

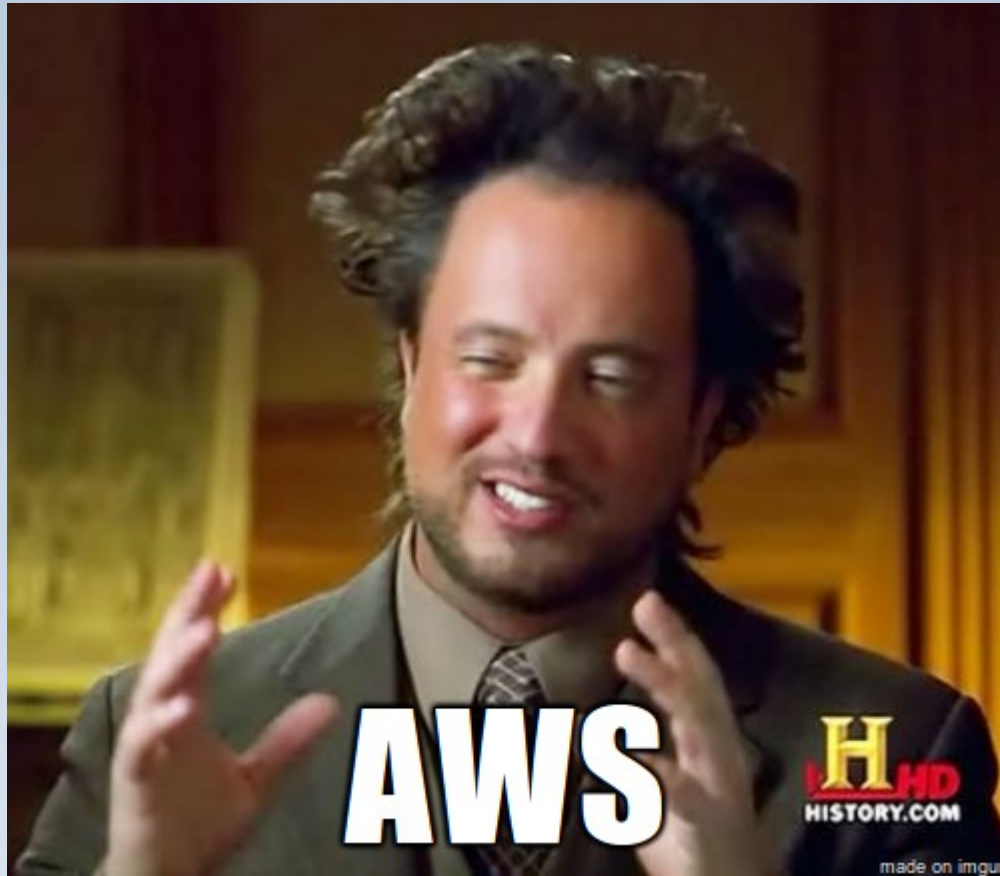
Difficulties of Python code development: packages, virtualenvs and package mangers

Michał Wodyński
pankejk@protonmail.com
23.07.2020

Agenda

- 1.Packages
- 2.Virtualenvs
- 3.Package mangers

Story 1



Single file

Bottle

Source code

- Zipped Python source code
- Preferred *.zip extension – win and unix support
- May contain C source files – ext_modules or libraries
- Scripts – defined in section scripts
- README.md, setup.py, setup.cfg
- package_data, meta_data

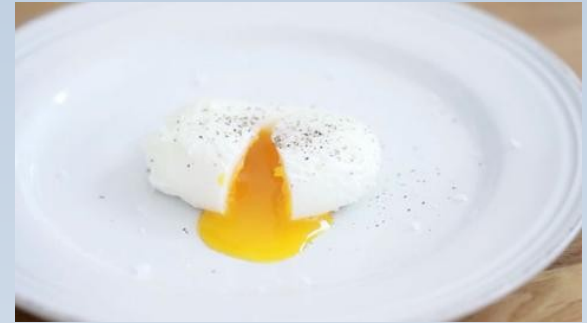
Egg



- Introduced by setuptools in 2004
- No need for building or compilation
- It is distribution format and runtime installation format
- *.egg-info directory importable

Egg

No official PEP



Wheel



- Also format that don't need building or compilation
- Introduced by [PEP427](#) in 2012
- Follows [PEP376](#) (distributions standard)
- [PEP426](#) (package metadata)
- Current standard for built and binary
- Packaging format

Wheel



- No *.pyc files but may contain other pre-compiled resources
- *.dist-info directory
- File name convention: Python version and implementation, ABIs, system architectures.

Wheel



- It is versioned
- Wheels can be importable but it is not supported now - [PEP427](#)

`pip-20.1.1-py2.py3-none-any.whl` – universal
`numpy-1.19.0-cp36-cp36m-macosx_10_9_x86_64.whl`

Wheel

- The Black Magic of the Python Wheels – Elena Hashman:
[video](#), [pdf](#)
- [Manylinux](#) project
- [Auditwheel](#) project

`python setup.py bdist_wheel`



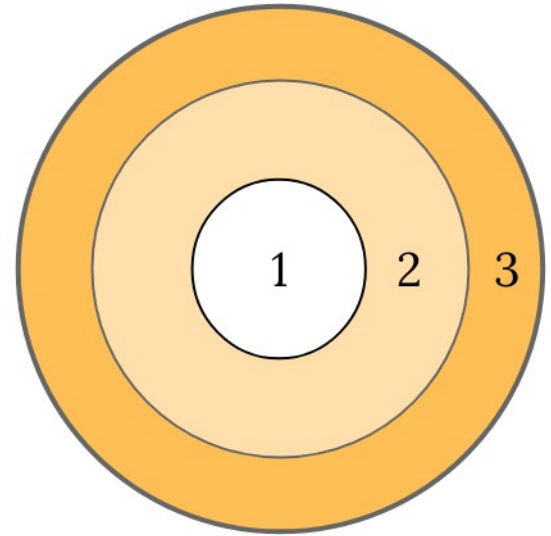
Packages

Packaging for Python **tools** and **libraries**



1. **.py** - standalone modules
2. **sdist** - Pure-Python packages
3. **wheel** - Python packages

(With room to spare for static vs. dynamic linking)



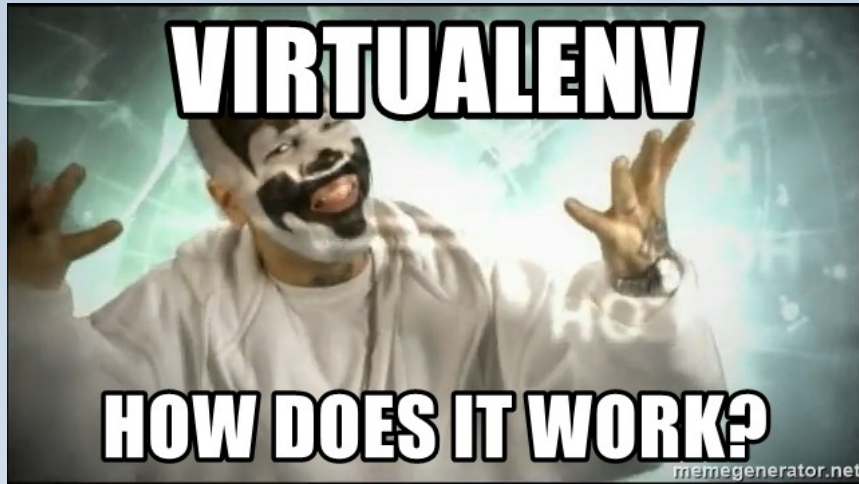
Story 2

```

09:31:29.239 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] INFO c.b.t.u.u.CreateTenantUsersAndGroups - Ple
ase check if user: Identity [username=admin, groups=[Administrators]] is already created in realm:/BmcRealm
09:31:29.309 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] ERROR c.b.t.u.u.CreateTenantUsersAndGroups - Err
or in creating Identity [username=service_admin, groups=[Model Administrators]].Received Http Status Code 500
09:31:29.310 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] INFO c.b.t.u.u.CreateTenantUsersAndGroups - Ple
ase check if user: Identity [username=service_admin, groups=[Model Administrators]] is already created in realm:/BmcRealm
09:31:29.338 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] ERROR c.b.t.u.u.CreateTenantUsersAndGroups - Err
or in creating Identity [username=csn_user, groups=[Administrators]].Received Http Status Code 500
09:31:29.338 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] INFO c.b.t.u.u.CreateTenantUsersAndGroups - Ple
ase check if user: Identity [username=csn_user, groups=[Administrators]] is already created in realm:/BmcRealm
09:31:29.366 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] ERROR c.b.t.u.u.CreateTenantUsersAndGroups - Err
or in creating Identity [username=user, groups=[Supervisors]].Received Http Status Code 500
09:31:29.366 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] INFO c.b.t.u.u.CreateTenantUsersAndGroups - Ple
ase check if user: Identity [username=user, groups=[Supervisors]] is already created in realm:/BmcRealm
09:31:29.392 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] ERROR c.b.t.u.u.CreateTenantUsersAndGroups - Err
or in creating Identity [username=event_admin, groups=[Monitoring Administrators]].Received Http Status Code 500
09:31:29.392 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] INFO c.b.t.u.u.CreateTenantUsersAndGroups - Ple
ase check if user: Identity [username=event_admin, groups=[Monitoring Administrators]] is already created in realm:/BmcRealm
09:31:29.419 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] ERROR c.b.t.u.u.CreateTenantUsersAndGroups - Err
or in creating Identity [username=bppmws_internal, groups=[Administrators]].Received Http Status Code 500
09:31:29.419 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueSightPServerPostInstallInstallationTask] INFO c.b.t.u.u.CreateTenantUsersAndGroups - Ple
ase check if user: Identity [username=bppmws_internal, groups=[Administrators]] is already created in realm:/BmcRealm
09:31:29.447 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueS
or in creating Identity [username=oper, groups=[Operators]].Received Http Status
09:31:29.448 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueS
ase check if user: Identity [username=oper, groups=[Operators]] is already creat
09:31:29.648 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueS
or in creating Identity [username=apiuser, groups=[API Group]].Received Http Sta
09:31:29.648 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueS
ase check if user: Identity [username=apiuser, groups=[API Group]] is already cr
09:31:29.648 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueS
rs are created successfully
09:31:29.672 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueS
t Installation cleanup invoked
09:31:30.780 [Thread-188 -- Task: com.bmc.install.product.truesighttpserver.TrueS
fully started the CSR, IdxServer and the DB processes
(Jul 07 2017 09:31:32.663 AM -0700),CONFIG,Command return code,
LOG EVENT {Description=[0]}
(Jul 07 2017 09:31:32.677 AM -0700),INFO,setup,
PROGRESS EVENT {Description=[Unpacked payload entry],Progress=[100],Detail=[1
(Jul 07 2017 09:31:32.693 AM -0700),INFO,setup,
LOG EVENT {Description=[Extracted payload entry count],Detail=[1]}

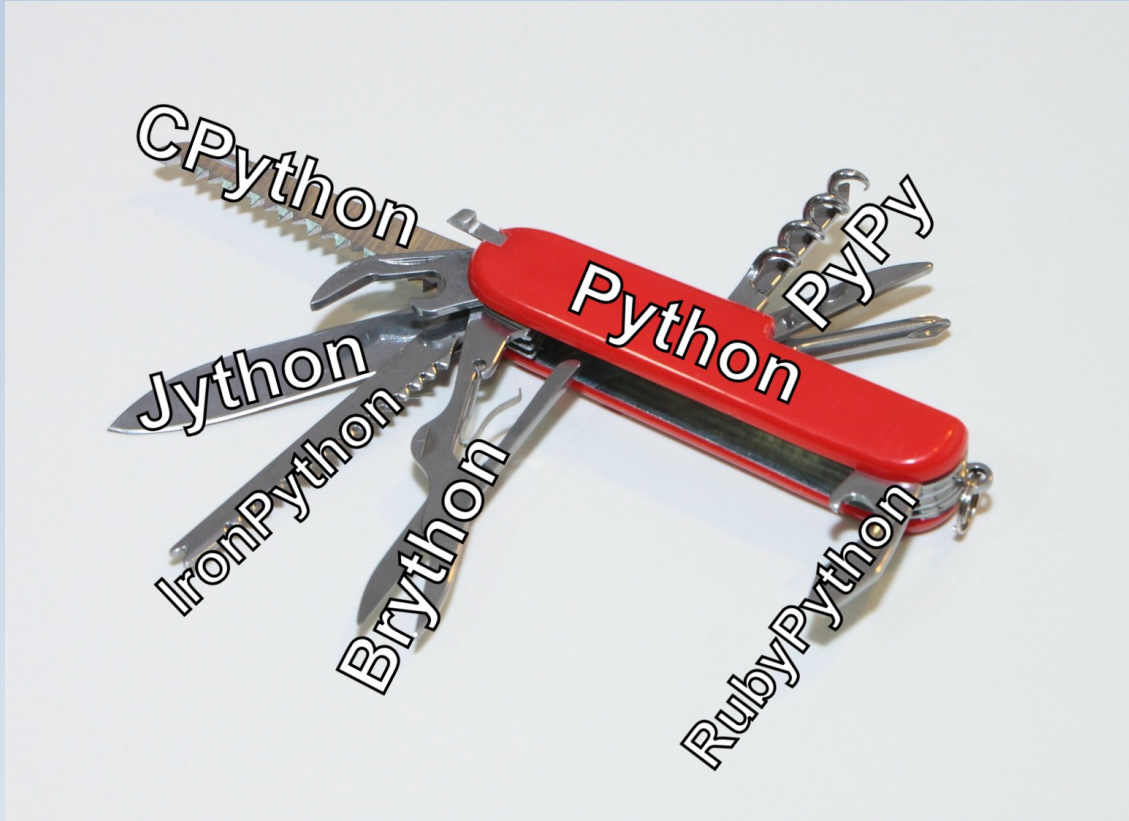
```

Virtualenvs



- Isolated environment for python projects
- Virtualenvwrapper
- Adding directory to begin of PATH env variable
- How test other Python versions

Python versions



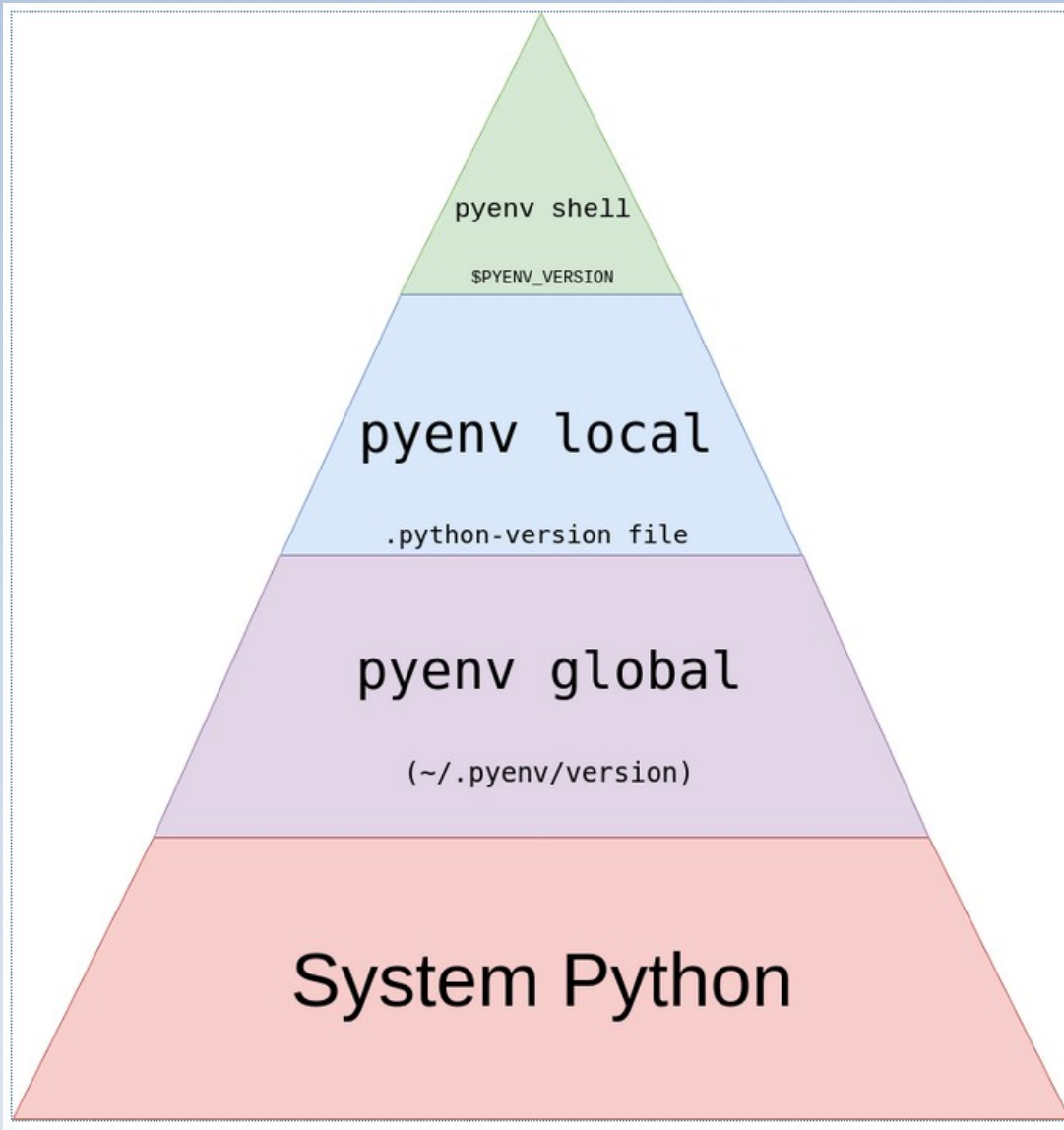
- Cpython
- Pypy
- Jython
- IronPython
- MicroPython
- Stackless

Pyenv

- Multiple versions of Python (latest)
- Switch between versions
- Virtualenvs
- Automatic activation of the Python versions and virtual environments

Pyenv

- **Installation** and **dependencies**
- `pyenv install --list`
- `pyenv install -v 3.7.2`
- `pyenv uninstall 2.7.15`
- `pyenv versions`
- `which python` vs `pyenv which python`
- `pyenv global 2.7.15`
- `pyenv local 2.7.13`
- `pyenv shell 3.8-dev`
- `pyenv activate <environment_name>`
- `pyenv deactivate`



Story 3



Easy install

- Released in 2004 as part of setuptools
- Installing packages and dependencies using requirements specifiers
- Installing from eggs

Pip



- Released in 2008, built on top of setuptools
- Not installing eggs but rather sdist or wheels
- Introduce requirements.txt

	pip	easy_install
Installs from Wheels	Yes	No
Uninstall Packages	Yes (pip uninstall)	No
Dependency Overrides	Yes (Requirements Files)	No
List Installed Packages	Yes (pip list and pip freeze)	No
PEP 438 Support	Yes	No
Installation format	‘Flat’ packages with egg-info metadata.	Encapsulated Egg format
sys.path modification	No	Yes
Installs from Eggs	No	Yes
pylauncher support	No	Yes 1
Multi-version installs	No	Yes
Exclude scripts during install	No	Yes
per project index	Only in virtualenv	Yes, via setup.cfg

Setup.py

- Describe module to use various commands to:
- Building
- distributing,
- installing

Setup.py

```
#!/usr/bin/env python
```

```
from distutils.core import setup
```

```
setup(name='Distutils',  
      version='1.0',  
      description='Python Distribution Utilities',  
      author='Greg Ward',  
      author_email='gward@python.net',  
      url='https://www.python.org/sigs/distutils-sig/',  
      packages=['distutils', 'distutils.command'],  
      )
```

Pip-tools

- Pip-compile – setup.py + requirements.in = requirements.txt
- Pip-sync – checks libraries in environment and compare it with requirements.txt

Pipenv



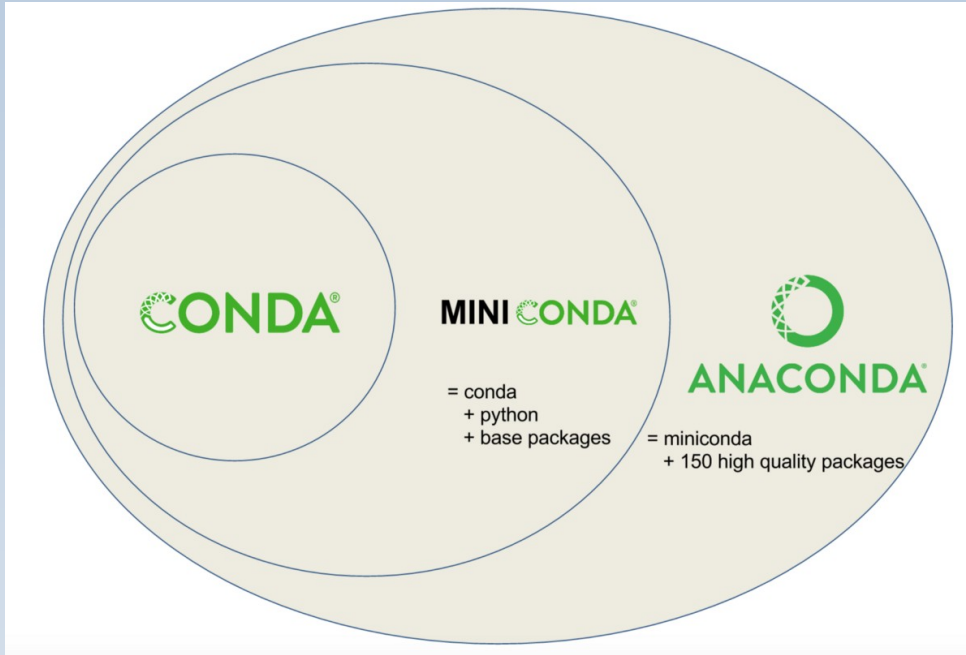
- Managing different environments
- Installing Python packages
- Environment reproducibility

Poetry



- Managing different environments
- Installing Python packages
- Environment reproducibility
- Packaging and publishing python packages

Anaconda, conda, miniconda



- No PEP
- Conda – package manager
- Is not using Pypi and is not supporting Python packages
- Conda packages not working in non-Conda environments

Anaconda, conda, miniconda



- Conda vs conda + pip (other package manger)
- Conda for binary packages and pip for python packages (latest packages)
- Scientific libraries
- Managing environments
- Precompiled packages
- Channels
- Not complete packages
- Libraries behind current versions
- Different maintainers of packages

PDM

- [PEP 582](#) local package installer and runner
- no use of virtualenv.
- Fast dependency resolver
- [PEP 517](#) build backend

PIP~~x~~



- Installing package
- Upgrade certain package or all packages
- Reinstall environment
- Installing package in isolation

	Installing python packages	Managing python versions	Managing virtual environments	Environment reproducibility
easy_install	yes	no	no	no
pip	yes	no	no	yes
Pip-tools	yes	no	no	yes
pipenv	yes	no	yes	yes
poetry	yes	yes	yes	yes
conda	yes	yes 1	yes	no
pdm	yes	yes 2	yes 3	yes
pipx	yes	no	no	no 4
venv	no	no	yes	no
pyenv	no	yes 1	yes	no

	Installation	dev/prod dependency	file format	dependency lock	dependency hashing
easy_install	preinstalled with python	no	no	no	no
pip	preinstalled with python	yes	requirements.txt	pip freeze	yes
Pip-tools	pip install pip- tools	yes 5	setup.py requirements.in requirements.txt	pip freeze	yes
pipenv	pip install pipenv	yes	Pipfile Pipfile.lock	pipenv lock	yes
poetry	pip install poetry	yes	poetry.lock	poetry lock	yes
conda	download miniconda or anaconda	no 6	no 6	no 6	no 6
pdm	pip install pdm	yes	pdm.lock	yes	yes
pipx3	pip install pipx	no	no	no	no
venv	preinstalled with python	no	no	no	no
pyenv	use installer from pyenv github	no	no	no	no

	resolving dependency conflicts	wheel support	egg support	Society support
easy_install	no	no	yes	Deprecated
pip	no	yes	no	Good
Pip-tools	yes	yes	no	Good
pipenv	yes	yes	no	Good
poetry	yes	yes	no	Good
conda	no	no	no	Good 6
pdm	yes	yes	no	Early stage
pipx3	no 4	yes	no	Good
venv	no	no	no	Good
pyenv	no	no	no	Good

- 1) Even though conda can install non-python packages, it can't replace your system package manager
- 2) pdm must cooperate with pyenv.
- 3) Virtualenv is placed in the same directory as project
- 4) pipx is installing only one package with its dependencies
- 5) Two Separate Python requirements.txt files
- 6) Anaconda channel is maintained by different people than real authors of certain packages

Installation benchmarks

Package manager	Time
pipenv	1m6,960s
poetry	0m11,981s
pdm	0m47,447s
pip	0m7,156s
pipx	0m5,201s

My setup

- 1) Get-pip.py script
- 2) Install pipenv/popetry and other tools
- 3) Install pyenv
- 4) Create virtualenv

Resources

- 1) <https://www.youtube.com/watch?v=02aAZ8u3wEQ>
- 2) <https://packaging.python.org/overview/>
- 3) <https://realpython.com/intro-to-pyenv/>
- 4) <https://packaging.python.org/discussions/wheel-vs-egg/>
- 5) <https://hashman.ca/pycon-2019/ehashman-pycon2019-slides.pdf>
- 6) <https://realpython.com/python-virtual-environments-a-primer/>
- 7) <https://docs.python.org/3/distutils/setupscript.html>
- 8) <https://modelpredict.com/python-dependency-management-tools>
- 9) <https://pypi.org/project/pip-tools/>
- 10) <https://github.com/frostming/pdm>

Questions?

Michał Wodyński
pankejk@protonmail.com
23.07.2020