
doc2dash Documentation

Release 2.0.0

Hynek Schlawack

August 14, 2014

1	User's Guide	3
1.1	Requirements and Installation	3
1.2	Supported Input Formats	3
1.3	Usage	4
1.4	Project Information	5
2	Indices and tables	9

Release v2.0.0 (*What's new?*).

doc2dash is an MIT-licensed extensible [Documentation Set](#) generator intended to be used with the [Dash.app](#) API browser for OS X or one of its many clones.

If you've never heard of Dash.app, you're missing out: together with doc2dash it's all your API documentation at your fingertips!

doc2dash's documentation lives at [Read the Docs](#), the code on [GitHub](#). It's tested on Python 2.7, 3.3+, and PyPy. Both Linux and OS X are supported although certain features are only available on OS X.

1.1 Requirements and Installation

The latest stable version can be always found on [PyPI](#).

Since it's a stand-alone tool, I recommend to install it using [pipsi](#).

Of course, you can also install it as usual using [pip](#):

```
$ pip install --user doc2dash
```

`doc2dash` runs on Python 2.7, and 3.3+, and PyPy. Both Linux and OS X are supported although certain features are only available on OS X.

Note: For best performance when converting large pieces documentation, I *strongly* recommend using PyPy as the interpreter of choice.

1.1.1 Viewer

To view the results, you will need a docset viewer, the most commonly known being [Dash.app](#) for OS X.

Other alternatives have been developed in cooperation with Dash.app's developer [Kapeli](#):

- [helm-dash](#) for Emacs,
- [lovelydocs](#) for Android,
- [velocity](#) for Windows,
- and [zeal](#) for Linux and Windows.

`doc2dash` is only tested against the original Dash.app though.

1.2 Supported Input Formats

Currently, `doc2dash` supports two documentation formats:

- *Sphinx*
- *pydoctor*

Feel free to help adding more! While `doc2dash` is implemented in Python, the scope for the supported documentation types is unlimited. So go on and submit a parser for your favourite Ruby, Haskell, Lisp, Erlang, JavaScript, ... format!

1.2.1 Sphinx

`Sphinx` is a very common documentation format in the Python world and beyond.

`doc2dash` offers two approaches to parsing it. The preferable one is used whenever a machine-readable `intersphinx` index file is present and it results in very precise and reliable parsing.

If none is found, `doc2dash` attempts to parse the HTML API index file (`genindex.html` or `genindex-all.html`). Simply point `doc2dash` at Sphinx's HTML output (usually `_build/html` if you built it yourself) and it will do the right thing.

1.2.2 pydoctor

Contrary to Sphinx, `pydoctor` is not a complete documentation format. Instead, it's focused on creating API documentation from Python docstrings. The most popular project employing is `Twisted` and its ecosystem.

Since pydoctor alas does not emit a machine-readable file, the `nameIndex.html` is always parsed. Fortunately, no theming is common in the pydoctor world, so the parsing is reliable nonetheless.

1.3 Usage

The usage is as simple as:

```
$ doc2dash -A <docdir>
```

`doc2dash` will create a new directory called `<docdir>.docset` in `~/Library/Application Support/doc2dash/DocSets` containing a Dash.app-compatible docset. When finished, the docset is automatically added to Dash.app.

1.3.1 Options and Arguments

The full usage is:

```
$ doc2dash [OPTIONS] SOURCE
```

The `SOURCE` is a directory containing the documentation you would like to convert. Valid `OPTIONS` are the following:

- n, --name**
Name the docset explicitly instead of letting `doc2dash` guess the correct name from the directory name of the source.
- d PATH, --destination PATH**
Put the resulting docset into `PATH`. Default is the current directory. Ignored if `--add-to-global` is specified.
- f, --force**
Overwrite docset if it already exists. Otherwise, `doc2dash` would exit with an error.
- i FILENAME, --icon FILENAME**
Add PNG icon `FILENAME` to docset that is used within Dash.app to represent the docset.

- I FILENAME, --index-page FILENAME**
Set the file that is shown when the docset is clicked within Dash.app.
- a, --add-to-dash**
Automatically add the resulting docset to Dash.app. Works only on OS X and when Dash.app is installed.
- A, --add-to-global**
Create docset in doc2dash's default global directory [~/Library/Application Support/doc2dash/DocSets] and add it to Dash.app Works only on OS X and when Dash.app is installed.
- q, --quiet**
Limit output to errors and warnings.
- v, --verbose**
Be verbose.
- version**
Show the version and exit.
- help**
Show a brief usage summary and exit.

1.4 Project Information

1.4.1 Changelog

2.0.0 (2014-08-14)

- Added a new parser for Sphinx documentation that uses [intersphinx](#) files that are machine readable. That should lead to more reliable parsing and a better deduction of symbol types. (#28)
- Added Sphinx-based documentation.
- Added colors, styles, and a progress bar to make output more comprehensible.
- `setup.py test` works now.
- Internally quite a few changes happened. Most prominently tuples and namedtuples have been replaced by real classes and parsers don't inherit from anything anymore. The fundamental working principal stayed the same though so porting your parsers is trivial.

1.2.1 (2014-07-24)

- Fix docset name deduction if the source path ends with a `/`. (#26)

1.2.0 (2014-01-07)

- Runs now on Python 3.3. This is achieved by upgrading dependencies that didn't play along well before on 3.3.
- Add `--index` option.

1.1.0 (2013-01-13)

- Use better dash types for modules and attributes.

1.0.0 (2012-10-14)

- Make tests pass on Python 2.7 too.
- Due to lack of known problems, pronouncing stable, thus 1.0.0.
- Please note that no code except for the tests has changed since 0.3.1.

0.3.1 (2012-06-28)

- Pronounced beta – happy testing!

0.3.0 (2012-06-28)

- Add table of contents links to docs to get a nice TOC in dash when inside of a module.
- Support DashDocSetFamily field.
- Add `--verbose` and `--quiet` options.
- Add `--destination` option.
- Add `--add-to-dash` option.
- Allow adding of an PNG icon to the docset (`--icon`).

0.2.2 (2012-06-16)

- Don't collect `()` as part of method/function names.
- Index whole names: symbols are searchable by the whole name, including the namespace.

0.2.1 (2012-06-15)

- Fix PyPI package: add missing MANIFEST.in and add missing packages to setup.py.

0.2.0 (2012-06-14)

- Add support for built-in constants and classes.
- Strip annotations from unused remembered names the are re-used in synonyms.

1.4.2 License and Hall of Fame

doc2dash is licensed under the permissive [MIT](#) license. The full license text can be also found in the [source code repository](#).

Credits

doc2dash is written and maintained by Hynek Schlawack and various contributors:

- Bogdan Popescu
- Dirkjan Ochtman
- Łukasz Langa

1.4.3 How To Contribute

Every open source project lives from the generous help by contributors that sacrifice their time and doc2dash is no different.

To make participation as pleasant as possible, this project adheres to the [Code of Conduct](#) by the Python Software Foundation.

Here are a few hints and rules to get you started:

- Add yourself to the [AUTHORS.rst](#) file in an alphabetical fashion. Every contribution is valuable and shall be credited.
- If your change is noteworthy, add an entry to the [changelog](#).
- No contribution is too small; please submit as many fixes for typos and grammar bloopers as you can!
- *Always* add tests and docs for your code. This is a hard rule; patches with missing tests or documentation won't be merged – if a feature is not tested or documented, it doesn't exist.
- Obey [PEP 8](#) and [PEP 257](#).
- Write [good commit messages](#).

Note: If you have something great but aren't sure whether it adheres – or even can adhere – to the rules above: **please submit a pull request anyway!**

In the best case, we can mold it into something, in the worst case the pull request gets politely closed. There's absolutely nothing to fear.

Thank you for considering to contribute to doc2dash!

Indices and tables

- *genindex*
- *search*