

[Help](#)[Sponsors](#)[Log in](#)[Register](#)

# dircsync 2.2.5

[Latest version](#)`pip install dircsync`

Released: Jul 31, 2020

Advanced directory tree synchronisation tool

## Navigation

[Project description](#)[Release history](#)[Download files](#)

## Project links

[Homepage](#)

## Statistics

GitHub statistics:

[★ Stars: 50](#)

## Project description

© 2014-2019 Thomas Khyn © 2003-2015 Anand B Pillai

Advanced directory tree synchronisation tool

based on [Python robocopier](#) by Anand B Pillai

If you like dircsync and are looking for a way to thank me and/or encourage future development, here is my BTC or BCH donation address: `1EwENyR8RV6tMc1hsLTkPURtn5wJgaBfG9`.


## Usage

From the command line:

```
dircsync <sourcedir> <targetdir> [options]
```

From python:

 **Forks: 19**


 **Open**  
**issues/PRs: 28**

View statistics for this project via [Libraries.io](#), or by using [our public dataset on Google BigQuery](#)

Meta

License: MIT License

Author: [Thomas Khyn](#)

 directory, folder, update, synchronisation

Maintainers

 [tkhyn](#)

Classifiers

Development Status

- 5 - [Production/Stable](#)

Environment

- [Console](#)

Intended Audience

- [Developers](#)
- [End Users/Desktop](#)

License

```
from dirsync import sync
sync(sourcedir, targetdir, action, **options)
```

Main Options

Chosing one option among the following ones is mandatory

<code>--diff, -d</code>	Only report difference between sourcedir and targetdir
<code>--sync, -s</code>	Synchronize content between sourcedir and targetdir
<code>--update, -u</code>	Update existing content between sourcedir and targetdir

If you use one of the above options (e.g. `sync`) most of the time, you may consider defining the `action` option in a [Configuration file](#) parsed by `dirsync`.

Additional Options

<code>--verbose, -v</code>	Provide verbose output
<code>--purge, -p</code>	Purge files when synchronizing (does not purge by default)
<code>--force, -f</code>	Force copying of files, by trying to change file permissions
<code>--twoway, -2</code>	Update files in source directory from target directory (only updates target from source by default)
<code>--</code>	Create target directory if it does not exist (By default,

- [OSI Approved :: MIT License](#)

## Operating System

- [OS Independent](#)

## Programming

### Language

- [Python](#)
- [Python :: 2](#)
- [Python :: 3](#)

## Topic

- [Desktop Environment](#)
- [System :: Archiving :: Backup](#)
- [System :: Archiving :: Mirroring](#)
- [Utilities](#)



Motional is a Contributing sponsor of the Python Software Foundation.

[PSF Sponsor](#) · [Served ethically](#)

<code>create,</code> <code>-c</code>	target directory should exist.)
<code>--ctime</code>	Also takes into account the source file's creation time (Windows) or the source file's last metadata change (Unix)
<code>--</code> <code>content</code>	Takes into account ONLY content of files. Synchronize ONLY different files. At two-way synchronization source files content have priority if destination and source are existed
<code>--</code> <code>ignore,</code> <code>-x</code> <code>patterns</code>	
	Regex patterns to ignore
<code>--only,</code> <code>-o</code> <code>patterns</code>	
	Regex patterns to include (exclude every other)
<code>--</code> <code>exclude,</code> <code>-e</code> <code>patterns</code>	
	Regex patterns to exclude
<code>--</code> <code>include,</code> <code>-i</code> <code>patterns</code>	
	Regex patterns to include (with precedence over excludes)

## Configuration file

### Note

Configuration files are only used when using the command line, and ignored when dirsync is called from within Python.

If you want to use predefined options all the time, or if you need specific options when 'dirsyncing' a specific source directory, dirsync looks for two configuration files, by order or priority (the last takes precedence):

```
~/dirsync
source/directory/.dirsync
```

### Note

A ~/dirsync configuration file is automatically created the first time dirsync is ran from the command line. It enables `sync` mode by default.

### Warning

Any `source/directory/.dirsync` file is automatically excluded from the files to compare. You have to explicitly include using the `--include` option if you want it to be covered by the comparison.

The command line options always override the values defined in the configuration files.

The configuration files must have a `defaults` section, and the options are as defined above. The only exception is for the option `action`, which can take 3 values `diff`, `sync` or `update`.

Example config file:

```
[defaults]
action = sync
create = True
```

## Custom logger

From python, you may not want to have the output sent to `stdout`. To do so, you can simply pass your custom logger via the `logger` keyword argument of the `sync` function:

```
sync(sourcedir, targetdir, action, logger=my_logger, **o
```



## Help

[Installing packages](#)

[Uploading packages](#)

[User guide](#)

[FAQs](#)

## About PyPI

[PyPI on Twitter](#)

[Infrastructure dashboard](#)

[Package index name retention](#)

[Our sponsors](#)

## Contributing to PyPI

[Bugs and feedback](#)

[Contribute on GitHub](#)

[Translate PyPI](#)

[Development credits](#)

## Using PyPI

[Code of conduct](#)

[Report security issue](#)

[Privacy policy](#)

[Terms of use](#)

---

Status: [All Systems Operational](#)

Developed and maintained by the Python community, for the Python community.

[Donate today!](#)

© 2022 [Python Software Foundation](#)

[Site map](#)

[Switch to desktop version](#)

► [English](#) [español](#) [français](#) [日本語](#) [português \(Brasil\)](#) [українська](#) [Ελληνικά](#) [Deutsch](#) [中文 \(简体\)](#)  
[中文 \(繁體\)](#) [русский](#) [עברית](#) [esperanto](#)