Package Managers and Virtual Environments

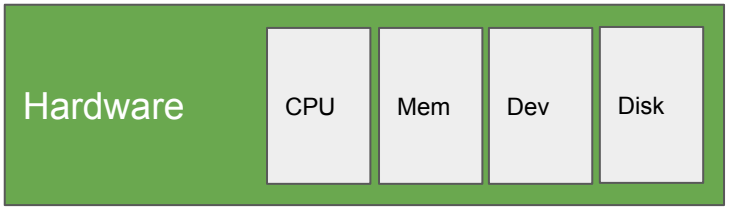
Layers in a computer

Layers in a computer

A working computer is a union between hardware and software. A computer is composed of three different layers.

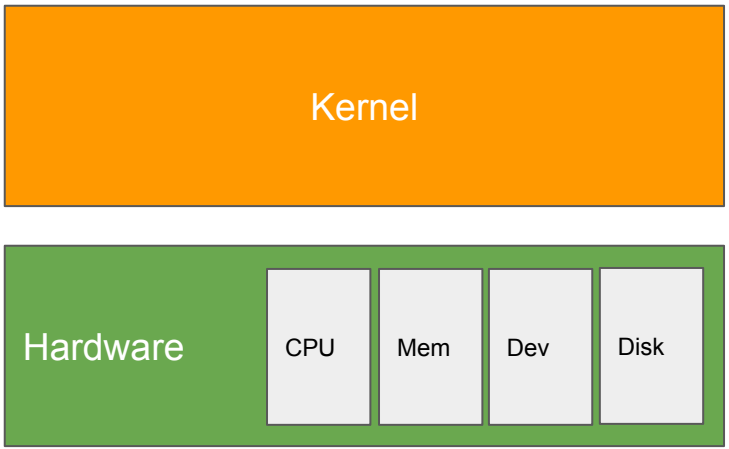
Layers in a computer

A working computer is a union between hardware and software. A computer is composed of three different layers. The hardware,



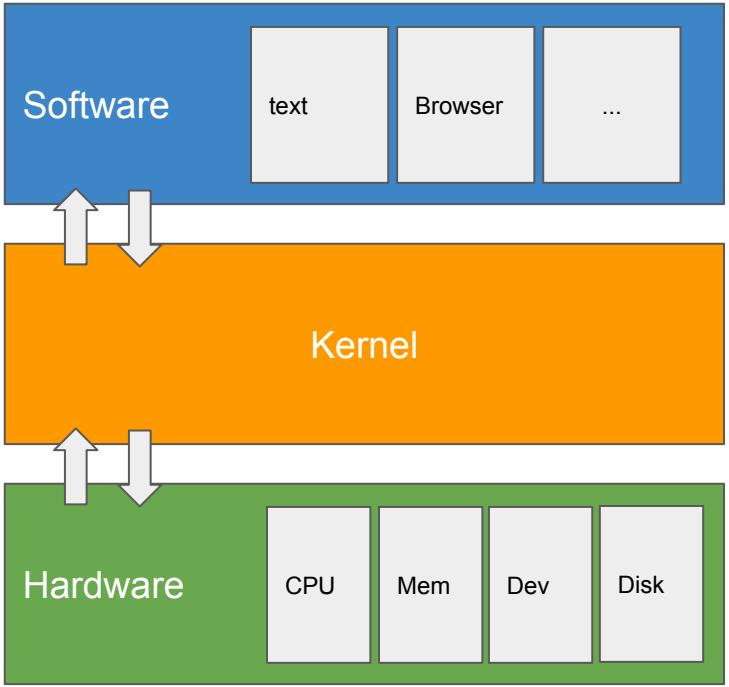
Layers in a computer

A working computer is a union between hardware and software. A computer is composed of three different layers. The hardware, the kernel



Layers in a computer

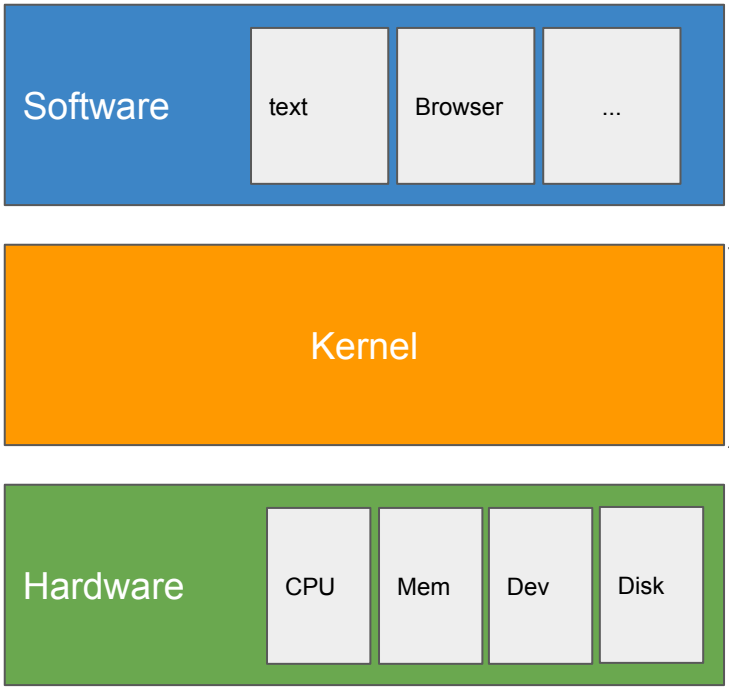
A working computer is a union between hardware and software. A computer is composed of three different layers. The hardware, the kernel, and the software.



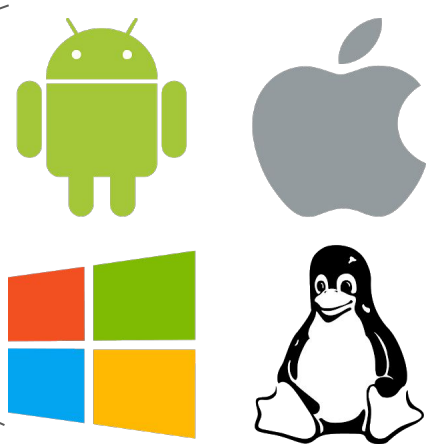
The kernel is the bridge between hardware and software. It is a translator between the layers.

Layers in a computer

A working computer is a union between hardware and software. A computer is composed of three different layers. The hardware, the kernel, and the software.

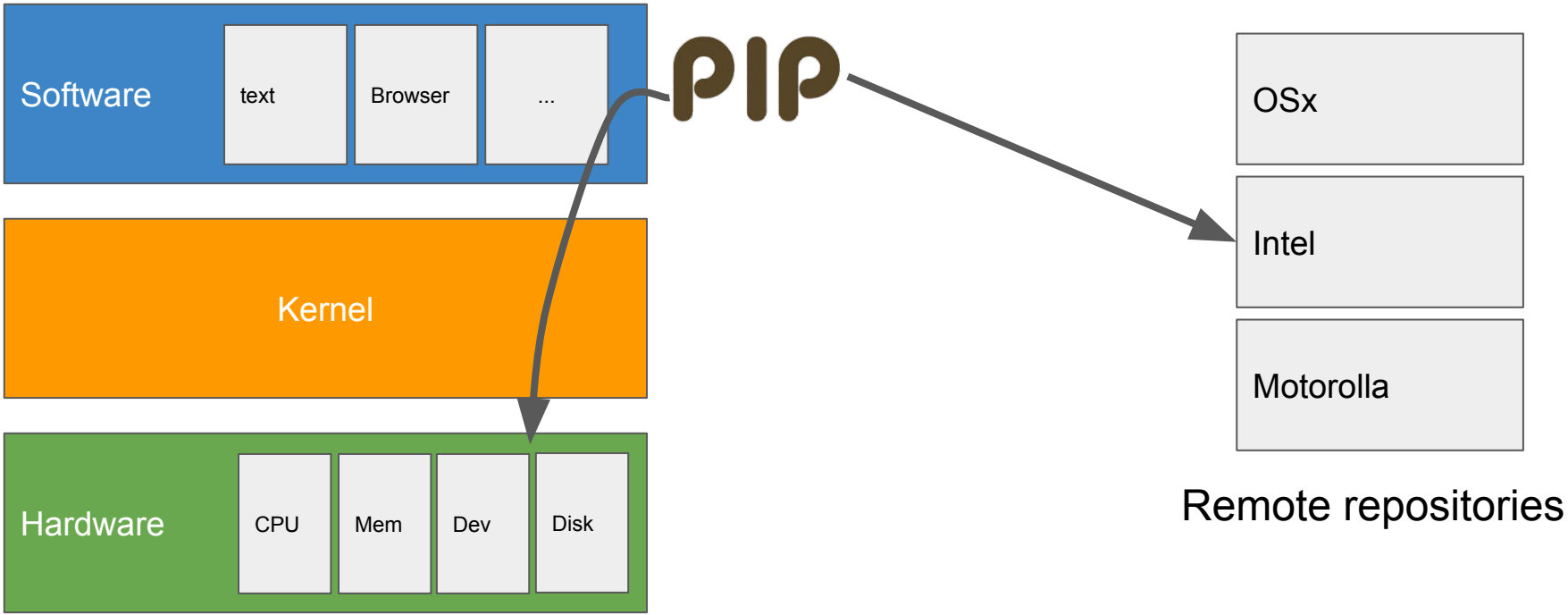


Different OS have different kernels.



Layers in a computer

A package manager knows where to get the appropriate pre-compiled software packages for your machine



pip + virtualenv

This is your software layer

- Browser
- Text editor
- Chat programme
- ...
- Python installation

This area is all you have on disk

This is your software layer

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.1
  - Pandas 0.25
  - ...

Your baseline python installation may come with some packages that may be obsolete

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.1
  - Pandas 0.25
  - ...

> pip install -U pandas

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.2
  - Pandas 1.0
  - ...

If you upgrade package **P** to version **X**, all dependencies of version **X** are also upgraded to their corresponding compatible version.
Pandas 0.25 into 1.0
Numpy 1.1 into 1.2
(this is an example)

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.1.6
  - Pandas 0.25
  - ...

> pip install pandas==0.25

If you downgrade a single package, it does not mean other packages also return to the original status.
Instead, pip will see what is the latest version still compatible with what you want to achieve.

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.1
  - Pandas 0.25
  - ...

Let’s assume you want to safely try out a new package or an upgrade of a package you already have

Notice:

We returned to the original status for the purposes of the demo

- Browser
- Text editor
- Chat programme
- ...
- Python installation
    - Numpy 1.1
    - Pandas 0.25
    - ...

Win:
> python -m venv testenv
Mac + Linux:
> python3 -m venv testenv

<u>Important:</u>

Although pip manages the packages, it is another programme, **virtualenv**, that creates virtual environments. Presently, pip and virtualenv are disconnected entities.

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.1
  - Pandas 0.25
  - ...

testenv

A directory is created with the name of the virtual environment you chose. This is just a directory, but holds everything a python distribution has.

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.1
  - Pandas 0.25
  - ...

Activate the environment
Win:
> .\testenv\Scripts\activate
Mac + Linux:
> source ./testenv/bin/activate

testenv

- Browser
- Text editor
- Chat programme
- ...

testenv

testenv

Your OS, at the moment, sees
“testenv” as your python distribution

- Browser
- Text editor
- Chat programme
- ...

testenv

testenv

You can keep installing and testing packages in your virtual environment. If you break something, **this is just a directory**. You can just deactivate the virtual environment and delete the directory.

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.1
  - Pandas 0.25
  - ...

Deactivate the environment and return to your standard install
> deactivate

testenv

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.1
  - Pandas 0.25
  - ...

Your can create many virtual environments. As a best practice for developing software, you never perform dangerous operations in your base VE. You create a new VE and work from there.

testenv

Python Virtual environment 1

Python Virtual environment 2

(pip+virtualenv) VS conda

- Browser
- Text editor
- Chat programme
- ...
- Python installation
  - Numpy 1.1
  - Pandas 0.25
  - ...

When you install conda, almost all settings of the conda installation supercede you native python installation.
You go from this setting...

testenv

Python Virtual environment 1

Python Virtual environment 2

- Browser
- Text editor
- Chat programme
- ...

testenv

conda

To something more like this.

- Browser
- Text editor
- Chat programme
- ...

testenv

conda

It is still possible to access your native python installation. However, as soon as you run the **navigator** or the **prompt**, conda takes over.

- Browser
- Text editor
- Chat programme
- ...

testenv

conda

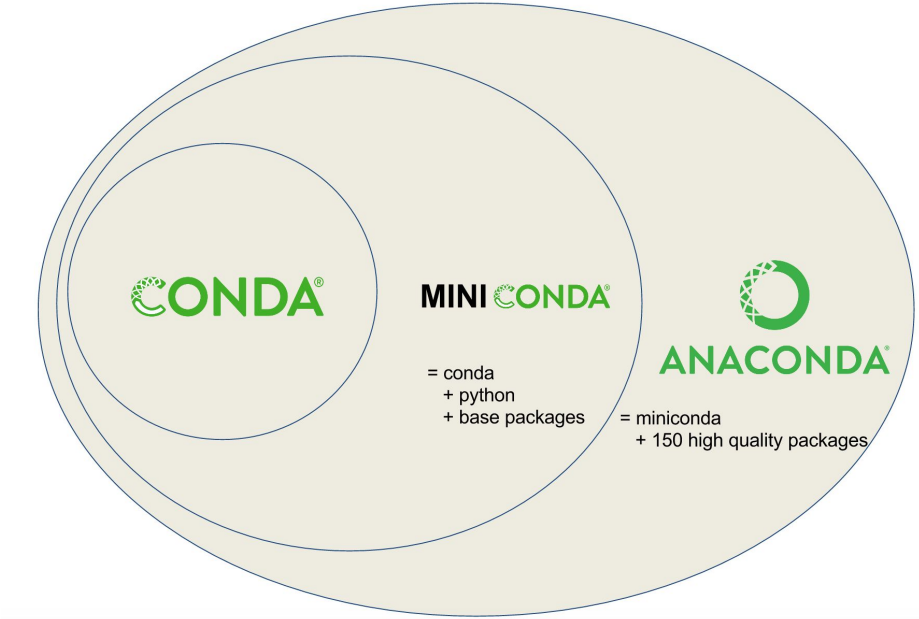
Virtual environment 1

Virtual environment 2

conda has many features for package management like pip, and also the ability to create **virtual environments.**

Conda naming: why the variants

With standard python management tools, you have separate programmes to manage packages and virtual environments.



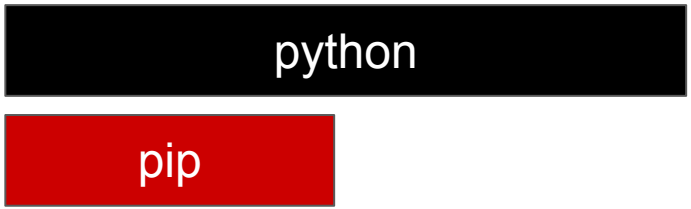
- **Conda** is the executable that manages packages and virtual environments.
- **Miniconda** is conda distributed with the bare minimum of packages you need to start your work.
- The **Anaconda distribution** is a super sized package with a GUI, the navigator.

(pip+virtualenv) VS conda

With standard python management tools, you have separate programmes to manage packages and virtual environments.

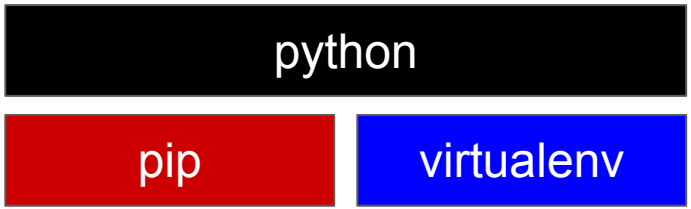
(pip+virtualenv) VS conda

With standard python management tools, you have separate programmes to manage packages and virtual environments.
You have pip to manage packages



(pip+virtualenv) VS conda

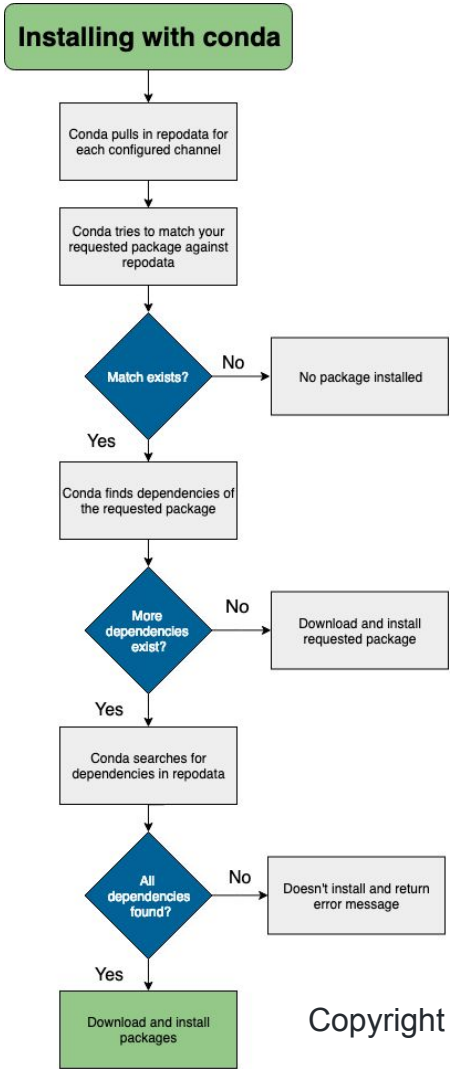
With standard python management tools, you have separate programmes to manage packages and virtual environments.
You have pip to manage packages and virtualenv to create virtual environments.



(pip+virtualenv) VS conda

With standard python management tools, you have separate programmes to manage packages and virtual environments.
You have pip to manage packages and virtualenv to create virtual environments.

With conda, you do both the management and the environment creation.



Package checking for speed

- Conda has a local cache in your machine.
- Everytime you ask to install something, conda will check if the dependencies are already local.
- If they are, conda uses the local packages.
- If not, conda downloads the newest version and adds it to your local cache.

End