

Productive Programming in Chapel: A Language for General, Locality-Aware Parallelism

Chapel Team, Cray Inc.

SC12: November 11th, 2011



SC12
Salt Lake City, Utah



What is Chapel?

- An emerging parallel programming language
 - Design and development led by Cray Inc.
 - in collaboration with academia, computing centers, industry
 - Initiated under the DARPA HPCS program

- **Overall goal:** Improve programmer productivity
 - Improve the **programmability** of parallel computers
 - Match or beat the **performance** of current programming models
 - Support better **portability** than current programming models
 - Improve the **robustness** of parallel codes

- A work-in-progress

Chapel's Implementation

- Being developed as open source at SourceForge
- Licensed as BSD software
- **Target Architectures:**
 - Cray architectures
 - multicore desktops and laptops
 - commodity clusters
 - systems from other vendors
 - (in-progress: CPU+accelerator hybrids, manycore, ...)

Today's Goals

- Introduce you to the Chapel language in-depth
 - motivating themes
 - central language concepts and features
 - project status
- Demonstrate the Chapel compiler interactively
- Point you toward resources for future reference
- Get your feedback on Chapel

Our Team For Today's Tutorial

- Brad Chamberlain



- Sung-Eun Choi



- Martha Dumler



Who Are You?

Type of Institution?

- Academic, Industry, HPC Lab, Gov't, ...

Role?

- Student, postdoc, faculty, developer, researcher, ...

Favorite Languages?

- Fortran, C, C++, Java, Matlab, Python, Perl, C#, ...

Parallel Programming Models?

- MPI, OpenMP, UPC, CAF, Pthreads, CUDA, ...

Ground Rules

- Please ask questions as we go
- Also feel free to ask us questions during the break and afterwards

This Morning You Should Receive

1. A Chapel USB Stick with...
 - the final tutorial slides
 - the Chapel release
 - Chapel documentation and materials
2. A spiffy Chapel backpack
3. A Chapel-specific survey on the tutorial and language
 - please return to us by the end of the session
 - **Note:** SC also has a survey you should complete today
 - return these to the student volunteers

This Morning's Plan

8:30 – Welcome

8:40 – [Background](#)

9:00 – [Base Language](#)

9:30 – [Data Parallelism](#)

10:00 – Break

10:30 – [Task Parallelism](#)

11:00 – [Locales](#)

11:20 – [Domain Maps](#)

11:40 – [Project Overview](#)

12:00 – Done!

Resources For After Today

Chapel project page: <http://chapel.cray.com>

- overview, papers, presentations, language spec, ...

Chapel SourceForge page: <https://sourceforge.net/projects/chapel/>

- release downloads, public mailing lists, code repository, ...

IEEE TCSC Blog Series:

- [*Myths About Scalable Parallel Programming Languages*](#)

Mailing Lists:

- chapel_info@cray.com: contact the team
- chapel-users@lists.sourceforge.net: user-oriented discussion list
- chapel-developers@lists.sourceforge.net: dev.-oriented discussion
- chapel-education@lists.sourceforge.net: educator-oriented discussion
- chapel-bugs@lists.sourceforge.net/chapel_bugs@cray.com : public/private bug forum



Surveys

Please take the time to fill out and return both ours and SC12's

Thanks!

For your interest in Chapel and your feedback





Any Final Questions or Comments?