Be More {Productive, Collaborative} with {Git, GitHub}

Robert McGibbon, Kyle Beauchamp

SimTk Meeting

October 16, 2012

git :: overview

git :: overview

- (distributed) version control system
  - cvs, svn, hg, bzr

git :: overview

- (distributed) version control system
  - cvs, svn, hg, bzr
- support for distributed, non-linear development

git :: overview

- (distributed) version control system
  - cvs, svn, hg, bzr
- support for distributed, non-linear development
- originally developed for managing the linux kernel by linus torvalds

git :: overview

- (distributed) version control system
  - cvs, svn, hg, bzr
- support for distributed, non-linear development
- originally developed for managing the linux kernel by linus torvalds
- extremely flexible, can be complicated

git :: simple workflow

```
# make a local copy of a repository
$ git clone <url>
$ vi new_module.py

# stage your work
$ git add new_module.py

# commit early and often -- local
$ git commit -m “Sweet new feature!”

# upload your changes to the remote
$ git push
```

github :: overview

github :: overview

- collaborative code hosting web app
  - used by facebook, twitter, mozilla, ...
  - hosts numpy, scipy, ipython, matplotlib, ...

github :: overview

- collaborative code hosting web app
  - used by facebook, twitter, mozilla, ...
  - hosts numpy, scipy, ipython, matplotlib, ...
- $100M in VC funding this summer at $750M valuation

github :: overview

- collaborative code hosting web app
  - used by facebook, twitter, mozilla, ...
  - hosts numpy, scipy, ipython, matplotlib, ...
- $100M in VC funding this summer at $750M valuation
- center of the open source “do-ocracy”
  - forkable, decentralized

github :: collaborative

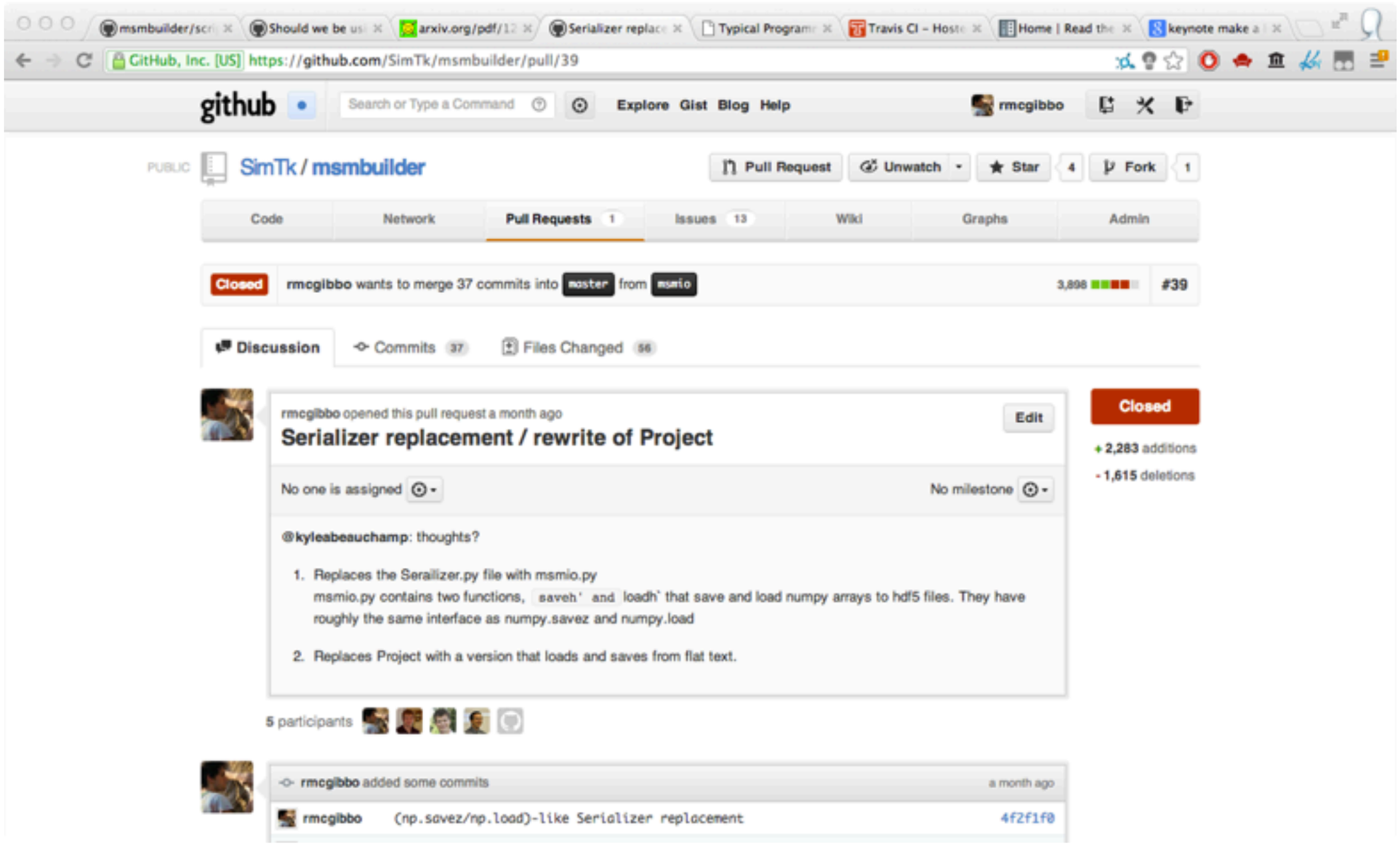
github :: collaborative

- pull requests
  - accept contributions from the community without giving up control.
  - manage concurrent development across features / branches

github :: collaborative

- pull requests
  - accept contributions from the community without giving up control.
  - manage concurrent development across features / branches
- issues
  - accept contributions from the community without giving up control.
  - manage concurrent development across features / branches

github :: pull requests



github :: pull requests

schwancr started a discussion in the diff 4 days ago

TestFiles/TestWrappers.py — View full changes

```
...  ...  @@ -181,16 +179,16 @@ def test_f_CalculateImpliedTimescales(self):
188       -        # make a predictable stream of random numbers by seeding the RNG with 42
189       -        random_source = np.random.RandomState(42)
190       -        randomconfs = GetRandomConfs.run(P1, Assignments, NumRandomConformations, random_source)
191       -
192       -        reference = Trajectory.load_trajectory_file(os.path.join(ReferenceDir, "2RandomConfs.lh5"))
193       -        self.assert_trajectories_equal(reference, randomconfs)
      182 +    # def test_g_GetRandomConfs(self):
      183 +    #     P1 = Project.load_from(ProjectFn)
      184 +    #     Assignments = io.loadh("Data/Assignments.Fixed.h5", 'arr_0')
      185 +    #
      186 +    #     # make a predictable stream of random numbers by seeding the RNG with 42
      187 +    #     random_source = np.random.RandomState(42)
      188 +    #     randomconfs = GetRandomConfs.run(P1, Assignments, NumRandomConformations, random_source)
      189 +    #
      190 +    #     reference = Trajectory.load_trajectory_file(os.path.join(ReferenceDir, "2RandomConfs.lh5"))
      191 +    #     self.assert_trajectories_equal(reference, randomconfs)
```

2

schwancr repo collab — 4 days ago

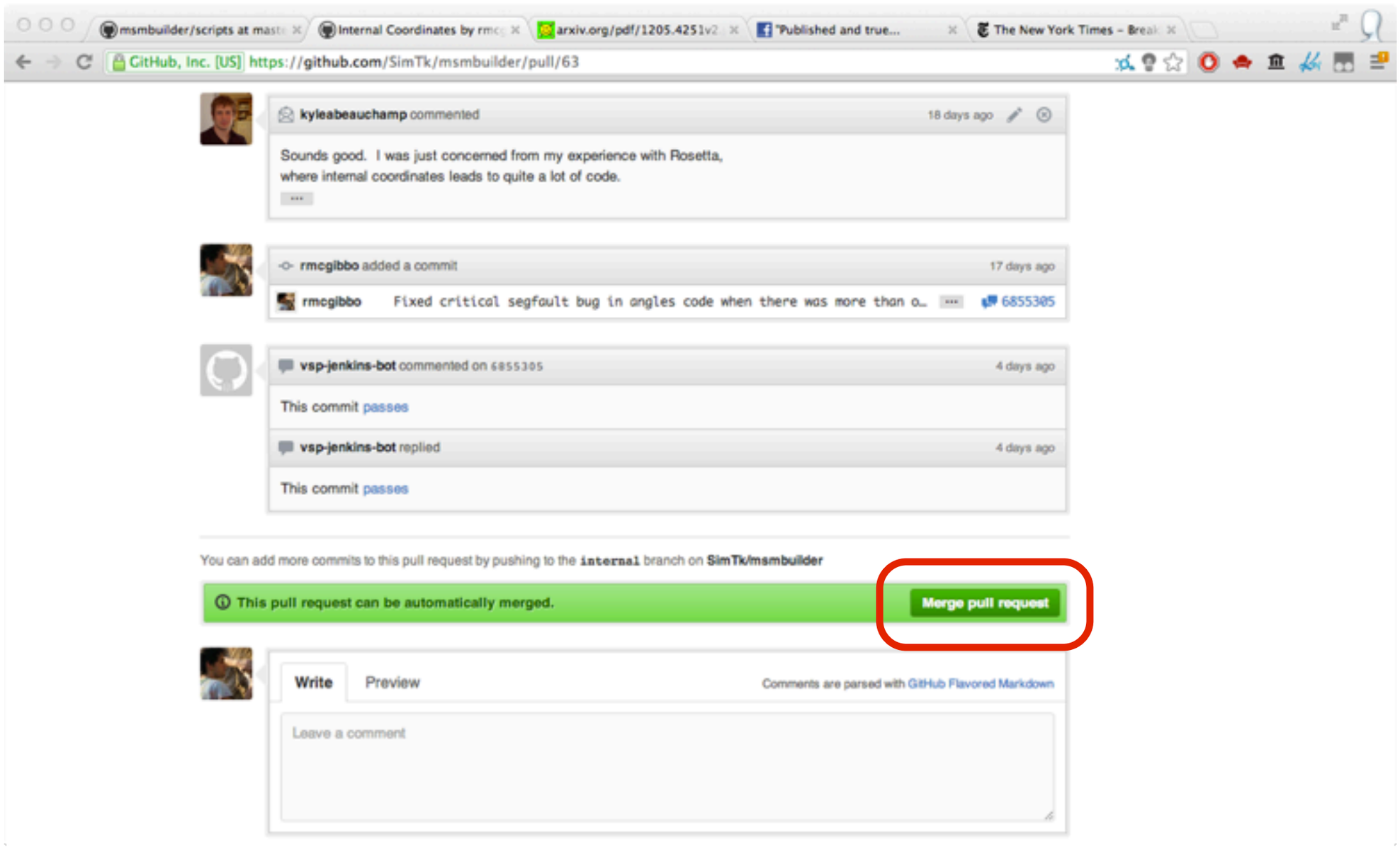
Why are these commented out?

rmcgibbo repo collab — 4 days ago

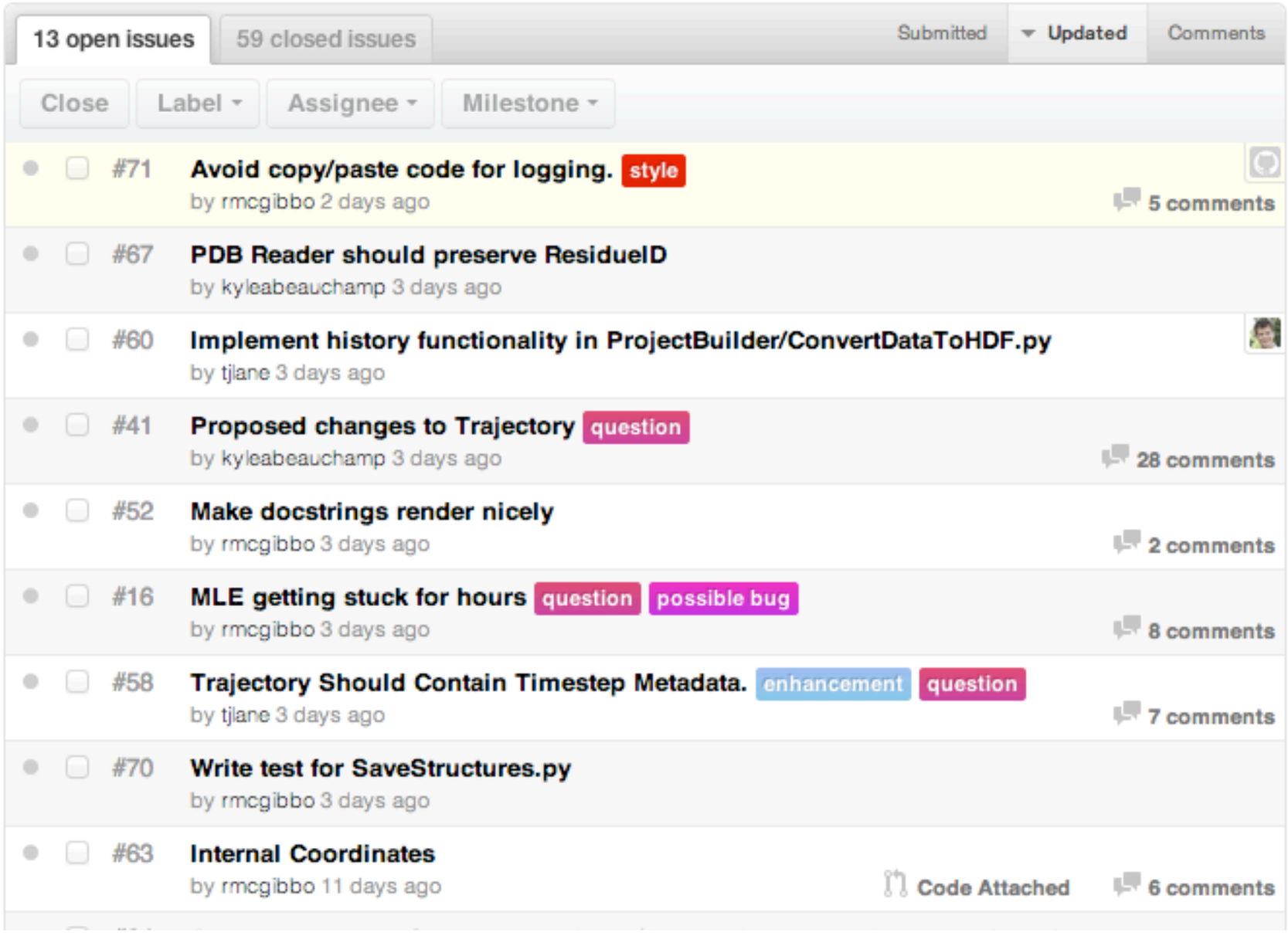
Because I was lazy, and when I deleted GetRandomConfs.py I didn't readjust them.

Add a line note

github :: pull requests



github :: issues



github :: issues

rmcgibbo commented — 25 days ago

@tjlane: Do you want to start hacking the fahproject stuff on this?

I think the `validators` mechanism here will generalize pretty well for the FaH stuff.

basically, you do

```
pb = ProjectBuilder(...)
pb.add_validator(validators.RMSDExplosionValidator(structure_or_filename='native.pdb', max_rmsd=0.1))

project = pb.get_project()
project.save('ProjectInfo.yaml')
```

tjlane commented — 25 days ago

Sweet! Nice work. Especially like the Validators -- I wouldn't worry about adding more, just documenting how they work so people can add them if necessary.

I'll get cracking on this in a bit.

github :: beautiful

github :: beautiful

- the little things make it useful
  - low friction, lightweight “project management”
  - code highlighting
  - diff discussion
  - reply to comments via email
  - web hooks

github :: extensible

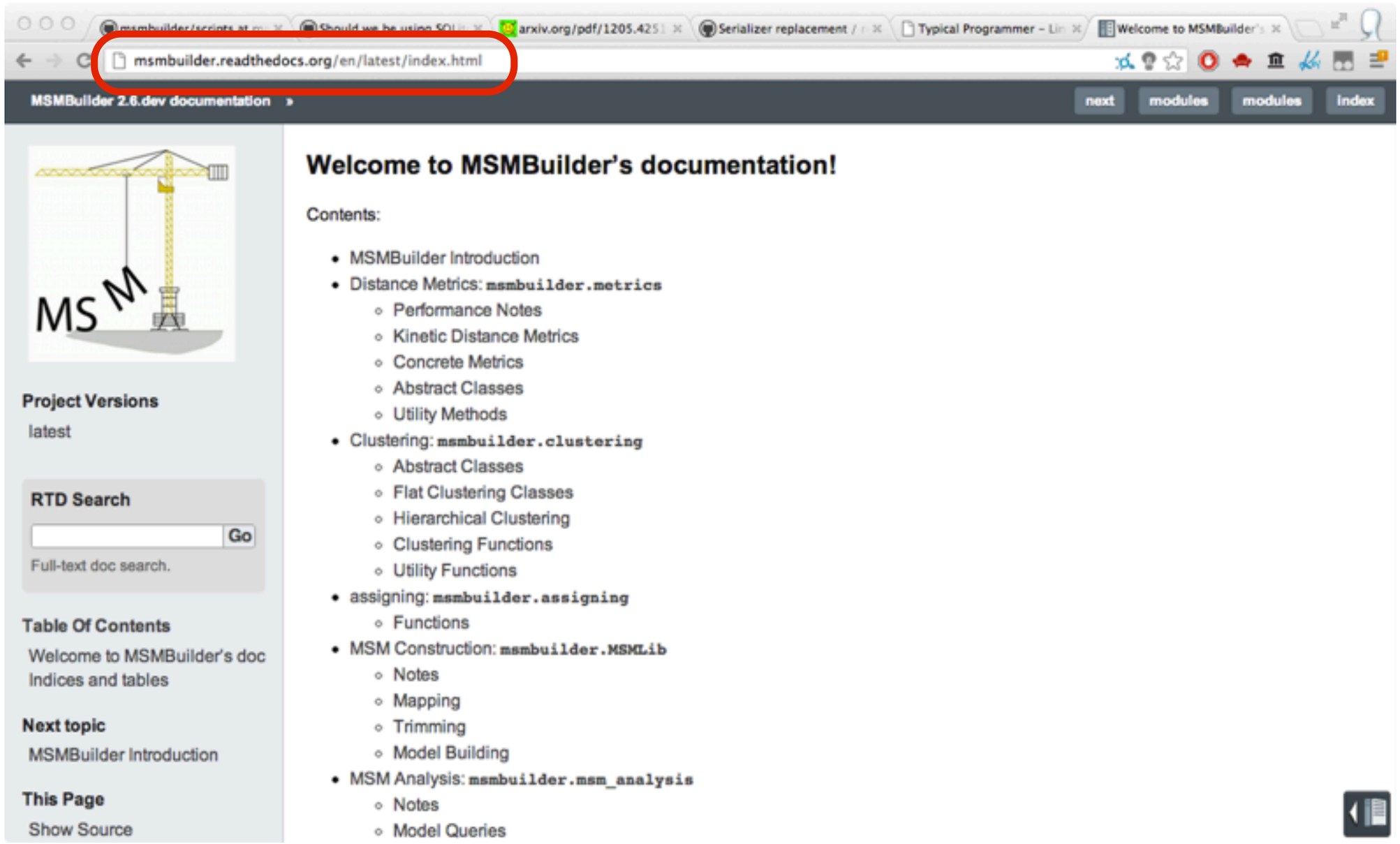
github :: extensible

- continuous integration
  - run the build/test suite after every commit across a matrix of configurations
  - TravisCI, CircleCI

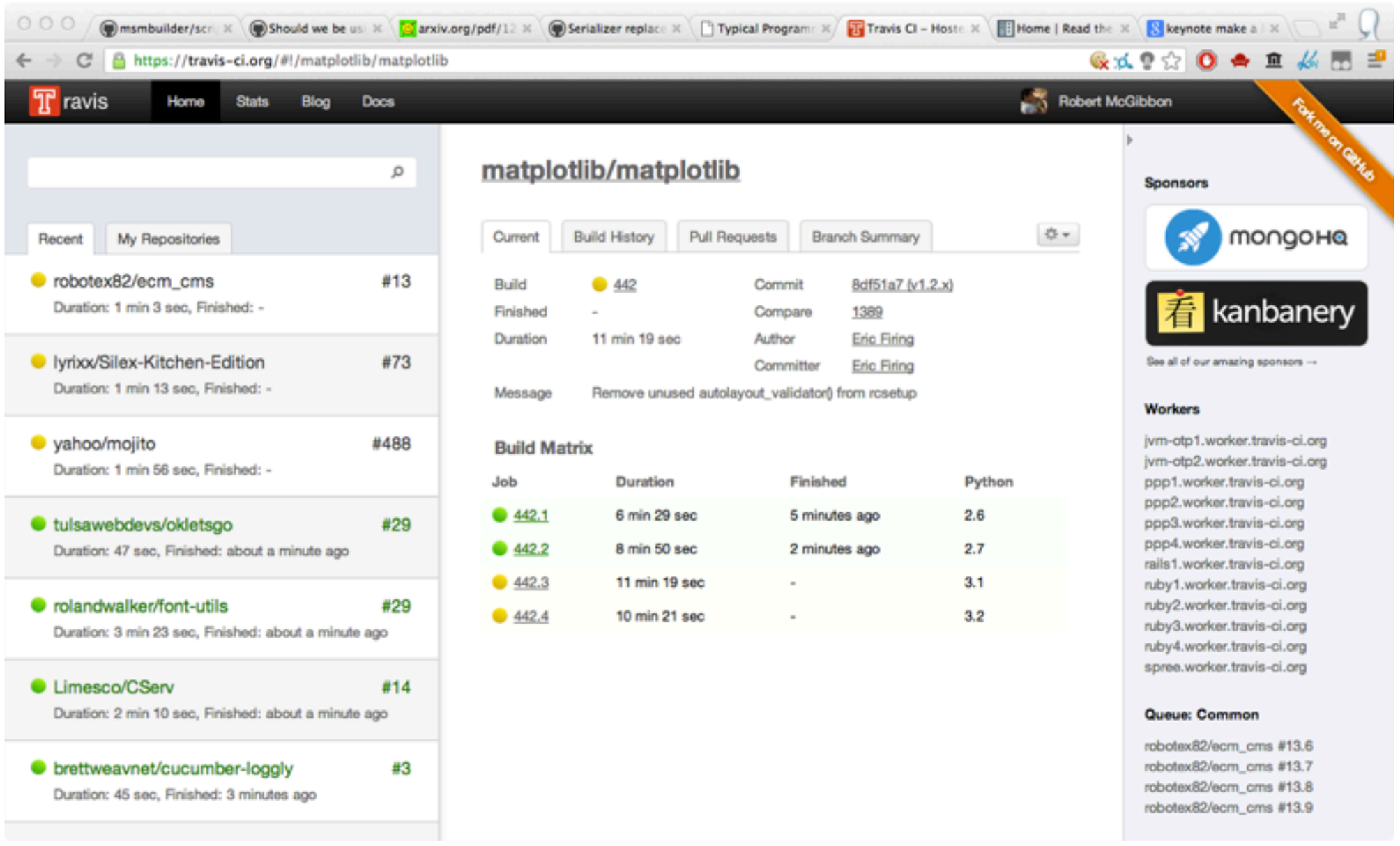
github :: extensible

- continuous integration
  - run the build/test suite after every commit across a matrix of configurations
  - TravisCI, CircleCI
- documentation generation and hosting
  - build the documentation, host it in the cloud
  - ReadTheDocs.org

github :: readthedocs.org



github :: travis-ci.org



github :: simtk.org synergy

github :: simtk.org synergy

- source code management is not our speciality; we can leverage external tools

github :: simtk.org synergy

- source code management is not our speciality; we can leverage external tools
- simtk.org and github have distinct and complementary core competencies
  - simtk.org as project “front end” to centralize and advertise
  - github as a “back end” for source control and developer tools

git :: criticisms

I didn't really expect anyone to use [Git] because it's so hard to use, but that turns out to be its big appeal. No technology can ever be too arcane or complicated for the black t-shirt crowd. - Fake Interview with Linus Torvalds