
Trashman Documentation

Release 1.5.0

Kwpolska

December 31, 2015

1	How to use	3
2	Features	5
3	Additional documents, indices and tables	7
3.1	Trashman	7
3.2	Trashman. A Python trash manager.	8
3.3	Global Functions and Modules List	9
3.4	Backends	10
3.5	Appendix A. Contribution rules	13
3.6	Appendix A. License for Trashman	15
3.7	CHANGELOG for Trashman	15
	Python Module Index	17

Note: This documentation is for Trashman version 1.5.0. Some parts of this documentation might not apply to other versions.

Trashman is a trash manager, i.e. an application which manages trash folders. It provides several backends.

How to use

Install it from PyPI or the AUR.

To trash a file, just type `trash FILE [FILE ...]` into your shell. To restore a file from the trash, add the `-r` (`--restore`) argument. To empty the trash and delete the files forever, use the `-e` (`--empty`) switch. Verbose output can be provided by `-v` (`--verbose`). The location of your trash will be printed by `-w` (`--trash-location`) and the files location will be presented by `-W` (`--files-location`); *w* as in where. Other switches, related to Trashman, are `-V` (`--version`) and `-h` (`--help`).

Features

Currently implemented in most backends:

- trashing
- trash conflict resolution
- restore
- localization support
- tests

Not yet ready:

- restore to inexistent directory (KDE refuses to do that, so we'll probably do so, too)

Useless features that are planned:

- CLI
- Qt GUI

Developers using Trashman as a library may want to look at the [Global functions list](#) or [Backends info](#) (and maybe [TODO](#)). Other people may want to look at the [README](#).

Additional documents, indices and tables

3.1 Trashman

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2015-03-07

Version 1.5.0

Manual section 1

Manual group Trashman manual

3.1.1 SYNOPSIS

trashman [-hVerlvwW] [FILE [FILE...]]

3.1.2 DESCRIPTION

Trashman is a trash manager, i.e. an application which manages trash folders. It provides several backends.

3.1.3 OPERATIONS

[FILE [FILE...]] Trashes the FILE(s).

-e, --empty Empties the trash.

-r, --restore Restores the FILE(s) from the trash.

-l, --list Lists the trash contents using ls.

-w, --trash-location Prints the trash root location.

-W, --files-location Prints the location of trashed files, which may be different from the root.

3.1.4 OPTIONS

-b BACKEND, --backend BACKEND Selects the backend to use. ‘auto’ chooses the backend automatically (fall-back: XDG), ‘config’ tries the config file (which is ‘auto’ by default) and ‘list’ displays a list of the possible backends. Can be configured in `~/.config/kwpolska/trashman/trashman.cfg`.

-h, --help Prints a help message.

-v, --verbose Turns on verbose mode.

-V, --version Prints the version in use.

3.1.5 EXAMPLES

trash foo Trashes the `foo` directory.

trash foo Restores the `foo` directory from trash.

trash -e Empties the trash.

3.1.6 BUGS

Bugs should be reported at the GitHub page (<<https://github.com/Kwpolska/trashman/issues>>). You can also send mail to <chris@chriswarrick.com>.

3.2 Trashman. A Python trash manager.

Info This is the README file for Trashman.

Author Chris Warrick <chris@chriswarrick.com>

Date 2015-03-07

Version 1.5.0

3.2.1 PURPOSE

This is a simple trash manager. It can do various things related to the trash.

3.2.2 INSTALLATION

PyPI or AUR.

3.2.3 COPYRIGHT

Copyright © 2011–2015, Chris Warrick. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the author of this software nor the names of contributors to this software may be used to endorse or promote products derived from this software without specific prior written consent.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

3.3 Global Functions and Modules List

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE or [Appendix B.](#))

Date 2015-03-07

Version 1.5.0

This is an auto-generated documentation of the Trashman suite. It is bare-bones, and you’d be better off reading the source code yourself.

3.3.1 `__init__.py`

trashman

A Python trash manager.

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE).

exception `trashman.__init__.TMErrror (src, info, msg)`

Exceptions raised by the Trashman.

`trashman.__init__.size_dir (sdir)`

Get the size of a directory. Based on code found online.

3.3.2 `tmds.py` (TMDs — Trashman Data Storage)

trashman.TMDs

Trashman Data Storage.

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE).

```
class trashman.tmds.TMDS
    Trashman Data Storage.

    critical(msg)
        Show a critical error and log it.

    error(msg)
        Show an error and log it.

    info(msg)
        Show an information and log it.

    warning(msg)
        Show a warning and log it.
```

3.3.3 main.py (main())

3.4 Backends

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2015-02-28

Version 1.5.0

Trashman uses backends for managing files. All the backends define the same functions and inherit from the `Trash` class.

Backend requirements:

- name ending in `Trash` (eg. `XDGTrash`, `DummyTrash`)
- PEP 8, docstrings
- inherit from `Trash`
- throw exceptions and log activity
- implement the functions listed below (do not re-implement `log` unless needed)
- do actual work (save for the dummy backend)
- have a `self.trashdir` variable with the trash location (`None` if impossible)
- have a `self.logger` instance (`import logging, logging.getLogger('NAME')`)

In order to choose a backend, use `trashman.backends.select(backend)`. If `backend` is `auto`, a choice will be made basing on the config, and if it also says `auto`, on the OS used. If it is `list`, a human-readable list of backends will be printed.

3.4.1 Functions

(this automatically-generated of the `Trash` class.)

```
class trashman.backends.trash.Trash
    Base Trash class.
```

empty (*verbose*)
Empty the trash.

list (*human=True*)
List the trash contents.

log (*msg, lvl='debug'*)
Log a message.

regenerate ()
Regenerate the trash and recreate metadata.

restore (*filename, verbose*)
Restore a file from trash.

trash (*filepath, verbose*)
Move specified file to trash.

3.4.2 Existing backends

DummyTrash

Name DummyTrash

Module dummytrash

Author Chris Warrick <chris@chriswarrick.com>

Specification n/a

OS any

Description A dummy backend, printing all the requests it gets.

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2015-02-28

Version 1.5.0

trashman.backends.dummytrash

A dummy backend, printing all the requests it gets.

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE).

class trashman.backends.dummytrash.DummyTrash
A dummy trash, printing and logging actions that would be performed.

empty (*verbose*)
Empty the trash.

list (*human=True*)
List the trash contents.

log (*msg, lvl='debug'*)
Log a message.

regenerate ()
Regenerate the trash and recreate metadata.

restore (*filename*, *verbose*)
Restore a file from trash.

trash (*filepath*, *verbose*)
Move specified file to trash.

XDGTrash

Name XDGTrash

Module xdgtrash

Author Chris Warrick <chris@chriswarrick.com>

Specification XDG/freedesktop <<http://freedesktop.org/wiki/Specifications/trash-spec>>

OS Linux (KDE, GNOME, Xfce and others conforming to this spec)

Description An XDG trash implementation.

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2015-02-28

Version 1.5.0

trashman.backends.xdgtrash

An XDG trash implementation.

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE).

class trashman.backends.xdgtrash.XDGTrash
XDG trash backend.

empty (*verbose*)
Empty the trash.

list (*human=True*)
List the trash contents.

regenerate ()
Regenerate the trash and recreate metadata.

restore (*filename*, *verbose*, *tocwd=False*)
Restore a file from trash.

trash (*filepath*, *verbose*)
Move specified file to trash.

3.5 Appendix A. Contribution rules

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011–2015, Chris Warrick.

License BSD (see /LICENSE or [Appendix B.](#))

Date 2015-03-07

Version 1.5.0

Do you want to contribute to this project? Great! I'd love to see some help, but you must comply with some rules.

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in RFC 2119.

3.5.1 Issue reporting

GitHub Issues are the recommended way to report an issue. If you do not have an account there, get one or mail me.

When pasting console sessions, you must paste them fully, *prompt-to-prompt*, to see all the messages and your input. Trim only stuff that you are 1000% sure that is not related to the project in question.

3.5.2 General preparations, rules and pull process

Prepare

A GitHub account is recommended. Patches by mail are accepted, but working with you through there is a better way.

Fork the repo first. Non-GitHub people, `git clone`.

Rules

1. Commits must have short, informative and logical messages. Signoffs and long messages are recommended. “fixes #xxx” is required if an issue exists.
2. The following fancy Unicode characters should be used when needed: `-- " ' ``
3. The ellipsis (`. . .`) character must not be used in program output for humans, but may be used elsewhere.
4. I'm a Grammar Nazi.

Request a Pull

Done? Go hit the **Pull Request** button over on GitHub! And if you don't use GitHub, `git format-patch`. Other formats are not accepted.

Your commit should be pulled up in a (longer) while. If I like it. Because some commits may be bad. So, do your best not to do those bad commits.

3.5.3 Details for specific contribution types

Code (Python)

1. PEP 8. `pip install pep8` is recommended.
2. `./tests.py`.
3. Localize all the strings (`_('string')`)
4. Strings must be quoted using `'str'`. Multi-line strings, `"""str"""`. Use the latter only if needed. Otherwise, do:

```
1 string = ('A very, very, very long string '  
2         'that's broken up into multiple lines.')3  
4 string = _('A very, very, very log string '  
5         'that's broken up into multiple lines '  
6         'and that is localized through gettext.'))
```

5. `str.format`. Braces should be empty or contain a name that is later passed on to the function. The format function should be inserted *after* the parenthesis for string localization. For example:

```
1 string = _('{ } is awesome').format('PKGBUILDer')  
2 string = _('{sth} is awesome').format(sth='PKGBUILDer')
```

6. Documentation is important. Please take care of it.

Code (non-Python)

Rejected.

Non-code contributions

Those are accepted. No specific rules exist. And don't remove any files without my permission (`docs/*.8.gz` in particular).

Localization

1. Run `mkdir -p locale/[CODE]/LC_MESSAGES` in your terminal, replacing `[CODE]` by your language code, as in `/usr/share/locale`.
2. Copy the `/messages.pot` file to `locale/[CODE]/LC_MESSAGES/[PROJECT].po` (replace `[PROJECT]` with the project name, all-lowercase)
3. Do your work. The comments will inform you where this string is, and the ones starting with 'TRANSLATORS:' are for you to read and make use of. Other comments come from my code and you should not care about them. And if it is directed for translators, let me know. The [Poedit](#) app may be of help. Please take care of the headers at the top of the file (with a text editor, do not use Poedit for that!) and modify them. The Last-Translator, Language-Team and Language are important, the others are auto-generated anyways. The general [Rules](#) apply, please take care of it!
4. Commit (`-s/--signoff` is required here).
5. Your translation will be added in the next release, or, if a release isn't planned in the near future, a new release will be made. Your addition will be appreciated. Note that I cannot translate new strings, and, as a result, I might ask you for additions in the future.

3.6 Appendix A. License for Trashman

Info This is the license for Trashman.

Author Chris Warrick <chris@chriswarrick.com>

Date 2015-03-07

Version 1.5.0

Copyright © 2011–2015, Chris Warrick. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the author of this software nor the names of contributors to this software may be used to endorse or promote products derived from this software without specific prior written consent.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

3.7 CHANGELOG for Trashman

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011–2015, Chris Warrick.

License BSD (see [/LICENSE](#) or [Appendix B.](#))

Date 2015-03-07

Version 1.5.0

3.7.1 GitHub holds releases, too

More information can be found on GitHub in the [releases](#) section.

3.7.2 Version History

1.5.0

1.5.0 A big modernization of Trashman.

1.0.0

1.0.4 Minor fixes.

1.0.3 I do not like the word “exit”.

1.0.2 Fixes #2, creating config dirs properly. (the app crashed if there was no ~/.config/kwpolska/ dir. This bug wasn't found due to having PKGBUILDer installed, which was creating that directory.)

1.0.1 Bugfix release.

1.0.0 A complete modification of the approach, now using a concept of “backends”, allowing multiple trash systems to be used.

0.2.0

0.2.4.1 (semi-official) Locale fix #3, only pypi/aur

0.2.4 Zsh completions.

0.2.3 Locale fix #2

0.2.2 Locale fix #1

0.2.1 Localization support.

0.2.0 Documentation. The project became a bit more mature now.

0.1.0

0.1.3 Fixed some problems with file restoration.

0.1.2 Ommited a character, also added restoration to –help.

0.1.1 Restoration magic.

0.1.0 Initial release.

- genindex
- modindex
- search

t

`trashman.__init__`, [9](#)
`trashman.backends.dummytrash`, [11](#)
`trashman.backends.xdgtrash`, [12](#)
`trashman.tmds`, [9](#)

B

backends, 10

C

CHANGELOG, 15

contributing, 13

critical() (trashman.tmds.TMDS method), 10

D

DummyTrash, 11

DummyTrash (class in trashman.backends.dummytrash),
11

E

empty() (trashman.backends.dummytrash.DummyTrash
method), 11

empty() (trashman.backends.trash.Trash method), 10

empty() (trashman.backends.xdgtrash.XDGTrash
method), 12

error() (trashman.tmds.TMDS method), 10

G

global functions, 9

I

info() (trashman.tmds.TMDS method), 10

issues, 13

L

LICENSE, 15

list() (trashman.backends.dummytrash.DummyTrash
method), 11

list() (trashman.backends.trash.Trash method), 11

list() (trashman.backends.xdgtrash.XDGTrash method),
12

locale, 14

log() (trashman.backends.dummytrash.DummyTrash
method), 11

log() (trashman.backends.trash.Trash method), 11

R

regenerate() (trashman.backends.dummytrash.DummyTrash
method), 11

regenerate() (trashman.backends.trash.Trash method), 11

regenerate() (trashman.backends.xdgtrash.XDGTrash
method), 12

restore() (trashman.backends.dummytrash.DummyTrash
method), 12

restore() (trashman.backends.trash.Trash method), 11

restore() (trashman.backends.xdgtrash.XDGTrash
method), 12

S

size_dir() (in module trashman.__init__), 9

T

TMDS (class in trashman.tmds), 10

TLError, 9

Trash (class in trashman.backends.trash), 10

trash() (trashman.backends.dummytrash.DummyTrash
method), 12

trash() (trashman.backends.trash.Trash method), 11

trash() (trashman.backends.xdgtrash.XDGTrash method),
12

trashman.__init__ (module), 9

trashman.backends.dummytrash (module), 11

trashman.backends.xdgtrash (module), 12

trashman.tmds (module), 9

W

warning() (trashman.tmds.TMDS method), 10

X

XDGTrash, 12

XDGTrash (class in trashman.backends.xdgtrash), 12