>0%</td>
</tr>
<tr>
<td>Other</td>
<td>0%</td>
</tr>
</tbody>
</table>
PGAS Programming Models

- PGAS: Partitioned Global Address space
- Chapel, UPC++, and Fortran with coarrays are PGAS programming models
- A programming model provides an interface and code patterns to a programmer along with a concept of how code will execute at runtime.

- Can access variables in global address space from each node
- Implemented with puts and gets (RMA: remote memory access)
- Can partition/organize data and computation to reduce RMA
This tutorial: Chapel, UPC++, Fortran with coarrays

- Shared example shown in all three: **2D heat diffusion**
- Then other examples per programming model
  - Chapel: k-mer counting, processing files in parallel
  - UPC++: 1-d Jacobi solver, distributed hash table
  - Fortran: task scheduling, hello world variants

- Example Codes You Can Try
  - Providing a Docker image and instructions for obtaining a tarball containing all example programs: [go.lbl.gov/sc23](https://go.lbl.gov/sc23)
  - You are encouraged to compile, run, and experiment with the examples throughout

- Q&A Protocol
  - Raise your hand!
  - Model experts also available to answer questions in Slack: [go.lbl.gov/sc23-slack](https://go.lbl.gov/sc23-slack)
Production Applications using these Programming Models

**CHAMPS: 3D Unstructured CFD**
(~100K lines of Chapel)
Éric Laurendeau, Simon Bourgault-Côté, Matthieu Parenteau, et al.
École Polytechnique Montréal

**ICAR:** Intermediate Complexity Atmospheric Research model written in Coarray Fortran
https://github.com/NCAR/icar

MetaHipMer, a genome assembler written in UPC++  https://exabiome.org/
Do you have any parallel programming experience? If so, what tools have you used?

When poll is active, respond at pollev.com/michellestrout402
Text MICHELLESTROUT402 to 22333 once to join
Shared Problem: 2D Heat Diffusion

- Specifically a 2D heat diffusion problem
  - 2D diffusion equation is above. Mathematical details: [wikipedia.org/wiki/Heat_equation](https://wikipedia.org/wiki/Heat_equation)
  - Discretization solving for the unknown at time step n+1 and spatial coordinate i,j

- Steps in sample codes
  - Set some initial conditions for $u_0$
  - Estimate $u$ over time and space as shown below
  - Show how to parallelize these computations

\[
\frac{\partial u}{\partial t} = \nu \frac{\partial^2 u}{\partial x^2} + \nu \frac{\partial^2 u}{\partial y^2}
\]

\[
\begin{align*}
u_{i,j}^{n+1} &= u_{i,j}^n + \frac{\nu \Delta t}{\Delta x^2} (u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n) \\
&+ \frac{\nu \Delta t}{\Delta y^2} (u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n)
\end{align*}
\]

Simplified form

assumes $\Delta x = \Delta y$, and let $\alpha = \nu \Delta t / \Delta x^2$

\[
\begin{pmatrix}
    u_{i,j}^{n+1} \\
    \vdots
\end{pmatrix} = \begin{pmatrix}
    u_{i,j}^n \\
    \vdots
\end{pmatrix} + \alpha \begin{pmatrix}
    u_{i+1,j}^n + u_{i-1,j}^n \\
    -4u_{i,j}^n + u_{i,j+1}^n + u_{i,j-1}^n
\end{pmatrix}
\]
What do you want to learn about Chapel, UPC++, or Coarray Fortran today?

💡 When poll is active, respond at pollev.com/michellestrout402
SMS Text MICHELLESTROUT402 to 22333 once to join
Schedule for Chapel, UPC++ and Fortran Tutorial

**Sun Nov 12th, 8:30am - noon (all times MST)**

- **8:30 - 8:40:** Tutorial Overview
- **8:40 - 9:40:** Parallel programming in Chapel
- **9:40 - 10:00:** Parallel programming with Fortran Coarrays, Part 1
- **10:00-10:30:** Coffee Break
- **10:30 - 11:05:** Parallel programming with Fortran Coarrays, Part 2
- **11:05 - noon:** Parallel programming with UPC++