Tools

Chapel Team, Cray Inc.

Chapel version 1.14

October 6, 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.

Outline

- chplvis Improvements
- chpldoc Improvements

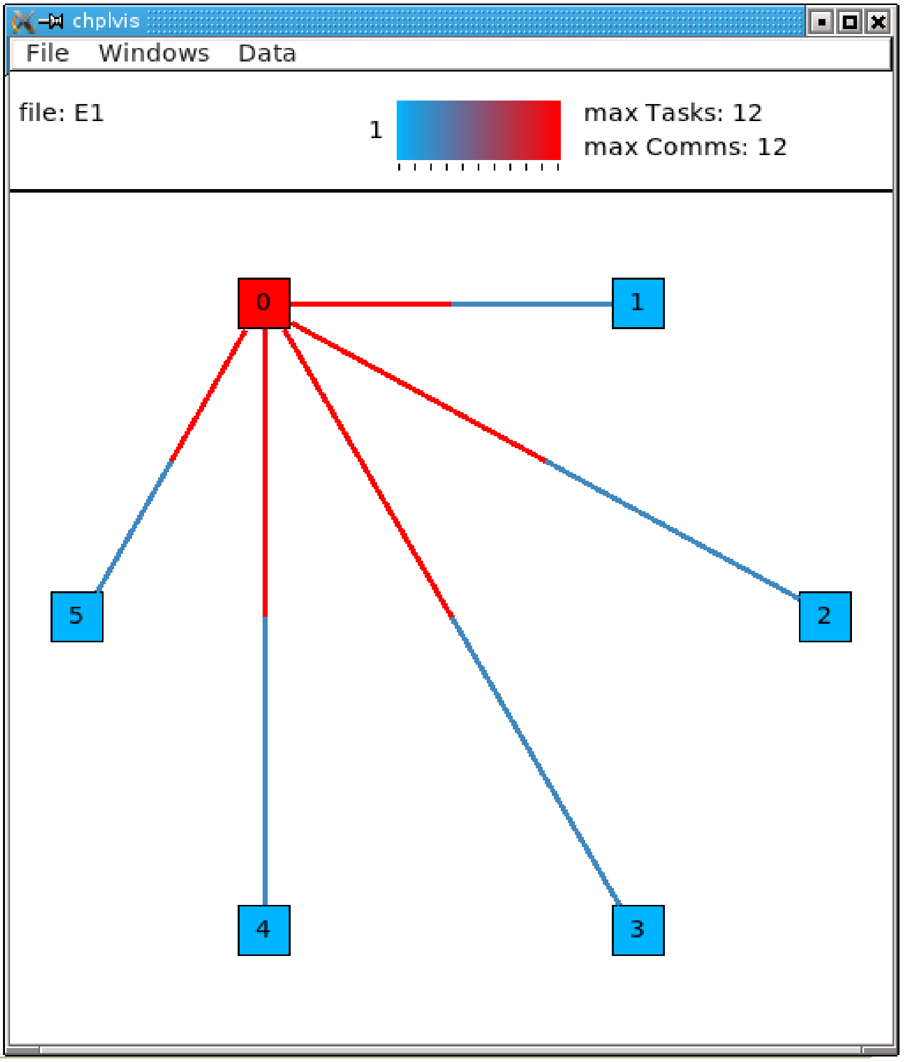
chplvis Improvements

chplvis: Background

- **chplvis first available with Chapel 1.12.0**
- **chplvis is a tool to visualize tasks and inter-locale communication in a graphical user interface**
- **Instrumentation done by programmer by adding calls to procedures in the VisualDebug module.**
  - At runtime, each locale writes a data file with events
  - Task events are: task creation, task begin and end execution
  - Communication events: gets, puts, "on calls"
- **chplvis reads data files written at run time and provides a post run view of the Chapel program run**
  - Shows CPU time, clock time, number of tasks and concurrency, and size or volume of communication events

chplvis: Original Example Display

- **Simple coforall and on-clause display, 6 locales**
  - Boxes are locales
  - Lines are communication
  - Color shows data value
    - Segment adjacent to box indicates number of communications *into* locale
- **Interactive display**
  - Click on boxes
  - Click on lines
  - Data menu changes display

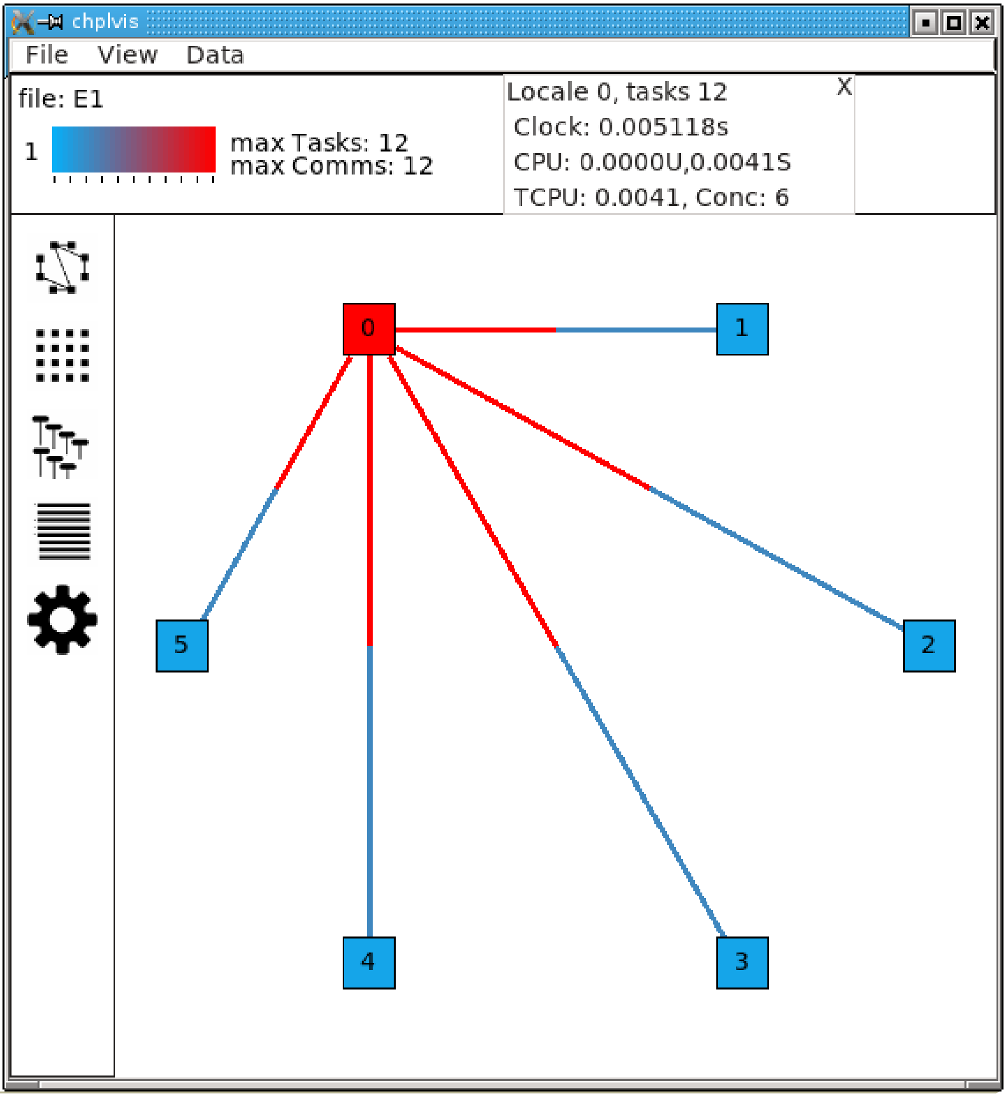


chplvis: VisualDebug Module / Runtime

- **Original version: VisualDebug module always “on”**
  - User added calls to VisualDebug module in their code
  - VisualDebug always ran unless commented out by user
- **Added VisualDebugOn**
  - Config const: VisualDebugOn – allows user to disable data collection
- **Completed callback mechanisms in runtime**
  - Task Events: Creation, Start execution, End execution
  - Communication Events: gets, puts, on Clause calls
- **Internal function names, filenames**
  - Source file name table
  - Internal function name table
  - Used by VisualDebug module for source view

chplvis: New Default Display

- **Original view now called graph view**
- **One window design**
  - Locale clicks bring up information box instead of a window
  - Communication clicks bring up comm (gets, puts, on calls) box
- **View zooms and scrolls**
- **View selection on left bar**

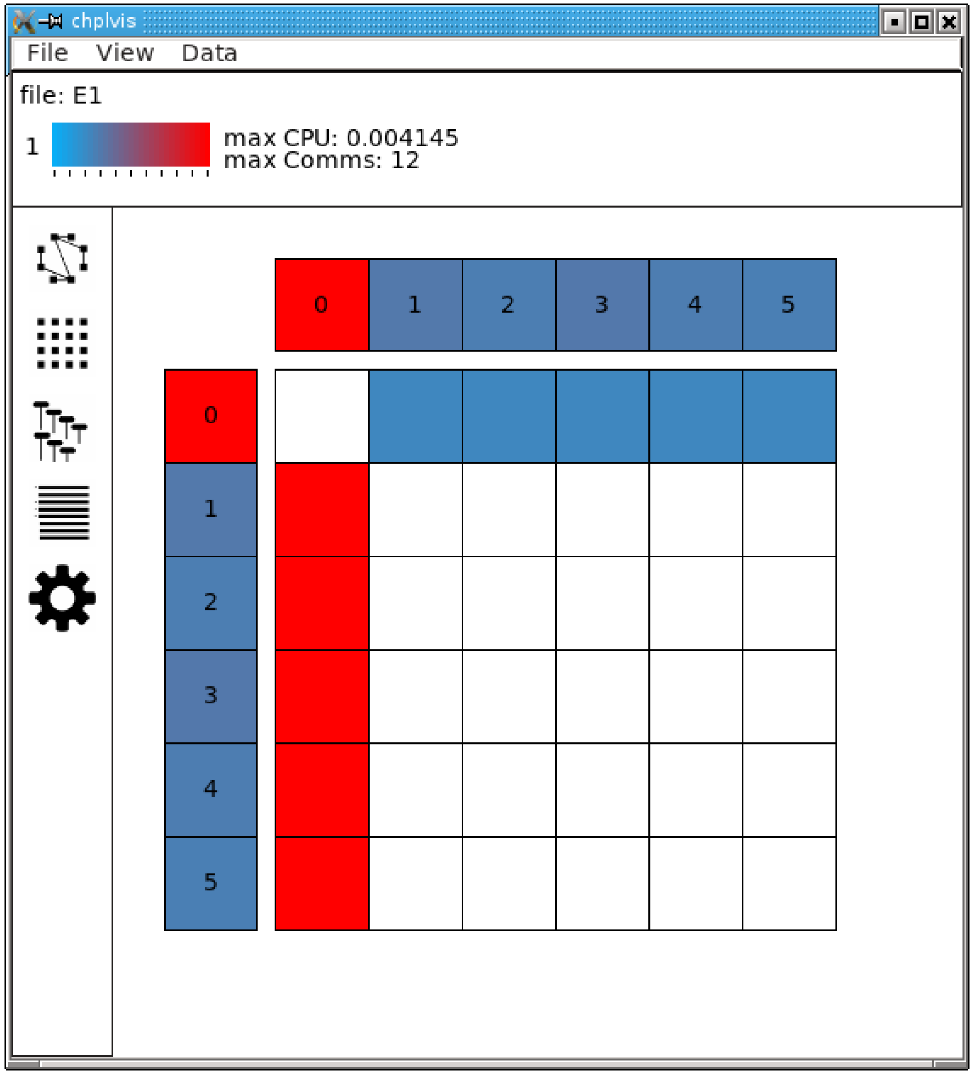


file: E1
max Tasks: 12
max Comms: 12

Locale 0, tasks 12
Clock: 0.005118s
CPU: 0.0000U,0.0041S
TCPU: 0.0041, Conc: 6

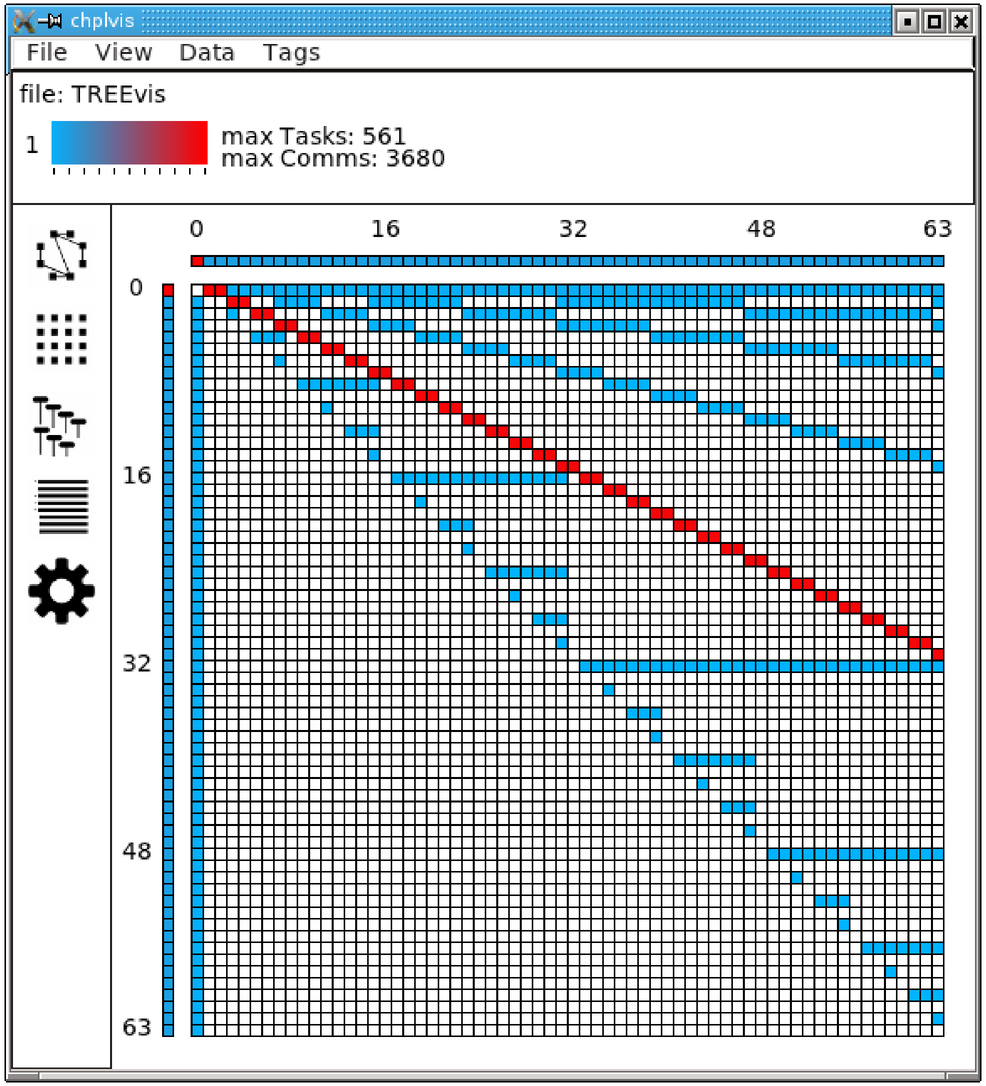
chplvis: Grid View

- Alternate view of same data
- Locales left and top
- Comm from left locale to top locale
- Red column is comm into locale 0
- Blue row is comm from locale 0 to others



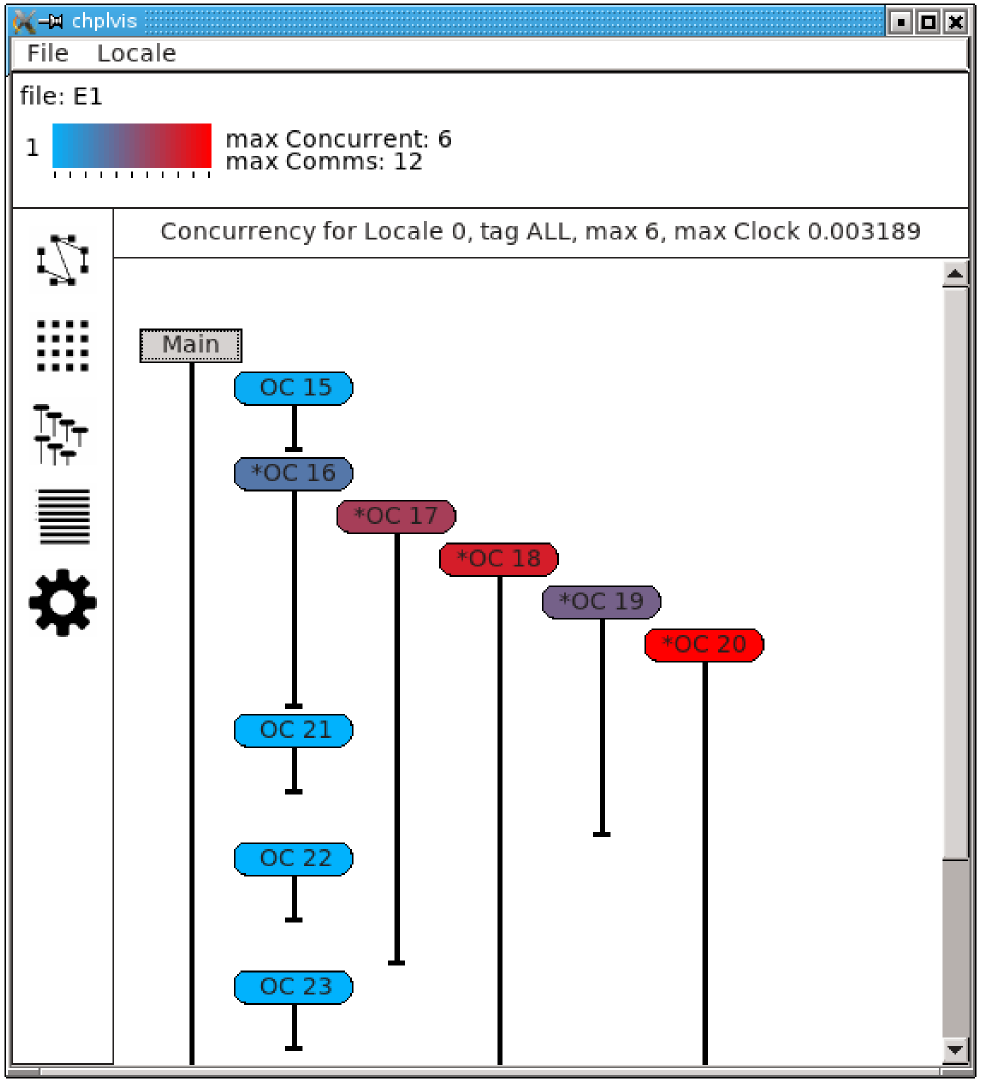
chplvis: More Than 32 Locales

- Grid view easily shows >32 locales
- Box size varies based on number of locales, window size
- View zooms and scrolls



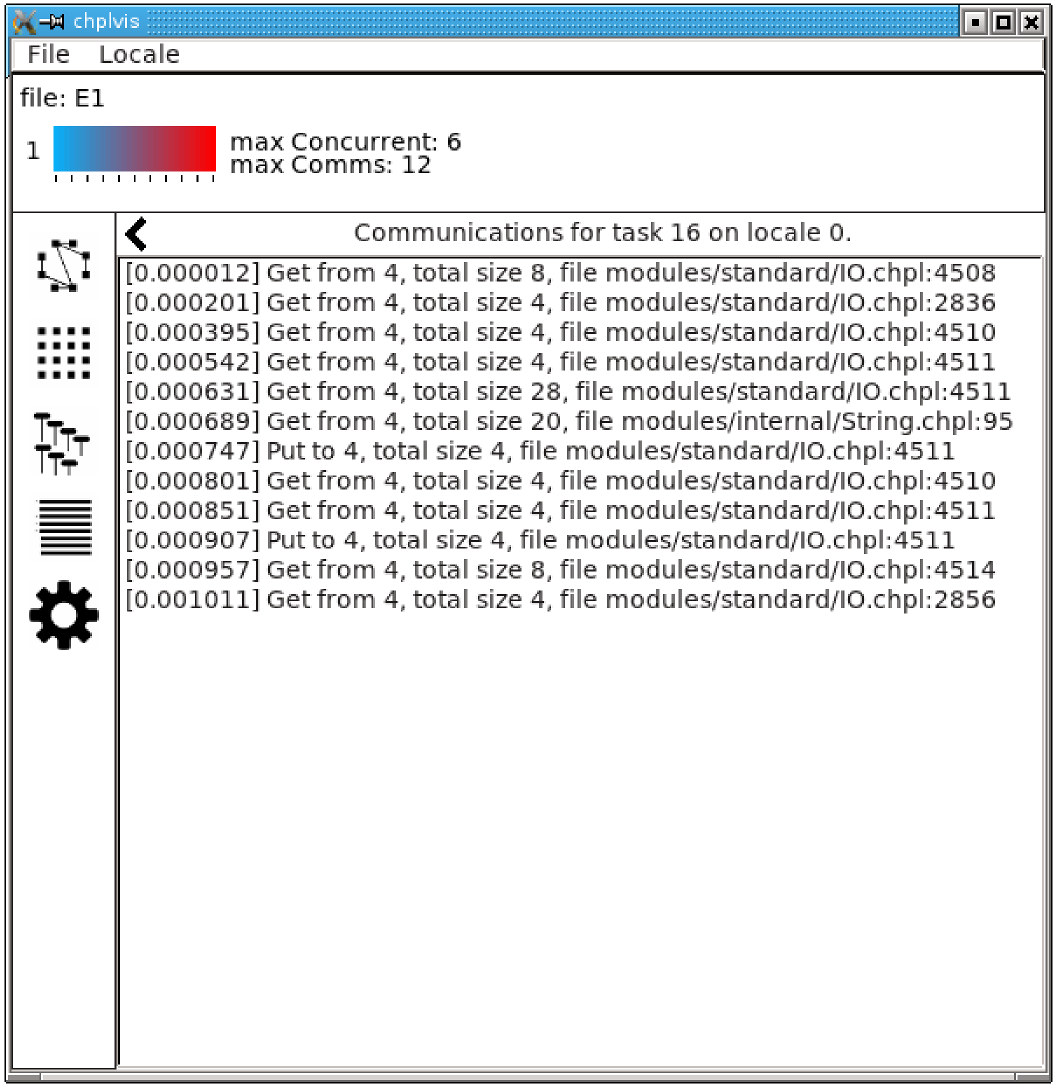
chplvis: Concurrency View

- View instead of a window
- tasks with communication now identified with '*' (*OC 16, *OC 17, …)
- Menus allow selection of locales and tags
- Click on task brings up task communication view



chplvis: Task Communication

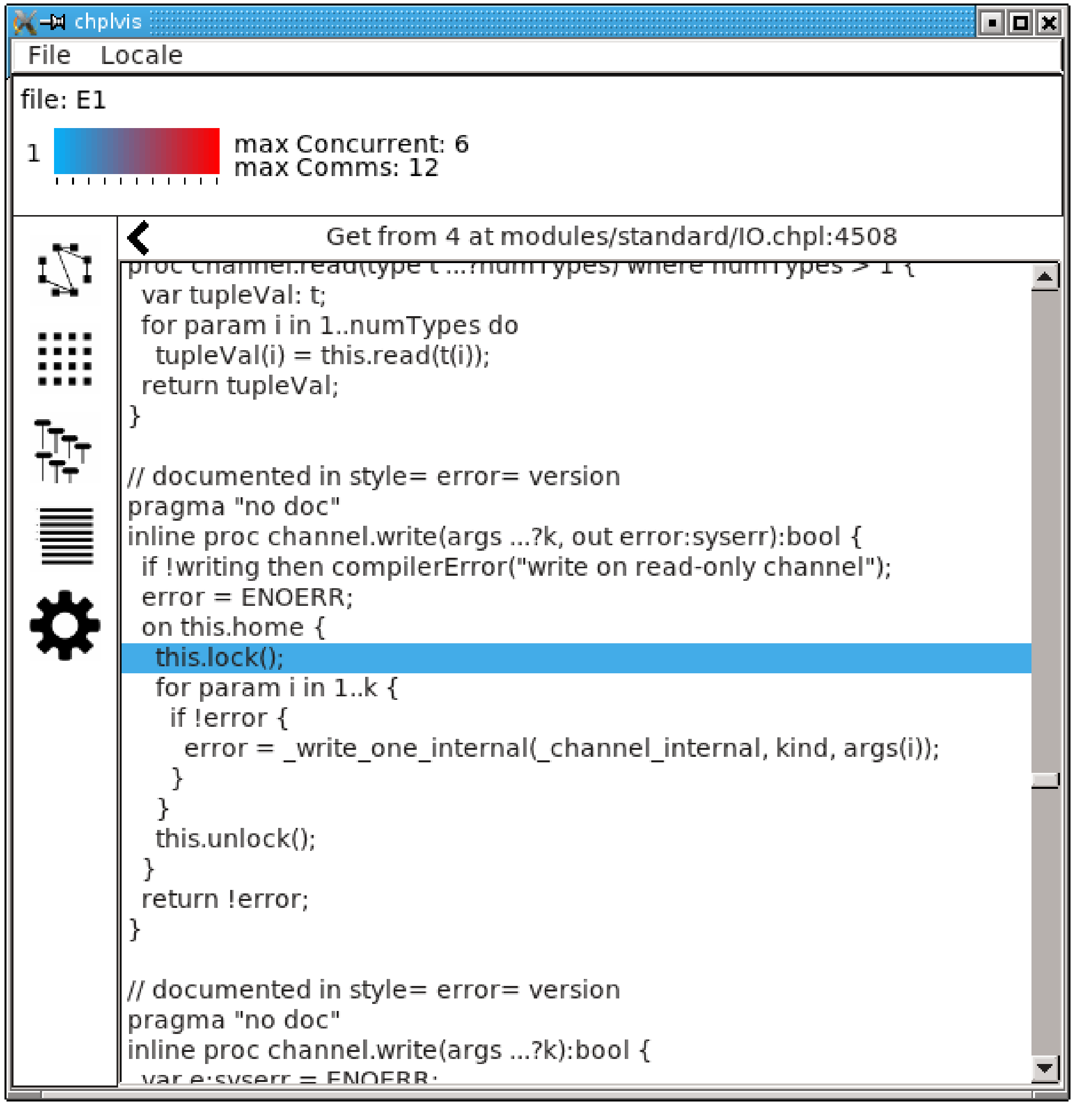
- Lists comm events for single task
- Click on comm brings up file at line number listed



```
[0.000012] Get from 4, total size 8, file modules/standard/IO.chpl:4508
[0.000201] Get from 4, total size 4, file modules/standard/IO.chpl:2836
[0.000395] Get from 4, total size 4, file modules/standard/IO.chpl:4510
[0.000542] Get from 4, total size 4, file modules/standard/IO.chpl:4511
[0.000631] Get from 4, total size 28, file modules/standard/IO.chpl:4511
[0.000689] Get from 4, total size 20, file modules/internal/String.chpl:95
[0.000747] Put to 4, total size 4, file modules/standard/IO.chpl:4511
[0.000801] Get from 4, total size 4, file modules/standard/IO.chpl:4510
[0.000851] Get from 4, total size 4, file modules/standard/IO.chpl:4511
[0.000907] Put to 4, total size 4, file modules/standard/IO.chpl:4511
[0.000957] Get from 4, total size 8, file modules/standard/IO.chpl:4514
[0.001011] Get from 4, total size 4, file modules/standard/IO.chpl:2856
```

chplvis: File View

- **Line highlighted for comm location or task definition**
- **Back arrow to get to view that led to file view**



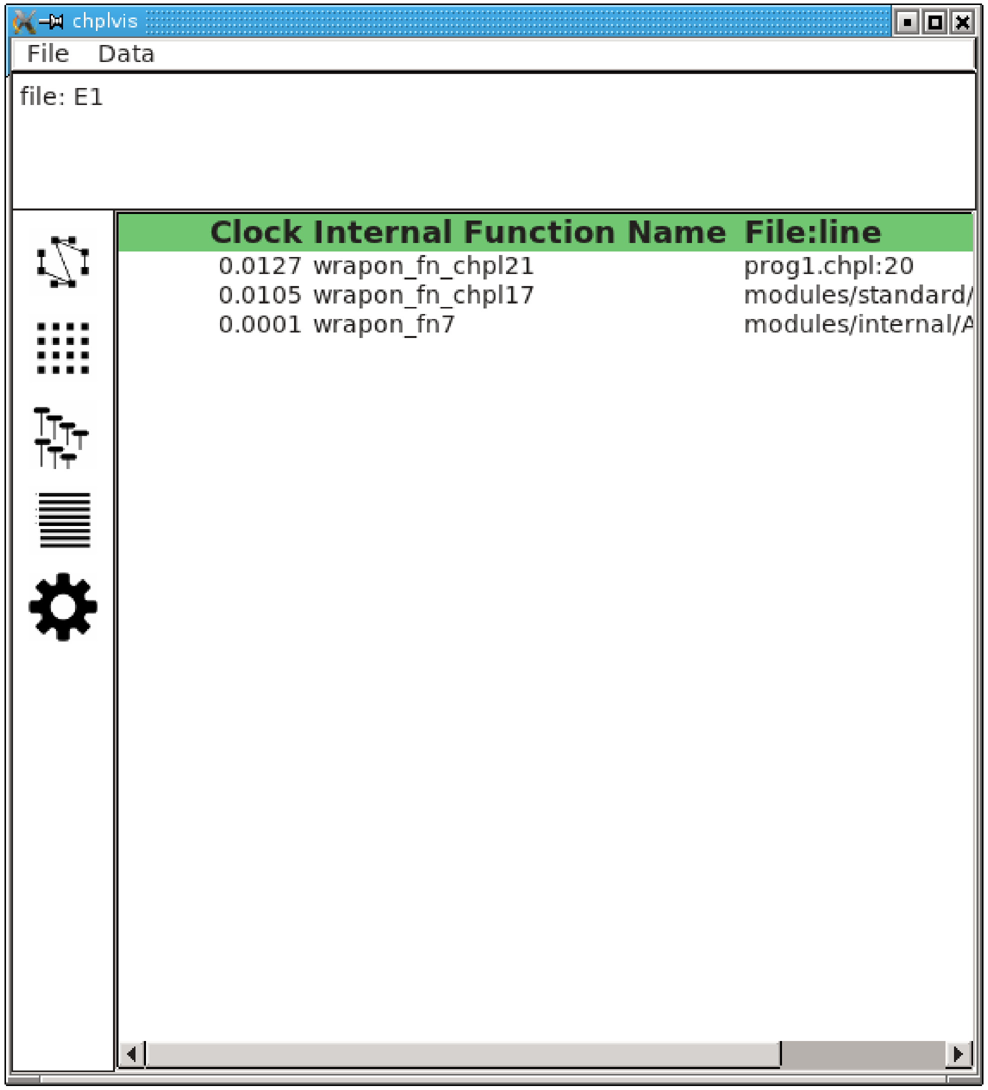
```
proc channel.read(type t ...?numTypes) where numTypes > 1 {
  var tupleVal: t;
  for param i in 1..numTypes do
    tupleVal(i) = this.read(t(i));
  return tupleVal;
}

// documented in style= error= version
pragma "no doc"
inline proc channel.write(args ...?k, out error:syserr):bool {
  if !writing then compilerError("write on read-only channel");
  error = ENOERR;
  on this.home {
    this.lock();
    for param i in 1..k {
      if !error {
        error = _write_one_internal(_channel_internal, kind, args(i));
      }
    }
    this.unlock();
  }
  return !error;
}

// documented in style= error= version
pragma "no doc"
inline proc channel.write(args ...?k):bool {
  var e:syserr = ENOERR;
```

chplvis: Profile View

- Profile view of internal functions
- Clicking on line brings up file view
- Profile data: clock, tasks, communication, on calls, gets, puts
- Selection via menu

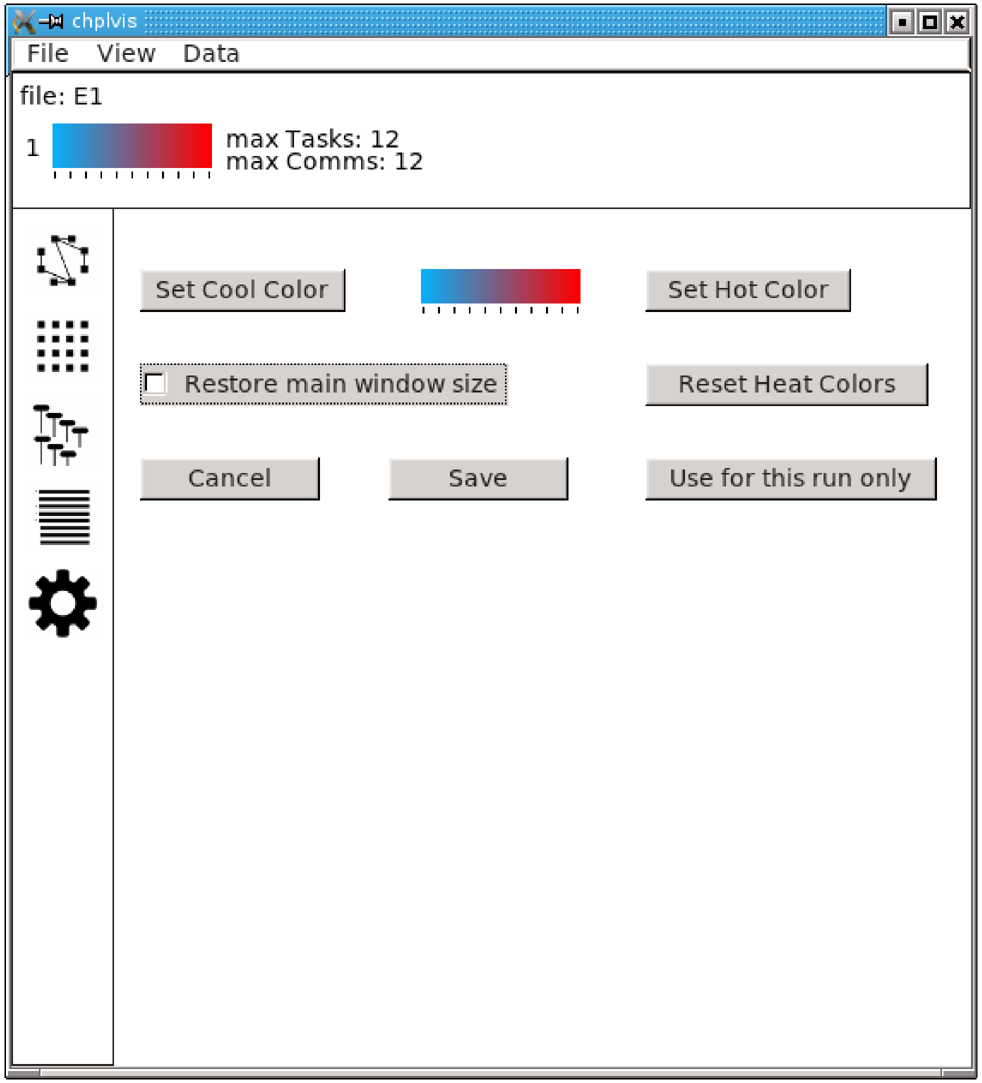


file: E1

```
Clock  Internal Function Name    File:line
0.0127 wrapon_fn_chpl21          prog1.chpl:20
0.0105 wrapon_fn_chpl17          modules/standard/
0.0001 wrapon_fn7                modules/internal/A
```

chplvis: Settings

- Heat colors selectable
- Restore window size on next run selectable



file: E1
1
max Tasks: 12
max Comms: 12

chplvis: Next Steps

- **Runtime / chplvis improvements**
  - Map more events back to user code in non-developer mode
  - Improve profile data
  - Find ways to gather and display more data
  - Look for ways to add animation to displays

- **More UI improvements**
  - Improve zooming
  - Add more settings for users, possible choices
    - background color
    - default view
    - set *CHPL_HOME* and compile directory for copied data files
  - Add a way to export a .png or .pdf of current view
  - Many more tweaks to UI possible

chpldoc Improvements

chpldoc: Background

- **Tool to generate documentation for .chpl files**
  - Originally added in Chapel 1.6.0
    - text output only
  - Improved significantly in 1.11.0 and onwards
    - HTML output revamped
    - Used to generate our online documentation for our library modules
- **Considered stable and reliable**
  - Handful of known bugs
  - Some feature requests

chpldoc: This Effort

- **Bug fixes:**
  - Stop generating documentation for symbols within:
    - Loops
    - Inner block statements

```
module M {
  var x = 10; // Should be documented
  while x > 3 {
    var inner = x / 2; // Shouldn't be documented
    … }
}
```

  - These symbols can't be accessed through a 'use' of the module
    - So they aren't part of the program's API and shouldn't be documented

chpldoc: This Effort

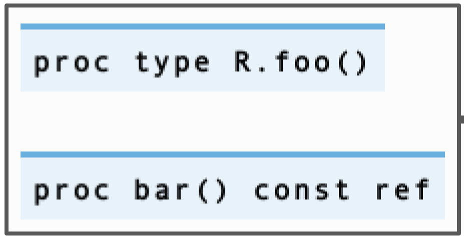
- **Bug fixes:**
  - Fix output of:
    - Dmapped variables
    - Expressions with the ‘new’ keyword

```
const rcDom = rcDomainBase dmapped new dmap(rcDomainMap);
// Turned into this instead:
const rcDom = appendExpr.Call09chpl__distributedrcDomainBase
```

chpldoc: This Effort

- **Bug fixes:**
  - HTML source highlighting for:
    - Functions with ‘const ref’ return intents
    - Type and param methods on types
  - Clean up leftover documentation of removed symbols in library docs
    - Built-ins had script to remove certain symbols, e.g. those named “chpl_*”
    - But accidentally left any documentation comments behind

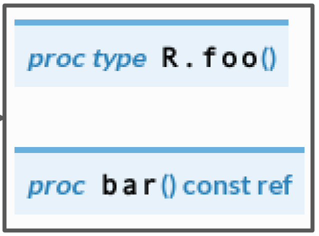
1.13 release



```
proc type R.foo()

proc bar() const ref
```

1.14 release



```
proc type R.foo()

proc bar() const ref
```

chpldoc: Next Steps

- **Change testing of chpldoc to validate .rst output**
  - Currently uses distinct text-oriented mode which could then be retired
- **Add class/record view, including inheritance**
  - Plus, class and record index
- **Support testing Chapel code used in chpldoc comments**
  - Similar to Python doctests
- **Link module documentation to source code**

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