
Hashdex Documentation

Release 0.6.0

Jeffrey Slort

Sep 24, 2019

Contents

1	Hashdex	3
1.1	As easy as..	3
1.2	Features	3
1.3	Credits	4
2	Installation	5
2.1	Stable release	5
2.2	From sources	5
3	Usage	7
3.1	Create an Hashdex	7
3.2	Check new files against index	7
3.3	Find duplicate files	8
3.4	Cleanup the index	8
4	Contributing	9
4.1	Types of Contributions	9
4.2	Get Started!	10
4.3	Pull Request Guidelines	11
4.4	Tips	11
5	Credits	13
5.1	Development Lead	13
5.2	Contributors	13
6	History	15
6.1	0.6.0 (2017-10-28)	15
6.2	0.5.0 (2017-09-28)	15
6.3	0.4.1 (2017-09-24)	15
6.4	0.4.0 (2017-09-24)	16
6.5	0.3.0 (2017-09-12)	16
6.6	0.2.0 (2017-09-08)	16
6.7	0.1.1 (2017-09-05)	16
6.8	0.1.0 (2017-09-04)	16
7	Indices and tables	17

Contents:

CHAPTER 1

Hashdex

A file indexer based on content hashes to quickly find duplicate files on your system. I created this tool because I always forget which pictures from my phone I already uploaded to my Dropbox account. Now I can just upload all pictures to an uploads directory check these files against the index and only organize the remaining files.

1.1 As easy as..

```
pip install hashdex
hashdex add /path/to/my-main-pictures-directory
hashdex check --rm /path/to/my-uploads-directory
```

- Free software: MIT license
- Documentation: <https://hashdex.readthedocs.io>.

1.2 Features

- create an index of your files
- find duplicate files on your filesystem
- check if files in a directory are already indexed

1.3 Credits

This package was created with [Cookiecutter](#) and the [audreyr/cookiecutter-pypackage](#) project template.

2.1 Stable release

To install Hashdex, run this command in your terminal:

```
$ pip install hashdex
```

This is the preferred method to install Hashdex, as it will always install the most recent stable release.

If you don't have [pip](#) installed, this [Python installation guide](#) can guide you through the process.

2.2 From sources

The sources for Hashdex can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/jeff-99/hashdex
```

Or download the [tarball](#):

```
$ curl -OL https://github.com/jeff-99/hashdex/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```


Hashdex is a command-line utility for quickly checking your files for duplicates

3.1 Create an Hashdex

To initialize an hashdex you need to add some files/directories to the index..

```
# directory
hashdex add /path/to/directory

# file
hashdex add /path/to/file.txt
```

This will create an index file in the users home folder (`~/.config/hashdex/index.db`) If you want to create an index in another location you can specify the `-index` option to the command

```
hashdex add --index /path/to/index.db /path/to/directory
```

Once you have added all the necessary directories to the index we can begin checking directories against the index. To check all files in a directory against the index execute the following

3.2 Check new files against index

```
# directory
hashdex check --index /path/to/index.db /path/to/directory/to/check

# file
hashdex check /path/to/file.txt
```

This will list all files in the given directory which are already indexed with the indexed file path. You can add the `-rm` flag to delete all files in the given directory which are found in the index, so you will be left with only new files. In

addition to the **-rm** flag you can also pass an **-mv** option with an existing path to move duplicate files to the given directory.

```
# remove duplicates
hashdex check --rm /path/to/directory/to/check

# move duplicates
hashdex check --mv ./duplicates /path/to/directory/to/check
```

3.3 Find duplicate files

To get a list of all duplicate files within already indexed files you can execute:

```
hashdex duplicates --index /path/to/index.db
```

3.4 Cleanup the index

Overtime files will be deleted from your system or moved to another location. So once in a while you should clean up the index with the following command:

```
hashdex cleanup --index /path/to/index.db
```

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/jeff-99/hashdex/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” and “help wanted” is open to whoever wants to implement it.

4.1.3 Implement Features

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

4.1.4 Write Documentation

Hashdex could always use more documentation, whether as part of the official Hashdex 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/jeff-99/hashdex/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 *hashdex* for local development.

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

```
$ git clone git@github.com:your_name_here/hashdex.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 hashdex
$ cd hashdex/
$ 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 hashdex tests
$ python setup.py test or 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, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/jeff-99/hashdex/pull_requests and make sure that the tests pass for all supported Python versions.

4.4 Tips

To run a subset of tests:

```
$ py.test tests.test_hashdex
```


5.1 Development Lead

- Jeffrey Slort <j_slort@hotmail.com>

5.2 Contributors

None yet. Why not be the first?

6.1 0.6.0 (2017-10-28)

- Added version option to main command
- Added option to move files while checking for duplicates
- Updated pytest-runner to 3.0.0
- Updated cryptography to 2.1.2
- Updated sphinx to 1.6.5
- Updated flake8 to 3.5.0
- Updated tox to 2.9.1
- Updated pytest to 3.2.3

6.2 0.5.0 (2017-09-28)

- added support for adding and checking files instead of directories
- moved default index file to users home directory
- Updated dependencies

6.3 0.4.1 (2017-09-24)

- adjusted for python 2.7 math.floor returning a float

6.4 0.4.0 (2017-09-24)

- renamed the index command to ‘add’
- refactored –dir option for add/check command to cli argument
- added cleanup command
- added docs
- updated dependencies

6.5 0.3.0 (2017-09-12)

- Added –rm flag to check cli command (without the flag the command will not delete files)
- Dropped python 2.6 & 3.3 support
- Updated dependencies

6.6 0.2.0 (2017-09-08)

- Refactored Indexer class
- Added Hasher class
- Added tests 100% code coverage
- Updated dependencies

6.7 0.1.1 (2017-09-05)

- Updated dependencies
- Fixed file listing bug

6.8 0.1.0 (2017-09-04)

- First release on PyPI.

CHAPTER 7

Indices and tables

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