
pyrpo Documentation

Release 0.2.1

Wes Turner

September 08, 2015

1	pyrpo	3
1.1	Features	3
1.2	Installing	3
1.3	Usage	4
1.4	License	4
2	Installation	5
3	Usage	7
4	pyrpo	9
4.1	pyrpo package	9
5	Contributing	11
5.1	Types of Contributions	11
5.2	Get Started!	12
5.3	Pull Request Guidelines	12
5.4	Tips	13
6	Credits	15
7	History	17
7.1	0.2.1 (2015-05-24)	17
7.2	0.2.0 (2015-04-25)	17
7.3	0.1.0 (2014-05-12)	17
8	License	19
9	Indices and tables	21
	Python Module Index	23

Contents:

[GitHub](#) | [PyPi](#) | [Warehouse](#) | [Documentation](#) | [Travis-CI](#) pyrpo: a shell command wrapper for hg, git, bzd, svn

1.1 Features

- Wrap and parse shell commands (largely as a reference)
- Walk for repository directories
- Generate reports for one or more repositories
- Call `hg status`, `git status`, etc.
- Generate mercurial `.hgsubs`
- Generate `git .git submodule`
- Generate `pip requirements.txt`
- Generate shell script (to rebuild environment)
 - TODO: replicate branches/tags/revisions
- Functional `namedtuples`, `iterators` `yield-ing` generators
- `optparse` argument parsing (`-h`, `--help`)
- `cookiecutter-pypackage` project templating

1.2 Installing

Install from PyPi:

```
pip install pyrpo
```

Install from [GitHub](#) as editable (add a `pyrpo.pth` in `site-packages`):

```
pip install -e git+https://github.com/westurner/pyrpo#egg=pyrpo
```

1.3 Usage

Print help:

```
pyrpo --help
```

Scan for files:

```
# Scan and print a shell report
pyrpo -s . -r sh
pyrpo
```

Generate a TortoiseHG thg-reporegistry.xml file:

```
pyrpo -s . --thg
```

Generate a pip report:

```
pyrpo -r pip
```

Generate a status report:

```
pyrpo -r status
```

Generate an *.hgsubs* file:

```
pyrpo -r hgsub
```

Generate a *.gitmodules* file:

```
pyrpo -r gitmodule
```

Generate an origin report:

```
pyrpo -r origin
```

Generate a string report:

```
pyrpo -r str
```

1.4 License

BSD Software License

Installation

At the command line:

```
$ easy_install pyrpo
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv pyrpo  
$ pip install pyrpo
```

Usage

To use pyrpo in a project:

```
import pyrpo
```

To use pyrpo as a shell command:

```
Usage: pyrpo [-h] [-v] [-q] [-s .] [-r <pip||full|status|hgsub|thg>] [--thg]
```

Options:

```
-h, --help          show this help message and exit
-s SCAN, --scan=SCAN Path(s) to scan for repositories
-r REPORTS, --report=REPORTS
                    pip || full || status || hgsub || thg
--thg              Write a thg-reporegistry.xml file to stdout
--template=REPORT_TEMPLATE
                    Report template
-v, --verbose
-q, --quiet
```


4.1 pyrpo package

4.1.1 Submodules

4.1.2 pyrpo.pyrpo module

4.1.3 Module contents

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:

5.1 Types of Contributions

5.1.1 Report Bugs

Report bugs at <https://github.com/westurner/pyrpo/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.

5.1.2 Fix Bugs

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

5.1.3 Implement Features

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

5.1.4 Write Documentation

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

5.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/westurner/pyrpo/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 :)

5.2 Get Started!

Ready to contribute? Here's how to set up *pyrpo* for local development.

1. Fork the *pyrpo* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/pyrpo.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 pyrpo
$ cd pyrpo/
$ 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 pyrpo 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.

5.3 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/westurner/pyrpo/pull_requests and make sure that the tests pass for all supported Python versions.

5.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_pyrpo
```

Credits

- Wes Turner – <https://github.com/westurner>

History

```
git log --format='* %s [%h]' master..develop
# [ ... ]
```

7.1 0.2.1 (2015-05-24)

- BUG: pyrpo.py: sh_full report: cat > %r << EOF [91d5fb7]
- BUG,CLN: pyrpo.py: logname='pyrpo' [e029abe]
- BLD: Makefile: pull, push, BROWSECMD lookups [59cbc66]
- BLD: Makefile: twine [d636e15]

7.2 0.2.0 (2015-04-25)

- Development: <https://github.com/westurner/pyrpo/commits/develop>
- Master: <https://github.com/westurner/pyrpo/commits/master>

7.3 0.1.0 (2014-05-12)

- First release on PyPI.
- Re-packaged from <https://github.com/westurner/dotfiles/blob/2813e4ad/scripts/repos.py>

License

Copyright (c) 2014, Wes Turner All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of pyrpo nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Indices and tables

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

p

pyrpo, 9

P

`pyrpo` (module), 9