Welcome to the Cheese Shop

setuptools, virtualenv and PyPUG

Dale Visser
https://dalevisser.wordpress.com/

Outline

- Cheese Shop?
- Setuptools
- Sdists, Eggs and Wheels
- Demo of pip and virtualenv
- Online Resources
- Conclusion

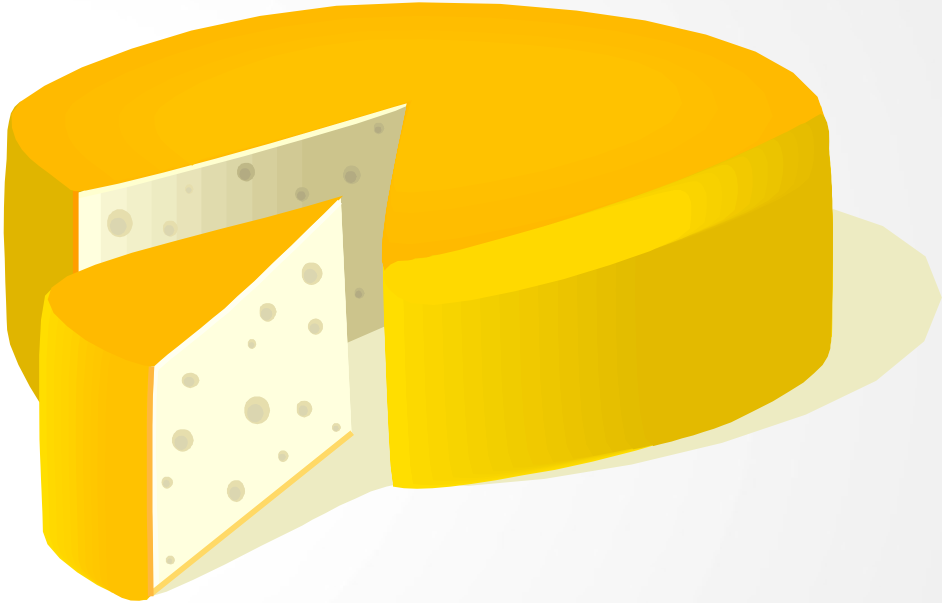


Image credit: http://www.clker.com/clipart-cheese-wheel.html

Cheese Shop?

- Name comes from Monty Python, *of course*.
  - For full history: https://youtu.be/pZTiAkwiPgI
  - For the sketch: https://youtu.be/B3KBuQHHKx0
- Corporate sponsor preferred something more formal, less likely to infringe copyright:
  - The Python Package Index, or PyPI
  - https://pypi.python.org/pypi

Setuptools (1/2)

https://bitbucket.org/pypa/setuptools

```
from setuptools import setup, find_packages
setup(
    name = "HelloWorld",
    version = "0.1",
    packages = find_packages(),
    scripts = ['say_hello.py'],

    # Project uses reStructuredText, so ensure that the docutils get
    # installed or upgraded on the target machine
    install_requires = ['docutils>=0.3'],

    package_data = {
        # If any package contains *.txt or *.rst files, include them:
        '': ['*.txt', '*.rst'],
        # And include any *.msg files found in the 'hello' package, too:
        'hello': ['*.msg'],
    },
```

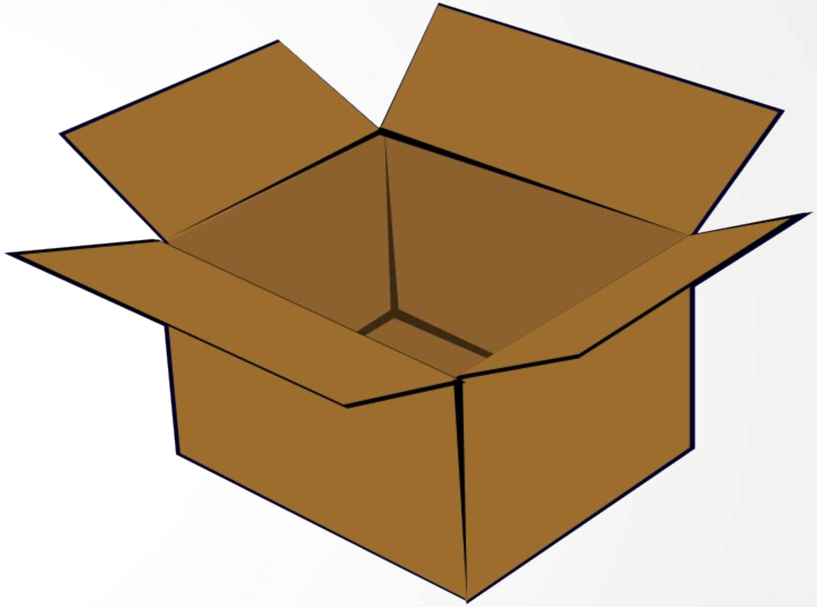
Setuptools (2/2)

```
    # metadata for upload to PyPI
    author = "Me",
    author_email = "me@example.com",
    description = "This is an Example Package",
    license = "PSF",
    keywords = "hello world example examples",
    url = "http://example.com/HelloWorld/",   # project home page, if any

    # could also include long_description, download_url, classifiers, etc.
)
```

Python Distributions – 2 kinds

- Source Distributions or “sdists”
  - python setup.py sdist
- Built Distributions (“bdists”)
  - Eggs
    - python setup.py bdist or
    - python setup.py bdist_egg
  - Wheels
    - python setup.py bdist_wheel
    - python setup.py bdist_wheel --universal



Egg and Wheel Distribution Formats

- Eggs - classic
  - .egg – zip archive or folder containing package and metadata
  - .egg-info – metadata folder that sits *alongside* installed package folder
  - Lacks *formal* specification, but *is* well-described: http://bit.ly/egg_format

- Wheels – pip and PyPI preferred
  - .whl – zip archive containing package and metadata
  - .dist-info – metadatafolder that sits *alongside* installed package folder
  - Relevant specifications:
    - PEP-376 - Database of Installed Python Distributions
    - PEP-426 - Metadata for Python Software Packages 2.0
    - PEP-427 - The Wheel Binary Package Format 1.0

Universal Wheels?

- Platform wheels
  - Contain compiled extensions
  - Specific to a platform
    - Linux
    - Windows
    - OSX
  - PyPI only supports Windows and OSX

- Pure Python wheels
  - No compiled extensions
  - Subtype: universal wheels
    - Support both Python 2 and 3

pip and virtualenv demo

Online Resources

- Your first, best resource is this: https://packaging.python.org/
  - It is a collaborative resource created by the PyPI people. E.g., find out about *twine*
- Also, you can “pip install” from git repos!: http://bit.ly/pip_vcs

Image credit: http://www.clker.com/clipart-9829.html

Conclusion

- This talk just scratched the surface.
- Keeping your library-dependent software secure:
  - Have a look at OWASP Dependency Check.
  - With any luck, a near-future release will be Python-aware.