



Data Scientists



ANACONDA.



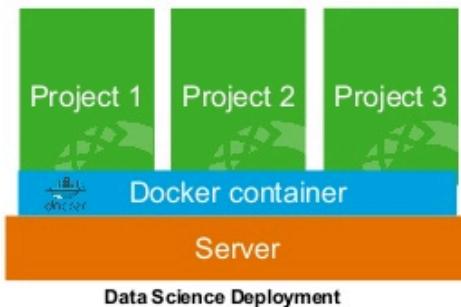
Data Science Development



DevOps



docker



DigitalHouse >
Coding School

DATA SCIENCE

MÓDULO 1

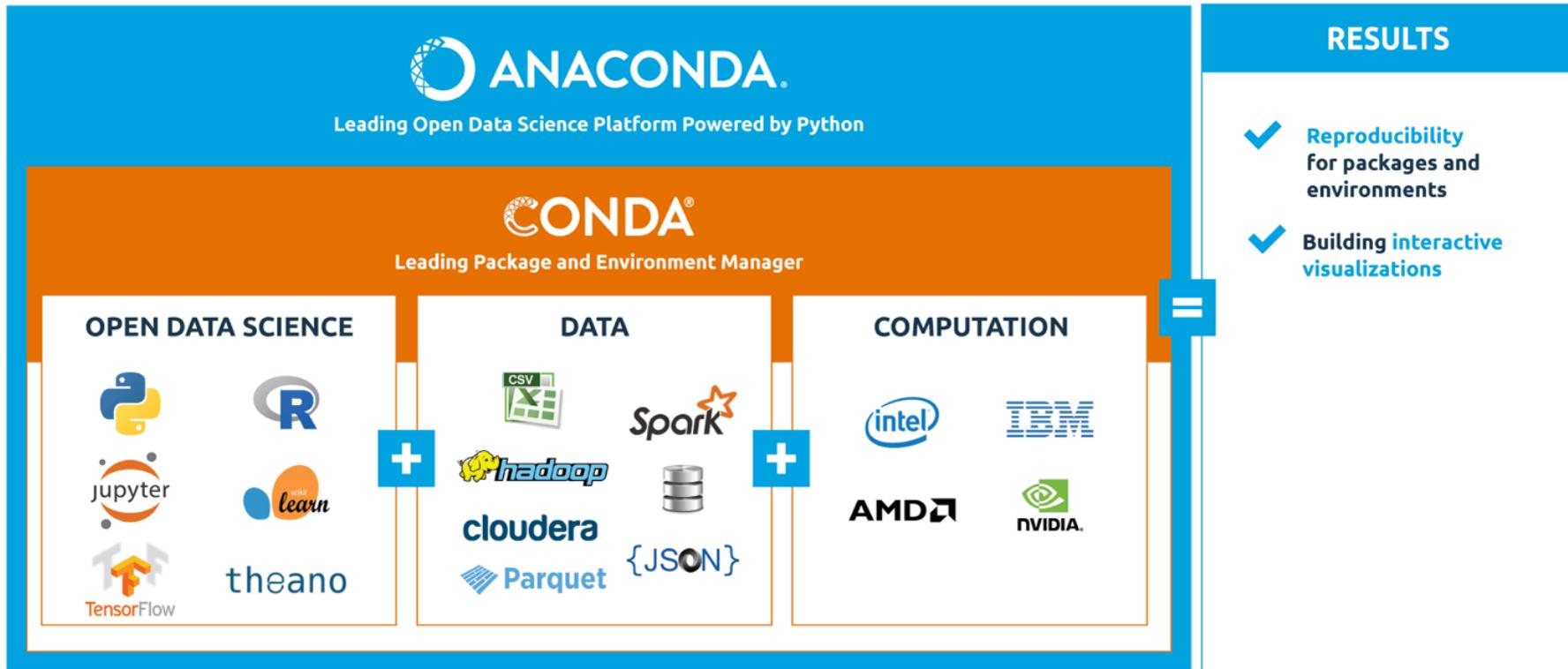
Entornos de trabajo a nivel usuario

- 1 Introducir nociones de Anaconda a nivel usuario.
- 2 Introducir Colaboratory nivel usuario.
- 3 Introducir nociones de Docker a nivel usuario.

ANACONDA



- **Anaconda** es una distribución gratuita y open source de Python y R para data science y machine learning.
- Dispone de un administrador de entorno y de paquetes (librerías) **Conda**.
- Cuenta con un entorno de desarrollo integrado para Python (IDE): **Spyder**.
- Es posible ejecutar notebooks interactivas con:
 - **Jupyter Notebooks**
 - **Jupyter Lab**
- Una interfaz interactiva para graficar el flujo de trabajo: **Orange**.



Anaconda: Spyder

The screenshot shows the Anaconda Spyder IDE interface. On the left, the code editor displays Python code for interpolation and Monte Carlo simulation. The variable explorer shows variables e and pi. The console window shows IPython 0.10.1 running and a welcome message for pylab. The object inspector provides details about the array variable.

Editor - C:\Documents and Settings\carlos\Mis documentos\Python\Interpolation.py

```
1 """
2 Interpolation of an II-D curve
3 From the SciPy Cookbook
4 """
5
6 from numpy import arange, cos, linspace, pi, sin, random
7 from scipy.interpolate import splprep, splev
8
9 # make ascending spiral in 3-space
10 t=linspace(0,1.75*2*pi,100)
11
12 x = sin(t)
13 y = cos(t)
14 z = t
15
```

Variable explorer

Name	Type	Size	Value
e	float	1	2.7182818284590451
pi	float	1	3.1415926535897931

Object inspector

Source Console Object array Options

array(...)
Function of numpy.core.multiarray module

array(object, dtype=None, copy=True, order=None, subok=False, ndmin=0)

Create an array.

Parameters

object: array_like
An array, any object exposing the array interface, an object whose `__array__` method returns an array, or any (nested) sequence.

dtype: data-type, optional
The desired data-type for the array. If not given, then the type will be determined as the minimum type required to hold

In [1]:

Permissions: RW | End-of-lines: LF | Encoding: UTF-8-GUESSED | Line: 7 | Column: 1

jupyter convnets Last Checkpoint: 08/29/2017 (unsaved changes) Logout

File Edit View Insert Cell Kernel Help

Code CellToolbar

Looking for Collisions in Training, Test and Validation Sets

```
In [47]: train_hashes = set(hash(i.tostring()) for i in train_dataset)
test_hashes = set(hash(i.tostring()) for i in test_dataset)
valid_hashes = set(hash(i.tostring()) for i in valid_dataset)
```

```
In [66]: with Timer('28x 28 Pixel Photo Set Sizes {} {} {}'.format(
    len(train_dataset), len(test_dataset), len(valid_dataset))):
    train_hashes = set(hash(i.tostring()) for i in train_dataset)
    test_hashes = set(hash(i.tostring()) for i in test_dataset)
    valid_hashes = set(hash(i.tostring()) for i in valid_dataset)

    result = np.ndarray(shape=(3, 3))
    hashes = [train_hashes, test_hashes, valid_hashes]

    for i, first_hashes in enumerate(hashes):
        for j, next_hashes in enumerate(hashes):
            result[i, j] = 100.0 * len(first_hashes.intersection(next_hashes)) / len(first_hashes)

    ax = sns.heatmap(result, annot=True)
    plt.title('Percentage overlap between sets (with floating point error)')
    ax.set_yticklabels(['Train', 'Test', 'Valid'])
    ax.set_xticklabels(['Train', 'Test', 'Valid'])
    plt.show();
```

Percentage overlap between sets (with floating point error)

	Train	Test	Valid
Train	9.7	0.56	1e+02
Test	12	0.62	0.51
Valid	1e+02	0.56	0.01

28x 28 Pixel Photo Set Sizes 200000 10000 10000 taken 0.89 secs

```
In [25]: import time

def softmax(x):
    return nn.exp(x) / nn.sum(nn.exp(x), axis=0)
```

Anaconda: Jupyter Lab

File Notebook Editor Terminal Console Help

Files

Name Last Modified

- data_clean 5 months ago
- data_raw 5 months ago
- img 2 months ago
- plots 5 months ago
- rconnect 2 months ago
- report.ipynb 2 months ago
- cleaning.R 5 months ago
- data_load.R 5 months ago
- data_water.Rproj 5 months ago
- eda.R 5 months ago
- helpers.R 5 months ago
- machine_learning.R 5 months ago
- maps.R 5 months ago
- notes.Rmd 5 months ago
- report.html 2 months ago
- report.Rmd 2 months ago

Commands

Launcher report.ipynb Python 2

In [2]:
import pandas as pd
import missingno

%matplotlib inline

In [4]:
collect data urls
train_features_url = "http://s3.amazonaws.com/drivendata/data/7/publ
train_labels_url = "http://s3.amazonaws.com/drivendata/data/7/publ
test_features_url = "http://s3.amazonaws.com/drivendata/data/7/publ

read in data
train_features = pd.read_csv(train_features_url)
train_labels = pd.read_csv(train_labels_url)
test_features = pd.read_csv(test_features_url)

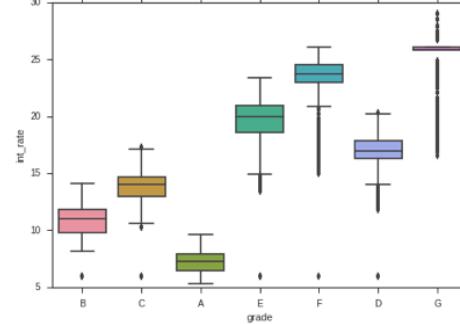
In [5]:
merge dataframes
train = pd.concat([train_labels, train_features], axis = 1)

In [6]:
missing data visualise
missingno.matrix(train)

lending_club.ipynb Python 2

In [6]: sns.boxplot(x = data_raw.grade, y = data_raw.int_rate)

Out[6]: <matplotlib.axes._subplots.AxesSubplot at 0x7fd42ebc6290>



In [7]: data_raw.shape

Out[7]: (887379, 74)

In [15]: sns.distplot(data_raw['loan_amnt']);



.ta/deep-water

```
-rw-r--r-- 1 boyanangelov staff 256 Feb 16 20:16 maps.R
-rw-r--r-- 1 boyanangelov staff 706 Feb 16 20:16 notes.Rmd
drwxr-xr-x 3 boyanangelov staff 7316 Jun 5 21:07 plots
-rw-r--r-- 1 boyanangelov staff 3136447 Jun 5 21:07 report.Rmd
-rw-r--r-- 1 boyanangelov staff 395119 Jun 5 21:00 report.ipynb
drwxr-xr-x 3 boyanangelov staff 102 Jun 5 21:00 rconnect

# boyanangelov @ mac-home in ~/ds/drivendata/deep-water on git:master x [11:54:12]
$
```

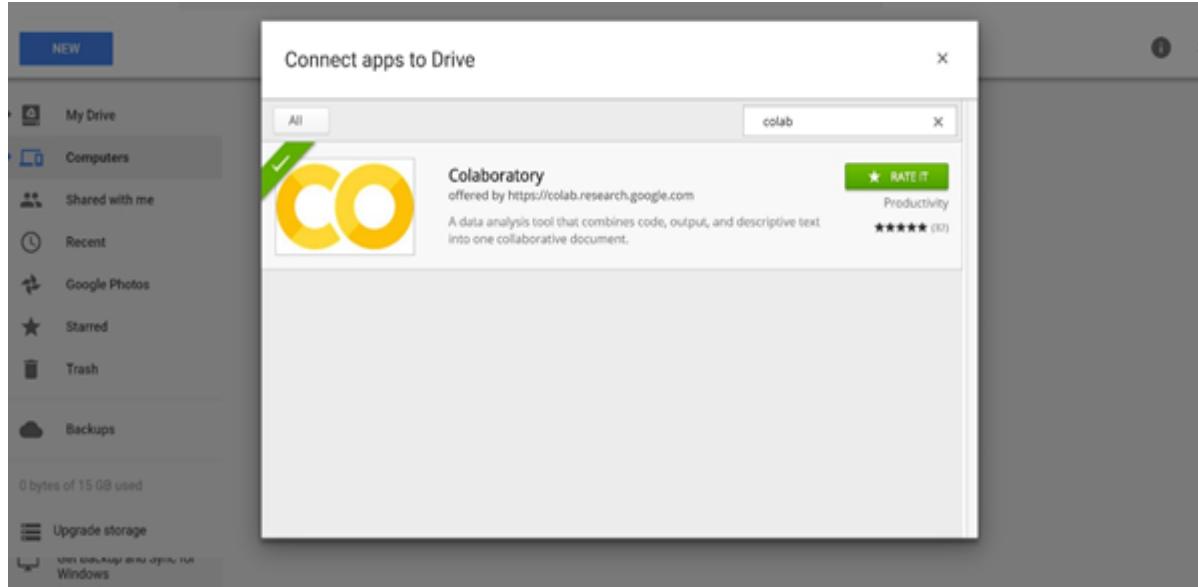
COLABORATORY



<https://colab.research.google.com/notebooks/welcome.ipynb>

- Es un entorno de Jupyter Notebook que no requiere configuración y que se ejecuta completamente en la nube.
- Los cuadernos de Colaboratory se almacenan en Google Drive, y puedes compartirlos como harías con Hojas de cálculo o Documentos de Google.
- Colaboratory es un servicio gratuito.

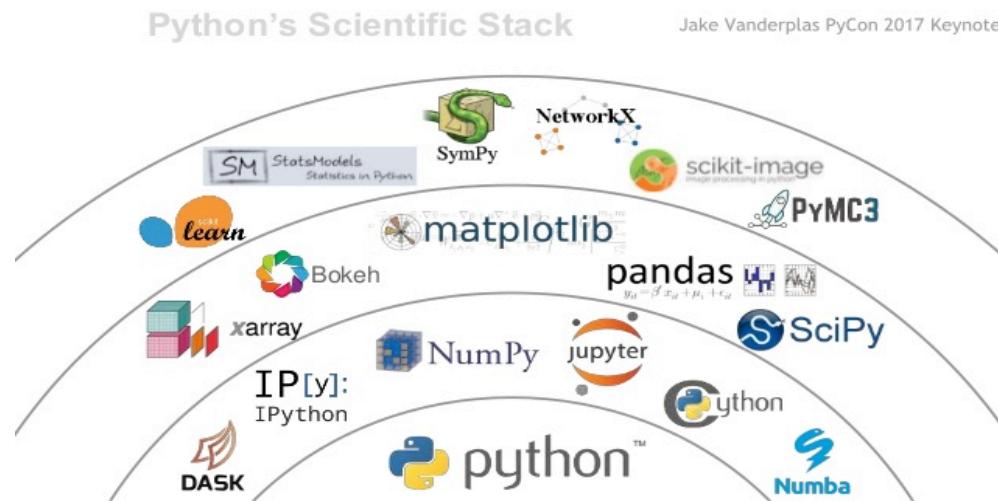
The screenshot shows the Google Drive web interface. At the top left is a 'NEW' button. The main title 'Computers' is centered above a large white area. In the center of this area is a circular icon containing a computer monitor, with the text 'No computers syncing' below it. To the right of the icon is a link: 'To sync folders on your computer with Google Drive, install Backup and Sync on your computer. [LEARN MORE](#)'. On the left side of the main area, there's a vertical sidebar with several options: 'Folder', 'File upload', 'Folder upload', 'Google Docs', 'Google Sheets', 'Google Slides', 'More', 'Backups', '0 bytes of 15 GB used', 'Upgrade storage', and a link to 'Windows'. A secondary dropdown menu is open under the 'More' option, listing 'Google Forms', 'Google Drawings', 'Google My Maps', 'Google Sites', and a 'Connect more apps' option at the bottom.



DOCKER



Diferentes programas o aplicaciones requieren múltiples recursos virtuales para funcionar correctamente...



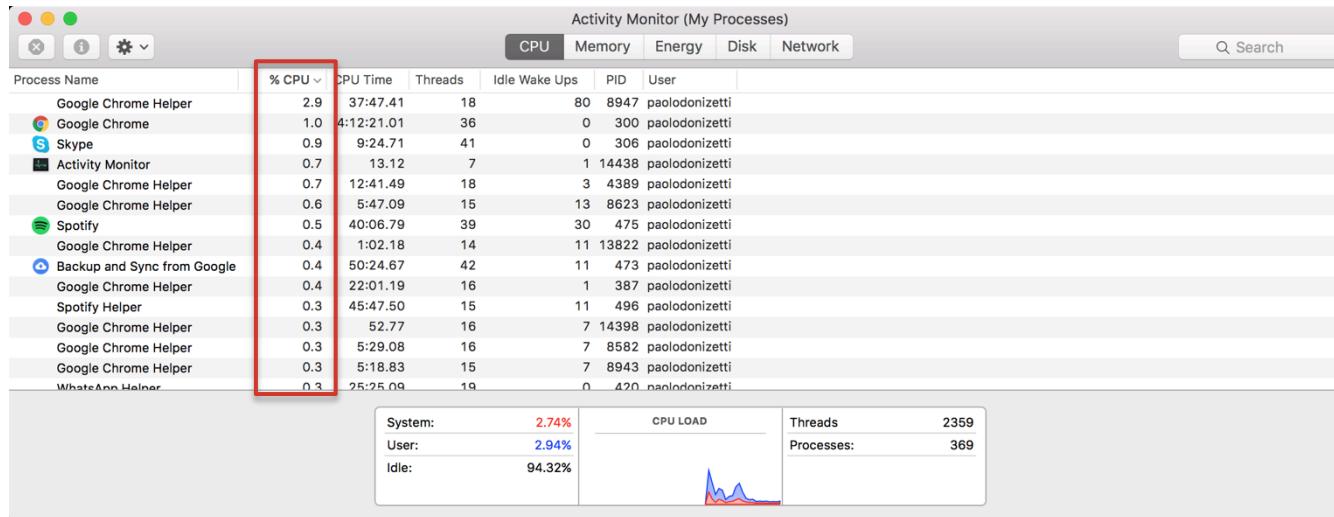
... Y estos recursos podrían no ser compatible entre sí...

Lidiar con la incompatibilidad dentro de los sistemas es bastante común en Data Science dada la cantidad de instancias:

- Estructura de datos
- Recursos Algorítmicos
- Gráficos
- Bases de Datos
- Formas de adquisición de la información
- Entornos de Desarrollo
- Recursos Matemáticos
- Mapas

Muchas formas de realizar una tarea en particular, pero con enfoques y técnicas que podrían diferir.

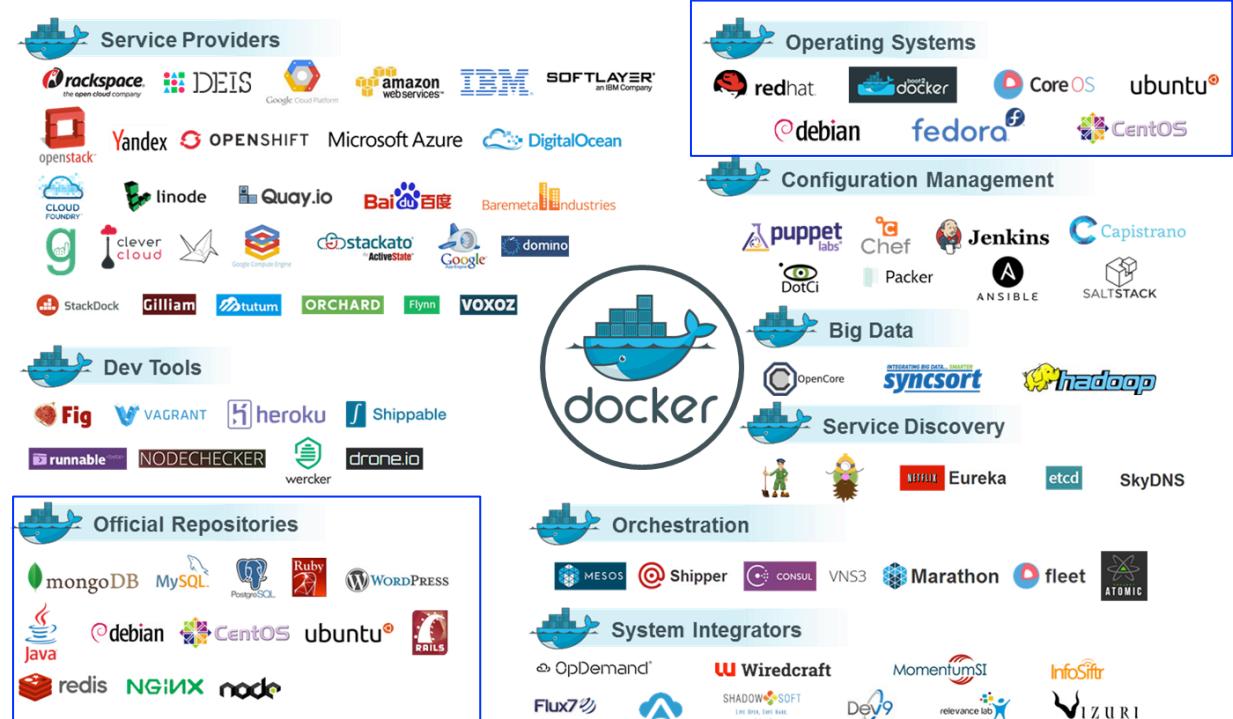
Los procesos compiten por los mismos ciclos de CPU, por la misma memoria y por el mismo disco.



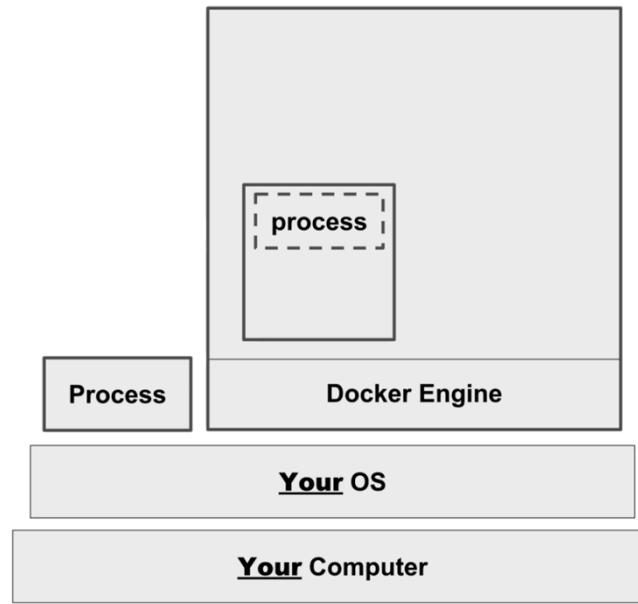
Esto implica que, si un proceso acapara todos los recursos, los demás procesos de mi computadora se van a ver afectados negativamente...

- Docker es una herramienta diseñada para facilitar la creación, implementación y ejecución de aplicaciones mediante el uso de **contenedores**.
- Los contenedores permiten a un desarrollador **empaquetar** una aplicación con todas las partes que necesita, como bibliotecas y otras dependencias, y enviarla como un solo paquete.
- Al hacerlo, gracias al contenedor, el desarrollador puede estar seguro de que la aplicación se ejecutará en cualquier otra máquina, independientemente de las configuraciones personalizadas que la máquina pueda tener (que difieran la máquina utilizada para escribir y donde se prueba el código).

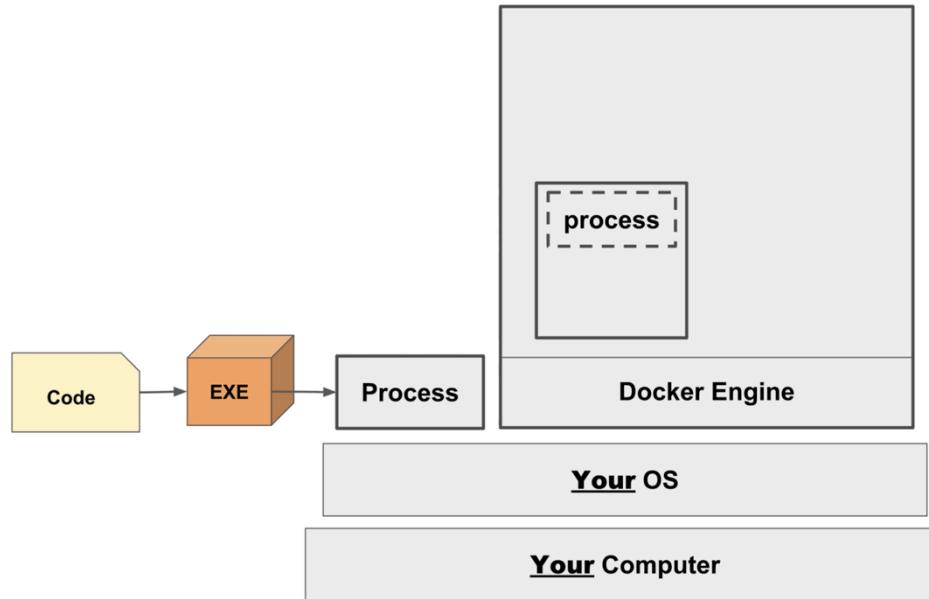
Un contenedor es un paquete liviano y autocontenido, con todo lo necesario para ser ejecutado, por la naturaleza virtual corre de igual forma, independiente del entorno.



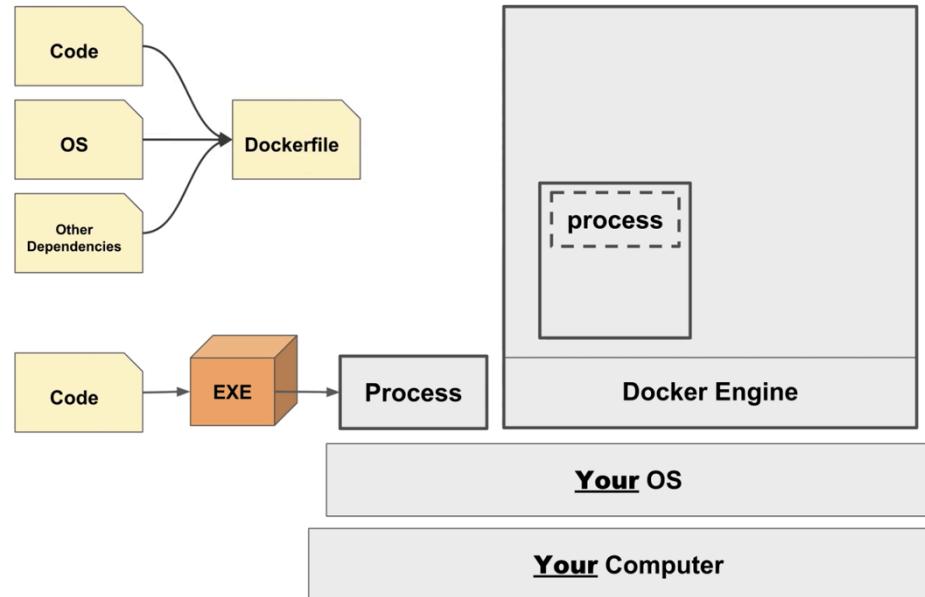
Comparemos cómo se crean un proceso regular y un contenedor de Docker...



Para el proceso regular comenzamos con un código que se compila como un ejecutable que corremos como un proceso en el OS.

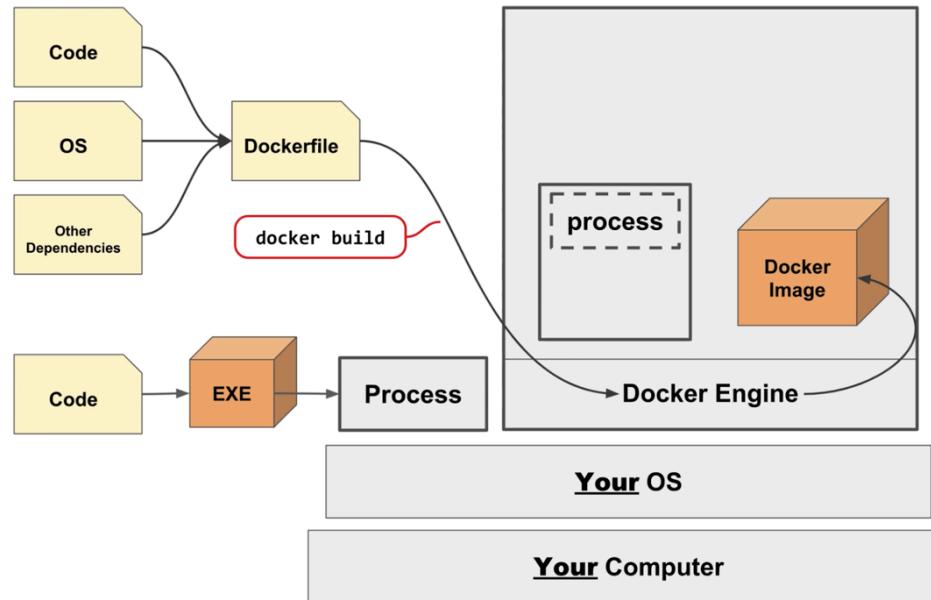


Para el contenedor, comenzamos con el código... también tenemos que agrupar el OS y las demás dependencias y crear un Dockerfile.



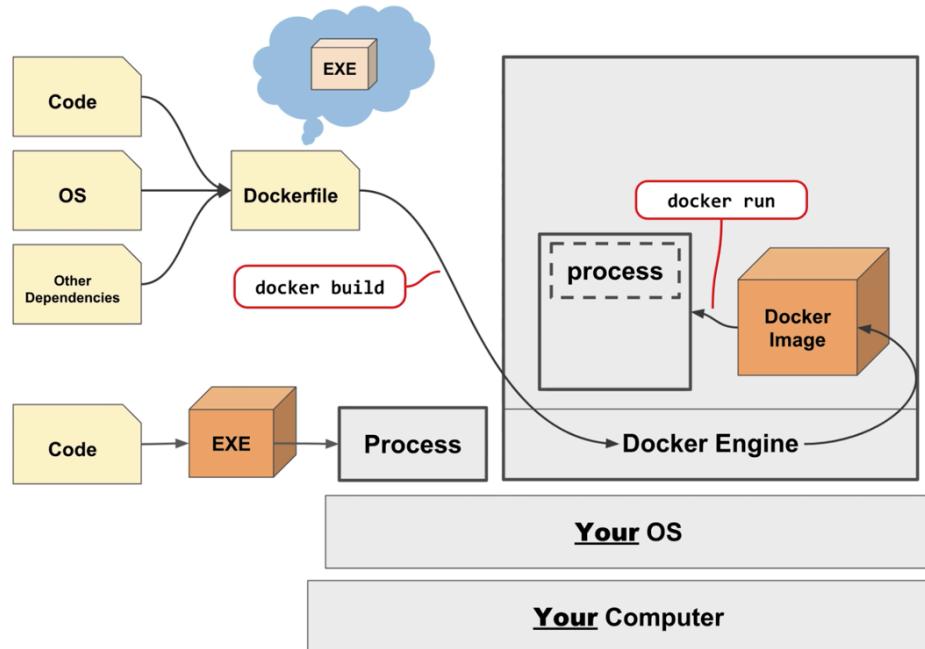
El Dockerfile es una lista de instrucciones que le pasamos al Docker Engine para que construya el contenedor.

Con el comando **docker build** le pasamos el Dockerfile al Docker Engine, que sigue las instrucciones para crear una Imagen de Docker que luego se usará para iniciar un contenedor.



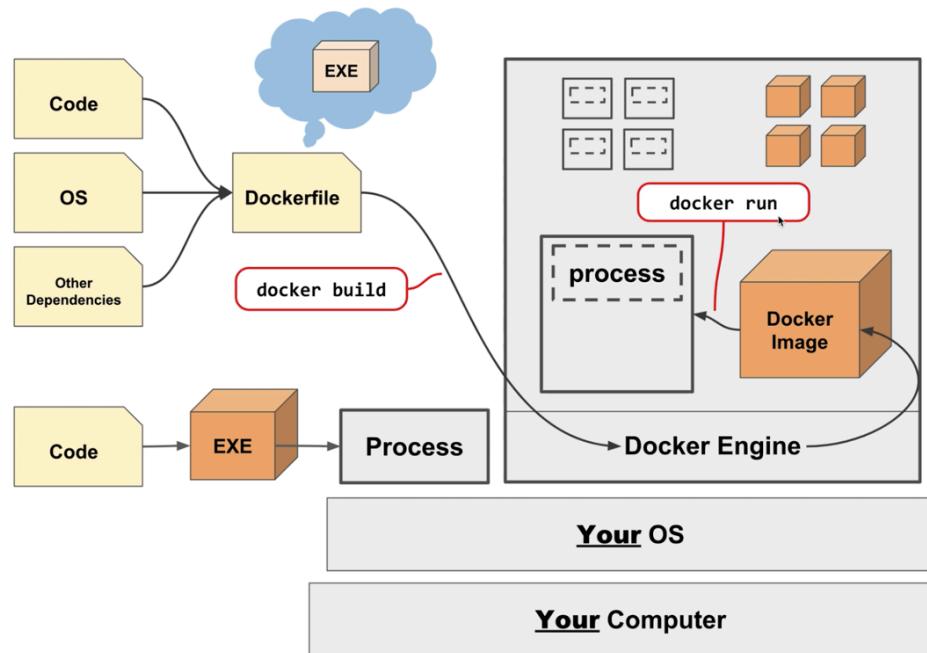
Un contenedor se construye a partir de un molde, este molde se conoce como imagen.

Con la Imagen de Docker iniciamos un contenedor con **docker run**.



Podemos compilar el ejecutable manualmente y pasarlo al Dockerfile o podemos describir como compilar el ejecutable en el Dockerfile como una instrucción.

El nuevo contenedor va a correr junto a los demás contenedores que tenga dentro de mi Docker Engine.

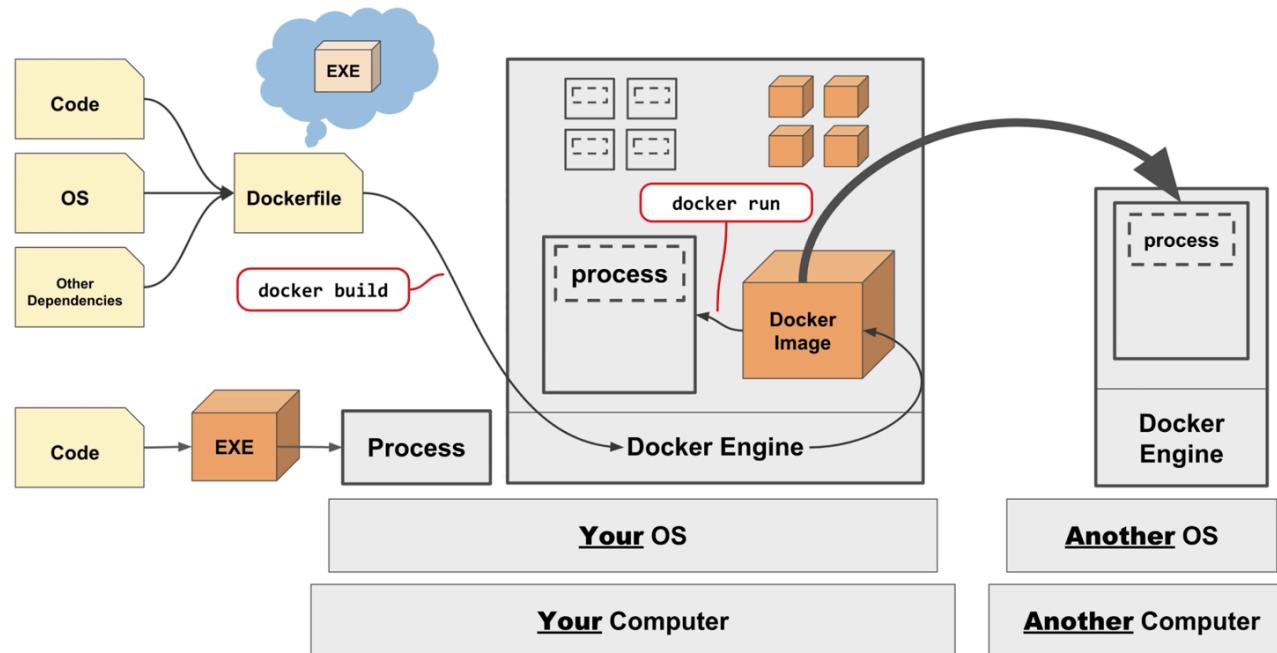


Los contenedores están aislados los unos de los otros: Docker Engine regula cuántos recursos asigna a cada uno.

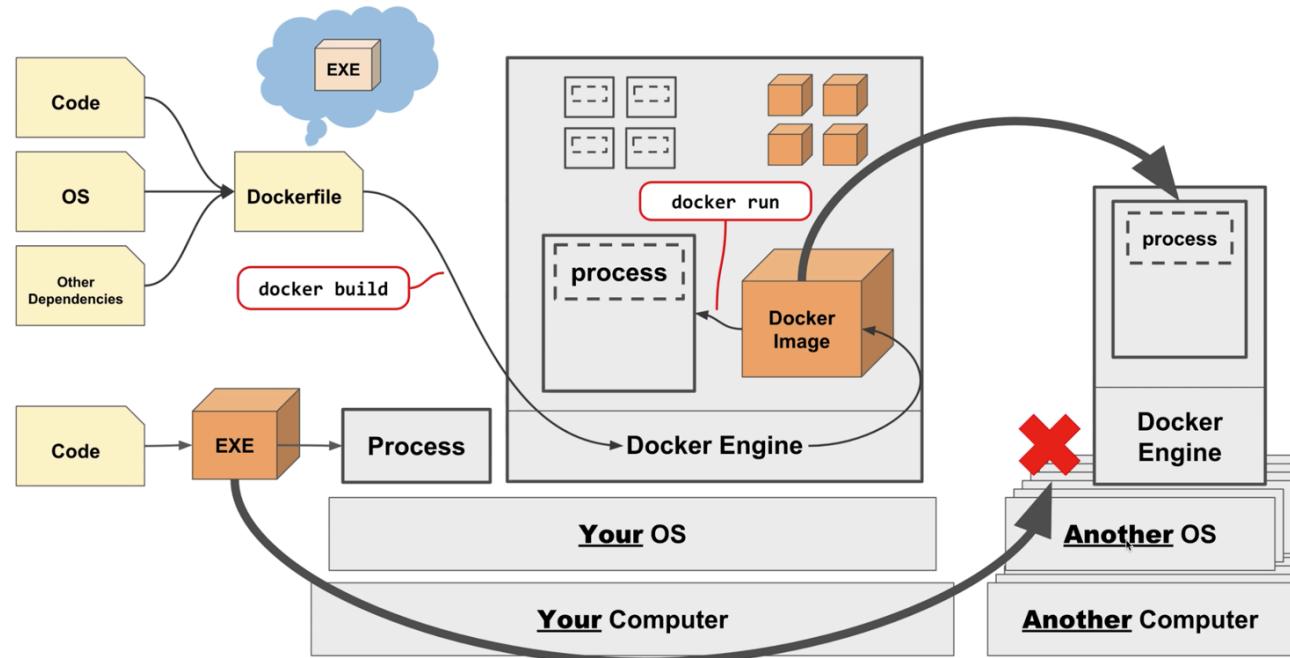
Cada vez que corremos **docker build** creamos una imagen nueva

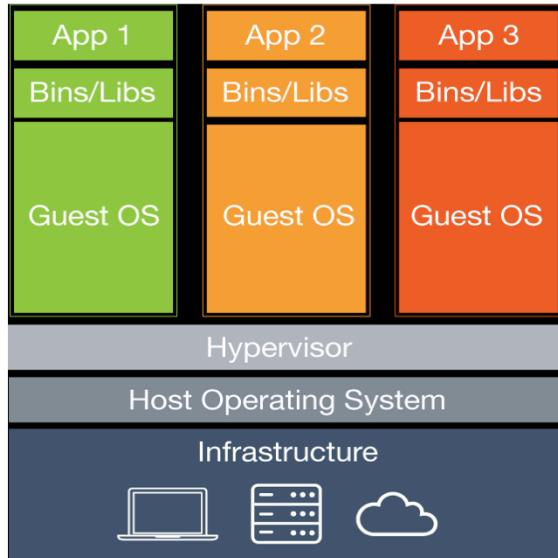
Docker: El contenedor.

Podemos correr una imagen de Docker como un contenedor en otro sistema. Docker nos asegura que este nuevo contenedor va a **correr como el original**.

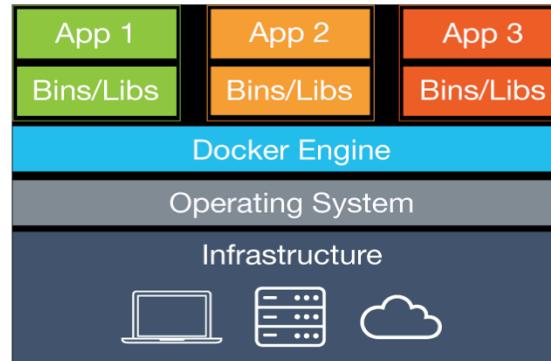


Esta **portabilidad** permite resolver el problema de compatibilidad entre sistemas que tendríamos si quisieramos correr el proceso directamente en otro sistema.





VM



Docker

USO DE DOCKER



- **Quickstart Terminal** es una terminal de Docker, contiene el entorno (comandos y herramientas) para realizar las tareas de gestión de los elementos de docker.

- Contenedores
 - Imágenes
 - Gestión de Virtualización

Comandos útiles para el manejo de contenedores:

run : Este comando se utiliza para la creación de un contenedor a partir de una imagen.
Una forma de usarlo es:

```
$ docker run [OPCIONES] imagen /comando
```

Las **Opciones** configuran el comportamiento del contenedor, la forma en la que se ejecuta y construye la imagen, e incluso qué pasa con la imagen después de su ejecución.

Dentro de las opciones más comunes dentro de **run** podemos encontrar:

- p : Asigna un puerto de salida al contenedor
- rmi : Borra la imagen después de la ejecución de la tarea
- rm : Borra un contenedor
- it : Abre una terminal interactiva que se comunica con el interior del contenedor
- user : Nombre de usuario que elegimos dentro del contenedor (tiene que existir dentro del contenedor)
- name : Nombre que elegimos para el contenedor

Ejemplos:

```
$ docker run hello-world
```

```
$ docker run --rm ubuntu /bin/echo "hola"
```

```
$ docker run -it --name mi_ubuntu ubuntu /bin/bash
```

```
$ docker run -p 8888:8888 jupyter/base-notebook
```

Comandos útiles para el manejo de contenedores:

ps -a : Lista los contenedores que hay en la máquina.

```
$ docker ps -a
```

stop : Detiene un contenedor en ejecución

```
$ docker stop contenedor
```

start : Ejecuta un contenedor.

```
$ docker start contenedor
```

rm : Elimina un contenedor (tiene que estar detenido)

```
$ docker rm contenedor
```

Comandos útiles para el manejo de imágenes:

images : Lista las imágenes que hay en la máquina.

```
$ docker images
```

rmi : Borra una imagen

```
$ docker rmi imagen
```

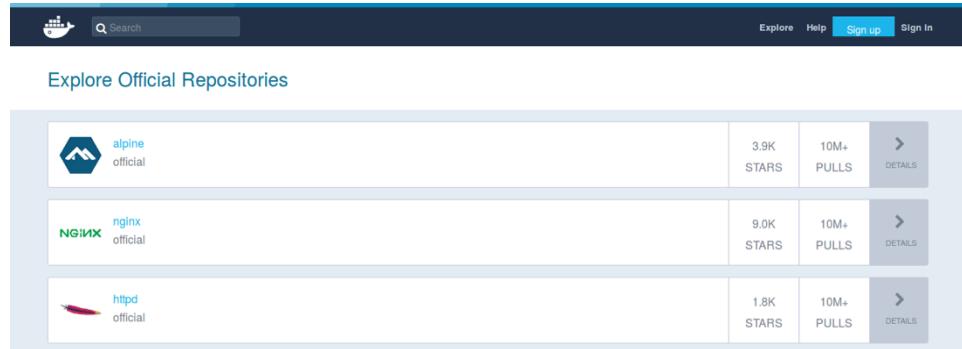
build : Construye una imagen en su contexto (en este caso local, notar el punto)

```
$ docker build .
```

Docker: Repositorio Docker Hub

Docker cuenta con un repositorio en donde se alojan las imágenes desarrolladas.

Puede ser público o privado y es posible subir y descargar una gran variedad de recursos, el repositorio cuenta con imágenes oficiales de los desarrolladores de las soluciones.



https://github.com/ds-dh/digital_data/blob/master/Dockerfile

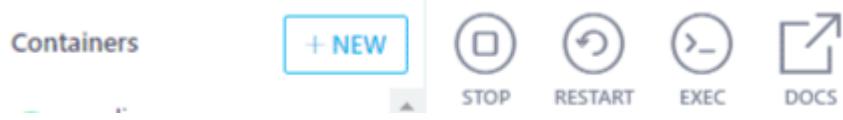
KITEMATIC



Para evitarnos el manejo a nivel de usuario se creó una aplicación gráfica para la gestión y ejecución de Docker: **Kitematic**

Permite buscar las imágenes (está conectado al repositorio Docker Hub), levantar, inicializar, detener, y borrar los contenedores, definir variables de entorno del contenedor, establecer sincronizaciones entre el contenedor y el host, configurar redes entre contenedores, etc.

Aún se encuentra en fase de desarrollo (actualmente está en versión alpha) y presenta algunos problemas.



+NEW : Construir un nuevo contenedor a partir de una imagen

Stop/Start : Para detener o iniciar un contenedor

Restart : Resetea un contenedor

Exec : Abre una terminal al interior del contenedor

Así se ve Docker Hub en Kitematic

Buscador de imágenes

Contenedor

Nombre de la imagen

Repositorio

Detalles del repositorio

Bajar imagen

Containers

+ NEW

Search for Docker Images from Docker Hub

FILTER BY All Recommended My Repos My Images

Recommended

- kitematic hello-world-nginx**
A light-weight nginx container that demonstrates the features of Kitematic
81 3M CREATE
- ghost**
ghost
Ghost is a free and open source blogging platform written in JavaScript
591 6M CREATE
- jenkins**
jenkins
Official Jenkins Docker image
2.9K 27M CREATE
- redis**
redis
Redis is an open source key-value store that functions as a data structure server.
3.9K 288M CREATE
- rethinkdb**
rethinkdb
RethinkDB is an open-source, document database that makes it easy to build and scale realtime...
411 7M CREATE
- solr**
solr
Solr is the popular, blazing-fast, open source enterprise search platform built on Apache...
400 3M CREATE
- elasticsearch**
elasticsearch
Elasticsearch is a powerful open source search and analytics engine that makes data easy to...
2.3K 77M CREATE
- postgres**
postgres
The PostgreSQL object-relational database system provides reliability and data integrity.
3.7K 65M CREATE

< >

<https://docs.docker.com/install/>

<https://docs.docker.com/engine/docker-overview/>

<https://opensource.com/business/14/7/guide-docker>

https://www.youtube.com/watch?time_continue=35&v=Q5POuMHxW-0