

# Productive Programming in Chapel: A Computation-Driven Introduction

## Project Status, Next Steps

Michael Ferguson and Lydia Duncan  
Cray Inc,  
SC15 November 15<sup>th</sup>, 2015



# Safe Harbor Statement

This presentation may contain forward-looking statements that are based on our current expectations. Forward looking statements may include statements about our financial guidance and expected operating results, our opportunities and future potential, our product development and new product introduction plans, our ability to expand and penetrate our addressable markets and other statements that are not historical facts. These statements are only predictions and actual results may materially vary from those projected. Please refer to Cray's documents filed with the SEC from time to time concerning factors that could affect the Company and these forward-looking statements.

# Outline

- ✓ Motivation
- ✓ Chapel Background and Themes
- ✓ Learning the Base Language with n-body
- ✓ Short Introduction to Task Par
- ✓ Hands-On 1: Hello World
- ✓ Short Introduction to Locality
- ✓ Data Parallelism with Jacobi
- ✓ Hands-On 2: Mandelbrot
- Project Status and Next Steps

# Chapel's 5-year push

- Based on positive user response to Chapel under HPCS, Cray undertook a five-year effort to improve it
  - we're partway through our third year
- Focus Areas:
  1. Improving **performance** and scaling
  2. **Fixing** immature aspects of the language and implementation
    - e.g., strings, memory management, error handling, ...
  3. **Porting** to emerging architectures
    - Intel Xeon Phi, accelerators, heterogeneous processors and memories, ...
  4. Improving **interoperability**
  5. Growing the Chapel user and developer **community**
    - including non-scientific computing communities
  6. Exploring transition of Chapel **governance** to a neutral, external body



# Chapel is a Collaborative, Community Effort



Colorado  
State  
University



RICE®



LABORATORY FOR  
TELECOMMUNICATIONS  
SCIENCES

ETH Zürich

 Lawrence Livermore  
National Laboratory

 Sandia National Laboratories



Lawrence Berkeley  
National Laboratory

Argonne  
NATIONAL LABORATORY



 OAK  
RIDGE  
National Laboratory

  
Pacific Northwest  
NATIONAL LABORATORY

Proudly Operated by Battelle Since 1965

 東京大学  
THE UNIVERSITY OF TOKYO

  
UNIVERSIDAD  
DE MÁLAGA



  
UNIVERSITY OF  
MARYLAND

AMD

(and many others as well...)

<http://chapel.cray.com/collaborations.html>

COMPUTE

STORE

ANALYZE

Copyright 2015 Cray Inc.



# A Year in the Life of Chapel

- **Two major releases per year** (April / October)
  - ~a month later: detailed release notes
- **SC** (Nov)
  - annual **Lightning Talks BoF** featuring talks from the community
  - annual **CHUG (Chapel Users Group) happy hour**
  - plus tutorials, panels, BoFs, posters, educator sessions, exhibits, ...
- **CHIUW: Chapel Implementers and Users Workshop** (May/June)
  - CHIUW 2014 held at IPDPS (Phoenix, AZ)
  - CHIUW 2015 held at PLDI/FCRC (Portland, OR)
  - CHIUW 2016 to be held at IPDPS (Chicago, IL)
- **Talks, tutorials, research visits, blog posts, ...** (year-round)

# Implementation Status -- Version 1.12.0 (Oct 2015)

## Overall Status:

- **User-facing Features:** generally in good shape
  - some receiving additional attention (e.g., strings, OOP, errors)
- **Multiresolution Features:** in use today
  - their interfaces are likely to continue evolving over time
- **Error Messages:** not always as helpful as one would like
  - correct code tends to work well, incorrect code can be puzzling
- **Performance:** hit-or-miss depending on the idioms used
  - ultimately, Chapel will support competitive performance
  - effort to-date has focused primarily on correctness

## This is a great time to:

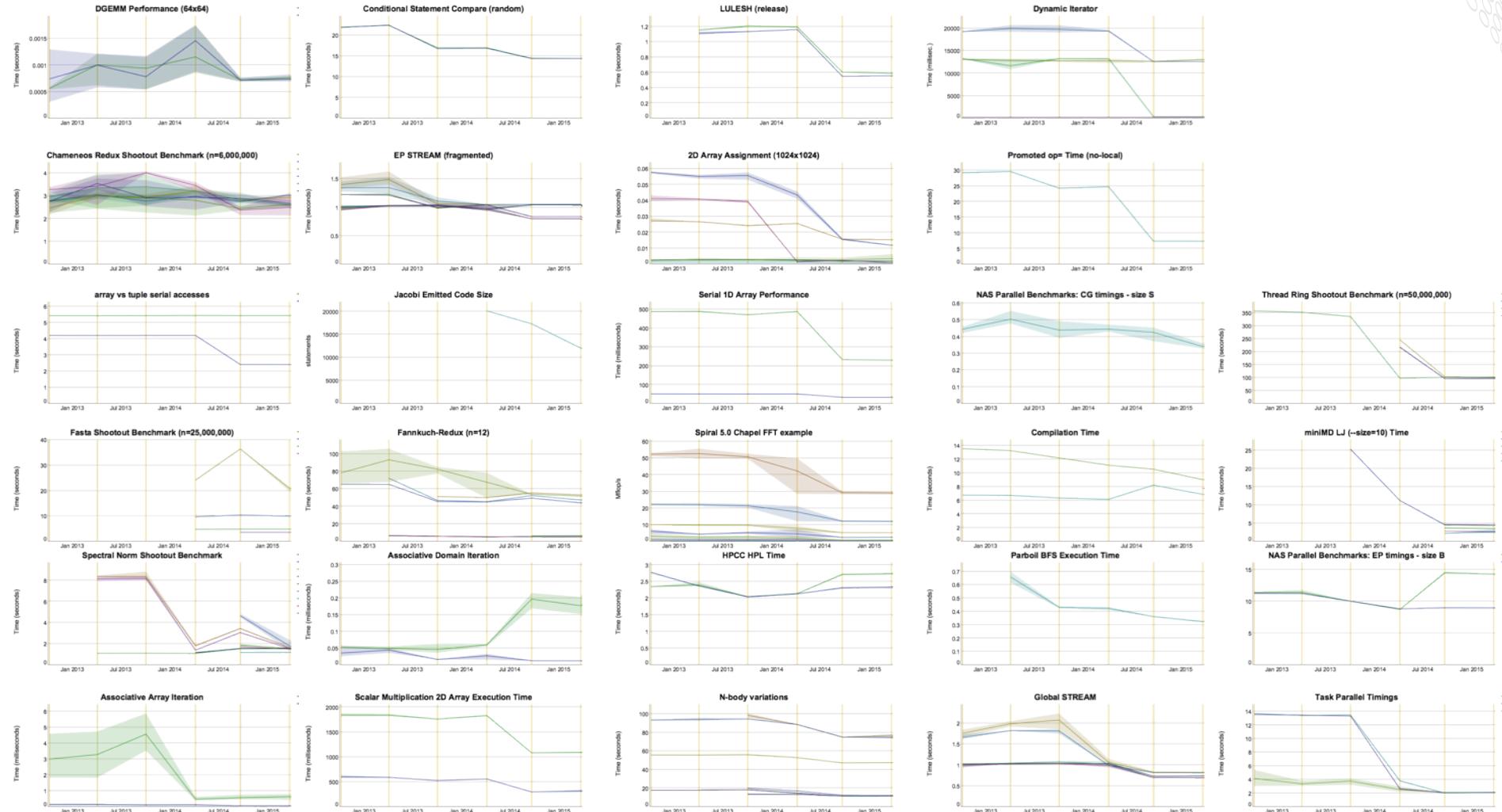
- Try out the language and compiler
- Use Chapel for non-performance-critical projects
- Give us feedback to improve Chapel
- Use Chapel for parallel programming education

# Chapel and Education

- When teaching parallel programming, I like to cover:
  - data parallelism
  - task parallelism
  - concurrency
  - synchronization
  - locality/affinity
  - deadlock, livelock, and other pitfalls
  - performance tuning
  - ...
- I don't think there's been a good language out there...
  - for teaching *all* of these things
  - for teaching *some* of these things well at all
  - *until now:* We believe Chapel can play a crucial role here  
(see <http://chapel.cray.com/education.html> for more information and  
<http://cs.washington.edu/education/courses/csep524/13wi/> for my use of Chapel in class)

# Single-Locale Execution Time is Improving

lower is better, yellow lines indicate releases (1.6-1.11)



COMPUTE

STORE

ANALYZE

# Suggested Reading

## Overview Papers:

- [A Brief Overview of Chapel](#), Chamberlain (early draft of a chapter for *A Brief Overview of Parallel Programming Models*, edited by Pavan Balaji, to be published by MIT Press in 2015).
  - *a detailed overview of Chapel's history, motivating themes, features*
- [The State of the Chapel Union \[slides\]](#), Chamberlain, Choi, Dumler, Hildebrandt, Iten, Litvinov, Titus. CUG 2013, May 2013.
  - *a higher-level overview of the project, summarizing the HPCS period*

# Lighter Reading

## Blog Articles:

- [Chapel: Productive Parallel Programming](#), [Cray Blog](#), May 2013.
  - *a short-and-sweet introduction to Chapel*
- [Why Chapel?](#) (part 1, part 2, part 3), [Cray Blog](#), June-October 2014.
  - *a series of articles answering common questions about why we are pursuing Chapel in spite of the inherent challenges*
- [Six Ways to Say “Hello” in Chapel](#) (part 1), [Cray Blog](#), September 2015.
  - *a series of articles to illustrate the basics of Chapel*
- [\[Ten\] Myths About Scalable Programming Languages](#),  
[IEEE TCSC Blog](#) ([index available on chapel.cray.com “blog articles” page](#)), April-November 2012.
  - *a series of technical opinion pieces designed to combat standard arguments against the development of high-level parallel languages*

# Online Resources

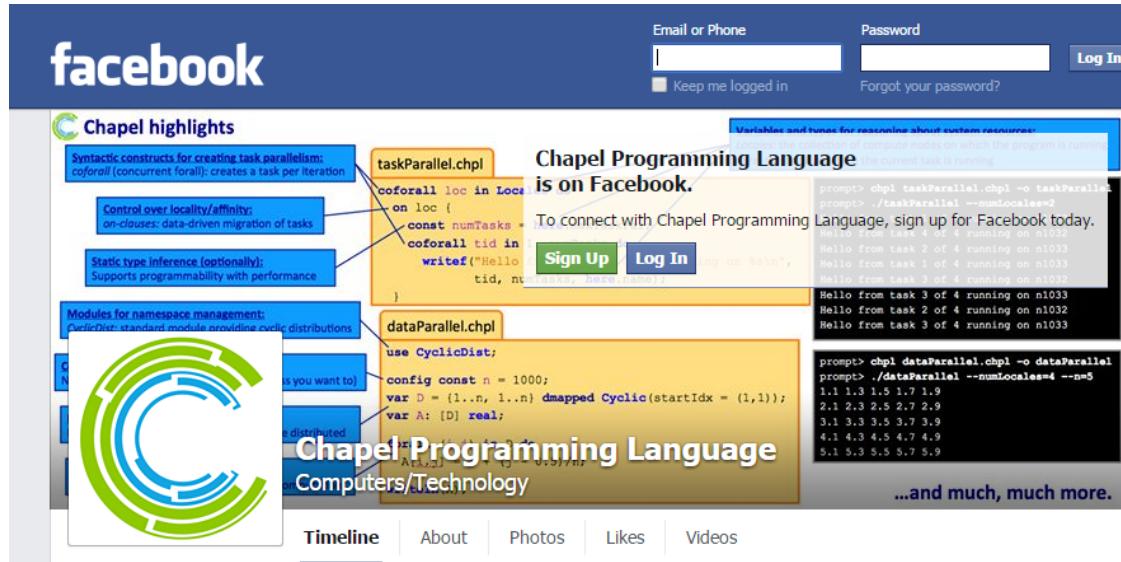
## Project page: <http://chapel.cray.com>

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

## GitHub page: <https://github.com/chapel-lang>

- download Chapel; browse source repository; contribute code

## Facebook page: <https://www.facebook.com/ChapelLanguage>



COMPUTE

STORE

ANALYZE

# Community Resources

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

- hosts community mailing lists  
(also serves as an alternate release download site to GitHub)

## Mailing Aliases:

### write-only:

- [chapel\\_info@cray.com](mailto:chapel_info@cray.com): contact the team at Cray

### read-only:

- [chapel-announce@lists.sourceforge.net](mailto:chapel-announce@lists.sourceforge.net): read-only announcement list

### read/write:

- [chapel-users@lists.sourceforge.net](mailto:chapel-users@lists.sourceforge.net): user-oriented discussion list
- [chapel-developers@lists.sourceforge.net](mailto:chapel-developers@lists.sourceforge.net): developer discussion
- [chapel-education@lists.sourceforge.net](mailto:chapel-education@lists.sourceforge.net): educator discussion
- [chapel-bugs@lists.sourceforge.net](mailto:chapel-bugs@lists.sourceforge.net): public bug forum

# Final Notes

## Surveys

Please take the time to fill out and return the surveys  
(both ours and SC15's)

## Thanks!

For your interest in Chapel and your feedback

# Legal Disclaimer

*Information in this document is provided in connection with Cray Inc. products. No license, express or implied, to any intellectual property rights is granted by this document.*

*Cray Inc. may make changes to specifications and product descriptions at any time, without notice.*

*All products, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.*

*Cray hardware and software products may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.*

*Cray uses codenames internally to identify products that are in development and not yet publically announced for release. Customers and other third parties are not authorized by Cray Inc. to use codenames in advertising, promotion or marketing and any use of Cray Inc. internal codenames is at the sole risk of the user.*

*Performance tests and ratings are measured using specific systems and/or components and reflect the approximate performance of Cray Inc. products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance.*

*The following are trademarks of Cray Inc. and are registered in the United States and other countries: CRAY and design, SONEXION, and URIKA. The following are trademarks of Cray Inc.: ACE, APPRENTICE2, CHAPEL, CLUSTER CONNECT, CRAYPAT, CRAYPORT, ECOPHLEX, LIBSCI, NODEKARE, THREADSTORM. The following system family marks, and associated model number marks, are trademarks of Cray Inc.: CS, CX, XC, XE, XK, XMT, and XT. The registered trademark LINUX is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis. Other trademarks used in this document are the property of their respective owners.*

# CRAY®



COMPUTE

| STORE

| ANALYZE

