NAME

pybuild − invokes various build systems for requested Python versions in order to build modules and extensions

SYNOPSIS

pybuild [ACTION] [BUILD SYSTEM ARGUMENTS] [DIRECTORIES] [OPTIONS]

DEBHELPER COMMAND SEQUENCER INTEGRATION

- build depend on *dh−python*,
- build depend on all supported Python interpreters, pybuild will use it to create a list of interpreters to build for. Recognized dependencies:
  - *python3−all−dev* − for Python extensions that work with Python 3.X interpreters,
  - *python3−all−dbg* − as above, add this one if you're building −dbg packages,
  - *python3−all* − for Python modules that work with Python 3.X interpreters,
  - *python3−dev* − builds an extension for default Python 3.X interpreter (useful for private extensions, use python3−all−dev for public ones),
  - *python3* − as above, used if headers files are not needed to build private module,
  - *python−all−dev* − for Python extensions that work with obsolete Python 2.X interpreters,
  - *python−all−dbg* − as above, add this one if you're building −dbg packages,
  - *python−all* − for Python modules that work with obsolete Python 2.X interpreters,
  - *pypy* − for PyPy 2.X interpreter.
- add *−−buildsystem=pybuild* to dh's arguments in debian/rules,
- if more than one binary package is build: add debian/python−foo.install files, or *export PYBUILD_NAME=modulename* (modulename will be used to guess binary package prefixes), or *export PYBUILD_DESTDIR* env. variables in debian/rules
- add *−−with=python3* or *−−with=python3,python2,pypy* to dh's arguments in debian/rules (see proper helper's manpage for more details) or add *dh−sequence−python3* (*dh−sequence−python2* for Python 2.X, *dh−sequence−pypy* for PyPy) to Build−Depends

debian/rules file example:

```
#! /usr/bin/make -f
export PYBUILD_NAME=foo
%:
	dh $@ --with python2,python3 --buildsystem=pybuild
```

OPTIONS

Most options can be set (in addition to command line) via environment variables. PyBuild will check:

- PYBUILD_OPTION_VERSIONED_INTERPRETER (f.e. PYBUILD_CLEAN_ARGS_python3.2)
- PYBUILD_OPTION_INTERPRETER (f.e. PYBUILD_CONFIGURE_ARGS_python3−dbg)
- PYBUILD_OPTION (f.e. PYBUILD_INSTALL_ARGS)

optional arguments

−h, **−−help**
: show this help message and exit

−v, **−−verbose**
: turn verbose mode on

−q, **−−quiet**
: doesn't show external command's output

−q_q_**, −−really−quiet**
be quiet

−−version
show program's version number and exit

ACTION

The default is to build, install and test the library using detected build system version by version. Selecting one of following actions, will invoke given action for all versions − one by one − which (contrary to the default action) in some build systems can overwrite previous results.

−−detect
return the name of detected build system

−−clean
clean files using auto−detected build system specific methods

−−configure
invoke configure step for all requested Python versions

−−build
invoke build step for all requested Python versions

−−install
invoke install step for all requested Python versions

−−test invoke tests for auto−detected build system

−−list−systems
list available build systems and exit

−−print
print pybuild's internal parameters

TESTS

unittest's discover from standard library (available in Python 2.7 and >= 3.2) is used in test step by default.

−−test−nose
use nose module in test step, remember to add python−nose and/or python3−nose to Build−Depends

−−test−pytest
use pytest module in test step, remember to add python−pytest and/or python3−pytest to Build−Depends

−−test−tox
use tox command in test step, remember to add tox to Build−Depends. Requires tox.ini file

testfiles

Tests are invoked from within build directory to make sure newly built files are tested instead of source files. If test suite requires other files in this directory, you can list them in *debian/pybuild.testfiles* file (you can also use *debian/pybuild−pythonX.testfiles* or *debian/pybuild−pythonX.Y.testfiles*) and files listed there will be copied before test step and removed before install step. By default only *test* and *tests* directories are copied to build directory.

BUILD SYSTEM ARGUMENTS

Additional arguments passed to the build system. −−system=custom requires complete command in −−foo−args parameters.

−−before−clean *COMMAND*
invoked before the clean command

−−clean−args *ARGUMENTS*
arguments added to clean command generated by build system plugin

−−after−clean *COMMAND*
invoked after the clean command

−−before−configure *COMMAND*
invoked before the configure command

−−configure−args *ARGUMENTS*
arguments added to configure command generated by build system plugin

−−after−configure *COMMAND*
invoked after the configure command

−−before−build *COMMAND*
invoked before the build command

−−build−args *ARGUMENTS*
arguments added to build command generated by build system plugin

−−after−build *COMMAND*
invoked after the build command

−−before−install *COMMAND*
invoked before the install command

−−install−args *ARGUMENTS*
arguments added to install command generated by build system plugin

−−after−install *COMMAND*
invoked after the install command

−−before−test *COMMAND*
invoked before the test command

−−test−args *ARGUMENTS*
arguments added to test command generated by build system plugin

−−after−test *COMMAND*
invoked after the test command

variables that can be used in *ARGUMENTS* **and** *COMMAND*

- *{version}* will be replaced with current Python version, you can also use *{version.major}*, *{version.minor}*, etc.
- *{interpreter}* will be replaced with current interpreter, you can also use *{interpreter.include_dir}*
- *{dir}* will be replaced with sources directory,
- *{destdir}* will be replaced with destination directory,
- *{home_dir}* will be replaced with temporary HOME directory, where plugins can keep their data (.pybuild/interpreter_version/ by default),
- *{build_dir}* will be replaced with build directory
- *{install_dir}* will be replaced with install directory.
- *{package}* will be replaced with suggested package name, if −−name (or PYBUILD_NAME) is set to *foo*, this variable will be replaced to *python−foo*, *python3−foo* or *pypy−foo* depending on interpreter which is used in given iteration.

DIRECTORIES

−d *DIR*, **−−dir** *DIR*
set source files directory − base for other relative dirs [by default: current working directory]

−−dest−dir *DIR*
set destination directory [default: debian/tmp]

−−ext−dest−dir *DIR*
set destination directory for .so files

−−ext−pattern *PATTERN*
regular expression for files that should be moved if −−ext−dest−dir is set [default: *.so(.[^/]*)?$*]

−−ext−sub−pattern *PATTERN*
regular expression for part of path/filename matched in −−ext−pattern that should be removed or replaced with −−ext−sub−repl

−−ext−sub−repl *PATTERN*
replacement for matches in −−ext−sub−pattern

−−install−dir *DIR*
set installation directory [default: .../dist−packages]

−−name *NAME*
use this name to guess destination directories (depending on interpreter, "foo" sets debian/python−foo, debian/python3−foo, debian/python3−foo−dbg, etc.)

variables that can be used in *DIR*

- *{version}* will be replaced with current Python version,
- *{interpreter}* will be replaced with selected interpreter.

LIMITATIONS

−s *SYSTEM*, **−−system** *SYSTEM*
select a build system [default: auto−detection]

−p *VERSIONS*, **−−pyver** *VERSIONS*
build for Python VERSIONS. This option can be used multiple times. Versions can be separated by space character. The default is all Python 3.X supported versions.

−i *INTERPRETER*, **−−interpreter** *INTERPRETER*
change interpreter [default: python{version}]

−−disable *ITEMS*
disable action, interpreter, version or any mix of them. Note that f.e. python3 and python3−dbg are two different interpreters, −−disable test/python3 doesn't disable python3−dbg's tests.

disable examples

- *−−disable test/python2.5−dbg* − disables tests for python2.5−dbg
- *−−disable '2.4 2.7'* − disables all actions for version 2.4 and 2.7
- *PYBUILD_DISABLE_python2=1* − disables all actions for Python 2.X
- *PYBUILD_DISABLE_python3.3=test* − disables tests for Python 3.3
- *PYBUILD_DISABLE=test/python3.3* − same as above
- *PYBUILD_DISABLE=configure/python3 2.4 pypy* − disables configure action for all python3 interpreters, all actions for version 2.4, and all actions for pypy

SEE ALSO

- dh_python2(1)
- dh_python3(1)
- *https://wiki.debian.org/Python/Pybuild*
- *http://deb.li/pybuild* − most recent version of this document

AUTHOR

Piotr Oarowski, 2012-2019