
PKGBUILDer Documentation

Release 4.2.10

Chris Warrick

2017-02-04T08:02:32+0000

1	User Documentation	3
1.1	PKGBUILDer	3
1.2	PBWrapper	5
1.3	PKGBUILDer. An AUR helper (and library) in Python 3.	6
2	Developer Documentation	9
2.1	aur module (AUR class)	9
2.2	build module	10
2.3	__main__ module	11
2.4	package module (Package, AURPackage, ABSPackage classes)	11
2.5	pbds module (PBDS class)	12
2.6	Transactions	15
2.7	ui module (UI class)	16
2.8	upgrade module	18
2.9	utils module	18
2.10	PBWrapper	19
2.11	Exceptions in PKGBUILDer	20
2.12	PKGBUILDer Sample Scripts	21
3	Appendices	23
3.1	Appendix A. Contribution rules	23
3.2	Appendix B. License for PKGBUILDer	24
3.3	Appendix C. Changelog	24
4	Indices and tables	31
	Bibliography	33
	Python Module Index	35

Note: This documentation is compatible with version 4.2.10 of PKGBUILDer. The current version on your system can be checked by running `pkgbuilder -v`.

Note: 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.

PKGBUILDer is an AUR helper, i.e. an application which builds AUR packages. It can be used in conjunction with `pacman` (with a special script). It uses various techniques to automatize the process as much as possible.

Since version 2.1.0, PKGBUILDer provides modules that can be used in other scripts.

For standalone use, see the [PKGBUILDer man page](#). Reading the [README](#) is also a good idea.

For the PBWrapper (`pb`), see the [PBWrapper man page](#).

For developers using PKGBUILDer as a Python module, see the respective class documents.

User Documentation

By reading those three documents, you will know your way around PKGBUILDer as an user.

1.1 PKGBUILDer

Subtitle An AUR helper (and library) in Python 3.

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2017-01-08

Version 4.2.10

Manual section 8

Manual group PKGBUILDer manual

1.1.1 SYNOPSIS

```
pkgbuilder [-hVcCdDvvy] [--debug|--nodebug] [--pgpcheck|--skippgpcheck] [--confirm|--noconfirm] [--deep|--shallow]
[--userfetch USER] [--SFisuUX] [PACKAGE [PACKAGE ...]]
```

1.1.2 DESCRIPTION

PKGBUILDer is an AUR helper, i.e. an application which builds AUR packages. It can be used in conjunction with pacman (with a special script). It uses various techniques to automatize the process as much as possible.

Since version 2.1.0.0, PKGBUILDer provides modules that can be used in other scripts.

Since version 2.1.5.6, PKGBUILDer also provides support for ABS packages. Passing an ABS package name to the `-S` option will result in a seamless detection and build process.

Notice: Running PKGBUILDer and/or PBWrapper as root can deal catastrophic damage to your system. Run it as a regular user, you will be prompted for the root password when one will be required (i.e. to run **pacman**).

1.1.3 CONFIGURATION

PKGBUILDer supports per-user configuration, in the file `~/config/kwpolska/pkgbuilder/pkgbuilder.ini`. It can also be configured on a per-usage basis via command-line arguments.

1.1.4 OPERATIONS

- S, --sync** Build packages in `/tmp` instead of CWD. Override with `--notmp`.
- F, --fetch** Fetch (and don't build) **PACKAGE**s in a fashion similar to `cower -d`. Override with `--nofetch`.
- userfetch USER** Fetch all AUR packages of an user.
- y, --refresh** A dummy option for pacman syntax compatibility.
- i, --info** Display info about **PACKAGE** in a fashion similar to pacman.
- s, --search** Search the AUR for packages with **PACKAGE** as the query.
- u, --sysupgrade** Check for package updates in the AUR. If updates are found, they will be installed or fetched if the user accepts. Pass twice to downgrade.
- U, --upgrade** Move pacman packages to the cache and install them.
- X, --runtx** Run transactions from `.tx` files. (created as part of the install process, usable to re-run an installation if it fails)

Additionally, parameters **-S, --sync, -y** and **--refresh** are accepted for pacman syntax compatibility. **-S/--sync** makes the script build its packages in `/tmp` instead of the current working directory (CWD).

1.1.5 OPTIONS

Most option have a negated version, to temporarily override a config setting. Only the non-default options are documented below.

- h, --help** Show the help message.
- V, --version** Show the version number.
- c, --clean** Clean the build directory after a finished build. (*makepkg -c*)
- C, --nocolors** Force the script to ignore the ANSI color codes.
- debug** Output debug information to stderr.
- d, --nodepcheck** Skip dependency checks. It may (and, most likely, will) break *makepkg*.
- D, --vcsupgrade** Upgrade all the VCS packages on the system. Requires **-u**.
- v, --novalidation** Skip package installation validation phase (checking if the package is installed).
- w, --buildonly** Skip package installation after building.
- skipgpcheck** Skip PGP checks.
- noconfirm** Do not ask for confirmation when installing packages.
- deep** Perform deep clones of git repositories. Override with `--shallow`.
- ignore [PACKAGE PACKAGE ...]** Ignore a package upgrade (can be used more than once, or use commas – follows pacman syntax)
- y, --refresh** Dummy option for pacman compatibility.

1.1.6 EXAMPLES

pkgbuilder hello Install the package hello from the AUR. It will be built in the CWD.

pkgbuilder -S hello Install hello, but builds the package in /tmp/pkgbuilder-UID.

pkgbuilder -F hello Fetch the AUR git repository for hello to the CWD.

pkgbuilder -SF hello Like above, but does it in /tmp/pkgbuilder-UID.

pkgbuilder python Python is a binary repo package, triggering a ABS download via rsync (no need for `extra/abs`). `-S` and/or `-F` are also accepted.

pkgbuilder -Syu Check for updates and offer installing them.

pkgbuilder -uF Check for updates and offer fetching them.

1.1.7 SEE ALSO

pb(8), a wrapper for pacman and PKGBUILDer, included with PKGBUILDer, also known as PBWrapper.

pacman(8), **makepkg(8)**, **PKGBUILD(5)**

You can visit the git repo at <https://github.com/Kwpolska/pkgbuilder> or the documentation at <https://pkgbuilder.readthedocs.org> for more info.

1.1.8 BUGS

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

1.2 PBWrapper

Subtitle A wrapper for pacman and PKGBUILDer.

Author Chris Warrick chris@chriswarrick.com

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2017-01-08

Version 4.2.10

Wrapper Version 0.5.3

Manual section 8

Manual group PKGBUILDer manual

1.2.1 SYNOPSIS

pb <operation> [options] [targets]

1.2.2 DESCRIPTION

PBWrapper (also known as just *pb*) is a wrapper for pacman and PKGBUILDer, i.e. an application which handles checking if a package is in the official repos or in the AUR and installs them according to that information.

It is a part of PKGBUILDer and is included since version 2.1.4.0 of PKGBUILDer.

Notice: Running PKGBUILDer and/or PBWrapper as root can deal catastrophic damage to your system. Run it as a regular user, you will be prompted for the root password when one will be required (i.e. to run **pacman**).

1.2.3 OPERATIONS AND OPTIONS

All operations and options from pacman and PKGBUILDer work. Operations other than **-S**, **-sync** are passed as-is to pacman. The **-S**, **-sync** switch activates a check for other parameters. If one of **-syu** (**-search -refresh -sysupgrade**) is present, the requests are passed to pacman and PKGBUILDer, in that order, with arguments that apply to the manager and nothing else. If any other operations are present (including **-i**, **-info**), the **targets** provided are checked for presence in the AUR. If a package is in the AUR, it is added to the queue that will be passed to PKGBUILDer, otherwise it is added to a pacman queue. Please note that brand new options will not work until the next PKGBUILDer/PBWrapper release.

The **-debug** option enables additional debug information from PBWrapper.

An additional option is **-unlock**, which unlocks the pacman database.

1.2.4 EXAMPLES

pb -S python hello Installs the Python interpreter from the repos (which is already on your system, especially if you have installed PKGBUILDer/PBWrapper) and GNU Hello from the AUR. Note that, if you used `pkgbuilder` instead of `pb`, the Python package would be fetched from the ABS (instead of the repositories) and built locally.

pb -Si python hello Shows information for the python and hello packages.

pb -Syu Checks for repo and AUR updates (in that order) and offers installing them.

1.2.5 SEE ALSO

pacman(8), **pkgbuilder(8)**, **makepkg(8)**, **PKGBUILD(5)**

You can visit the `git` repo at <https://github.com/Kwpolska/pkgbuilder> or the documentation at <https://pkgbuilder.readthedocs.org> for more info.

1.2.6 BUGS

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

1.3 PKGBUILDer. An AUR helper (and library) in Python 3.

Info This is the README file for PKGBUILDer.

Author Chris Warrick chris@chriswarrick.com

Date 2017-01-08

Version 4.2.10

1.3.1 PURPOSE

Building and installing AUR packages.

1.3.2 INSTALLATION

There are two ways to install PKGBUILDer:

1. Get it from the AUR: <https://aur.archlinux.org/packages/pkgbuilder/>
2. Add the PKGBUILDer unofficial repository: https://wiki.archlinux.org/index.php/Unofficial_user_repositories#pkgbuilder

After adding the repository, you need to run:

```
# pacman-key -r 5EAAEA16
# pacman-key --lsign 5EAAEA16
# pacman -Syu
```

1.3.3 BASIC USAGE

`pkgbuilder` is a command-line application. It takes various options and package names as arguments. The following options are needed for basic usage:

- `-S` to work in `/tmp` instead of the current directory
- `-F` to fetch packages instead of installing them
- `-s` to search for packages in the AUR
- `-i` to get info about an AUR package
- `-u` to upgrade all AUR packages on your system

PKGBUILDer also comes with `pb`, a wrapper that works with both `pacman` and the AUR.

For more information, refer to the `-h` command, the `pkgbuilder(8)` man page, or the online documentation at <https://pkgbuilder.readthedocs.org/>.

1.3.4 COPYRIGHT

Copyright © 2011-2017, 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.

Developer Documentation

Those documents will let a developer do something. Note that most of them is auto-generated from the Python source and you are actually better off reading that.

2.1 aur module (AUR class)

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

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

Date 2017-01-08

Version 4.2.10

New in version 2.1.0.0. Call the AUR API.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

class `pkgbuilder.aur.AUR`

Call the AUR API.

Valid request types for `request()` (and `jsonrequest()`):

name	purpose
<code>info</code>	get info about <i>arg</i>
<code>search</code>	search for <i>arg</i> in the AUR
<code>msearch</code>	show packages maintained by <i>arg</i>

A more accurate list might be available at the AUR RPC website: <https://aur.archlinux.org/rpc.php>

`multiinfo` is implemented in another function, `multiinfo()`.

Note: Most people don't actually want this and will prefer to use `pkgbuilder.utils.{info,search,msearch}()` instead.

jsonmultiinfo (*args*)

Make a `multiinfo` request and return plain JSON data.

jsonreq (*rtype, arg, search_by=None*)

Make a request and return plain JSON data.

multiinfo (*args*)

Make a multiinfo request and return the AURDict.

request (*rtype, arg, search_by=None*)

Make a request and return the AURDict.

rpc

Return the RPC URL.

search (*search_by, arg*)

Search the AUR and return the AURDict.

2.2 build module

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2017-01-08

Version 4.2.10

New in version 2.1.0.0.

Changed in version 3.0.0: Class removed, its methods became module-level functions Build AUR packages.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

`pkgbuilder.build.auto_build` (*pkgname, performdepcheck=True, pkginstall=True, completelist=None*)

A function that builds everything, that should be used by everyone.

This function makes building AUR deps possible. If you can, use it.

Note: This function returns a list of packages to install with pacman -U. Please take care of it. Running PKGBUILDer/PBWrapper standalone or `__main__.main()` will do that.

`pkgbuilder.build.clone` (*pkgbase*)

Clone or update a git repo.

New in version 4.0.0.

`pkgbuilder.build.rsync` (*pkg, quiet=False*)

Run rsync for a package.

`pkgbuilder.build.prepare_deps` (*srcinfo_path, pkgname=None*)

Get (make)depends from a .SRCINFO file and returns them.

(pkgname is now discarded, because it messes up one-build split packages.)

Changed in version 4.0.1.

In the past, this function used to get data via `bash -c`.

`pkgbuilder.build.depcheck` (*depends, pkgobj=None*)
Perform a dependency check.

`pkgbuilder.build.fetch_runner` (*pkgnames, preprocessed=False*)
Run the fetch procedure.

`pkgbuilder.build.build_runner` (*pkgname, performdepcheck=True, pkginstall=True*)
A build function, which actually links to others.
DO NOT use it unless you re-implement `auto_build!`

2.3 `__main__` module

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2017-01-08

Version 4.2.10

Changed in version 4.0.0.

New in version 2.1.3.0. Main routine of PKGBUILDer.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

`pkgbuilder.__main__.main` (*source='AUTO', quit=True*)
Main routine of PKGBUILDer.

2.4 package module (Package, AURPackage, ABSPackage classes)

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2017-01-08

Version 4.2.10

New in version 3.0.0.

Added in 3.0.0, the Package family of classes is the core of PKGBUILDer. They are responsible for storing package data in a consistent way. All class entries are Pythonic. The Package class, the most important class in PKGBUILDer.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

class `pkgbuilder.package.Package` (***kwargs*)
The base class for packages.

This is base class. It shouldn't be used directly. It contains the following attributes:

Name	Type	De- fault	Description
is_abs	bool	None	A boolean informing of the package type, to prevent <code>isinstance(pkg, ABSPackage)</code> checks.
name	str	None	The name of the package.
version	str	None	Current package version.
description	str	None	Package description.
repo	str	None	The repo (ABS) or category (AUR) of the package in question.
url	str	None	The upstream URL specified in the PKGBUILD.
licenses	list	[]	Licenses specified in the PKGBUILD.
human	str	None	The packager (ABS) or maintainer (AUR) of the package in question.

class `pkgbuilder.package.AURPackage` (**kwargs)
 An AUR package.

classmethod `from_aurdict` (*aurdict*)
 Create an instance of AURPackage from AUR RPC data.

This class contains all the attributes of `Package`, `is_abs = False` and the following attributes:

Name	Type	De- fault	Description
id	int	None	ID of the AUR package.
is_outdated	bool	None	Package OoD flag status in the AUR.
outdated_since	date	None	A date (<code>datetime.datetime()</code> , aware UTC) of OoD flagging date OR None.
added	date	None	A date (<code>datetime.datetime()</code> , aware UTC) representing package addition time.
modified	date	None	A date (<code>datetime.datetime()</code> , aware UTC) representing the last modification time.
votes	int	None	Count of AUR votes.
urlpath	str	None	The URL of the tarball, sans <code>https://aur.archlinux.org</code> . Not used in AURv4.
_category_id	int	None	AUR Category ID, not supposed to be used by most people (hence the underscore).

class `pkgbuilder.package.AURPackage` (**kwargs)
 An AUR package.

classmethod `from_aurdict` (*aurdict*)
 Create an instance of AURPackage from AUR RPC data.

This class has a total of 29 attributes (21 if you exclude the `Package` ones) and `is_abs = True`. For more information, consult either the attribute names or documentation for/code of `libalpm` and `pyalpm`.

2.5 pbds module (PBDS class)

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

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

Date 2017-01-08

Version 4.2.10

2.5.1 PBDS

New in version 2.1.0.0.

class **PBDS**

This is the class used for storing data. Currently, it stores the following information (not including information humans should not touch and care about):

variable	contents/usage	default/acceptable values
colors	colors currently used in the script	<i>[colors]</i>
pacman	using wrapper-friendly behavior? <i>[pacman]</i>	False
validate	validating package installation?	True
depcheck	checking if deps are installed?	True
pkginst	if makepkg should install packages	True
inttext	text shown while interrupting (^C)	<i>[inttext]</i>
wrapperinttext	text shown while wrapper is interrupting (^C)	<i>[wrapperinttext]</i>
paccommand	Pacman command to use	pacman
hassudo	If <code>sudo</code> is present (see <code>sudo()</code>)	(bool)
uid	Current UID	<code>os.geteuid()</code>
confhme	configuration home	<code>~/.config/</code>
kwdir	directory used by all projects by yours truly	<code>~/.config/kwpolska</code>
confdir	configuration directory	<code>~/.config/kwpolska/pkgbuilder</code>
log	logger object (e.g. <code>PBDS.log.info</code>)	logger object
ui	an instance of <code>pkgbuilder.ui.UI</code>	None or <code>pkgbuilder.ui.UI</code>
pyc	a <code>pycman</code> instance	None or <code>pycman</code> instance

```

colors = {
    'all_off':    '\x1b[1;0m',
    'bold':      '\x1b[1;1m',
    'blue':      '\x1b[1;1m\x1b[1;34m',
    'green':     '\x1b[1;1m\x1b[1;32m',
    'red':       '\x1b[1;1m\x1b[1;31m',
    'yellow':    '\x1b[1;1m\x1b[1;33m'
}

```

PKGBUILDer Data Storage.

Copyright © 2011-2017, Chris Warrick.

License BSD (see `/LICENSE`).

class `pkgbuilder.pbds.PBDS`

PKGBUILDer Data Storage.

colorsoff ()

Disable colors.

colorson ()

Enable colors.

debugmode (*nochange=False*)

Print all the logged messages to `stderr`.

fancy_error (*text*)

Display error messages.

fancy_error2 (*text*)

Display error sub-messages.

fancy_msg (*text*)

Display main messages.

fancy_msg2 (*text*)

Display sub-messages.

fancy_warning (*text*)

Display warning messages.

fancy_warning2 (*text*)

Display warning sub-messages.

get_setting (*name, config_section, config_option, positive, negative*)

Get the value of a setting, based on config file and arguments.

Parameters

- **name** (*str*) – name of setting, used for error on conflict
- **config_section** (*bool*) – configuration file section
- **config_option** (*bool*) – configuration file option
- **positive** (*bool*) – positive argument
- **negative** (*bool*) – negative argument

Returns the value requested by the user, defaults to config

Return type bool

pyc

Return a pycman handle, initializing one if necessary.

pycreload ()

Reload pycman.

root_crash ()

Crash if running as root.

run_command (*args, prepend=None, asonearg=False*)

Run a command.

Note: Accepts only one command. *shell=False*, for safety. *asonearg* is for `su -c` and most people don't need nor want it.

Note: since version 2.1.6.2, *args* must be a list.

sudo (*args*)

Run as root.

Uses `sudo` if present, `su -c` otherwise, nothing if already running as root.

Note: Accepts only one command. *shell=False*, for safety.

2.6 Transactions

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

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

Date 2017-01-08

Version 4.2.10

New in version 4.1.0.

To install built packages, PKGBUILDer uses transactions. A transaction stores:

- package names (as requested by user), used for the validation step
- package file names, moved to pacman cache and installed through `pacman -U`
- signature file names, moved to pacman cache
- options for running the transaction
- a reference to the transaction file name (if any; `.tx` files are autosaved when status changes occur)
-

Package installation transactions.

New in version 4.1.0.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

`pkgbuilder.transaction.generate_filename` (*absolute=True*)

Generate a filename for the transaction.

class `pkgbuilder.transaction.Transaction` (*pkgnames, pkgpaths, sigpaths, asdeps=True, uopt='', filename=None, delete=False, status=None, pacmanreturn=-1, invalid=-1*)

A package transaction.

exitcode

Provide the most appropriate exit code.

classmethod `fromjson` (*jsondata*)

Create a transaction from JSON data.

The following fields exist:

- `__PBTX__` — PKGBUILDer version
- `pkgnames` — names of packages to install
- `pkgpaths` — paths to packages to install
- `sigpaths` — paths to attached signature files
- `asdeps` — whether or not this is a dependency install
- `uopt` — special (local) options to `pacman -U`
- `status` — status code of the transaction
- `pacmanreturn` — pacman return code

- `invalid` — number of invalid packages

install (*sudo_tested=False, quiet=False*)

Install packages through `pacman -U`.

Parameters

- **sudo_tested** (*bool*) – if sudo was tested (password prompt)
- **quiet** (*bool*) – suppress messages

Returns `pacman` return code

Return type `int`

classmethod load (*filename*)

Load a transaction file.

move (*sudo_tested=False, quiet=False*)

Move package and signature files to `pacman` cache.

Parameters

- **sudo_tested** (*bool*) – if sudo was tested (password prompt)
- **quiet** (*bool*) – suppress messages

Returns 0 on success, +mv return, -failed files

Return type `int`

pacman_pkgpaths

Return package paths, augmented for `pacman`.

run (*standalone=True, quiet=False, validate=True*)

Run a transaction.

save (*filename=None*)

Save a transaction file.

tojson ()

Serialize a transaction to JSON.

validate (*quiet*)

Check if packages were installed.

Parameters **quiet** (*bool*) – suppress messages

Returns number of packages that were not installed

Return type `int`

class `pkgbuilder.transaction.TransactionStatus`

Transaction status.

2.7 ui module (UI class)

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

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

Date 2017-01-08

Version 4.2.10

New in version 3.0.0. The User Interface.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

`pkgbuilder.ui.get_termwidth` (*default=None*)
Get the width of this terminal.

New in version 3.3.0.

Changed in version 4.2.9.

`pkgbuilder.ui.hanging_indent` (*text, intro, termwidth=None, change_spaces=True, in-
trowidth=None*)
Produce text with a hanging indent.

New in version 3.3.0.

Changed in version 4.0.0.

`pkgbuilder.ui.mlist` (*items, sep=' ', change_spaces=True, termwidth=None, indentwidth=17*)
Output a list of strings, complete with a hanging indent.

New in version 3.3.0.

Changed in version 4.0.0.

class `pkgbuilder.ui.Progress` (*total=1*)
A static progress indicator with numbers.

Usage:

```
pm = Progress(total=2)
pm.msg('Doing step 1...')
step1()
pm.msg('Doing step 2...')
step2()
```

msg (*msg, single=False*)
Print a progress message.

class `pkgbuilder.ui.Throbber` (*msg, finalthrob='*', printback=True*)
A nice animated throbber.

Usage:

```
with Throbber('Doing important stuff...'):
    dostuff()
```

throbber_alive
Check the status of a throbber.

class `pkgbuilder.ui.ProgressThrobber` (*msg, total=1*)
An animated progress throbber.

Similar to Progress, but the / is animated.

Usage:

```
with ProgressThrobber('Working...', total=2) as pt:
    dostuff()
    pt.bump('Cleaning up...')
    cleanup()
```

bump (*msg*)
Change the displayed message.

2.8 upgrade module

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

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

Date 2017-01-08

Version 4.2.10

New in version 2.1.0.0.

Changed in version 3.0.0: Class removed, its methods became module-level functions Tools for performing upgrades of AUR packages.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

`pkgbuilder.upgrade.gather_foreign_pkgs` ()
Gather a list of all foreign packages.

`pkgbuilder.upgrade.list_upgradable` (*pkglist*, *vcsup=False*, *aurcache=None*, *ignorelist=None*)
Compare package versions and returns upgradable ones.

Changed in version 4.2.9.

`pkgbuilder.upgrade.auto_upgrade` (*downgrade=False*, *vcsup=False*, *fetchonly=False*, *ignorelist=None*)
Human friendly upgrade question and output.
Returns packages — should be passed over to builder functions.

2.9 utils module

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

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

Date 2017-01-08

Version 4.2.10

New in version 2.1.0.0.

Changed in version 3.0.0: Class removed, its methods became module-level functions Common global utilities, used mainly for AUR data access.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

`pkgbuilder.utils.info` (*pkgnames*)

Return info about AUR packages.

Changed in version 3.0.0.

`pkgbuilder.utils.search` (*pkgname, search_by='name-desc'*)

Search for AUR packages.

Changed in version 3.0.0.

`pkgbuilder.utils.msearch` (*maintainer*)

Search for AUR packages maintained by a specified user.

New in version 3.0.0.

`pkgbuilder.utils.print_package_search` (*pkg, cachemode=False, prefix='', prefixp=''*)

Output/return a package representation.

Based on *pacman -Ss*.

Changed in version 4.0.0.

`pkgbuilder.utils.print_package_info` (*pkgs, cachemode=False*)

Output/return a package representation.

Based on *pacman -Ss*.

Changed in version 3.3.0.

2.10 PBWrapper

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

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

Date 2017-01-08

Version 4.2.10

PBWrapper Version 0.2.3

New in version 2.1.4.0. A wrapper for *pacman* and *PKGBUILDer*, also known as *PBWrapper* or *pb*.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

`pkgbuilder.wrapper.main` ()

Run the *PBWrapper* main function.

`pkgbuilder.wrapper.wrapper` (*source='AUTO'*)

A wrapper for *pacman* and *PKGBUILDer*.

This is the wrapper for *pacman* and *PKGBUILDer*, `bin/pb`. It is a complete mess, but at least it works. All the arguments it gets are passed to *pacman*, unless the operation is `-S`. Then, an additional check is made.

For `-S` and `-Si` requests, the packages are checked, one by one, if they are in the AUR. If one is, `pkgbuilder -Si` is requested. If not, all error handling is left to *pacman*.

For `-Ss` and `-Syu` requests, they are passed to both managers, complete with additional arguments applicable to one or another.

Note: Actually, pkgbuilder isn't called as a subprocess, the arguments are just passed to the main() function, so no time is wasted on reloading everything.

2.11 Exceptions in PKGBUILDer

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2017-01-08

Version 4.2.10

New in version 3.0.0.

Starting with version 3.0.0, the exceptions were split up into multiple ones, all inheriting from `PBException`. All exceptions have two guaranteed attributes: `msg`, which is a more-or-less human-friendly message, and `src`, which explains where the problem came from (string or instance of an object, depending on the exception — it is clearly visible in the source code). Defines PKGBUILDer exceptions.

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE).

exception `pkgbuilder.exceptions.PBException` (*msg, source, exit=True, *args, **kwargs*)
Base exception for PKGBUILDer.

exception `pkgbuilder.exceptions.AURERROR` (*msg, exit=True, *args, **kwargs*)
AUR-related errors.

exception `pkgbuilder.exceptions.MakepkgError` (*retcode, exit=True, *args, **kwargs*)
makepkg return codes.

exception `pkgbuilder.exceptions.NetworkError` (*msg, source, exit=True, *args, **kwargs*)
Network-related errors.

exception `pkgbuilder.exceptions.ConnectionError` (*msg, source, exit=True, *args, **kwargs*)
A connection error.

exception `pkgbuilder.exceptions.HTTPError` (*source, origin, exit=True, *args, **kwargs*)
An HTTP error.

exception `pkgbuilder.exceptions.PackageError` (*msg, source, exit=True, *args, **kwargs*)
Package-related errors.

exception `pkgbuilder.exceptions.PackageNotFoundError` (*name, source, exit=True, *args, **kwargs*)
Errors raised when a package cannot be found.

exception `pkgbuilder.exceptions.SanityError` (*msg, source, exit=True, *args, **kwargs*)
When PKGBUILDer or one of its friends goes insane.

exception `pkgbuilder.exceptions.CloneError` (*returncode, exit=True, *args, **kwargs*)
A git clone failed.

2.12 PKGBUILDer Sample Scripts

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

License BSD (see /LICENSE or Appendix B.)

Date 2017-01-08

Version 4.2.10

PKGBUILDer, since version 2.1.5.0, includes some sample scripts that are utilizing it as a library. Currently included:

- `aur-ood-orphans` — lists the locally installed packages that are marked as “out-of-date” or are orphans in the AUR. (Mind using the results to get something done about those packages? Thanks in advance, from everyone in this univ^H^H^H^Hcommunity.)
- `list-non-aur` — lists packages that don’t exist in any sync repo nor the AUR.
- `upgrade` — lists or shows the count of possible AUR upgrades. For use eg. in conky. `-h/--help` for details.

3.1 Appendix A. Contribution rules

Info Those are the contribution rules for PKGBUILDer.

Copyright © 2012-2017, Chris Warrick.

License 3-clause BSD

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.

3.1.1 Issue reporting

GitHub Issues are the recommended way to report an issue.

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.

3.1.2 General preparations, rules and pull process

Prepare

A GitHub account is recommended. Patches by e-mail are accepted, but I'd prefer to work via GitHub.

Rules

1. Commits must have short, informative and logical messages. Signoffs and long messages are recommended. “Fix #xxx” is required if an issue exists.
2. The following fancy Unicode characters should be used when needed: -- ` " \ ' should not appear in console output, but may appear elsewhere.
3. For Python code, use the PEP 8 coding style and PEP 257 documentation style. For other languages, K&R style applies. Braces are mandatory in all blocks (even one-line blocks). Braces are on the same lines as class names and function signatures. Use 4-space indents.

Request a Pull

Done? Go hit the **Pull Request** button over on GitHub! Your request should be accepted shortly (assuming there are no major errors).

Localization

Translations are managed on [Transifex](#).

3.2 Appendix B. License for PKGBUILDer

Info This is the license for PKGBUILDer.

Author Chris Warrick <chris@chriswarrick.com>

Date 2017-01-08

Version 4.2.10

Copyright © 2011-2017, 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 Appendix C. Changelog

Info This is the changelog for PKGBUILDer.

Author Chris Warrick <chris@chriswarrick.com>

Copyright © 2011-2017, Chris Warrick.

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

Date 2017-01-08

Version 4.2.10

3.3.1 Versioning scheme

PKGBUILDer uses the following versioning schemes:

3.0.0+ major.minor.revision

2.99.x.x 2.99.stage.revision (3.0.0 test versions)

2.1.1.0–2.1.6.3 generation.major.minor.revision

2.1.0 generation.major.minor

1.0–2.0 generation.revision

Where:

- **generation** 1 is the first Perl version, 2 is the Python version (dropped for 3.0.0).
- **major** basic release number.
- **minor** sub-release number.
- **revision** changes that aren't important enough to be new minor versions.

3.3.2 GitHub holds releases, too

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

3.3.3 Version History

4.2.10

- Fix #50 – prevent infinite loop when building with `-w, --buildonly` (crashes early and explains why)

4.2.9

- Add an option to force `chdir` to a place before building (Issue #45)
- Support `--ignore` for upgrades (Issue #46)
- Use `shutil.get_terminal_size()` instead of `stty size` (Issue #47)
- Fix ABS package builds (Issue #48)
- Use new PyPI URL scheme in PKGBUILDer's PKGBUILD

4.2.8

- Show help message for `pb -Sh` (Issue #44)

4.2.7

- Add support for AUR keywords (Issue #43)

4.2.6

- Fix dependency loop for split packages

4.2.5 *AUR RPC v5 compatibility * Pacman 5.0 compatibility

4.2.4

- Fix “out of date” flag check

4.2.3

- Translate “Running transaction...” line correctly
- Support `--noconfirm` in `-Syu`
- Pull existing git repos instead of cloning new ones or aborting (Issue #36)
- Display “skipping package” message when exceptions occur
- Support architecture-specific dependencies (Issue #37)

4.2.2

- Display a better message if some packages fail to build.

4.2.1

- Default to empty strings in Packages (Issue #35; thanks @mfussenegger)
- Fix duplicate `-confirm` argument in `pb`

4.2.0

- Support for config files!
- Change domain to `aur.archlinux.org`

4.1.0

- Add transactions framework — users can retry failed installs; less code duplication
- Add `-noconfirm` option
- Fix issue with git clones by adding explicit clone destination

4.0.3

- Add `--deep` option to create deep clones (especially useful for `-F` and `--userfetch`)

4.0.2

- Update documentation (man pages, README)
- Support `upgrade-fetch (-uF)` properly
- Drop fancy-style upgrade messages (use pacman-style messages only, no matter what mode of operation was used)
- Add `git` to PKGBUILD dependencies

4.0.1

- Fix `prepare_deps` for split PKGBUILDS (take all packages’ data, something `makepkg` actually does **not** do)

4.0.0

- Use `RPCv4`
- Use git clones instead of source tarballs (#34)
- Display Popularity field
- Delete Category field
- Use `.SRCINFO` files instead of running PKGBUILDS (via `python-srcinfo`)
- Don’t build split packages twice (happened in eg. `pkgbuilder -Syu`)
- Run `sudo true` before attempting to move files so that `mv` failures don’t cause an infinite loop
- Add `--skipgpcheck` option (mirrors `makepkg`’s)

- API change: use one `__main__.py` file instead of two different files
- API change: move some UI functions from `pkgbuilder.utils` to `pkgbuilder.ui`
- API change: cd out of the directory in `build_runner`

3.5.1

- Fix AURv4 support (use `URLPath`)

3.5.0 AURv4 compatibility and minor bugfixes.

3.4.0 Pacman v4.2.0 compatibility.

3.3.2 Various minor fixes.

3.3.1

- Don't crash on new RPC fields
- Add `PackageBaseID` field (in AUR v3.0.0+1)

3.3.0 PKGBUILDer is now compatible with AURv3. Note this is preliminary support, and as such, there might still be bugs.

3.2.0

- Downgrade listings have been fixed (Issue #31)
- Packages are now moved to `/var/cache/pacman/pkg/` and installed from there (previously, they were copied and installed from `/tmp`)

3.1.13

- Fix packages being built when `-i` was issued (Issue #29)
- Fix some instances where PKGBUILDer would quit, even though it shouldn't (as requested while running the main function — internal only)
- Modify downgrade messages to fit pacman more
- Update the translations

3.1.12 Make `setup.py` work properly and have PKGBUILDer install.

3.1.11

- Fix AUR/HTTP exceptions reporting. (via Issue #28)

3.1.10

- Inexistent packages don't crash badly anymore
- AUR-dependency-builds do not crash everything either
- `-Syu` can work without `stty size` working
- `$CARCH` is defined for dependency checks (Issue #28)

3.1.9 Something went wrong last release, and it did not fix what it was supposed to...

3.1.8 Fixed `pkgbuilder -F` (broken one version ago)

3.1.7 AUR v2.3.0 compatibility (fonts category)

3.1.6 Added `-userfetch`.

3.1.5 Added Vietnamese.

3.1.4 Fixed a mess.

3.1.3 Fixes to the safeupgrade; added Italian, Spanish and Turkish.

3.1.2 Modified timestamp generation in the Package classes.

3.1.1 A quick bugfix update.

3.1.0 Added some magic to AUR dependency building.

3.0.1 A lot of tiny fixes.

Also known as release *three point oh point **ell***, blame Consolas.

3.0.0 A new major release, introducing many new wonderful features.

2.99.6.0 Package classes done.

2.99.5.0 Exceptions 2.0 fully implemented.

2.99.4.0 First four stages done.

2.1.6.0–2.1.6.3 VerbosePkgLists, DS.run_command() and subsequent fixes to the latter

2.1.5.14 Fixing a quite important bug in the install process

2.1.5.13 2013! Oh, and the revision number is 13, too!

2.1.5.11—2.1.5.12 Fixes to the AUR v2.0 magic.

2.1.5.10 AUR v2.0 support.

2.1.5.9 And another issue that I have not noticed, in a tiny change of Update behavior. Sorry for all those updates, but I do not do testing on everything, just on stuff I think could break a lot (have you seen a bugfix for the ABS build magic? I haven't. Well, the validation fix was partially related to the ABS magic, but it was detected during a run of `pb -S` with an inexistent package that I thought might exist. I actually revised all the changes that happened since 2.1.5.5 (a release without those bugs) and I think I'm done with everything now.

2.1.5.8 A bug in the wrapper fixed. Sorry, but sometimes I forget to test certain things, and I forgot that the protocol choice in PBWrapper is implemented through a workaround. Also, fixed the installation validation behavior.

2.1.5.7 Fixed some bugs that managed to slip through while working on 2.1.5.6.

2.1.5.6 Added ABS support.

2.1.5.4—2.1.5.5 Applying patches from vadmium/pkgbuilder, also adding a few other fixes and changing the `pb` version number up to 0.2.0.

2.1.5.3 A bugfix for package copying and installation (signatures were passed to `-U`) broke the installation mechanism so only one package got installed. Also, fixing a bug with a `STDIN` that is not a terminal (eg. `xargs`, and I hope nobody is using it to search for stuff)

2.1.5.2 Fixed a bug where an error in `makepkg` while running an Upgrade crashed PB and thrown an unhelpful traceback.

2.1.5.1 More tiny bugfixes.

2.1.5.0 A release including the sample scripts, among other stuff. This is a release which now has all the functionality I want it to have. And it's time to move onto a new project, the `aurqt` interface for the AUR. Or maybe something else? [Update from the future: it wasn't all I wanted. Moreover, PKGBUILDer is a dependency of `aurqt`.]

2.1.4.9 Some more bugfixes.

2.1.4.8 Introducing a backwards-compatibility-breaking change of `.utils.Utils.info()`

2.1.4.7 Quite a lot of changes.

2.1.4.5-2.1.4.6 Fixes some bugs.

2.1.4.4 The mature release, including downgrades, excluding mess.

2.1.4.2-2.1.4.3 Bug fixes, thanks to fosskers (from aura, another AUR helper).

2.1.4.1 Dropped the useless msgcodes, which made no sense at all.

2.1.4.0 pb wrapper!

2.1.3.7 depcheck ignores empty deps now.

2.1.3.2-2.1.3.6 little, unimportant fixes, for docs and locale and whatnot.

2.1.3.1 print_package_*

2.1.3.0 Now divided into modules.

2.1.2.33 Bugfix release, final release of 2.1.2 series.

2.1.2.32

Test suite introduced. (unittests, nosetests were used in the very beginning)

2.1.2.31 The big changes begin. Introducing requests.

2.1.2.1-2.1.2.30 Tiny, unimportant bugfixes. Somehow, my version numbering broke, stuff went completely apeshit, and I do not understand it.

2.1.2.0 Support for the new pyalpm.

2.1.1.8 Fixed the license.

2.1.1.7 Some little changes.

2.1.1.6 Fixed AUR dep detection. (not released into git.)

2.1.1.5 Some fixes for locale support.

2.1.1.4 Locale support!

2.1.1.0-2.1.1.3 Little changes and refinements.

2.1.0 First OOP-based release. Including -Syu, BSD License, our own AUR class, documentation, module usage-friendliness.

2.1.0-prerelease A prerelease build of 2.1.0. This one still works with the AUR class by Xyne.

2.0 First release.

1.1 A more advanced version, never released publicly, and I do not even have any backups. Nobody cares anyways.

1.0 First and only release.

Indices and tables

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

[colors] Code below.

[pacman] *wrapper-friendly behavior* (-S): building in /tmp; Uutils.print_package() says aur/name

[inttext] Used by /bin/pkgbuilder, internationalized, looks like this: Aborted by user! Quitting...

[wrapperinttext] Used by /bin/pb, internationalized, looks like this: Interrupt signal received\n

[conf] In order: ~/.config/, ~/.config/kwpolska, ~/.config/kwpolska/pkgbuilder (may differ depending on system config)

p

- `pkgbuilder.__main__`, 11
- `pkgbuilder.aur`, 9
- `pkgbuilder.build`, 10
- `pkgbuilder.exceptions`, 20
- `pkgbuilder.package`, 11
- `pkgbuilder.pbds`, 13
- `pkgbuilder.transaction`, 15
- `pkgbuilder.ui`, 17
- `pkgbuilder.upgrade`, 18
- `pkgbuilder.utils`, 18
- `pkgbuilder.wrapper`, 19

A

ABSPackage, 11
 AUR
 RPC, 9
 AUR (class in pkgbuilder.aur), 9
 AURError, 20
 AURPackage, 11
 AURPackage (class in pkgbuilder.package), 12
 auto_build() (in module pkgbuilder.build), 10
 auto_upgrade() (in module pkgbuilder.upgrade), 18

B

build, 10
 build_runner() (in module pkgbuilder.build), 11
 bump() (pkgbuilder.ui.ProgressThrobber method), 17

C

CHANGELOG, 24
 clone() (in module pkgbuilder.build), 10
 CloneError, 20
 colorsoff() (pkgbuilder.pbds.PBDS method), 13
 colorson() (pkgbuilder.pbds.PBDS method), 13
 ConnectionError, 20
 contributing, 23

D

debugmode() (pkgbuilder.pbds.PBDS method), 13
 depcheck() (in module pkgbuilder.build), 10

E

Exceptions, 20
 exitcode (pkgbuilder.transaction.Transaction attribute),
 15

F

fancy_error() (pkgbuilder.pbds.PBDS method), 13
 fancy_error2() (pkgbuilder.pbds.PBDS method), 13
 fancy_msg() (pkgbuilder.pbds.PBDS method), 14
 fancy_msg2() (pkgbuilder.pbds.PBDS method), 14
 fancy_warning() (pkgbuilder.pbds.PBDS method), 14

fancy_warning2() (pkgbuilder.pbds.PBDS method), 14
 fetch_runner() (in module pkgbuilder.build), 11
 from_aurdict() (pkgbuilder.package.AURPackage class
 method), 12
 fromjson() (pkgbuilder.transaction.Transaction class
 method), 15

G

gather_foreign_pkgs() (in module pkgbuilder.upgrade),
 18
 generate_filename() (in module pkgbuilder.transaction),
 15
 get_setting() (pkgbuilder.pbds.PBDS method), 14
 get_termwidth() (in module pkgbuilder.ui), 17

H

hanging_indent() (in module pkgbuilder.ui), 17
 HTTPError, 20

I

info() (in module pkgbuilder.utils), 18
 install() (pkgbuilder.transaction.Transaction method), 16
 issues, 23

J

jsonmultiinfo() (pkgbuilder.aur.AUR method), 9
 jsonreq() (pkgbuilder.aur.AUR method), 9

L

LICENSE, 24
 list_upgradable() (in module pkgbuilder.upgrade), 18
 load() (pkgbuilder.transaction.Transaction class method),
 16
 locale, 24

M

main, 11
 main() (in module pkgbuilder.__main__), 11
 main() (in module pkgbuilder.wrapper), 19
 MakepkgError, 20

mlist() (in module pkgbuilder.ui), 17
move() (pkgbuilder.transaction.Transaction method), 16
msearch() (in module pkgbuilder.utils), 19
msg() (pkgbuilder.ui.Progress method), 17
multiinfo() (pkgbuilder.aur.AUR method), 10

N

NetworkError, 20

P

Package, 11
Package (class in pkgbuilder.package), 11
PackageError, 20
PackageNotFoundError, 20
pacman_pkgpaths (pkgbuilder.transaction.Transaction attribute), 16
PBDS, 13
PBDS (built-in class), 13
PBDS (class in pkgbuilder.pbds), 13
PBException, 20
PBWrapper, 19
pkgbuilder.__main__ (module), 11
pkgbuilder.aur (module), 9
pkgbuilder.build (module), 10
pkgbuilder.exceptions (module), 20
pkgbuilder.package (module), 11
pkgbuilder.pbds (module), 13
pkgbuilder.transaction (module), 15
pkgbuilder.ui (module), 17
pkgbuilder.upgrade (module), 18
pkgbuilder.utils (module), 18
pkgbuilder.wrapper (module), 19
prepare_deps() (in module pkgbuilder.build), 10
print_package_info() (in module pkgbuilder.utils), 19
print_package_search() (in module pkgbuilder.utils), 19
Progress (class in pkgbuilder.ui), 17
ProgressThrobber (class in pkgbuilder.ui), 17
pyc (pkgbuilder.pbds.PBDS attribute), 14
pycreload() (pkgbuilder.pbds.PBDS method), 14

R

request() (pkgbuilder.aur.AUR method), 10
root_crash() (pkgbuilder.pbds.PBDS method), 14
RPC, 9
rpc (pkgbuilder.aur.AUR attribute), 10
rsync() (in module pkgbuilder.build), 10
run() (pkgbuilder.transaction.Transaction method), 16
run_command() (pkgbuilder.pbds.PBDS method), 14

S

sample scripts, 21
SanityError, 20
save() (pkgbuilder.transaction.Transaction method), 16

search() (in module pkgbuilder.utils), 19
search() (pkgbuilder.aur.AUR method), 10
sudo() (pkgbuilder.pbds.PBDS method), 14
Syu, 18

T

Throbber (class in pkgbuilder.ui), 17
throbber_alive (pkgbuilder.ui.Throbber attribute), 17
tojson() (pkgbuilder.transaction.Transaction method), 16
transaction, 15
Transaction (class in pkgbuilder.transaction), 15
TransactionStatus (class in pkgbuilder.transaction), 16

U

UI, 17
upgrade, 18
utils, 18

V

validate() (pkgbuilder.transaction.Transaction method), 16

W

wrapper, 19
wrapper() (in module pkgbuilder.wrapper), 19