
UpMe Documentation

Release 0.3.0

David Zuber

August 27, 2014

1	Documentation	3
2	Features	5
2.1	Welcome to UpMe 's documentation!	5
3	Contents:	7
3.1	Installation	7
3.2	Usage	7
3.3	Usermanual	7
3.4	Developer's Documentation	7
3.5	API Doc	7
3.6	Contributing	9
3.7	Credits	10
3.8	History	11
4	Feedback	13
4.1	Indices and tables	13
	Python Module Index	15

Detects outdated packages and updates them inside python scripts

Documentation

The full documentation is at <http://upme.rtd.org>.

Features

- Check if package or dependencies are outdated and update them. No commandline, plain python. Easy to integrate in scripts.

2.1 Welcome to UpMe's documentation!

Contents:

3.1 Installation

At the command line either via `easy_install` or `pip`:

```
$ easy_install upme
$ pip install upme
```

Or, if you have `virtualenvwrapper` installed:

```
$ mkvirtualenv upme
$ pip install upme
```

3.2 Usage

To use UpMe in a project:

```
from upme.main import is_outdated, update, restart

# check if the package is outdated
if is_outdated('pip'):
    # update a package and all dependencies
    # provide additional arguments for the pip install command
    update('pip', args=['-v'])
    # restart the application
    restart()
```

3.3 Usermanual

3.4 Developer's Documentation

Welcome to the developer's documentation. All necessary information for contributors who want to extend the project.

3.5 API Doc

Automatic generated Documentation by `apidoc` and `autodoc`.

3.5.1 upme

Submodules

`main`

`upme.main.get_required(dist)`

Return a set with all distributions that are required of `dist`

This also includes subdependencies and the given distribution.

Parameters `dist` (`pkg_resources.Distribution` | `str`) – the distribution to query. Can also be the name of the distribution

Returns a list of distributions that are required including the given one

Return type `set` of `pkg_resources.Distribution`

Raises `class:pkg_resources.DistributionNotFound`

`upme.main.is_outdated(dist, dep=False)`

Return a dict with outdated distributions

If the given distribution has dependencies, they are checked as well.

Parameters

- **dist** (`pkg_resources.Distribution` | `str`) – a distribution to check
- **dep** – If `True`, also return all outdated dependencies. If `False`, only check given `dist`.

Returns dictionary of all distributions that are outdated and are either dependencies of the given distribution or the distribution itself. Keys are the outdated distributions and values are the newest parsed versions.

Return type `dict` of `pkg_resources.Distribution`

Raises `class:pkg_resources.DistributionNotFound`

`upme.main.update(dist, args=None)`

Update the given distribution and all of its dependencies

Parameters

- **dist** (`pkg_resources.Distribution` | `str`) – the distribution to check
- **args** (`list`) – extra arguments for the install command. this is somewhat equivalent to: `pip install -U <dist> args`

Returns `None`

Return type `None`

Raises `class:pkg_resources.DistributionNotFound`

`upme.main.restart()`

Restart the application the same way it was started

Returns `None`

Return type `None`

Raises `SystemExit`

Module contents

3.6 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:

3.6.1 Types of Contributions

Report Bugs

Report bugs at <https://github.com/storax/upme/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.

Fix Bugs

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

Implement Features

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

Write Documentation

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

Submit Feedback

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

3.6.2 Get Started!

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

1. Fork the *upme* repo on GitHub.

2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/upme.git
```

3. Create a branch for local development:

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

Now you can make your changes locally.

4. When you're done making changes, check that your changes pass style and unit tests, including testing other Python versions with tox:

```
$ tox
```

To get tox, just pip install it.

5. 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
```

6. Submit a pull request through the GitHub website.

3.6.3 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/storax/upme> under pull requests for active pull requests or run the `tox` command and make sure that the tests pass for all supported Python versions.

3.6.4 Tips

To run a subset of tests:

```
$ py.test test/test_upme.py
```

3.7 Credits

3.7.1 Development Lead

- David Zuber <zuber.david@gmx.de>

3.7.2 Contributors

None yet. Why not be the first?

3.8 History

3.8.1 0.1.0 (2014-08-25)

- First release on PyPI.

3.8.2 0.2.0 (2014-08-26)

- Core functionality

Feedback

If you have any suggestions or questions about **UpMe** feel free to email me at zuber.david@gmx.de.

If you encounter any errors or problems with **UpMe**, please let me know! Open an Issue at the GitHub <https://github.com/storax/upme> main repository.

4.1 Indices and tables

- *genindex*
- *modindex*
- *search*

U

upme, 9

upme.main, 8