

# Treinamento Big Data

## Ambiente de Desenvolvimento

### Sumário

1 - Requisitos para instalação.....	2
2 - Ambiente – Instalação WSL2 Windows .....	2
3 - Ambiente – Docker.....	4
4 - Ambiente – Instalação Docker – Windows .....	4
5 - Ambiente Instalação Docker – Mac.....	5
6 - Ambiente Instalação Docker – Linux .....	5
7 - Ambiente Instalação Docker Engine - Linux .....	5
8 - Instalação Cluster Big Data.....	7

## 1 - Requisitos para instalação Containers

- Software

Docker e Docker Compose

## Sistema Operacional de 64 bits

- Windows
- Linux
- Mac
- Hardware

Memória RAM de 8 GB

Acesso à internet

HD com no mínimo 20 GB de espaço livre

## 2 - Ambiente – Instalação WSL2 Windows

Windows/System/Settings/About

- OS Build < 18362

Sem suporte a WSL 2

### Especificações do Windows

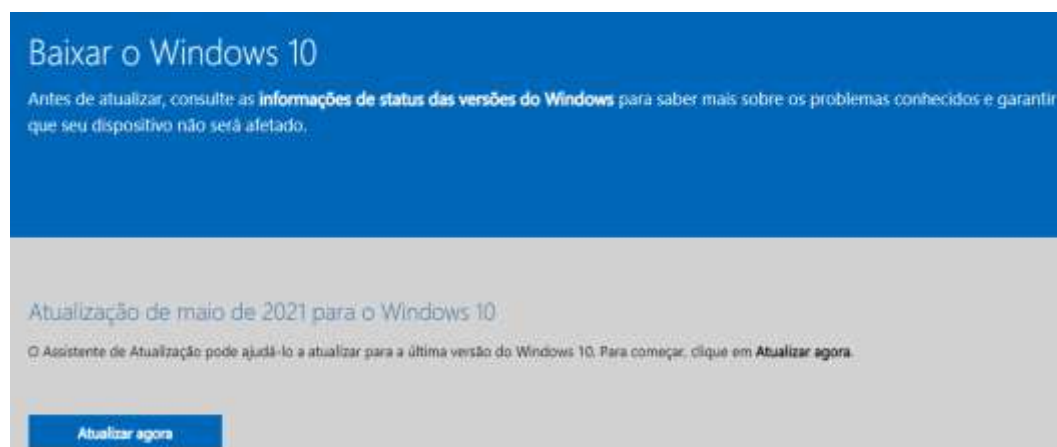
Edição	Windows 10 Pro
Versão	20H2
Instalado em	13/03/2021
Compilação do SO	19042.1288
Experiência	Windows Feature Experience Pack 120.2212.3920.0

Tutorial Oficial Microsoft

<https://docs.microsoft.com/pt-br/windows/wsl/install-win10>

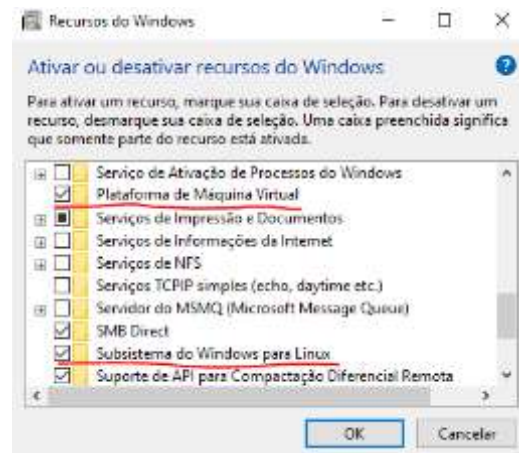
Link de Atualização Windows 10 (20.04)

<https://www.microsoft.com/pt-br/software-download/windows10>



Habilitar o Subsistema do Windows para Linux (WSL) e a Virtual Machine Platform

- Pesquisar no Windows por Ativar ou Desativar Recursos do Windows



## Download WSL 2

[https://wslstorestorage.blob.core.windows.net/wslblob/wsl\\_update\\_x64.msi](https://wslstorestorage.blob.core.windows.net/wslblob/wsl_update_x64.msi)

Execute o comando no PowerShell (Administrador)

Definir o WSL 2 como padrão ao instalar uma nova distribuição do Linux

`wsl --set-default-version 2`

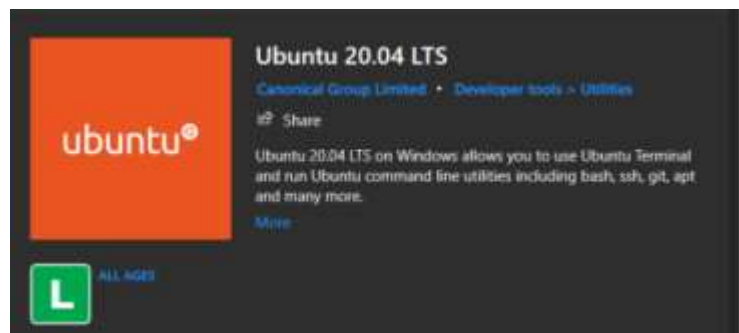
Definir o WSL 2 para uma distribuição já existente

`wsl --set-version <distribution name> 2`

## Instalar Distribuição Linux

Microsoft Store/Linux

Ubuntu 20.04 LTS (500 mb)



## Verificar versão distribuição

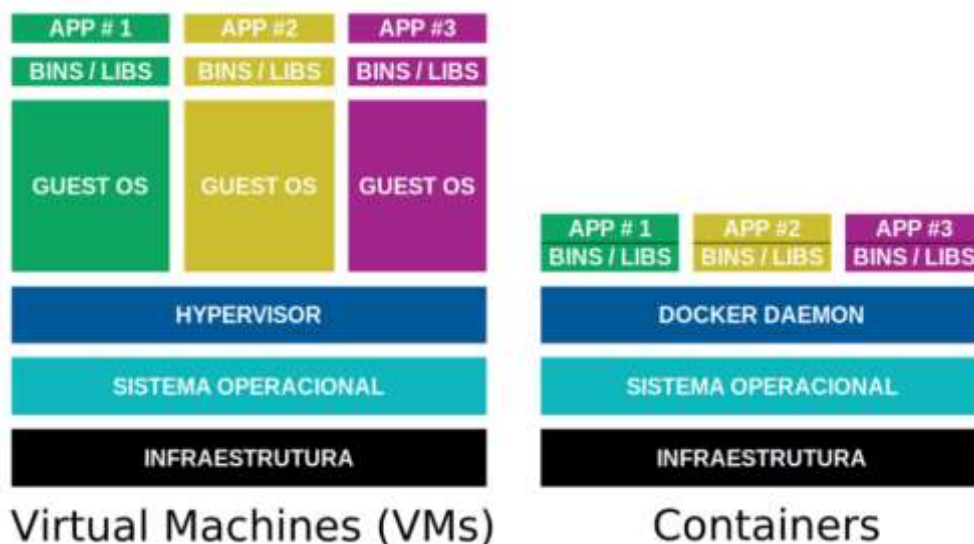
`wsl -l -v`

```
PS (PS C:\Users\aluisf> wsl -l -v
  NAME                STATE      VERSION
  * Ubuntu             Running    2
```

Caso a versão esteja 1 executar `wsl --set-version Ubuntu 2`

### 3 - Ambiente – Docker

Docker é uma plataforma de código aberto, desenvolvido na linguagem Go e criada pela própria Docker, Inc. Por ser de alto desempenho, o software garante maior facilidade na criação e administração de ambientes isolados, garantindo a rápida disponibilização de programas para o usuário final.

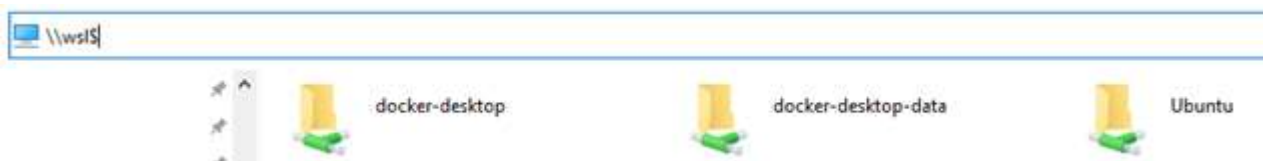


### 4 - Ambiente – Instalação Docker – Windows

<https://hub.docker.com/editions/community/docker-ce-desktop-windows/>



### Visualizar o disco do WSL \\wsl\$



## 5 - Ambiente Instalação Docker – Mac

<https://docs.docker.com/get-docker/>

**Get Docker**

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.

You can download and install Docker on multiple platforms. Refer to the following section and choose the best installation path for you.



### Docker Desktop for Mac

A native application using the macOS sandbox security model which delivers all Docker tools to your Mac.



### Docker Desktop for Windows

A native Windows application which delivers all Docker tools to your Windows computer.



### Docker for Linux

Install Docker on a computer which already has a Linux distribution installed.

## 6 - Ambiente Instalação Docker – Linux

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

**Server**

Docker provides `.deb` and `.rpm` packages from the following Linux distributions and architectures:

Platform	x86_64 / amd64	ARM	ARM64 / AARCH64
CentOS	•		•
Debian	•	•	•
Fedora	•		•
Raspbian		•	•
Ubuntu	•	•	•

## 7 - Ambiente Instalação Docker Engine - Linux

- Instalar o docker

```
$ sudo apt-get update
```

```
$ sudo apt-get install docker-ce docker-ce-cli containerd.io
```

- Alterar permissão

```
$ sudo usermod -aG docker $(whoami)
```

```
$ sudo reboot
```

- Verificar o status

```
sudo systemctl status docker
```

## Docker Engine no Ubuntu

- Atualizar e instalar alguns pacotes

```
$ sudo apt-get update
```

```
$ sudo apt-get install apt-transport-https ca-certificates curl  
gnupg-agent software-properties-common
```

- Adicionar Chave GPG do Docker

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg  
| sudo apt-key add -
```

- Configurar repositório

```
$ sudo add-apt-repository "deb [arch=amd64]
```

```
https://download.docker.com/linux/ubuntu $(lsb_release -  
cs) stable"
```

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



## Docker Compose no Ubuntu

- Baixar

```
$ sudo curl -L
```

```
"https://github.com/docker/compose/releases/download/1.28.2/docker-  
compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

- Alterar permissão

```
$ sudo chmod +x /usr/local/bin/docker-compose
```

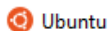
- Testar instalação

```
$ docker-compose --version
```

## 8 - Instalação Cluster Big Data

Baixar conteúdo do Cluster

\$ git clone <https://github.com/tiandrefreitas/docker-bigdata.git>



```
aluisf@ALUISF10:~/treinamento_indra$ git clone https://github.com/tiandrefreitas/docker-bigdata.git
```

Baixar as imagens

- docker-compose pull

Listar as imagens

- docker image ls

Executar os containers

- docker-compose up -d

```
$ docker-compose up -d
Creating network "docker-bigdata_default" with the default driver
Creating zookeeper      ... done
Creating database       ... done
Creating spark          ... done
Creating namenode      ... done
Creating hbase-master   ... done
Creating datanode       ... done
Creating hive-metastore-postgresql ... done
Creating hive_metastore ... done
Creating hive-server    ... done
```

Listar os containers

- docker container ls

```
$ docker container ls
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS
PORTS         NAMES
0c0871b0ecd4   fjardim/hive                        "entrypoint.sh /bin/..." About a minute ago Up About a minute
0.0.0.0:10000->10000/tcp, 10002/tcp   hive-server
3589dba8a75b   fjardim/hive                        "entrypoint.sh /opt/..." About a minute ago Up About a minute
10000/tcp, 0.0.0.0:9083->9083/tcp, 10002/tcp   hive_metastore
d2c0f331694e   fjardim/hive-metastore              "/docker-entrypoint...." About a minute ago Up About a minute
5432/tcp                                           hive-metastore-postgresql
2480e1629054   fjardim/hbase-master                "/entrypoint.sh /run..." About a minute ago Up About a minute
16000/tcp, 0.0.0.0:16010->16010/tcp   hbase-master
d9f035edd7ac   fjardim/datanode                    "/entrypoint.sh /run..." About a minute ago Up About a minute
0.0.0.0:50075->50075/tcp               datanode
```

## 9 - Comandos Docker

Iniciar todos os serviços em background (-d)

\$ docker-compose up -d

o Parar os serviços

\$ docker-compose stop

o Iniciar os serviços

\$ docker-compose start



## Visualizar os containers

- Ativos

\$ docker ps

- Todos

\$ docker ps -a

## Executar comandos no container

- \$ docker exec -it <container> <comando>

## Visualizar os logs

- \$ docker logs <container>

- \$ docker-compose logs

## Enviar arquivos

- \$ docker cp <diretório> <container>:/<diretório>

## Acesso o container namenode

- docker exec -it namenode bash

## Acesso o container do hive

- docker exec -it hive-server bash

## Ao Término do treinamento

- Matar os serviços

\$ docker-compose down

- Apagar todos os volumes sem uso

\$ docker volume prune

- Apagar tudo (image, volume, network)

\$ docker system prune --all