



# 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, bzt

# git :: overview

---

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

# git :: overview

---

- (distributed) version control system
  - cvs, svn, hg, bzt
- 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, bzz
- 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

---

Fork me on GitHub



# 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

---

Fork me on GitHub

# 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

The screenshot shows a GitHub pull request for the repository `SimTk/msmbuilder`. The pull request is titled "Serializer replacement / rewrite of Project" and is in a "Closed" state. It was opened by user `rmcgibbo` a month ago. The pull request description includes a list of changes:

- 1. Replaces the `Serializer.py` file with `msmio.py`. `msmio.py` contains two functions, `saveh` and `loadh`, that save and load numpy arrays to hdf5 files. They have roughly the same interface as `numpy.savez` and `numpy.load`.
- 2. Replaces `Project` with a version that loads and saves from flat text.

The pull request shows 37 commits and 56 files changed. It has 4 stars and 1 fork. The commit hash `4f2f1f0` is visible at the bottom.

# 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?

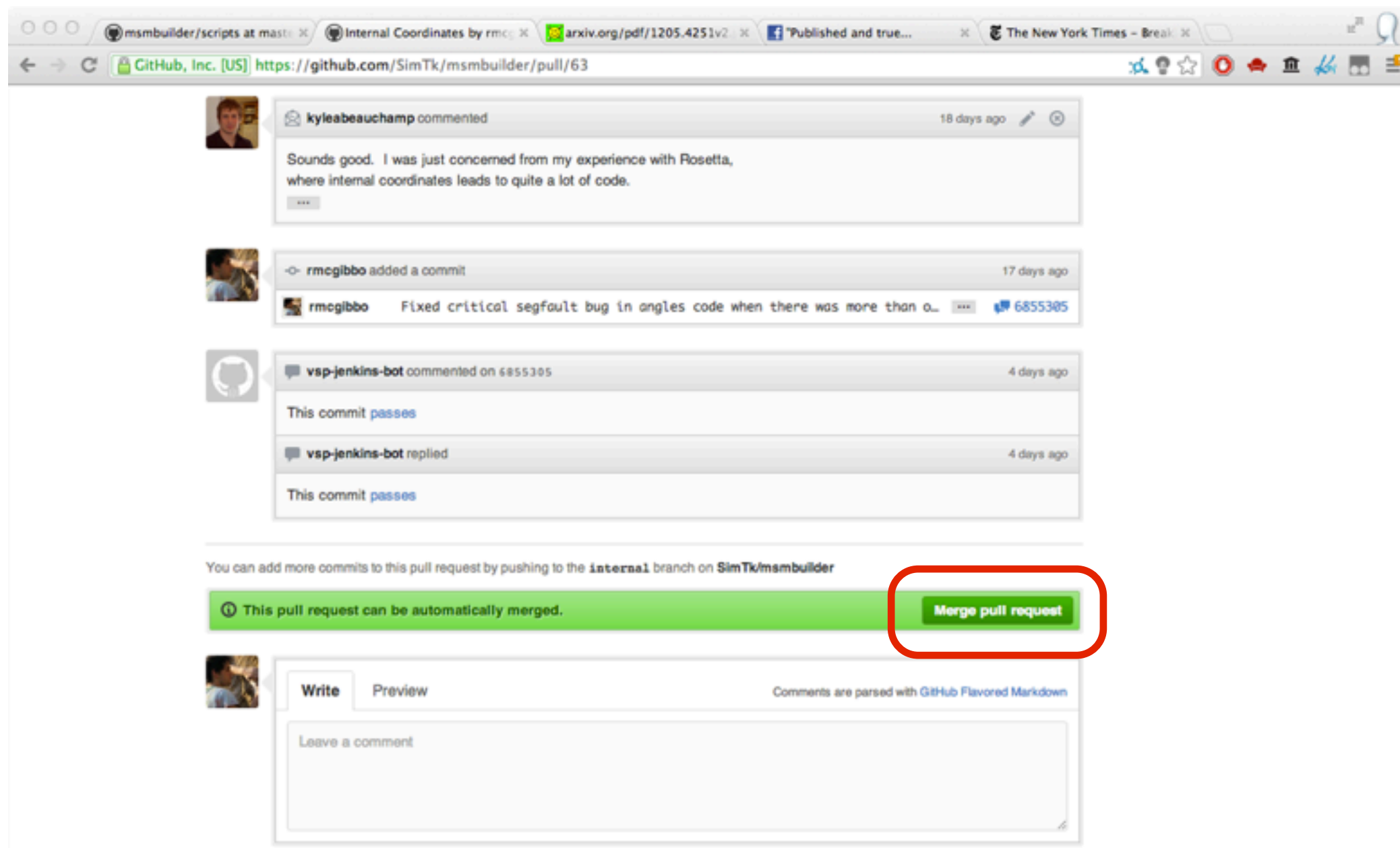
 rmcgibbo [repo collab](#) 4 days ago  

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

[Add a line note](#)



# github :: pull requests



# github :: issues

The screenshot shows a GitHub issues page with 13 open issues and 59 closed issues. The issues are listed in a table with columns for status, number, title, author, time, and comments. The issues are sorted by 'Updated'.

Status	Number	Title	Author	Time	Comments
Open	#71	Avoid copy/paste code for logging. <b>style</b>	rmcgibbo	2 days ago	5 comments
Open	#67	PDB Reader should preserve ResidueID	kyleabeauchamp	3 days ago	
Open	#60	Implement history functionality in ProjectBuilder/ConvertDataToHDF.py	tjlane	3 days ago	
Open	#41	Proposed changes to Trajectory <b>question</b>	kyleabeauchamp	3 days ago	28 comments
Open	#52	Make docstrings render nicely	rmcgibbo	3 days ago	2 comments
Open	#16	MLE getting stuck for hours <b>question</b> <b>possible bug</b>	rmcgibbo	3 days ago	8 comments
Open	#58	Trajectory Should Contain Timestep Metadata. <b>enhancement</b> <b>question</b>	tjlane	3 days ago	7 comments
Open	#70	Write test for SaveStructures.py	rmcgibbo	3 days ago	
Open	#63	Internal Coordinates	rmcgibbo	11 days ago	Code Attached 6 comments

# 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

---

Fork me on GitHub

# 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

---

Fork me on GitHub

# github :: extensible

---

Fork me on GitHub

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

# github :: extensible

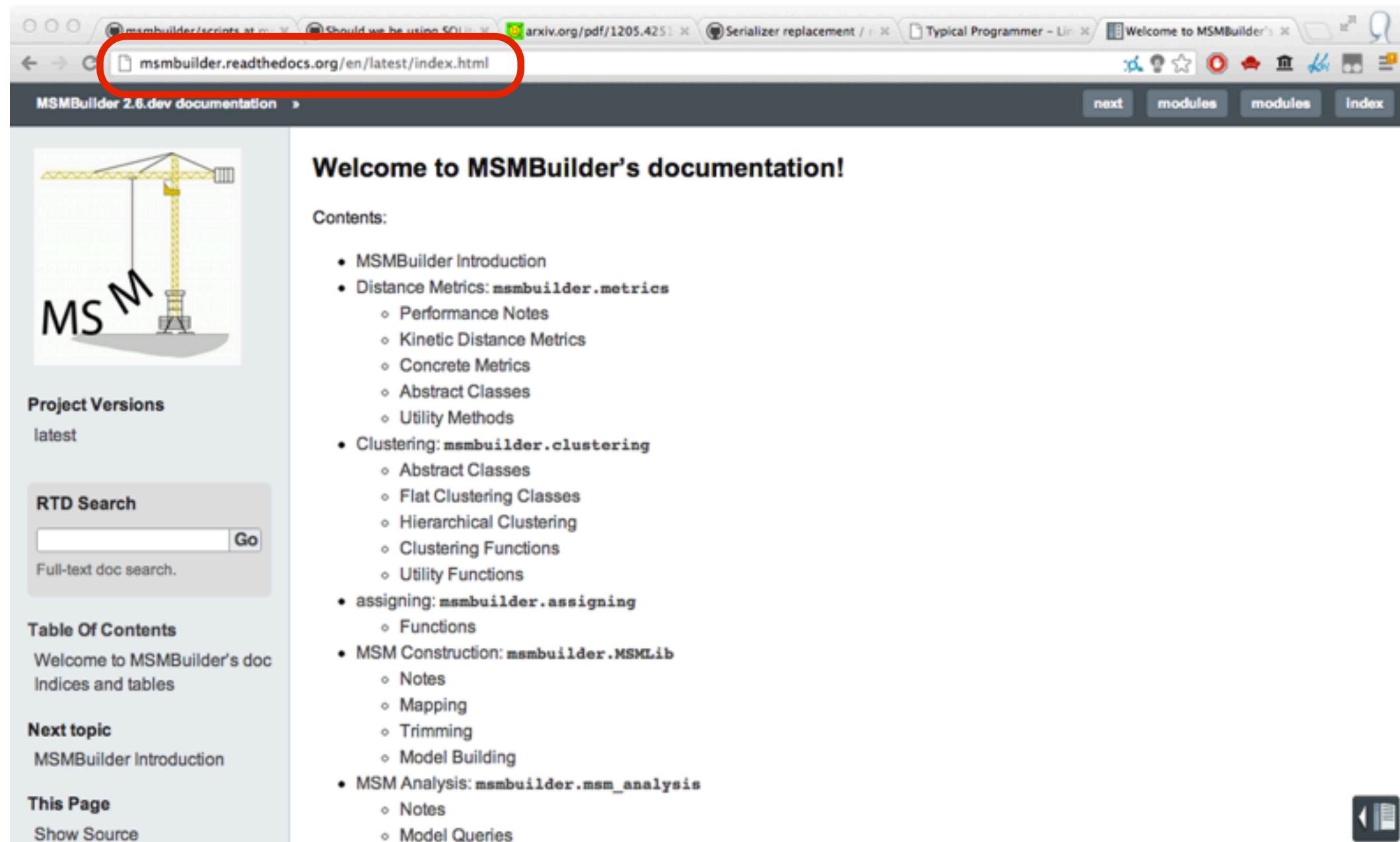
---

- 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

Fork me on GitHub



The screenshot shows a web browser with the address bar containing `msmbuilder.readthedocs.org/en/latest/index.html`, which is highlighted with a red circle. The page title is "MSMBuilder 2.6.dev documentation". The main content area is titled "Welcome to MSMBuilder's documentation!" and lists the following contents:

- MSMBuilder Introduction
- Distance Metrics: `msmbuilder.metrics`
  - Performance Notes
  - Kinetic Distance Metrics
  - Concrete Metrics
  - Abstract Classes
  - Utility Methods
- Clustering: `msmbuilder.clustering`
  - Abstract Classes
  - Flat Clustering Classes
  - Hierarchical Clustering
  - Clustering Functions
  - Utility Functions
- assigning: `msmbuilder.assigning`
  - Functions
- MSM Construction: `msmbuilder.MSMLib`
  - Notes
  - Mapping
  - Trimming
  - Model Building
- MSM Analysis: `msmbuilder.msm_analysis`
  - Notes
  - Model Queries

The left sidebar contains a logo with "MSM" and a crane, "Project Versions" (latest), an "RTD Search" box, a "Table Of Contents" link, a "Next topic" link (MSMBuilder Introduction), and a "This Page" link (Show Source).

# github :: travis-ci.org

The screenshot shows the Travis CI web interface for the `matplotlib/matplotlib` repository. The interface is divided into several sections:

- Left Sidebar:** A list of recent builds. Each entry includes a status icon (yellow for pending, green for successful), the repository name, build number, duration, and completion status.
 

Repository	Build #	Status	Duration	Finished
robotex82/ecm_cms	#13	Pending	1 min 3 sec	-
lyrix/Silex-Kitchen-Edition	#73	Pending	1 min 13 sec	-
yahoo/mojito	#488	Pending	1 min 56 sec	-
tulsawebdevs/okletsgo	#29	Success	47 sec	about a minute ago
rolandwalker/font-utils	#29	Success	3 min 23 sec	about a minute ago
Limesco/CServ	#14	Success	2 min 10 sec	about a minute ago
brettweavnet/cucumber-loggly	#3	Success	45 sec	3 minutes ago
- Center Panel:** Detailed view of the current build (442). It includes tabs for Current, Build History, Pull Requests, and Branch Summary. The build status is pending (yellow circle). The commit is `8df51a7 (v1.2.x)`. The build duration is 11 min 19 sec. The message is "Remove unused autolayout\_validator() from rcsetup".
 

Job	Duration	Finished	Python
442.1	6 min 29 sec	5 minutes ago	2.6
442.2	8 min 50 sec	2 minutes ago	2.7
442.3	11 min 19 sec	-	3.1
442.4	10 min 21 sec	-	3.2
- Right Sidebar:** Contains sponsors (mongoHQ, kanbanery), a list of workers (e.g., `jvm-otp1.worker.travis-ci.org`), and a queue of common builds.
 

**Queue: Common**

  - robotex82/ecm\_cms #13.6
  - robotex82/ecm\_cms #13.7
  - robotex82/ecm\_cms #13.8
  - robotex82/ecm\_cms #13.9

github :: simtk.org synergy

---

Fork me on GitHub

# github :: simtk.org synergy

---

Fork me on GitHub

- 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