
Tox Matrix Documentation

Release 0.1.1

Slawek Ehlert

February 03, 2017

1	Tox Matrix	3
1.1	Quick start	3
1.2	Features	4
2	Installation	5
3	Usage	7
4	Contributing	9
4.1	Types of Contributions	9
4.2	Get Started!	10
4.3	Pull Request Guidelines	10
4.4	Tips	11
5	Credits	13
5.1	Development Lead	13
5.2	Contributors	13
6	History	15
6.1	0.1.1 (2014-03-26)	15
6.2	0.1.0 (2014-03-23)	15
7	Indices and tables	17

Contents:

Tox Matrix

Project for creating `tox.ini` files with multiple dependencies.

WARNING! As of version 1.8.0 of tox it is possible to create generative envlist declarations. Described in [tox documentation](#). That feature of tox practically makes this project obsolete.

- Free software: BSD license
- Documentation: <http://tox-matrix.rtfid.org>.

A little utility script to generate `tox.ini` files. It should be handy in case where your project have multiple different dependencies with different version and you would like to test all available configurations of them.

1.1 Quick start

Install it:

```
pip install tox-matrix
```

Now let's assume you'd like to test your project against different versions of Django (latest 1.5.X and 1.6.X) and Fabric (latest 1.7.X and 1.8.X) using python2.7 and python3.3. You could do something like this:

```
tox-matrix generate -d Django -v 1.5,1.6,1.7 -d Fabric -v 1.7,1.8,1.9 -p 2.7 -p 3.3
```

Which should generate this `tox.ini` file content to stdout:

```
[tox]
envlist = py27-A, py27-B, py27-C, py27-D, py33-A, py33-B, py33-C, py33-D

[testenv]
commands = py.test

[testenv:py27-A]
basepython = python2.7
deps = {[testenv]deps}
      Django>=1.5,<1.6
      Fabric>=1.7,<1.8

[testenv:py27-B]
basepython = python2.7
deps = {[testenv]deps}
      Django>=1.5,<1.6
      Fabric>=1.8,<1.9
```

```
[testenv:py27-C]
basepython = python2.7
deps = {[testenv]deps}
      Django>=1.6,<1.7
      Fabric>=1.7,<1.8

[testenv:py27-D]
basepython = python2.7
deps = {[testenv]deps}
      Django>=1.6,<1.7
      Fabric>=1.8,<1.9

[testenv:py33-A]
basepython = python3.3
deps = {[testenv]deps}
      Django>=1.5,<1.6
      Fabric>=1.7,<1.8

[testenv:py33-B]
basepython = python3.3
deps = {[testenv]deps}
      Django>=1.5,<1.6
      Fabric>=1.8,<1.9

[testenv:py33-C]
basepython = python3.3
deps = {[testenv]deps}
      Django>=1.6,<1.7
      Fabric>=1.7,<1.8

[testenv:py33-D]
basepython = python3.3
deps = {[testenv]deps}
      Django>=1.6,<1.7
      Fabric>=1.8,<1.9
```

1.2 Features

- Generate tox.ini files
- Pin versions either with exact ones or by range
- Use a Jinja2 template for your tox.ini file

Installation

At the command line:

```
$ easy_install tox-matrix
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv tox-matrix  
$ pip install tox-matrix
```

Usage

To use Tox Matrix in a project:

```
import tox-matrix
```

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

4.1 Types of Contributions

4.1.1 Report Bugs

Report bugs at <https://github.com/slafs/tox-matrix/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

4.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

4.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

4.1.4 Write Documentation

Tox Matrix could always use more documentation, whether as part of the official Tox Matrix docs, in docstrings, or even on the web in blog posts, articles, and such.

4.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/slafs/tox-matrix/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

4.2 Get Started!

Ready to contribute? Here's how to set up *tox-matrix* for local development.

1. Fork the *tox-matrix* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/tox-matrix.git
```

3. Install your local copy into a virtualenv. Assuming you have *virtualenvwrapper* installed, this is how you set up your fork for local development:

```
$ mkvirtualenv tox-matrix
$ cd tox-matrix/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass *flake8* and the tests, including testing other Python versions with *tox*:

```
$ flake8 tox-matrix tests
$ python setup.py test
$ tox
```

To get *flake8* and *tox*, just *pip* install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

4.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in *README.rst*.
3. The pull request should work for Python 2.6, 2.7, and 3.3, and for PyPy. Check https://travis-ci.org/slafs/tox-matrix/pull_requests and make sure that the tests pass for all supported Python versions.

4.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_tox-matrix
```

Credits

5.1 Development Lead

- Slawek Ehlert <slafs@op.pl>

5.2 Contributors

None yet. Why not be the first?

History

6.1 0.1.1 (2014-03-26)

- Fixes for installation and tests

6.2 0.1.0 (2014-03-23)

- First release on PyPI.

Indices and tables

- `genindex`
- `modindex`
- `search`