# Process Improvements

Chapel Team, Cray Inc.

Chapel version 1.12

October 1st, 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

- Issue Tracking: JIRA
- Chapel Improvement Proposals (CHIPs)

# Issue Tracking: JIRA

# JIRA: Background

- **Chapel has historically lacked an issue tracker**
  - Use “futures” to track certain bugs, feature requests, etc.
    - essentially tests that we run, but don’t expect to work yet
    - can compliment, but not replace an issue tracker
  - Relied on email for “issue tracking”

- **Many reasons to use an actual issue tracker**
  - Better for developers:
    - shared central location
    - SCM and regression testing integration
    - ownership, prioritization, categorization
    - easy access to comments, status, history, etc
  - Better for users:
    - easy to find, track, and upvote existing issues

# JIRA: This Effort

- **Decided on JIRA as our issue tracker**
  - Surveyed popular issue trackers
    - narrowed choice down to JIRA and GitHub issues
  - Ultimately chose JIRA because of flexibility
    - highly configurable and has rich plugin support

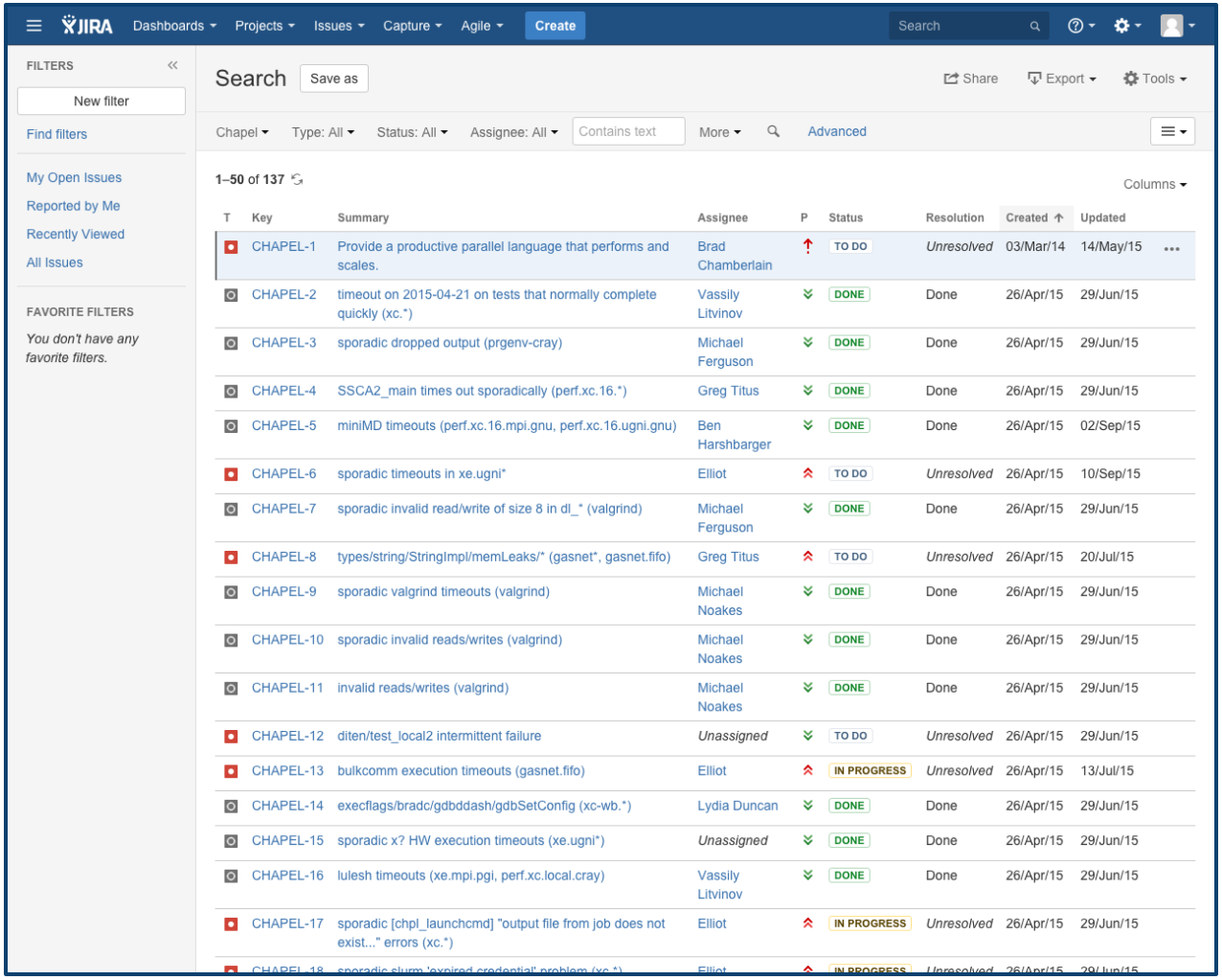
- **Started tracking regression testing with JIRA**
  - We historically used a text file under source control
    - simple, but cumbersome and completely manual
  - Regression testing is developer-oriented
    - made for a good trial run
    - improve our process on non-user-facing issues

# JIRA: Impact

- **Made triage easier**
  - Updates are immediately visible to all developers
  - No need to prune/clean-up old information manually
    - which also makes tracking sporadic issues easier

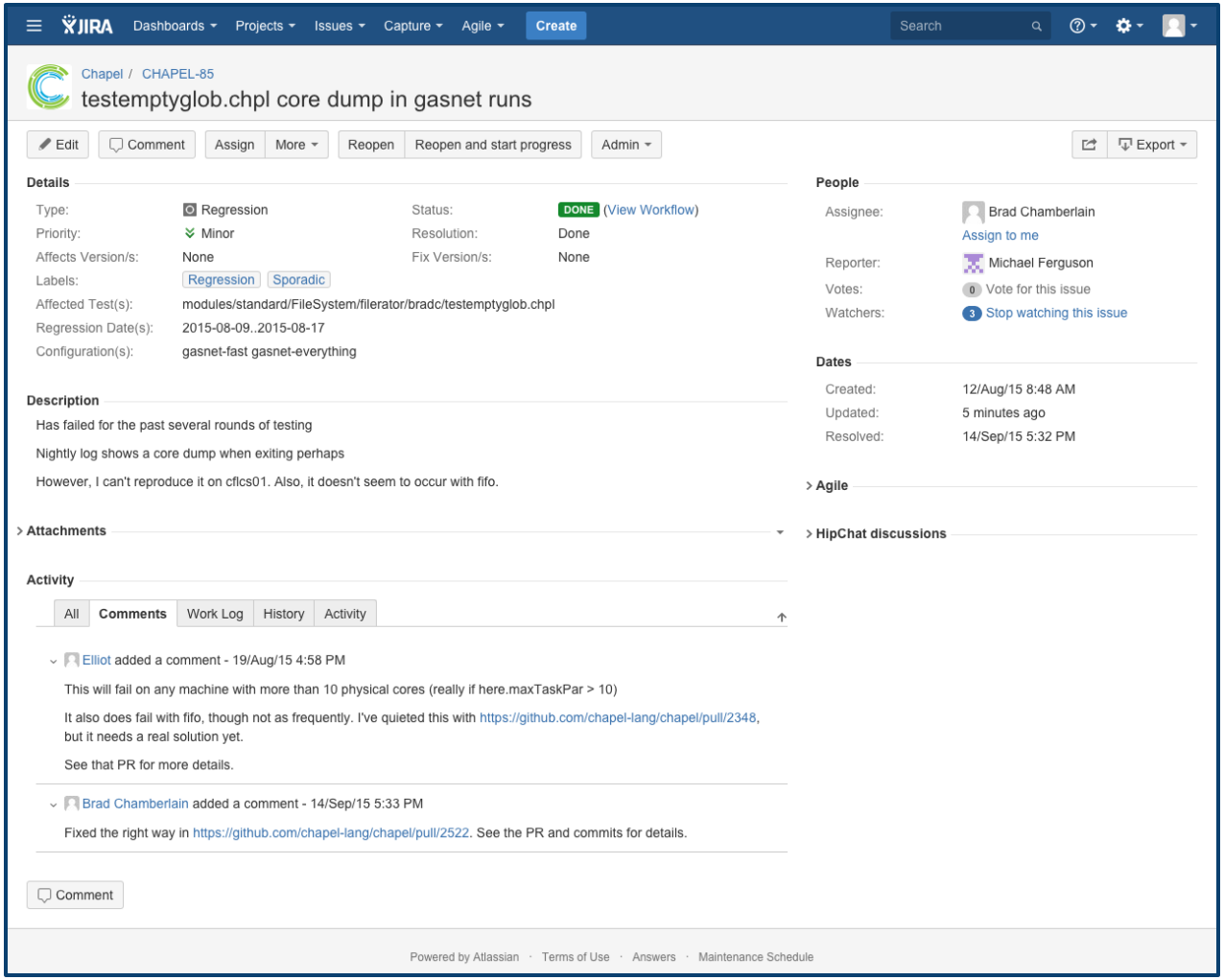
- **Improved bug fixing process in general**
  - Now much easier to…
    - … collaborate with other developers
    - … add comments to an issue
    - … identify related issues
    - … track the progress of an issue
    - … share an issue with others

# JIRA: This Effort




https://chapel.atlassian.net/projects/CHAPEL/issues/

# JIRA: This Effort



# JIRA: Status and Next Steps

## Status:

- Successfully using JIRA to track regression testing
- Available online at: https://chapel.atlassian.net/projects/CHAPEL/summary
- Recently started tracking string-as-rec issues

## Next Steps:

- Make the JIRA project more user-oriented
  - add issues for existing futures and user bugs
  - start using voting mechanism
  - but leave issue creation for developers initially
    - explore options for users to file issues directly – e.g., web portal?

# Chapel Improvement Proposals (CHIPs)

# CHIPs: Background

- **There are many ideas for improving Chapel**
- **However…**
  …significant time may pass before implementation starts
  …the people involved may change before implementation is complete
- **Not all good ideas make it past these barriers**

# CHIPs: What is a CHIP?

- **Significant changes should go through these steps:**
  1. Clear communication of the idea
  2. Discussion of the idea
  3. Implementation of the idea

- **A Chapel Improvement Proposal is:**
  - a way to record an idea to aid its progress through these steps
  - a lightweight document
  - a place to record the progress of an idea

# CHIPs: Impact, Status, and Next Steps

## Impact:

- Project can confidently separate ideas from implementation
- Ideas will not be lost to history

## Status:

- CHIPs stored in doc/chips in the Chapel git repository
- Examples of current CHIPs:
  - Chapel Improvement Proposals
  - Constrained Generics
  - ZeroMQ Integration
  - Constructor Syntax and Semantics
  - Implementing Object Copying
  - Tuple Semantics

## Next Steps:

- Further develop CHIP decision-making process
- Create new CHIPs; complete existing CHIPs

# 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, URIKA, and YARCDATA. 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
THE SUPERCOMPUTER COMPANY

http://chapel.cray.com chapel_info@cray.com http://sourceforge.net/projects/chapel/