

# Q & A, Project Status, and Wrap-up



# Any Questions About the Language?



COMPUTE | STORE | ANALYZE

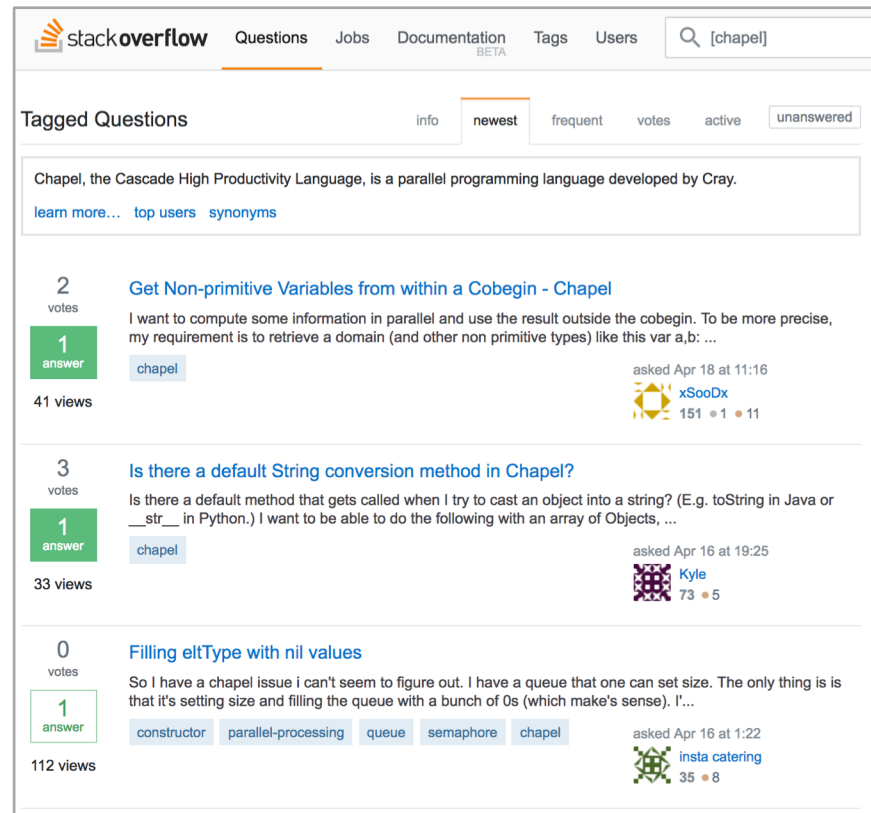
# How can I...

## ...ask questions about the language and its usage?

Stack Overflow: [chapel] tags followed by core developers

chapel-users@lists.sourceforge.net: user-oriented discussion list

#chapel (irc.freenode.net): user-oriented IRC channel



The screenshot shows the Stack Overflow search results for the tag 'chapel'. The page includes a navigation bar with links to Questions, Jobs, Documentation, Tags, and Users. A search bar contains the text '[chapel]'. Below the navigation bar, there are tabs for 'Tagged Questions', 'info', 'newest', 'frequent', 'votes', 'active', and 'unanswered'. The 'newest' tab is selected. The main content area displays three questions:

- Get Non-primitive Variables from within a Cobegin - Chapel**  
2 votes, 1 answer, 41 views. Asked Apr 18 at 11:16 by xSooDx. 151 • 1 • 11.
- Is there a default String conversion method in Chapel?**  
3 votes, 1 answer, 33 views. Asked Apr 16 at 19:25 by Kyle. 73 • 5.
- Filling elType with nil values**  
0 votes, 1 answer, 112 views. Asked Apr 16 at 1:22 by insta catering. 35 • 8.



# How can I...

## ...track the project?

[chapel-announce@lists.sourceforge.net](mailto:chapel-announce@lists.sourceforge.net): low-frequency announcements

<http://facebook.com/ChapelLanguage>: high-frequency announcements

<http://twitter.com/ChapelLanguage>: high-frequency announcements

<https://www.youtube.com/channel/UCHmm27bYjhknK5mU7ZzPGsQ/> : Chapel videos



The collage displays various Chapel Language project assets:

- Facebook Page:** Chapel Programming Language, April 21 at 5:47pm. Post text: "We're pleased to note that Chapel is currently ranked 1st in the Computer Language Benchmarks Game's 'fast' category. That said, we're even prouder of how clear and concise our programs are relative to other entries that performed well." Includes a box plot titled "How many times slower" comparing Chapel to other languages like C, C++, Fortran, Java, etc.
- Twitter Profile:** Chapel Language (@ChapelLanguage), 222 tweets, 12 following, 129 followers, 32 likes. Bio: "Chapel is a productive parallel programming language designed for large-scale computing whose development is being led by @cray\_inc".
- Twitter Post:** "Doing interesting applications work in Chapel or another language? Submit it to the PAW 2017 workshop at @SC17. [sourceinstitute.github.io/PAW/](http://sourceinstitute.github.io/PAW/)".
- Website:** Chapel Parallel Programming Language. Features a video titled "The 2nd Annual PGAS Applications Workshop" and a video titled "Chapel Productive, Multiresolution Parallel Programming | Brad Chamberlain, Cray, Inc.".
- YouTube Channel:** Chapel Parallel Programming Language. Videos include "SC16 Chapel Tutorial Promo" and "Chapel Productive, Multiresolution Parallel Programming | Brad Chamberlain, Cray, Inc.".



COMPUTE | STORE | ANALYZE

Copyright 2018 Cray Inc.

# Where to..

## Submit bug reports:

GitHub issues for chapel-lang/chapel: public bug forum  
 chapel\_bugs@cray.com: for reporting non-public bugs

## Discuss Chapel development

chapel-developers@lists.sourceforge.net: developer discussions  
 #chapel-developers (irc.freenode.net): developer-oriented IRC channel

## Discuss Chapel's use in education

chapel-education@lists.sourceforge.net: educator discussions

## Directly contact Chapel team at Cray

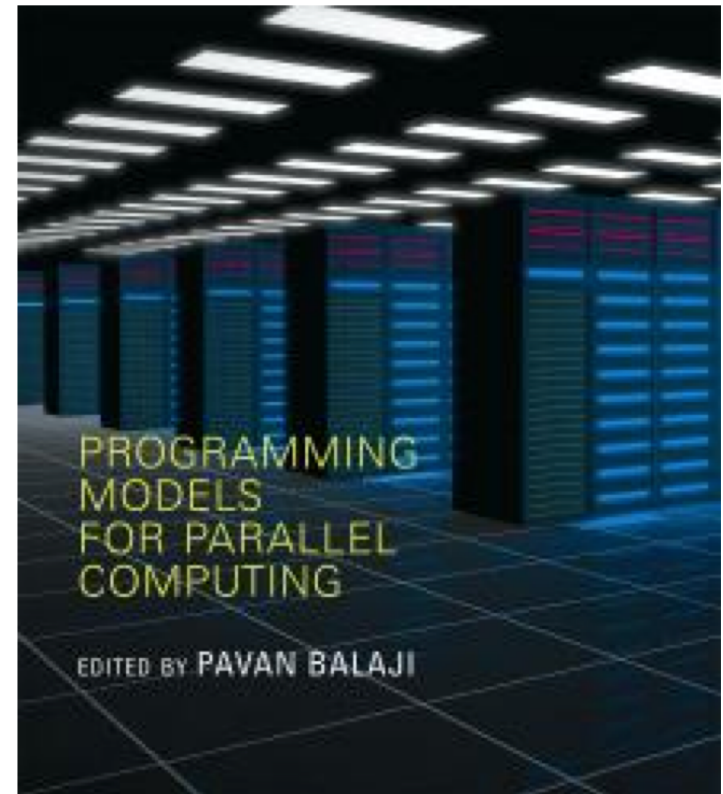
chapel\_info@cray.com



# Suggested Reading

Chapel chapter from [Programming Models for Parallel Computing](#)

- a detailed overview of Chapel's history, motivating themes, features
- published by MIT Press, November 2015
- edited by Pavan Balaji (Argonne)
- chapter is now also available [online](#)



Other Chapel papers/publications available at <http://chapel.cray.com/papers.html>



COMPUTE | STORE | ANALYZE

Copyright 2018 Cray Inc.

# Chapel Blog Articles

**[Chapel: Productive Parallel Programming](#)**, [Cray Blog](#), May 2013.

- *a short-and-sweet introduction to Chapel*

**[Six Ways to Say “Hello” in Chapel](#)** (parts [1](#), [2](#), [3](#)), [Cray Blog](#), Sep-Oct 2015.

- *a series of articles illustrating the basics of parallelism and locality in Chapel*

**[Why Chapel?](#)** (parts [1](#), [2](#), [3](#)), [Cray Blog](#), Jun-Oct 2014.

- *a series of articles answering common questions about why we are pursuing Chapel in spite of the inherent challenges*

**[\[Ten\] Myths About Scalable Programming Languages](#)**, [IEEE TCSC Blog](#) ([index available on chapel.cray.com “blog articles” page](#)), Apr-Nov 2012.

- *a series of technical opinion pieces designed to argue against standard reasons given for not developing high-level parallel languages*



# What's Next?

- **Complete new initializer, error-handling features**
- **Continue to improve multi-locale performance & scaling**
  - particularly in the context of application codes
- **Improve support for vectorization and GPUs**
- **Revamp the compiler architecture**
  - make it more approachable to developers
  - improve compilation time
  - support separate compilation and/or incremental recompilation
  - support interactive Chapel programming (e.g., REPL / interpreter)
- **Continue growing set of libraries**
- **Deploy a package manager for Chapel**





# Any Final Questions For Today?



COMPUTE | STORE | ANALYZE

# 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.*

