
doc2dash Documentation

Release 2.2.0

Hynek Schlawack

Jun 12, 2017

Contents

1	User's Guide	3
2	Indices and tables	9

Release v2.2.0 (*What's new?*).

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

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, 3.4+, and PyPy. Both Linux and macOS are tested although certain features are only available on macOS.

Requirements and Installation

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

Warning: Since `doc2dash` is an application, it has hard-coded dependency versions. I strongly discourage from installing it into your global `site-packages` because it will inevitably lead to `conflicts`. Either use `pip` or create a `virtualenv` by hand.

`doc2dash` runs on Python 2.7, and 3.4+, and PyPy. Both Linux and macOS are supported although certain features are only available on macOS.

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

Viewer

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

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

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

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

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!

Sphinx

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

Building Sphinx documentation is usually easy: after installing the `sphinx` package, `cd` into their documentation directory (usually `docs` or `doc`) and enter `make html`.

Warning: Do **not** attempt to run doc2dash over pre-built HTML documentation downloaded from Read The Docs. Those downloads aren't direct equivalents of the actual, pristine builds and indexing will not work.

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 parsed. Fortunately, no theming is common in the pydoctor world, so the parsing is reliable nonetheless.

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.

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 macOS 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 macOS and when Dash.app is installed.
- j, --enable-js**
Enable bundled and external javascript.
- u, --online-redirect-url**
As of Dash 3.0 users can open the online version of pages from within docsets. To enable this, you must set this value to the base URL of your online documentation. e.g. `https://doc2dash.readthedocs.io/`
- parser**
Specify a the import path of a custom parser class (implementing the `doc2dash.parsers.utils.IParser` interface) to use. For example, `--parser doc2dash.parsers.intersphinx.InterSphinxParser` will use the default `InterSphinxParser`. Default behavior is to auto-detect documentation type.
- 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.

Project Information

Changelog

2.2.0 (2017-06-12)

- `InterSphinxParser` is now open to extension. #59
- Support a `--parser` option to force the use of a custom parser class. #60

2.1.0 (2016-03-14)

- Remove legacy Sphinx parsing. The intersphinx parser is much more robust so the old way shouldn't be needed anymore.
 - Add support for InterSphinx constants. [#49](#)
 - Fix handling in-docs links with multiple anchors. [#47](#)
 - Add support for `--enable-js` and `--online-redirect-url` options. [#43](#)
 - Better support for relative paths. [#37](#), [#41](#)
-

2.0.2 (2014-09-24)

- Fix detection of `pydoctor 0.5`. [#31](#), [#39](#)
-

2.0.1 (2014-09-16)

- Better Unicode support. The move from `unicode_literals` to explicit prefixes broke some things that are fixed now. [#29](#), [#30](#)
-

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 that are re-used in synonyms.

License and Hall of Fame

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

Authors

doc2dash is written and maintained by Hynek Schlawack.

The development is kindly supported by [Variomedia AG](#).

A full list of contributors can be found on [GitHub's overview](#).

How To Contribute

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

Here are a few guidelines to get you started:

- Try to limit each pull request to one change only.
- To run the test suite, all you need is a recent [tox](#). It will ensure the test suite runs with all dependencies against all Python versions just as it will on [Travis CI](#). If you lack some Python version, you can always limit the environments like `tox -e py27, py35` (in that case you may want to look into [pyenv](#) that makes it very easy to install many different Python versions in parallel).
- Make sure your changes pass our CI. You won't get any feedback until it's green unless you ask for it.
- If your change is noteworthy, add an entry to the [changelog](#). Use present tense, semantic newlines, and add link to your pull request.
- No contribution is too small; please submit as many fixes for typos and grammar bloopers as you can!
- Don't break backward compatibility.
- *Always* add tests and docs for your code. This is a hard rule; patches with missing tests or documentation won't be merged.
- Write [good test docstrings](#).
- Obey [PEP 8](#) and [PEP 257](#).
- If you address review feedback, make sure to bump the pull request. Maintainers don't receive notifications if you push new commits.

Please note that this project is released with a Contributor [Code of Conduct](#). By participating in this project you agree to abide by its terms. Please report any harm to [Hynek Schlawack](#) in any way you find appropriate.

Thank you for considering to contribute to doc2dash!

CHAPTER 2

Indices and tables

- `genindex`
- `search`

Symbols

-help
 command line option, 5

-parser
 command line option, 5

-version
 command line option, 5

-A, --add-to-global
 command line option, 5

-I FILENAME, --index-page FILENAME
 command line option, 5

-a, --add-to-dash
 command line option, 5

-d PATH, --destination PATH
 command line option, 5

-f, --force
 command line option, 5

-i FILENAME, --icon FILENAME
 command line option, 5

-j, --enable-js
 command line option, 5

-n, --name
 command line option, 4

-q, --quiet
 command line option, 5

-u, --online-redirect-url
 command line option, 5

-v, --verbose
 command line option, 5

-f, --force, 5

-i FILENAME, --icon FILENAME, 5

-j, --enable-js, 5

-n, --name, 4

-q, --quiet, 5

-u, --online-redirect-url, 5

-v, --verbose, 5

C

command line option

- help, 5
- parser, 5
- version, 5
- A, --add-to-global, 5
- I FILENAME, --index-page FILENAME, 5
- a, --add-to-dash, 5
- d PATH, --destination PATH, 5