

Python Wheels and AIX - Part 2

Contributed by Michael Felt

Chapter 2 - Getting started with wheels

Creating a wheel is extremely easy

```
(pve) michael@x066:[/home/michael]pip wheel six
```

Collecting six

Using cached

```
https://files.pythonhosted.org/packages/73/fb/00a976f728d0d1fecfe898238ce23f502a721c0ac0ecfedb80e0d88c64e9/six-1.12.0-py2.py3-none-any.whl
```

```
Saved ./six-1.12.0-py2.py3-none-any.whl
```

Skipping six, due to already being wheel.

```
(pve) michael@x066:[/home/michael]ls -l
```

```
total 24
```

```
drwxr-xr-x  5 michael staff    256 Oct 31 07:27 pve
```

```
-rw-r--r--  1 michael staff  10586 Oct 31 07:45 six-1.12.0-py2.py3-none-any.whl
```

But, now that I have a wheel (.whl) - how can I identify what is in the file?

What's in a wheel name?

First, I should probably mention (again) what a wheel is. A wheel is a pre-packaged python module. So to identify the contents of a wheel file (.whl) it's name has several components (sub-tags?).

The first two (sub-)tags are the module name and module version, for example (six) and (1.12.0).

Note: I call the fields separated by the character '-' sub-tags. The next sub-tag relates to the Python version(s) supported. In this case py2.py3' indicates all versions of Python2 and Python3 are supported.

The 4th sub-tag is something mysterious to me, and safe to ignore for now.

The 5th sub-tag is the so-called pep425 platform tag. See PEP 425 -- Compatibility Tags for Built Distributions

The pep425 platform specification specifies platform specifics such as operating system (e.g., windows, linux, aix, freebsd) and binary mode (32-bit, 64-bit).

When the 4th and 5th tags are 'none' and 'any' there are no additional binary dependencies. I expect this to be 'the rule' for pure-python modules.

So, what does my (pep425) tag look like?

```
$ pip help
```

Shows lots of useful information.

```
$ pip debug # Only in the very latest version of pip!
```

Seperated by the character '-' this shows potential combinations of "3rd, 4th, and 5th" tags.

After updating pip - you will probably see something like:

```
root@x071:[/root]pip debug
```

```
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7. More details about Python 2 support in pip, can be found at https://pip.pypa.io/en/latest/development/release-process/#python-2-support
```

```
WARNING: This command is only meant for debugging. Do not use this with automation for parsing and getting these details, since the output and options of this command may change without notice.
```

```
pip version: pip 19.3.1 from /opt/lib/python2.7/site-packages/pip (python 2.7)
```

```
sys.version: 2.7.16 (default, Mar 4 2019, 14:15:42) [C]
```

```
sys.executable: /opt/bin/python
```

```
sys.getdefaultencoding: ascii
```

```
sys.getfilesystemencoding: ISO8859-1
```

```
locale.getpreferredencoding: ISO8859-1
```

```
sys.platform: aix5
```

```
sys.implementation:
```

```
name:
```

Compatible tags: 14

cp27-cp27m-aix_6_1

cp27-none-aix_6_1

py2-none-aix_6_1

cp27-none-any

cp2-none-any

py27-none-any

py2-none-any

py26-none-any

py25-none-any

py24-none-any

Note the "AIX" specific tags - where the pep425 part is "aix_6_1".

- This is broken (imho) - and in my Python3 releases I have updated the part of python that creates the pep425 tag. With that fix installed also submitted as a PR (pull-request) to CPython) you should see something like:

```
(pve) michael@x066:~/home/michael$ pip debug
```

```
WARNING: This command is only meant for debugging. Do not use this with automation for parsing and getting these details, since the output and options of this command may change without notice.
```

```
pip version: pip 19.3.1 from /home/michael/pve/lib/python3.6/site-packages/pip (python 3.6)
```

```
sys.version: 3.6.9 (default, Oct 25 2019, 15:49:23) [C]
```

```
sys.executable: /home/michael/pve/bin/python3.6
```

```
sys.getdefaultencoding: utf-8
```

```
sys.getfilesystemencoding: iso8859-1
```

```
locale.getpreferredencoding: ISO8859-1
```

```
sys.platform: aix6
```

```
sys.implementation:
```

```
  name: cpython
```

Compatible tags: 18

cp36-cp36m-AIX_6106_1043_64

cp36-abi3-AIX_6106_1043_64
cp36-none-AIX_6106_1043_64
cp35-abi3-AIX_6106_1043_64
cp34-abi3-AIX_6106_1043_64
cp33-abi3-AIX_6106_1043_64
cp32-abi3-AIX_6106_1043_64
py3-none-AIX_6106_1043_64
cp36-none-any
cp3-none-any
...

- Here the AIX pep425 tag includes the TLSP, builddate, and "bit-size". Note: the tag you see is based on the system you run the command on! You should see your current TLSP, build-date and bitness of your python executable.

In closing

I hope the purpose and labeling of wheels is understandable. Also, understandable is how the current official pep425 tag for AIX is not sufficient for actual use - in a distributed environment.

Note: the workaround has been to provide either RPM's (AIXTOOLBOX, Michael Perzl, BULL freeware) or installp (AIXTOOLS) packages. These only install to the SYSTEM area however - unsuited for a virtualenv.

Coming UP! - Part3

PIP, wheels, and installation (as non-root!)