

New Chaste Infrastructure

Martin Robinson

04/11/2015

New Version Control: Subversion -> Git

New build system: Scons -> CMake

New Continuous Integration: Buildbot



Figure : <https://git-scm.com>

- Chaste developers dispersing to other institutions
- Git provides a decentralised VC system
- more widely used for open-source projects

Obligatory xkcd comic



Figure : xkcd comic 1597

New Git Infrastructure

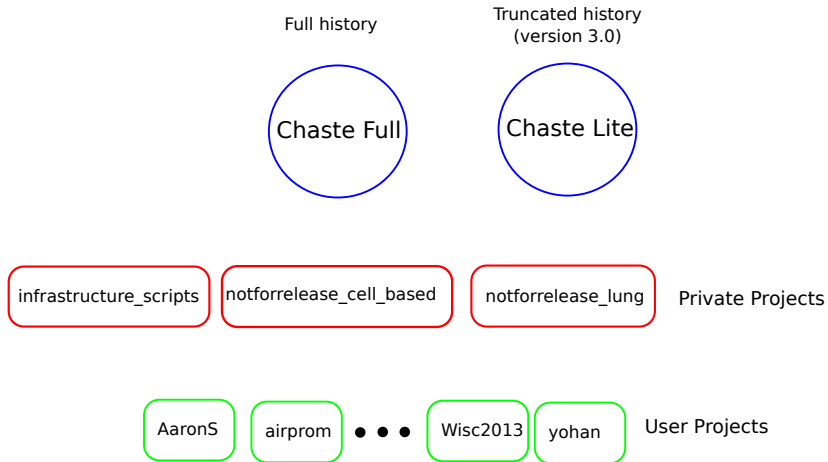


Figure : Chaste Git Repos

multiple branches

- **master**: latest commit that passed *all* tests
- **develop**: all development occurs here, starting point for CI testing
- **passed_continuous**: latest commit that passed continuous test pack
- **passed_nightly**: latest commit that passed continuous and nightly test packs
- **passed_lofty**: latest commit that passed continuous, nightly and lofty test pack
- **user_name/feature_name**: feature branches created as needed by developers (can be tested through buildbot web interface)

Example Workflows

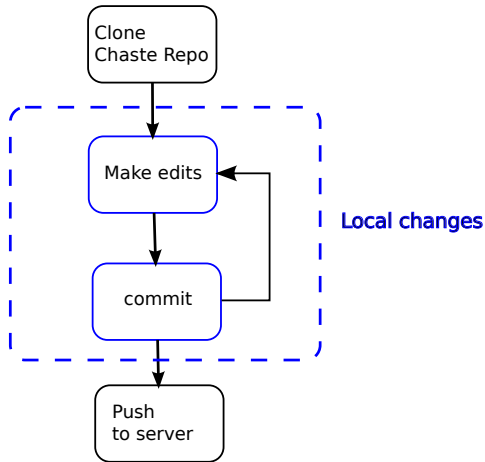


Figure : Minor edits

Example Workflows

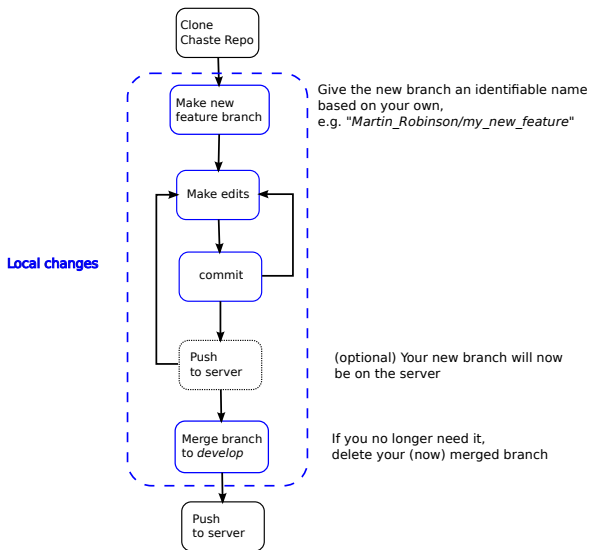


Figure : New features


```
$ git clone http://trix/git/chaste.git
```

```
$ git clone http://trix/git/projects/FunctionalCuration.git
```

Git resources

- Git documentation and book (<http://git-scm.com/doc>)
- Atlassian tutorials
(<https://www.atlassian.com/git/tutorials>)
- Software Carpentry Foundation
(<http://software-carpentry.org/>)



Figure : <https://cmake.org>

- better cross-platform support
- more widely used (e.g. many packages supply CMake Find scripts)
- faster? (leverages specialised build systems, e.g. Make)

- a build generator system
 - produces Makefiles, MSVC, Xcode, Eclipse projects
- 2-step process: 1) configuration, then 2) build
 - configure once, build often...
- command-line tool: `cmake`
- curses "Gui": `ccmake`
- also has testing (`ctest`) and packaging (`cpack`) applications

Chaste Dependencies

- For standard system layouts, should detect automatically
 - tested on Ubuntu 14.04, Mac OSX Homebrew, Windows (just scratch, with Petsc for Windows)
 - testers needed. other unix distros, other Mac build managers (e.g. MacPorts), Windows!!
- Can set environment variables to point to specific locations of dependencies
 - <https://chaste.cs.ox.ac.uk/trac/wiki/ChasteGuides/CmakeBuildGuide>

two options:

- 1 “Chaste Classic”: as normal, put your project in the `projects/` directory.
- 2 use Chaste as a system library (Can use whatever build system you want, but useful CMake macros are provided).

“Chaste Classic”

- 1 Put your project in the `projects/` directory. e.g.

```
$ cd projects
```

```
$ git clone http://trix/git/projects/myproject.git
```

- 2 Copy the following to
`projects/myproject/CMakeLists.txt`

```
find_package(Chaste COMPONENTS ${components})  
chaste_do_project(myproject)
```

- 3 Copy the following to
projects/myproject/test/CMakeLists.txt

```
chaste_do_test_project(myproject)
```

- 4 Copy the following to
projects/myproject/apps/CMakeLists.txt

```
chaste_do_test_apps(myproject)
```

- 5 Build it

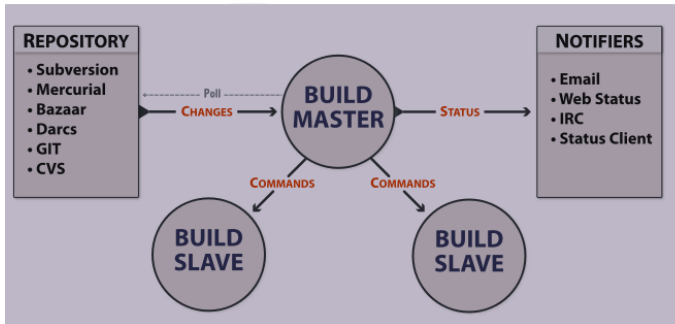
```
$ cd ..  
$ mkdir build  
$ cd build  
$ cmake ..  
$ make
```

Testers Needed

- checkout Chaste (subversion or git) and try it out
- Chaste CMake wiki page:
 - `https://chaste.cs.ox.ac.uk/trac/wiki/ChasteGuides/CmakeBuildGuide`
- feedback/bugs to:
 - `martin.robinson@cs.ox.ac.uk`
 - `chaste-users@maillist.ox.ac.uk`
 - raise ticket on trac

Buildbot CI

- replace custom python scripts with Buildbot CI server
- many extra features for “free”:
 - scheduling, tracking of individual commits
 - cross-platform support for build slaves
 - simplifies adding new slaves or builders
 - web interface



http://trix:8010/

The screenshot displays the Buildbot Waterfall interface. At the top, there's a browser window showing the URL `http://trix:8010/waterfall`. Below the browser window, there's a navigation bar with a search box and a 'login' button. The main area is a grid representing build activity over time. The columns represent different buildsets, and the rows represent time slots. Each cell in the grid contains a colored box indicating the build's status (e.g., 'successful', 'failed', 'waiting'). To the right of the grid, there's a vertical list of buildsets with their current status and a 'Work Pool' section.

GMT	changes	Continuous - DevOps	Continuous - GCC	Continuous - GCC-Parallel	Continuous - Intel	Continuous - Intel-Parallel	Continuous - Mac OS X	Luffy - T0xy	Luffy - Monday	Luffy - Saturday	Luffy - Sunday	Luffy - Thursday	Luffy - Tuesday	Luffy - Wednesday	Nightly - Coverage	Nightly - DevOps Coverage	Nightly - GCC-Parallel	Nightly - Intel-Nightly	Nightly - Mac OS X	Nightly - MemTest	Waterfall	Work Pool
Thu 29 Oct 2015		successful	successful	successful	successful	successful	failed test	waiting next in ~ 47 hrs 29 mins at 11:35	waiting next in ~ 126 hrs 19 mins at 11:35	waiting next in ~ 71 hrs 19 mins at 11:35	waiting next in ~ 95 hrs 19 mins at 11:35	waiting next in ~ 23 hrs 19 mins at 11:35	waiting next in ~ 143 hrs 19 mins at 11:35	waiting next in ~ 167 hrs 19 mins at 11:35	waiting next in ~ 22 hrs 5 mins at 10:01	waiting next in ~ 22 hrs 5 mins at 10:01	waiting next in ~ 22 hrs 5 mins at 10:01	waiting next in ~ 22 hrs 5 mins at 10:01	waiting next in ~ 22 hrs 5 mins at 10:01	waiting next in ~ 22 hrs 5 mins at 10:01	waiting next in ~ 71 hrs 19 mins at 11:35	waiting next in ~ 71 hrs 19 mins at 11:35
17:36:14																						
17:36:02																						
17:36:20																						
17:36:06																						
17:21:05																						
17:15:02																						
17:11:53																						
17:11:13																						
17:04:15																						
17:01:06																						
16:59:20																						
16:58:01																						
16:57:44																						
16:57:07																						
16:56:30																						
16:56:20																						
16:56:07																						
16:56:03																						
11:31:40																						
11:19:10																						
11:15:17																						

The End

- transition to Git occurs soon (Christmas 2015) after which subversion will be read-only
- for now, git repos are synced to subversion regularly (after today every weekend).
- CMake stuff is there now, try it out
- push a commit to the git repo and see it tested here