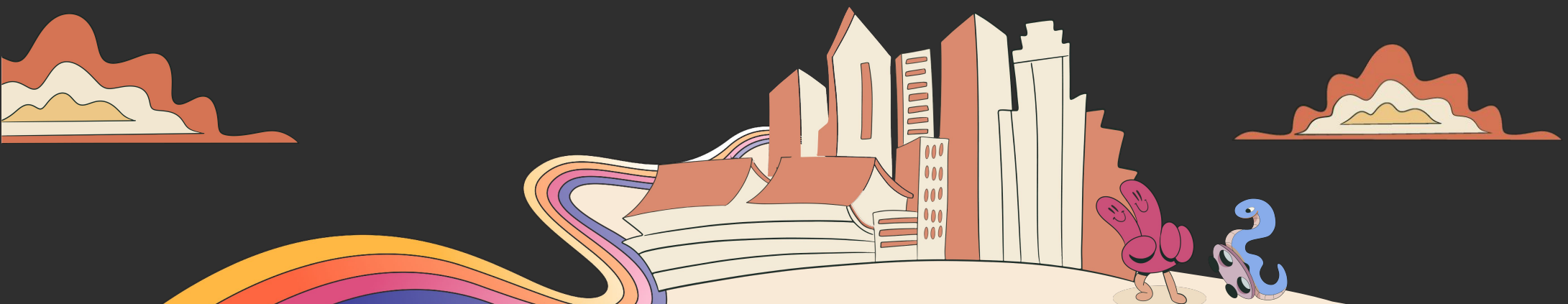

PyConUS
2025
Pittsburgh

Future PEP: Wheel Variants

Finer grained operational environment compatibility

Jonathan Dekhtiar

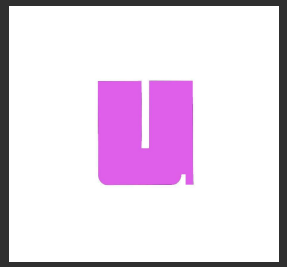
PyCon 2025 - Packaging Summit



PyConUS 2025

1 | WheelNext

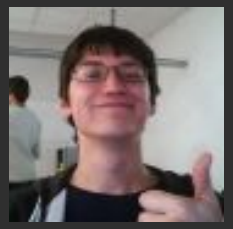
WheelNext - Who are we ?



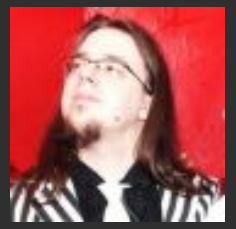
PyConUS 2025

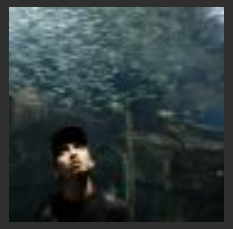
The work of so many

Alphabetic Order





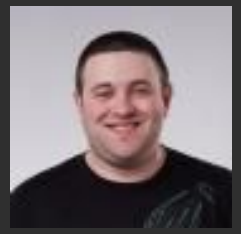



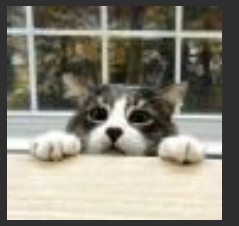


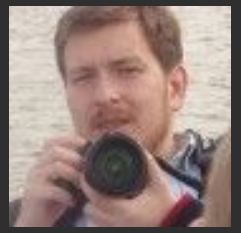


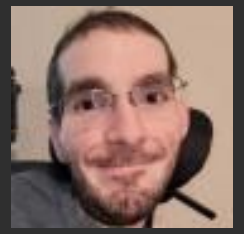






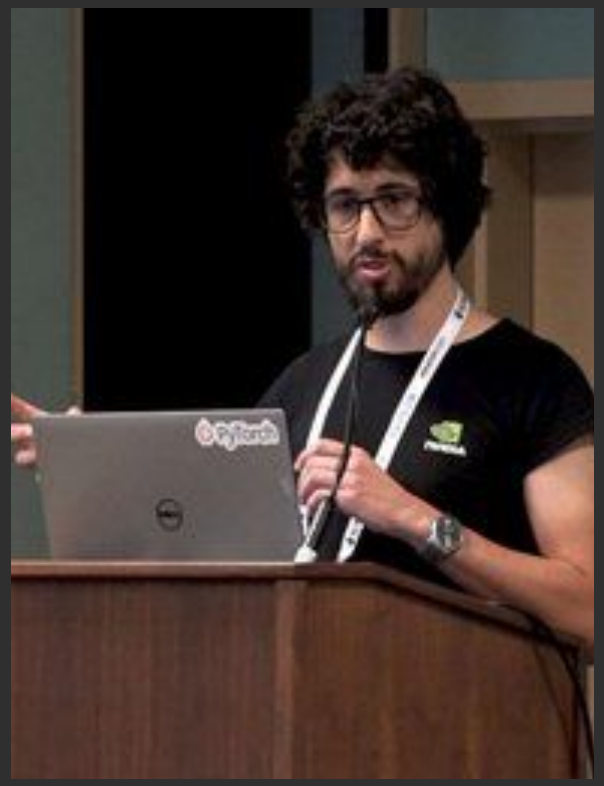





PyConUS 2025

Key Contributors

VariantLib



Michał Górny - Quansight

Jonathan Dekhtiar - NVIDIA

PyConUS 2025

2 | Wheel Variants

Problem: Specialized "Hardware"

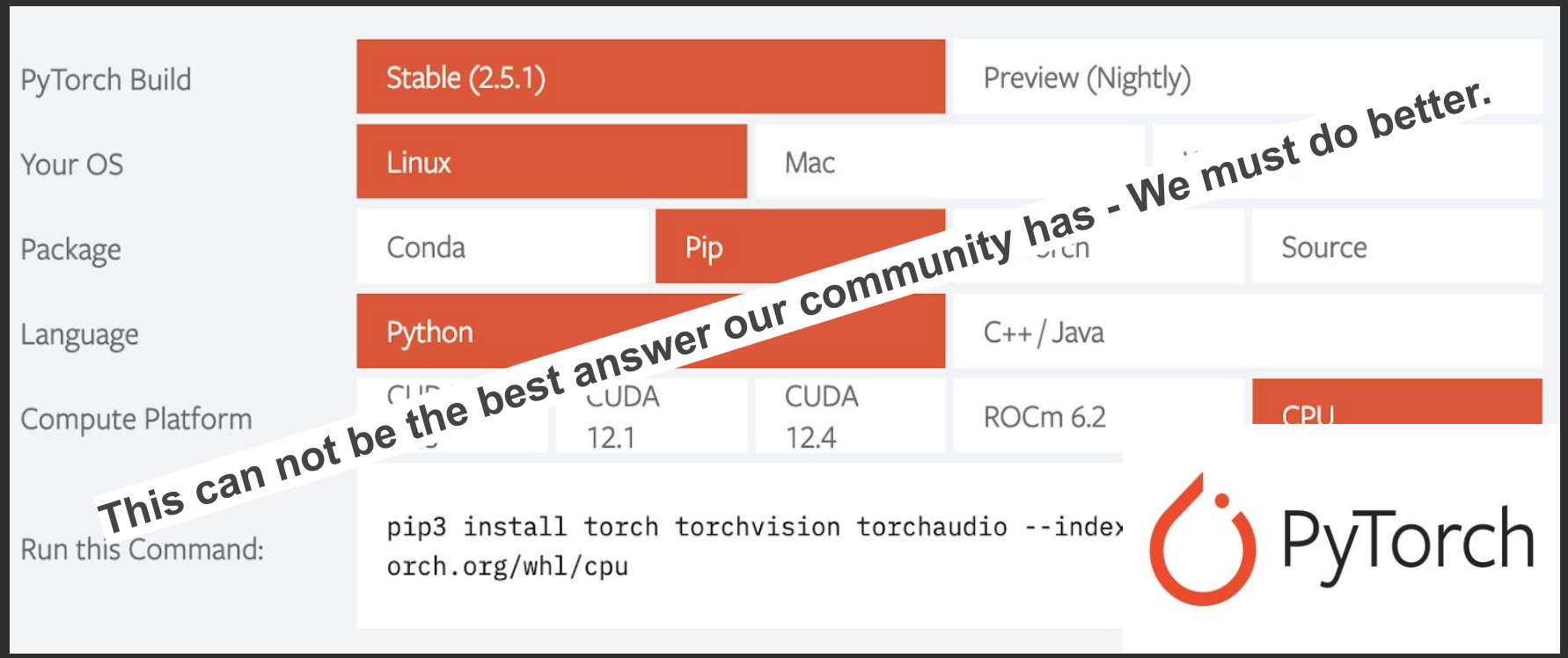
Going beyond Python version + ABI + Platform

- No way to more finely describe the operating environment
  - What type of hardware do you have *(e.g. GPU, FPGA, ASIC, etc.) ?*
  - What x86-64 / ARM version (e.g. x86-64v3, ARMv7, ARMv8, etc.)?
  - What special instruction sets (e.g. AVX512)?
  - Specialized libraries (BLAS, MPI, etc.)

PyConUS 2025

Problem: Specialized "Hardware"

Going beyond Python version + ABI + Platform



This can not be the best answer our community has - We must do better.

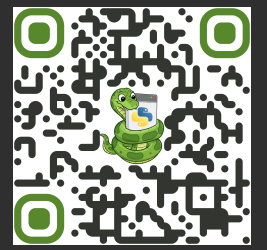
PyConUS 2025

Problem: Specialized "Hardware"

Going beyond Python version + ABI + Platform

	Linux, x86_64	Linux, aarch64	Mac, x86_64	Mac, aarch64	Windows, x86_64	Windows WSL2, x86_64
CPU	yes	yes	jax≤0.4.38 only	yes	yes	yes
NVIDIA GPU	yes	yes	no	n/a		experimental
Google Cloud TPU	yes	n/a		n/a	n/a	n/a
AMD GPU		no	experimental	n/a	no	no
GPU	n/a	no	n/a	experimental	n/a	
Intel GPU	experimental	n/a	n/a	n/a	no	

This can not be the best answer our community has - We must do better.

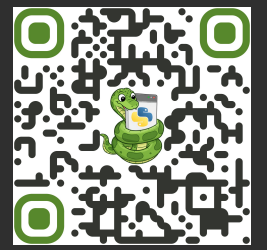


PyConUS 2025

User Rationale

Going beyond Python version + ABI + Platform

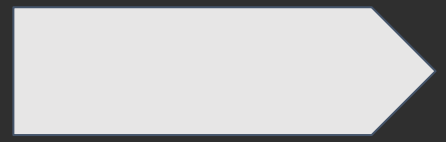
- A user wants NumPy specialized for their CPU architecture.
- A user wants PyTorch / JAX / vLLM accelerated for their GPU architecture.
- A user wants mpi4py built for their MPI implementation (OpenMPI, MPICH, etc.)
- A user wants SciPy built against different BLAS libraries (OpenBLAS, MKL, etc.)
- Manylinux cannot express x86-64-v2 requirements in Manylinux_2_34



PyConUS 2025

What is “the (future) Wheel Variant PEP” ?

- A **descriptive language** of externally-defined descriptors
  - “x86_64 :: level :: v4”
  - “aarch64 :: version :: 8.4a”
  - “nvidia :: cuda :: 12.8”



Variant Properties

- Standardized Plugin Interface designed around `typing.Protocol` :
  - we enforce an API - not a type / inheritance / language
- Variant Resolver comes after “classic version resolver”
  - Variant Resolver = filtering + sorting - [platform specific]
  - +/- 20ms for complete resolution with over 100 variants

PyConUS 2025

3 | Let’s get “hands on”

Proposed: Future PEP

Wheel Variants

https://variants-demo.wheelnext.dev

PyConUS 2025

Proposed: Future PEP

Wheel Variants

```
# pyproject.toml - torch

[variant.default-priorities]
# the plugins corresponding to these namespaces will be auto-installed at installation
namespace = ["nvidia", "x86_64"]     # optional - if empty nothing is auto-installed
feature = []                         # optional - 98%+ packages won't ever need
property = []                        # optional - 98%+ packages won't ever need


[variant.providers.nvidia]
requires = ["nvidia-variant-provider == 1.0.0"]
plugin-api = "nvidia_variant_provider.plugin:NvidiaVariantPlugin"

[variant.providers.x86_64]
requires = ["provider-variant-x86-64 == 1.0.0"]
plugin-api = "provider_variant_x86_64.plugin:X8664Plugin"
```

Proposed: Future PEP

Wheel Variants

.dist-info/METADATA

```
Metadata-Version: 2.4
Name: torch
Version: 1.0.0

Variant-hash: 9240ade1

Variant-property: nvidia :: cuda :: 12.8
Variant-property: nvidia :: sm_arch :: 90
Variant-property: x86_64 :: level :: 4

Variant-requires: nvidia: nvidia-variant-provider == 1.0.0
Variant-plugin-api: nvidia: nvidia_variant_provider.plugin:NvidiaVariantPlugin

Variant-requires: x86_64: provider-variant-x86-64 == 1.0.0
Variant-plugin-api: x86_64: provider_variant_x86_64.plugin:X8664Plugin

Variant-default-namespace-priorities: nvidia, x86_64
```

Proposed: Future PEP

Wheel Variants

```
Metadata-Version: 2.4
Name: torch
Version: 1.0.0

Variant-hash: 9240ade1

Variant-property: nvidia :: cuda :: 12.8
Variant-property: nvidia :: sm_arch :: 90
Variant-property: x86_64 :: level :: 4

Variant-requires: nvidia: nvidia-variant-provider == 1.0.0
Variant-plugin-api: nvidia: nvidia_variant_provider.plugin:NvidiaVariantPlugin

Variant-requires: x86_64: provider-variant-x86-64 == 1.0.0
Variant-plugin-api: x86_64: provider_variant_x86_64.plugin:X8664Plugin

Variant-default-namespace-priorities: nvidia, x86_64
```

torch-2.7.0-cp310-cp310-manylinux_2_28_x86_64-9240ade1.whl

Proposed: Future PEP

Wheel Variants

```
Metadata-Version: 2.4
Name: torch
Version: 1.0.0

Variant-hash: 9240ade1

Variant-property: nvidia :: cuda :: 12.8
Variant-property: nvidia :: sm_arch :: 90
Variant-property: x86_64 :: level :: 4

Variant-requires: nvidia: nvidia-variant-provider == 1.0.0
Variant-plugin-api: nvidia: nvidia_variant_provider.plugin:NvidiaVariantPlugin

Variant-requires: x86_64: provider-variant-x86-64 == 1.0.0
Variant-plugin-api: x86_64: provider_variant_x86_64.plugin:X8664Plugin

Variant-default-namespace-priorities: nvidia, x86_64
```

Proposed: Future PEP

Wheel Variants

```
Metadata-Version: 2.4
Name: torch
[variant.providers.nvidia]
requires = ["nvidia-variant-provider == 1.0.0"]
plugin-api = "nvidia_variant_provider.plugin:NvidiaVariantPlugin"

[variant.providers.x86_64]
requires = ["provider-variant-x86-64 == 1.0.0"]
plugin-api = "provider_variant_x86_64.plugin:X8664Plugin"

Variant-requires: nvidia: nvidia-variant-provider == 1.0.0
Variant-plugin-api: nvidia: nvidia_variant_provider.plugin:NvidiaVariantPlugin

Variant-requires: x86_64: provider-variant-x86-64 == 1.0.0
Variant-plugin-api: x86_64: provider_variant_x86_64.plugin:X8664Plugin

Variant-default-namespace-priorities: nvidia, x86_64
```

Proposed: Future PEP

Wheel Variants

```
Metadata-Version: 2.4
Name: torch
Version: 1.0.0

Variant-hash: 9240ade1

Variant-property: nvidia :: cuda :: 12.8
Variant-property: nvidia :: sm_arch :: 90
Variant-property: x86_64 :: level :: 4
```

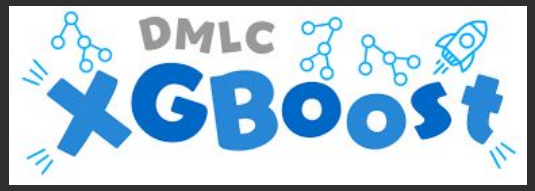
```
[variant.default-priorities]
# the plugins corresponding to these namespaces will be auto-installed at installation
namespace = ["nvidia", "x86_64"]     # optional - if empty nothing is auto-installed
feature = []                         # optional - 98%+ packages won't ever need
property = []                        # optional - 98%+ packages won't ever need
```

```
Variant-default-namespace-priorities: nvidia, x86_64
```

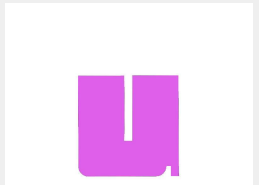
PyConUS 2025

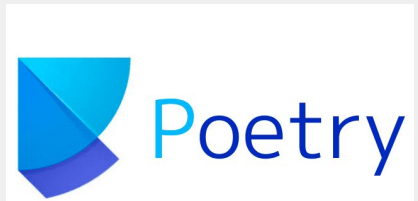
Proposed: Future PEP

Wheel Variants



(Hopefully) Very Soon - Work in Progress

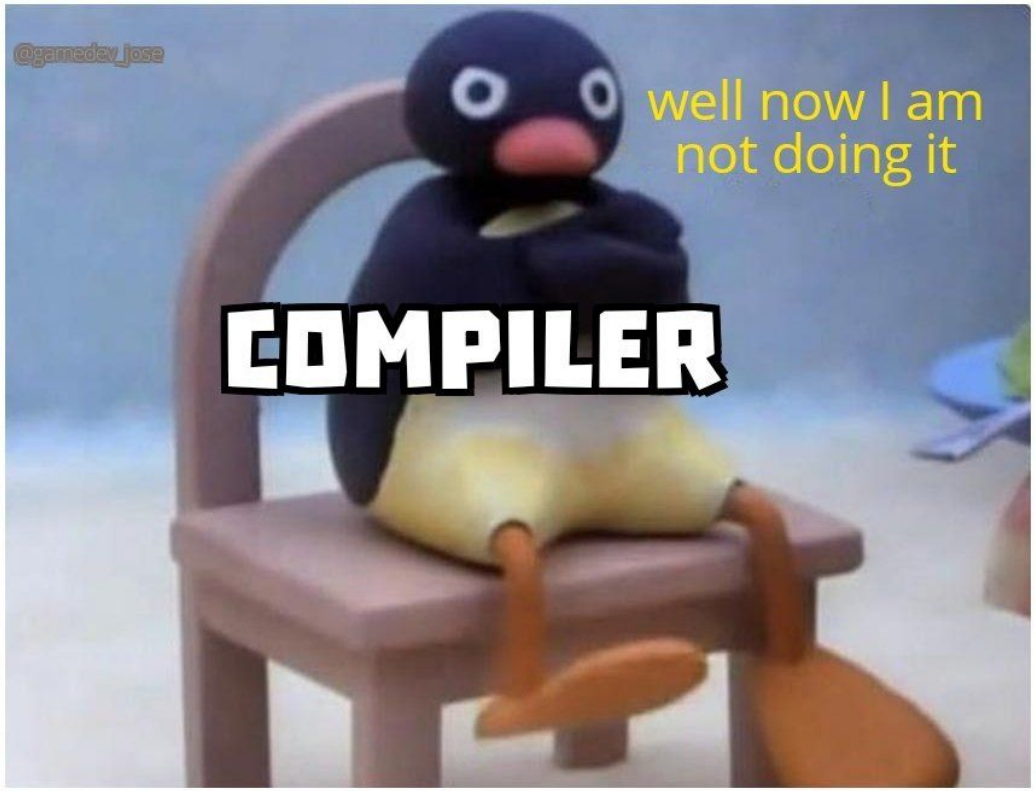




PyConUS 2025

The package “build story”
Package Maintainer - Experience

* Build successful *
* Imports package *



PyConUS 2025

Proposed: Future PEP

Wheel Variants

```
# pyproject.toml - torch

[variant.default-priorities]
# the plugins corresponding to these namespaces will be auto-installed at installation
namespace = ["nvidia", "x86_64"]     # optional - if empty nothing is auto-installed
feature = []                         # optional - 98%+ packages won't ever need
property = []                        # optional - 98%+ packages won't ever need


[variant.providers.nvidia]
requires = ["nvidia-variant-provider == 1.0.0"]
plugin-api = "nvidia_variant_provider.plugin:NvidiaVariantPlugin"

[variant.providers.x86_64]
requires = ["provider-variant-x86-64 == 1.0.0"]
plugin-api = "provider_variant_x86_64.plugin:X8664Plugin"
```

Proposed: Future PEP

Wheel Variants


```
# pyproject.toml

[build-system]
requires = ['flit_core >=3,<4']
build-backend = 'flit_core.buildapi'
requires = [
    "flit-core @ https://github.com/wheelnext/flit/archive/wheel_variants.tar.gz",
]
```

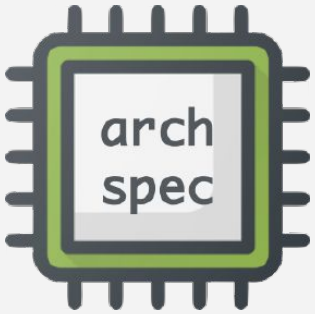
```
$ flit build --format wheel \
        -p "nvidia :: cuda :: 12.8" \
        -p "nvidia :: sm_arch :: 90" \
        -p "x86_64 :: level :: 4"
```

Proposed: Future PEP

Wheel Variants

```toml
# pyproject.toml

[build-system]
build-backend = "mesonpy"
requires = [
    "meson-python @ github.com/wheelnext/meson-python/archive/wheel-variants.tar.gz",
]
```

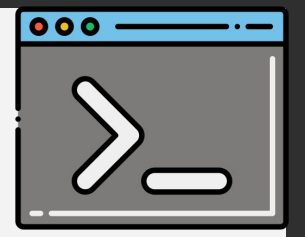
```
python3 -m build -w -Cvariant-name=x86_64::level::v3
```



Proposed: Future PEP

Wheel Variants

```
variantlib make-variant \
    -f dist/torch-2.7.0-cp310-cp310-manylinux_2_28_x86_64.whl \
    -o dist/ \
    --pyproject-toml pyproject.toml \
    -p "nvidia :: cuda :: 12.8" \
    -p "nvidia :: sm_arch :: 90" \
    -p "x86_64 :: level :: 4"

INFO - Loading plugin via nvidia_variant_provider.plugin:NvidiaVariantPlugin
INFO - Loading plugin via provider_variant_x86_64.plugin:X8664Plugin
INFO - Variant Wheel Created: `dist/torch-2.7.0-cp310-cp310-manylinux_2_28_x86_64-9240ade1.whl`
```



The User “install story”

User - Experience

PyConUS 2025

Proposed: Future PEP

Wheel Variants

```
$ pip install numpy

Installing variant-provider-plugins in current environment:
  - provider-variant-aarch64 == 0.0.1;

  Variant `09300f2f` rejected `[aarch64 :: version :: 8.4a]` is not supported.
  Variant `c87a4099` rejected `[aarch64 :: version :: 8.5a]` is not supported.

  Total Number of Compatible Variants: 4

######################### Selected Variant: `522ebbc7` #########################
Variant-property: aarch64 :: version :: 8.3a
################################################################################

Collecting numpy
  numpy-2.2.5-cp312-cp312-macosx_14_0_arm64-522ebbc7.whl (5.1 MB)

Installing collected packages: numpy
Successfully installed numpy-2.2.5-522ebbc7
```

ARM v8.3a

NumPy

PyConUS 2025

Proposed: Future PEP

Wheel Variants

```
$ pip install numpy

Installing variant-provider-plugins in current environment:
  - provider-variant-aarch64 == 0.0.1;

  Variant `09300f2f` rejected `[aarch64 :: version :: 8.4a]` is not supported.
  Variant `c87a4099` rejected `[aarch64 :: version :: 8.5a]` is not supported.

  Total Number of Compatible Variants: 4

######################### Selected Variant: `522ebbc7` #########################
Variant-property: aarch64 :: version :: 8.3a
################################################################################

Collecting numpy
  numpy-2.2.5-cp312-cp312-macosx_14_0_arm64-522ebbc7.whl (5.1 MB)

Installing collected packages: numpy
Successfully installed numpy-2.2.5-522ebbc7
```

ARM v8.3a

NumPy

Proposed: Future PEP

Wheel Variants

```
$ pip install numpy

Installing variant-provider-plugins in current environment:
  - provider-variant-aarch64 == 0.0.1;

  Variant `09300f2f` rejected `[aarch64 :: version :: 8.4a]` is not supported.
  Variant `c87a4099` rejected `[aarch64 :: version :: 8.5a]` is not supported.

  Total Number of Compatible Variants: 4

######################### Selected Variant: `522ebbc7` ########################
Variant-property: aarch64 :: version :: 8.3a
###############################################################################

Collecting numpy
  numpy-2.2.5-cp312-cp312-macosx_14_0_arm64-522ebbc7.whl (5.1 MB)

Installing collected packages: numpy
Successfully installed numpy-2.2.5-522ebbc7
```

ARM v8.3a

NumPy

Proposed: Future PEP

Wheel Variants

```
$ pip install numpy

Installing variant-provider-plugins in current environment:
  - provider-variant-aarch64 == 0.0.1;

  Variant `09300f2f` rejected `[aarch64 :: version :: 8.4a]` is not supported.
  Variant `c87a4099` rejected `[aarch64 :: version :: 8.5a]` is not supported.

  Total Number of Compatible Variants: 4

######################### Selected Variant: `522ebbc7` ##########################
Variant-property: aarch64 :: version :: 8.3a
#################################################################################

Collecting numpy
  numpy-2.2.5-cp312-cp312-macosx_14_0_arm64-522ebbc7.whl (5.1 MB)

Installing collected packages: numpy
Successfully installed numpy-2.2.5-522ebbc7
```

ARM v8.3a

NumPy

PyConUS 2025

Proposed: Future PEP

Wheel Variants

```
$ pip install numpy

Installing variant-provider-plugins in current environment:
  - provider-variant-x86-64 == 0.0.1;

  Variant `fa7c1393` rejected `[x86_64 :: level :: v3]` is not supported.
  Variant `cfdbe307` rejected `[x86_64 :: level :: v4]` is not supported.

  Total Number of Compatible Variants: 2

######################### Selected Variant: `40aba78e` #########################
Variant-property: x86_64 :: level :: v2
################################################################################

Collecting numpy
  numpy-2.2.5-cp312-cp312-linux_x86_64-40aba78e.whl (17.6 MB)

Installing collected packages: numpy
Successfully installed numpy-2.2.5-40aba78e
```

X86-64 v2

NumPy

PyConUS 2025

Proposed: Future PEP

Wheel Variants

```
$ pip install torch

Installing variant-provider-plugins in current environment:
  - nvidia-variant-provider == 0.0.1;

  Variant `1065b45d` rejected `[nvidia :: cuda :: 11.8]` is not supported.

  Total Number of Compatible Variants: 3

######################### Selected Variant: `d5784ad6` ##########################
Variant-property: nvidia :: cuda :: 12.8
#################################################################################

Collecting torch
  torch-2.7.0-cp312-cp312-manylinux_2_28_x86_64-d5784ad6.whl (1096.4 MB)

Installing collected packages: torch
Successfully installed torch-2.2.5-d5784ad6
```

NVIDIA CUDA

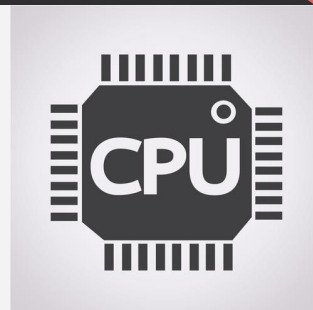
CUDA 12.8

PyTorch

Proposed: Future PEP

Wheel Variants

```
$ pip install torch

Installing variant-provider-plugins in current environment:
  - nvidia-variant-provider == 0.0.1;

  Variant `1065b45d` rejected `[nvidia :: cuda :: 11.8]` is not supported.
  Variant `43331073` rejected `[nvidia :: cuda :: 12.6]` is not supported.
  Variant `d5784ad6` rejected `[nvidia :: cuda :: 12.8]` is not supported.

  Total Number of Compatible Variants: 0

Collecting torch
  torch-2.7.0-cp312-cp312-manylinux_2_28_x86_64.whl (175.4 MB)

Installing collected packages: torch
Successfully installed torch-2.2.5
```



CPU “only”

“No variant found”

PyTorch

Proposed: Future PEP

Wheel Variants

```
$ pip install --no-variant torch

Collecting torch
  torch-2.7.0-cp312-cp312-manylinux_2_28_x86_64.whl (175.4 MB)

Installing collected packages: torch
Successfully installed torch-2.2.5
```

Proposed: Future PEP

Wheel Variants

```
$ pip install numpy==2.2.5

Installing variant-provider-plugins in current environment:
  - provider-variant-aarch64 == 0.0.1;

  Variant `09300f2f` rejected `[aarch64 :: version :: 8.4a]` is not supported.
  Variant `c87a4099` rejected `[aarch64 :: version :: 8.5a]` is not supported.

  Total Number of Compatible Variants: 4

######################### Selected Variant: `522ebbc7` ##########################
Variant-property: aarch64 :: version :: 8.3a
#################################################################################

Collecting numpy
  numpy-2.2.5-cp312-cp312-macosx_14_0_arm64-522ebbc7.whl (5.1 MB)

Installing collected packages: numpy
Successfully installed numpy-2.2.5-522ebbc7
```

ARM v8.3a

NumPy

PyConUS 2025

Proposed: Future PEP

Wheel Variants

```
$ pip install numpy==2.2.5#802e12ea
Looking in indexes: https://variants-index.wheelnext.dev/

Forced Variant selection: 802e12ea
Fetching https://variants-index.wheelnext.dev/numpy/numpy-2.2.5-variants.json

#################### numpy==2.2.5; variant_hash=802e12ea ####################
Variant-property: aarch64 :: version :: 8.1a
#############################################################################

Collecting numpy==2.2.5
     numpy-2.2.5-cp312-cp312-macosx_14_0_arm64-802e12ea.whl (5.0 MB)
Successfully installed numpy-2.2.5-802e12ea
```

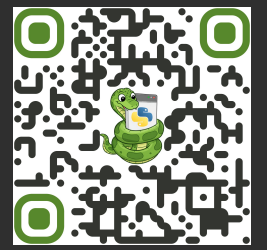
ARM v8.3a

NumPy

4 | Call to Action

Join us!

- Let's hear from you
  - wheelnext.dev & GitHub (Use the QR code)
- PyCon 2025
  - Sprints
  - WheelNext Open Space
  - Language/Packaging Summits
- Try `variants-demo.wheelnext.dev`



PyConUS 2025

WheelNext Resources

https://contribute.wheelnext.dev

https://github.com/wheelnext

https://mailing.wheelnext.dev

PyConUS 2025