Welcome to CHIUW 2016!

Brad Chamberlain
Chapel Team, Cray Inc.
CHIUW 2016, May 27, 2016

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.

What is Chapel?

Chapel: An emerging parallel programming language

- portable
- open-source
- a collaborative effort
- a work-in-progress

Goals:

- Support general parallel programming
  - "any parallel algorithm on any parallel hardware"
- Make parallel programming far more productive

A Year in the Life of Chapel

- **Two major releases per year** (April / October)
  - **~a month later:** detailed release notes available online
- **CHIUW:** Chapel Implementers and Users Workshop (May/June)
- **SC** (Nov)
  - tutorials, BoFs, panels, posters, educator sessions, exhibits, …
  - annual **CHUG** (Chapel Users Group) happy hour
  - For SC16: Hope to re-establish the **Chapel Lightning Talks BoF**
    - concept: propose 4 (CHIUW?) talks at submission time, 2 wildcards
- **Talks, tutorials, collaborations, social media, …** (year-round)

A Year in the Life of Chapel

- **Two major releases per year** (April / October)
  - **~a month later:** detailed release notes available online
- **CHIUW:** Chapel Implementers and Users Workshop (May/June)
- **SC** (Nov)
  - tutorials, BoFs, panels, posters, educator sessions, exhibits, …
  - annual **CHUG** (Chapel Users Group) happy hour
  - For SC16: Hope to re-establish the **Chapel Lightning Talks BoF**
    - concept: propose 4 (CHIUW?) talks at submission time, 2 wildcards
- **Talks, tutorials, collaborations, social media, …** (year-round)

A Brief History of CHIUW

CHIUW 2014:

- held in conjunction with IPDPS 2014
  - Phoenix, AZ, May 23
- keynote by Robert Harrison (SBU/BNL)
- 8 talks from the community

CHIUW 2015:

- held in conjunction with PLDI / FCRC 2015
  - Portland, OR, June 13-14
- keynote by Bill Carlson (IDA)
- 6 full-length talks, 6 “hot topic” talks
  - accepted talks from Cray team based on CHIUW 2014 feedback

program materials archived online: http://chapel.cray.com/CHIUW.html

CHIUW 2016: What's new?

Changes for CHIUW 2016:

- solicited research papers in addition to talk-only submissions
  - accepted papers to be published in IEEE IPDPS workshop proceedings
  - received 6 paper submissions, accepted 4
  - received 12 talk submissions, accepted 10

- managed submissions via EasyChair
  - took formality up a notch in terms of procedure, conflicts of interest, etc.
  - all submissions received 3+ reviews from PC members

CHIUW 2016 Schedule (Day One)

8:00: **Chapel Boot Camp** (optional pre-workshop session)

8:30: **Welcome, State of the Project**

9:00: **Research Papers I: Benchmarking and Optimization**

10:00: **Break** (catered)

10:30: **Short Talks I: Benchmarking and Optimization**

11:45: **Lunch** (on our own)

1:00: **Keynote: Nikhil Padmanabhan** (Yale University)
Chapel in the (Cosmological) Wild

2:00: **Research Papers II: Chapel Improvements**

3:00: **Break** (catered)

3:30: **Short Talks II: Chapel Improvements and Applications**

4:45: **Community Discussion**

5:30: **Dinner** (on our own)

CHIUW 2016 Schedule (Day Two)

8:30: Chapel Code Camp ("Haymarket" room)

pair-/small-team programming and design on projects of interest

Proposed Activities:

- targeting Chapel to GPUs
  (Mike Chu, Abhisek Pan, Michael Ferguson)
- explore symbolic computation code
  (Chris Taylor + Brad Chamberlain)
- support for hardware placement queries across language abstractions
  (Richard Johnson + Lydia Duncan)
- Block distributions in the context of resilience
  (Konstantina P. + David Iten)
- multigrid in Chapel
  (Nikhil Padmanhabhan + Ben Harshbarger)
- Chapel best practices / exposing impacts of implementation choices
  (Brian Guarraci + Ben Albrecht)
- enabling binary rewriting from compiler-generated code
  (Jens Breitbart + Brad Chamberlain + Lydia Duncan)

12:00: Adjourn/Lunch (on our own)

CHIUW 2016 Planning Committee

General Chairs: Tom MacDonald and Greg Titus, *Cray Inc.*

Program Committee:

- **Brad Chamberlain** (chair), *Cray Inc.*
- **Mike Chu** (co-chair), *AMD*
- **Richard Barrett**, *Sandia National Laboratories*
- **Jens Breitbart**, *Technische Universität München*
- **Brian Guarraci**, *Cricket Health*
- **Jeff Hammond**, *Intel*
- **Rob Neely**, *Lawrence Livermore National Laboratory*
- **Phil Nelson**, *Western Washington University*
- **Michelle Strout**, *University of Arizona*
- **Didem Unat**, *Koç Universitesi*

CHIUW 2016 Keynote Talk

Chapel in the (Cosmological) Wild 1:00 – 2:00

Nikhil Padmanabhan, *Yale University Professor, Physics & Astronomy*

Abstract: This talk aims to present my personal experiences using Chapel in my research. My research interests are in observational cosmology; more specifically, I use large surveys of galaxies to constrain the evolution of the Universe and to probe the physics underlying that evolution. Operationally, this involves measuring a number of spatial statistics of the distribution of galaxies, both on actual observations, but also on large numbers of simulated universes. I'll start by presenting a whirlwind introduction to cosmology, the problems that keep me up at night and our approaches to solving these. I'll then discuss what attracted me to Chapel—the ability to prototype algorithms quickly and the promised ease and flexibility of writing parallel programs. I'll then present a worked example of Chapel being used in a real-world application, discussing some of these aspects as well highlighting its interoperability with existing libraries, as well as some of the challenges. I'll conclude with what it would take for me to switch over to using Chapel all of the time.

CHIUW 2016 Research Paper Talks

9:00-10:00

Optimizing Chapel for Single-Node Environments
Richard Johnson and **Jeff Hollingsworth** (*University of Maryland*)

PGAS Access Overhead Characterization in Chapel
Engin Kayraklioglu, **Olivier Serres, Ahmad Anbar, Hashem Elezabi,** and **Tarek El-Ghazawi** (*The George Washington University*)

2:00-3:00

chplvis: A Communication and Task Visualization Tool for Chapel
Philip Nelson (*Western Washington University*) and **Greg Titus** (*Cray Inc.*)

Transparently Resilient Task Parallelism for Chapel
Konstantina Panagiotopoulou and **Hans-Wolfgang Loidl** (*Heriot-Watt University*)

CHIUW 2016 Short Talks (Morning Session)

10:30-11:45

LCALS Benchmark in Chapel
David Iten and **Elliot Ronaghan** (*Cray Inc.*)

Chapel With Polyhedral Transformation Using Autotuning
Tuowen Zhao and **Mary Hall** (*University of Utah*)

Binary Rewriting at Runtime for Efficient Dynamic Domain Map Implementations
Josef Weidendorfer and **Jens Breitbart** (*Technical University of Munich*)

Enabling Sparse Matrix Computation in Multi-locale Chapel
Amer Tahir, Milton Halem (*University of Maryland Baltimore County*)**,** and
Tyler Simon (*Laboratory for Physical Sciences*)

ISx in Chapel: Early Observations and Results
Ben Harshbarger and **Brad Chamberlain** (*Cray Inc.*)

CHIUW 2016 Short Talks (Afternoon Session)

3:30-4:45

Chapel-on-HSA: Towards Seamless Acceleration of Chapel Code using the Heterogeneous System Architecture

Abhisek Pan and **Michael Chu** (*Advanced Micro Devices*)

The Use and I: Transitivity of Module Uses and its Impact

Lydia Duncan (*Cray Inc.*)

Using Chapel for Natural Language Processing and Interactions

Brian Guarraci (*Cricket Health*)

Developing a Big Data Chapel

Chris Taylor (*Department of Defense*)

Social Network Analysis on Twitter with Chapel

Ben Albrecht and **Michael Ferguson** (*Cray Inc.*)

Questions about CHIUW?

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 CHAPEL

CRAY

THE SUPERCOMPUTER COMPANY