Containers

Corso di: Big Data Architectures

Claudio Badii, Paolo Nesi

Department of Systems and Informatics, University of Florence
Via S. Marta 3, 50139, Firenze, Italy
tel: +39-055-2758515, fax: +39-055-2758570

DISIT Lab, Sistemi Distribuiti e Tecnologie Internet

http://www.disit.dinfo.unifi.it/

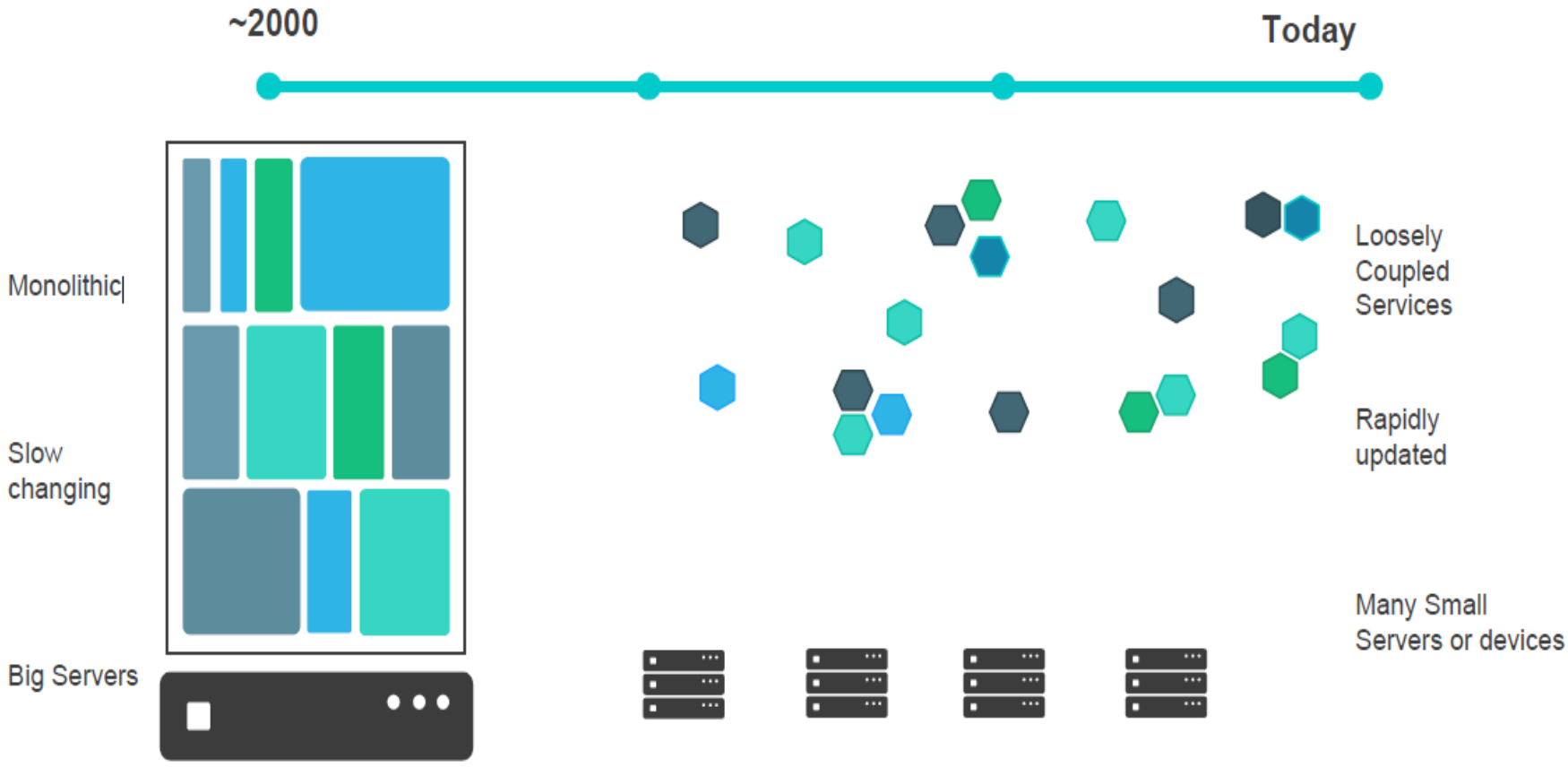
paolo.nesi@unifi.it
http://www.disit.dinfo.unifi.it/nesi

The path to Containers

- Most of the processes have been moved to Cloud
- Cloud costs are high since the hypervisors are expensive and the orchestrator are very expensive
- The costs for cloud management are mainly:
  - Per CPU
  - Per Cores
  - Per hosts
  - Per VM
- This implies a costs per process.

- *The next slides have been taken from Docker*

Applications are transforming



From Docker

Application Modernization

Developer Issues:

- Minor code changes require full re-compile and re-test
- Application becomes single point of failure
- Application is difficult to scale

Microservices: Break application into separate operations

12-Factor Apps: Make the app independently scalable, stateless, highly available by design

	MICROSERVICES	TRADITIONAL APPS
Cloud or New Infrastructure	You are either here..	
Old Infrastructure		…or here

History of Docker

2004
Solaris Containers / Zones technology introduced

2008
Linux containers (LXC 1.0) introduced

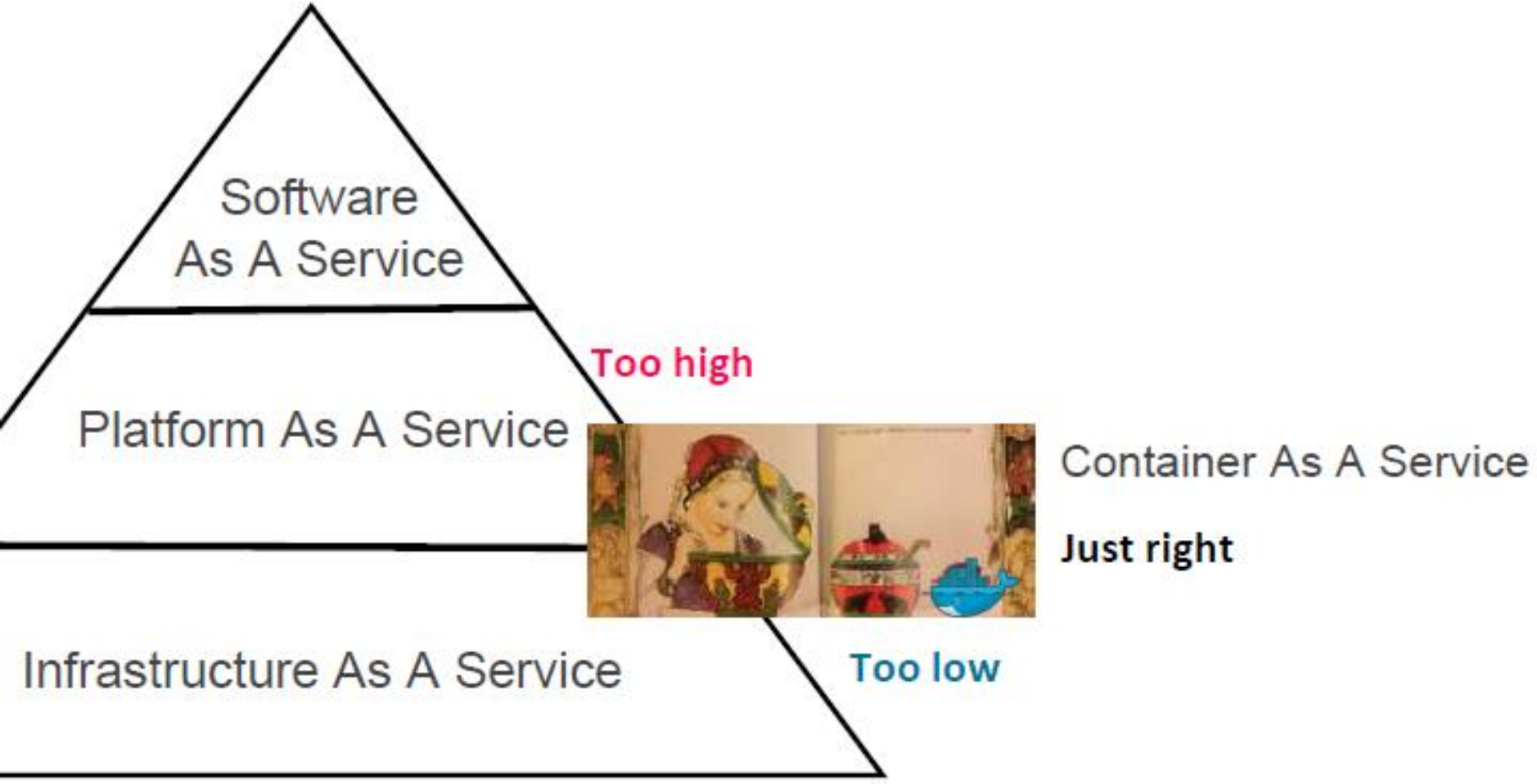
2013
Solomon Hykes starts Docker as an internal project within dotCloud

Mar 2013
Docker released to open source

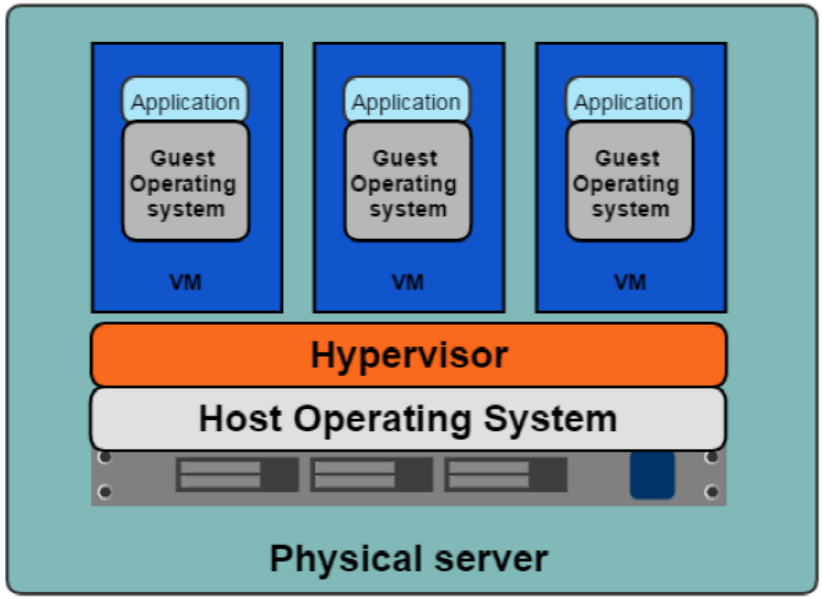
Feb 2016
Docker introduces first commercial product – now called Docker Enterprise Edition

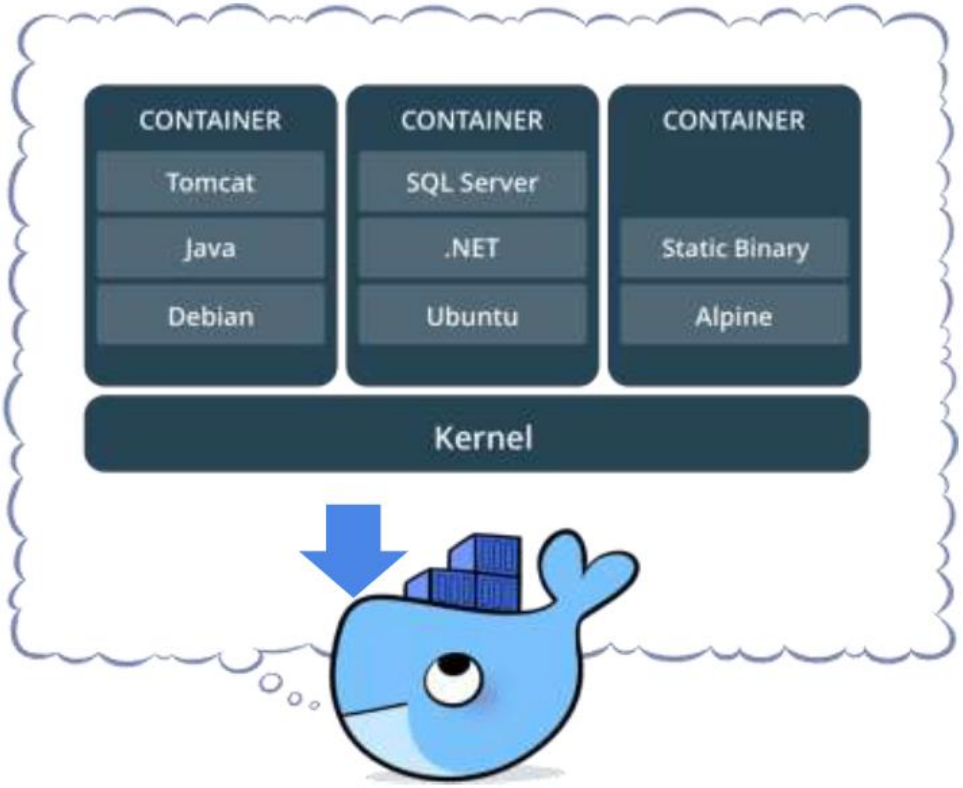
Today
Open source community includes:
- 3,300+ contributors
- 43,000+ stars
- 12,000+ forks

The 4 XaaS


Software
As A Service
Too high
Platform As A Service
Container As A Service
Just right
Infrastructure As A Service
Too low

Cloud vs Docker





Cloud

Pros and Cons of Cloud and VM

PROS

- Better resource pooling
  - One physical machine divided into multiple virtual machines
- Easier to scale
  - Elastically V/H
- VMs in the cloud
  - Rapid elasticity
  - as a service

CONS

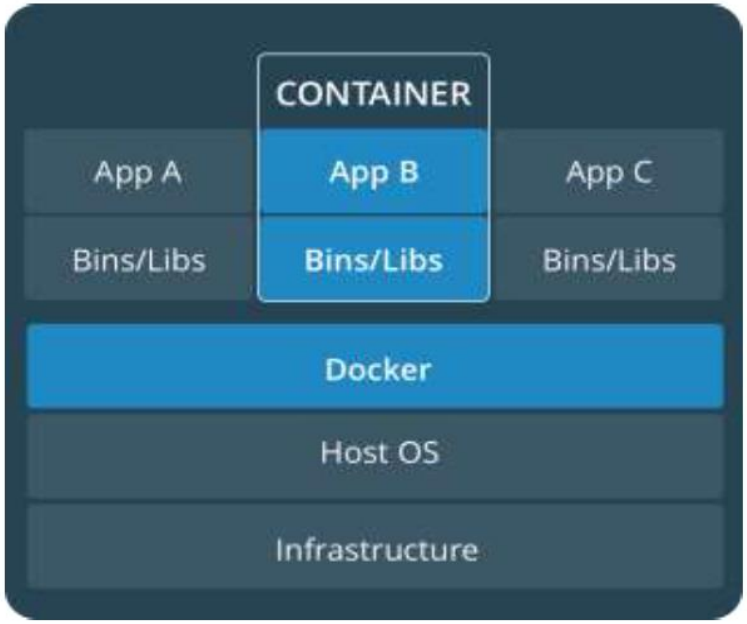
- Each VM stills requires
  - CPU allocation
  - Storage
  - Memory
  - An entire guest operating system
- more VMs, more resources you need
- Guest OS means wasted resources
- Application portability not guaranteed

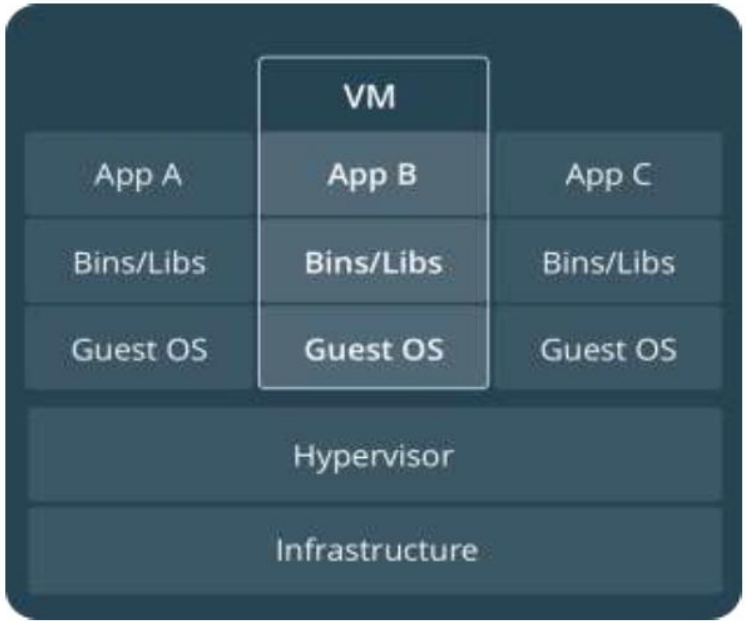
Pros of Containers

- Standardized packaging for software and dependencies
- Isolate apps from each other
- Share the same OS kernel
- Works with all major Linux and Windows Server

- Speed: No OS to boot = applications online in seconds
- Portability: Less dependencies between process layers =
  - ability to move between infrastructure
- Efficiency: Less OS overhead; Improved VM density

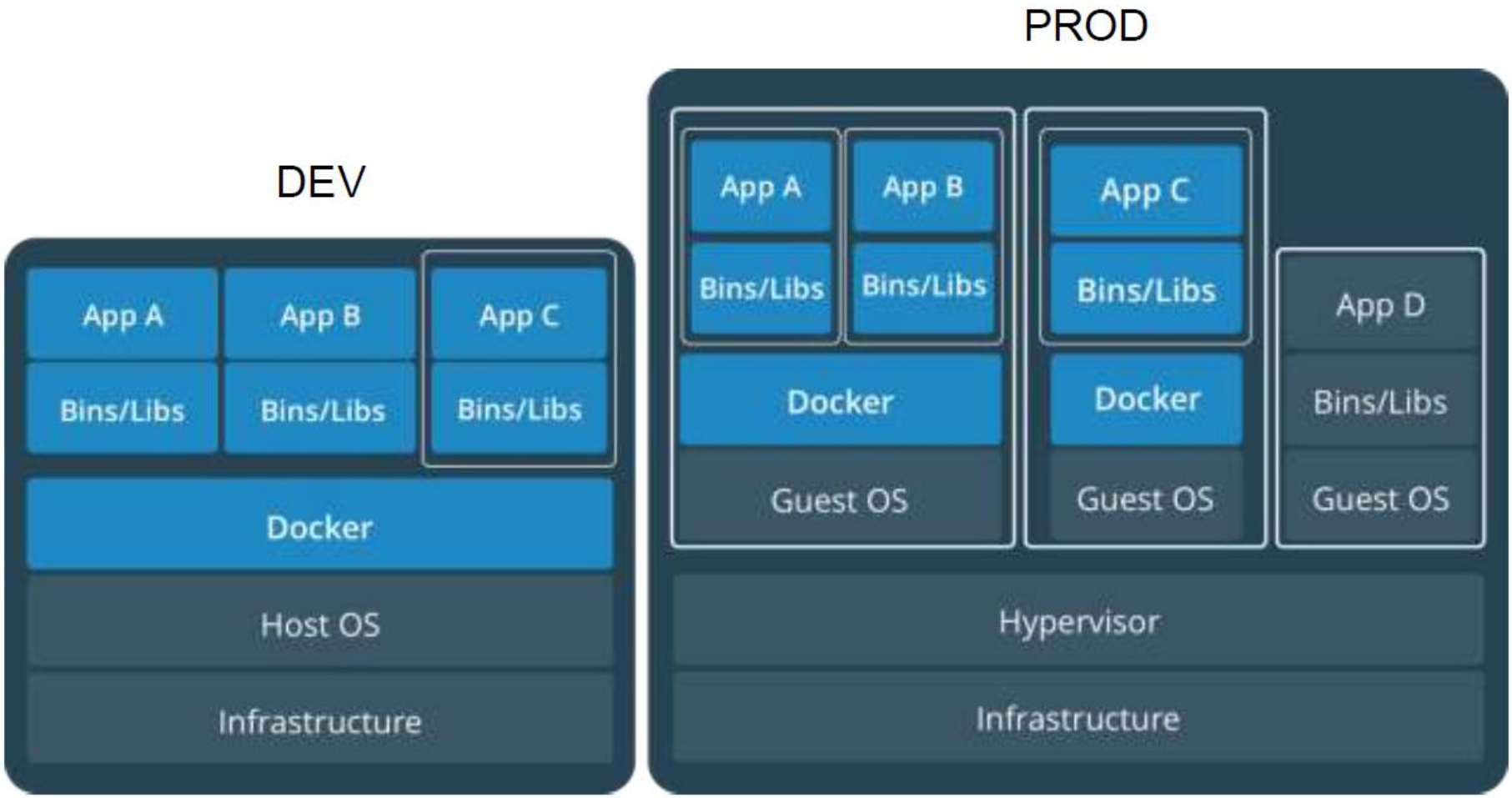
Comparing Container and VMs





Containers are an app level construct

VMs are an infrastructure level construct to turn one machine into many servers



Containers and VMs together provide a tremendous amount of flexibility for IT to optimally deploy and manage apps.

Docker Basics

Image

The basis of a Docker container. The content at rest.

Container

The image when it is 'running.' The standard unit for app service

Engine

The software that executes commands for containers. Networking and volumes are part of Engine. Can be clustered together.

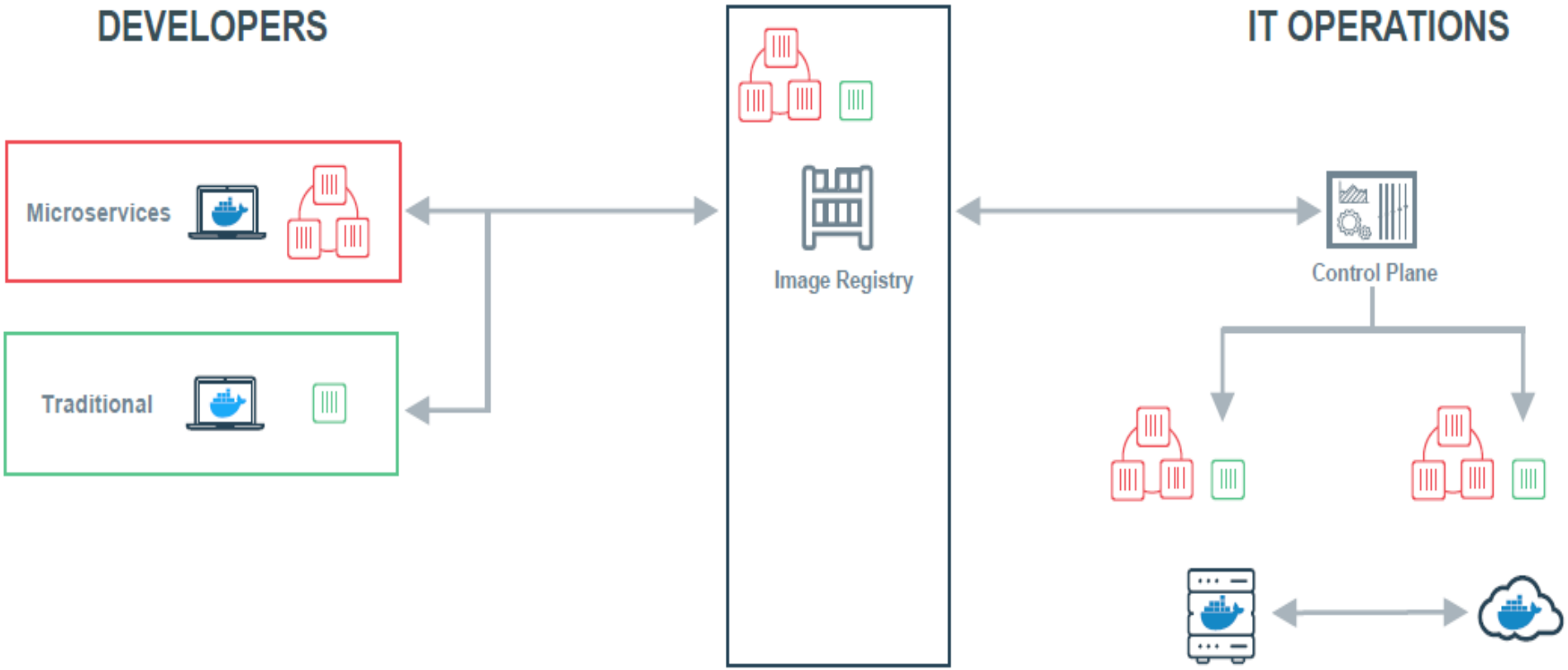
Registry

Stores, distributes and manages Docker images

Control Plane

Management plane for container and cluster orchestration

Building a Software Supply Chain



Docker Engine

Desktop

Platform	x86_64 / amd64
Docker Desktop for Mac (macOS)	✓
Docker Desktop for Windows	✓

Server

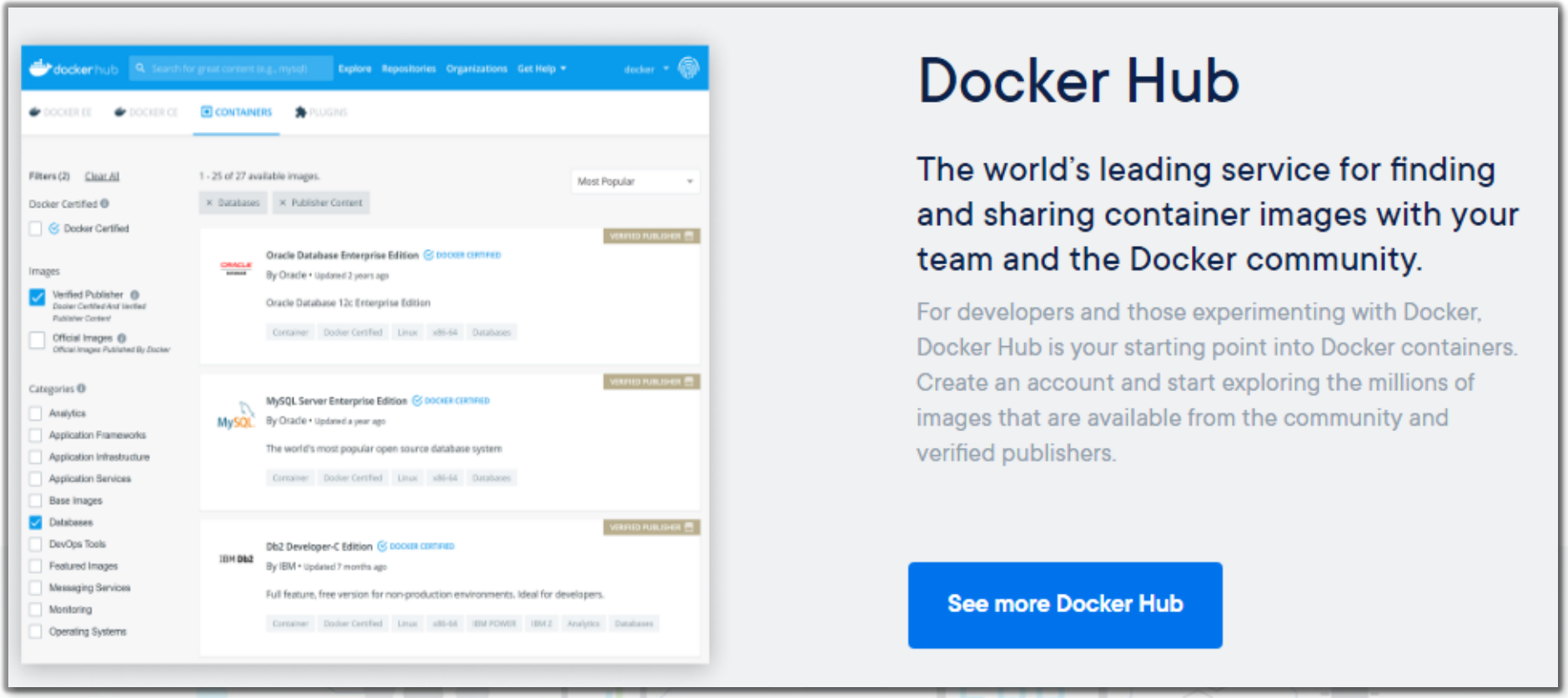
Docker provides `.deb` and `.rpm` packages from the following Linux distributions and architectures:

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

```
C:\Users\paolo>docker --version
Docker version 18.06.1-ce, build e68fc7a
```

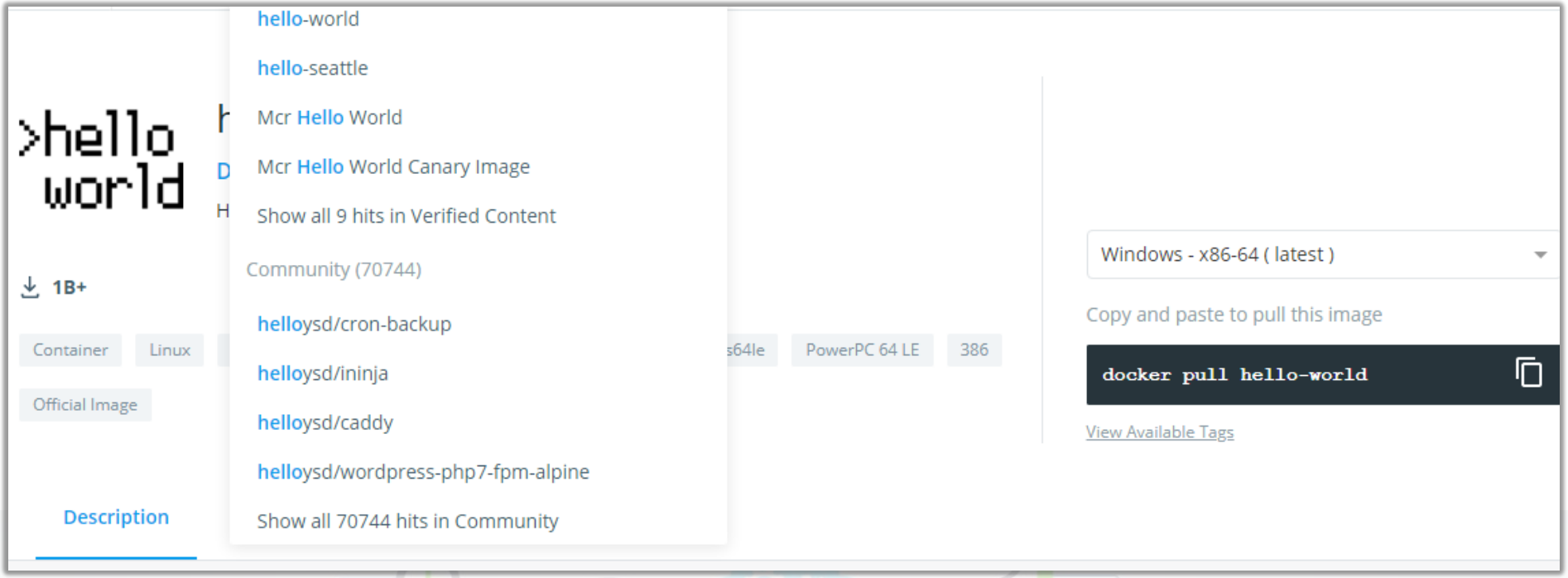
https://docs.docker.com/engine/install/

Docker Hub



docker pull imageName:imageVersion

Image hello-world



docker pull hello-world

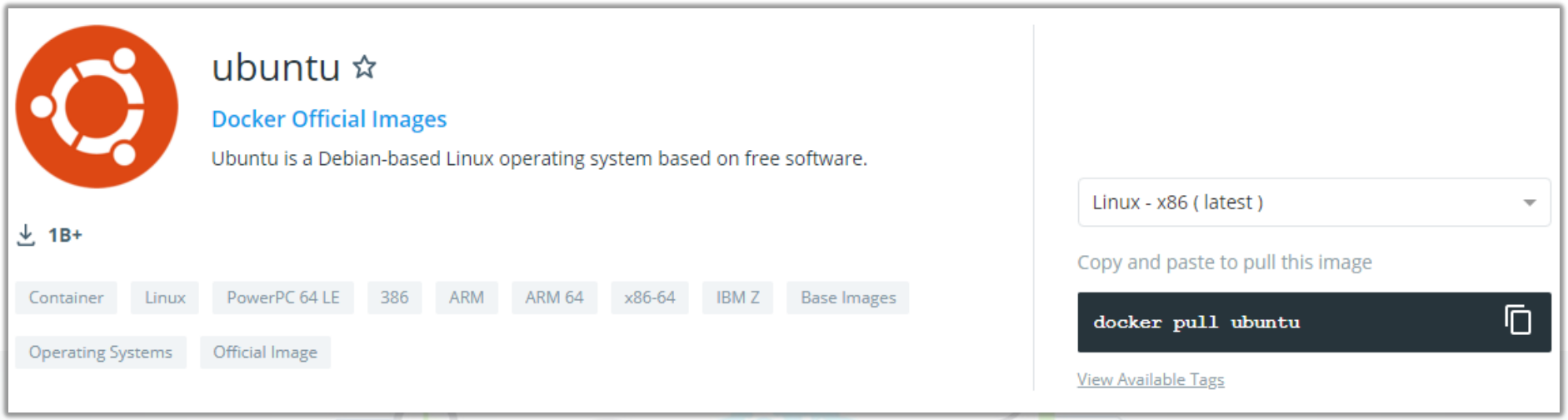
Image hello-world

```
C:\Users\paolo>docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
d1725b59e92d: Pull complete
Digest: sha256:0add3ace90ecb4adbf7777e9aacf18357296e799f81cabc9fde470971e499788
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```

docker run hello-world

docker --version

Image ubuntu



docker pull ubuntu

Image ubuntu

```
C:\Users\paolo>docker run -it ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
473ede7ed136: Pull complete
c46b5fa4d940: Pull complete
93ae3df89c92: Pull complete
6b1eed27cade: Pull complete
Digest: sha256:29934af957c53004d7fb6340139880d23fb1952505a15d69a03af0d1418878cb
Status: Downloaded newer image for ubuntu:latest
root@a5e45c9653f4:/# _
```

```
root@a5e45c9653f4:/# ls -la
total 72
drwxr-xr-x   1 root root 4096 Oct 29 18:19 .
drwxr-xr-x   1 root root 4096 Oct 29 18:19 ..
-rwxr-xr-x   1 root root    0 Oct 29 18:19 .dockerenv
drwxr-xr-x   2 root root 4096 Oct 18 21:03 bin
drwxr-xr-x   2 root root 4096 Apr 24  2018 boot
drwxr-xr-x   5 root root  360 Oct 29 18:19 dev
drwxr-xr-x   1 root root 4096 Oct 29 18:19 etc
drwxr-xr-x   2 root root 4096 Apr 24  2018 home
drwxr-xr-x   8 root root 4096 Oct 18 21:02 lib
drwxr-xr-x   2 root root 4096 Oct 18 21:02 lib64
drwxr-xr-x   2 root root 4096 Oct 18 21:02 media
drwxr-xr-x   2 root root 4096 Oct 18 21:02 mnt
drwxr-xr-x   2 root root 4096 Oct 18 21:02 opt
dr-xr-xr-x 138 root root    0 Oct 29 18:19 proc
drwx------   2 root root 4096 Oct 18 21:03 root
drwxr-xr-x   1 root root 4096 Oct 19 00:47 run
drwxr-xr-x   1 root root 4096 Oct 19 00:47 sbin
drwxr-xr-x   2 root root 4096 Oct 18 21:02 srv
dr-xr-xr-x  13 root root    0 Oct 29 18:19 sys
drwxrwxrwt   2 root root 4096 Oct 18 21:03 tmp
drwxr-xr-x   1 root root 4096 Oct 18 21:02 usr
drwxr-xr-x   1 root root 4096 Oct 18 21:03 var
root@a5e45c9653f4:/# _
```

```
root@a5e45c9653f4:/dev# ls -la
total 4
drwxr-xr-x 5 root root    360 Oct 29 18:19 .
drwxr-xr-x 1 root root   4096 Oct 29 18:19 ..
crw--w---- 1 root tty  136, 0 Oct 29 18:21 console
lrwxrwxrwx 1 root root     11 Oct 29 18:19 core -> /proc/kcore
lrwxrwxrwx 1 root root     13 Oct 29 18:19 fd -> /proc/self/fd
crw-rw-rw- 1 root root   1, 7 Oct 29 18:19 full
drwxrwxrwt 2 root root     40 Oct 29 18:19 mqueue
crw-rw-rw- 1 root root   1, 3 Oct 29 18:19 null
lrwxrwxrwx 1 root root      8 Oct 29 18:19 ptmx -> pts/ptmx
drwxr-xr-x 2 root root      0 Oct 29 18:19 pts
crw-rw-rw- 1 root root   1, 8 Oct 29 18:19 random
drwxrwxrwt 2 root root     40 Oct 29 18:19 shm
lrwxrwxrwx 1 root root     15 Oct 29 18:19 stderr -> /proc/self/fd/2
lrwxrwxrwx 1 root root     15 Oct 29 18:19 stdin -> /proc/self/fd/0
lrwxrwxrwx 1 root root     15 Oct 29 18:19 stdout -> /proc/self/fd/1
crw-rw-rw- 1 root root   5, 0 Oct 29 18:19 tty
crw-rw-rw- 1 root root   1, 9 Oct 29 18:19 urandom
crw-rw-rw- 1 root root   1, 5 Oct 29 18:19 zero
root@a5e45c9653f4:/dev#
```

docker run –it ubuntu bash

ls –la

Containers

```
Usage:  docker container COMMAND

Manage containers

Commands:
  attach      Attach local standard input, output, and error streams to a running container
  commit      Create a new image from a container's changes
  cp          Copy files/folders between a container and the local filesystem
  create      Create a new container
  diff        Inspect changes to files or directories on a container's filesystem
  exec        Run a command in a running container
  export      Export a container's filesystem as a tar archive
  inspect     Display detailed information on one or more containers
  kill        Kill one or more running containers
  logs        Fetch the logs of a container
  ls          List containers
  pause       Pause all processes within one or more containers
  port        List port mappings or a specific mapping for the container
  prune       Remove all stopped containers
  rename      Rename a container
  restart     Restart one or more containers
  rm          Remove one or more containers
  run         Run a command in a new container
  start       Start one or more stopped containers
  stats       Display a live stream of container(s) resource usage statistics
  stop        Stop one or more running containers
  top         Display the running processes of a container
  unpause     Unpause all processes within one or more containers
  update      Update configuration of one or more containers
  wait        Block until one or more containers stop, then print their exit codes

Run 'docker container COMMAND --help' for more information on a command.
```

Containers and Images

```
root@debian9:/home/badii# docker container ls --all
```

```
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS                      NAMES
7578cb426abd        ubuntu              "/bin/bash"         4 minutes ago       Exited (0) 4 minutes ago    vigorous_banzai
d204dd66ecad        hello-world         "/hello"            4 minutes ago       Exited (0) 4 minutes ago    trusting_hopper
root@debian9:/home/badii#
```

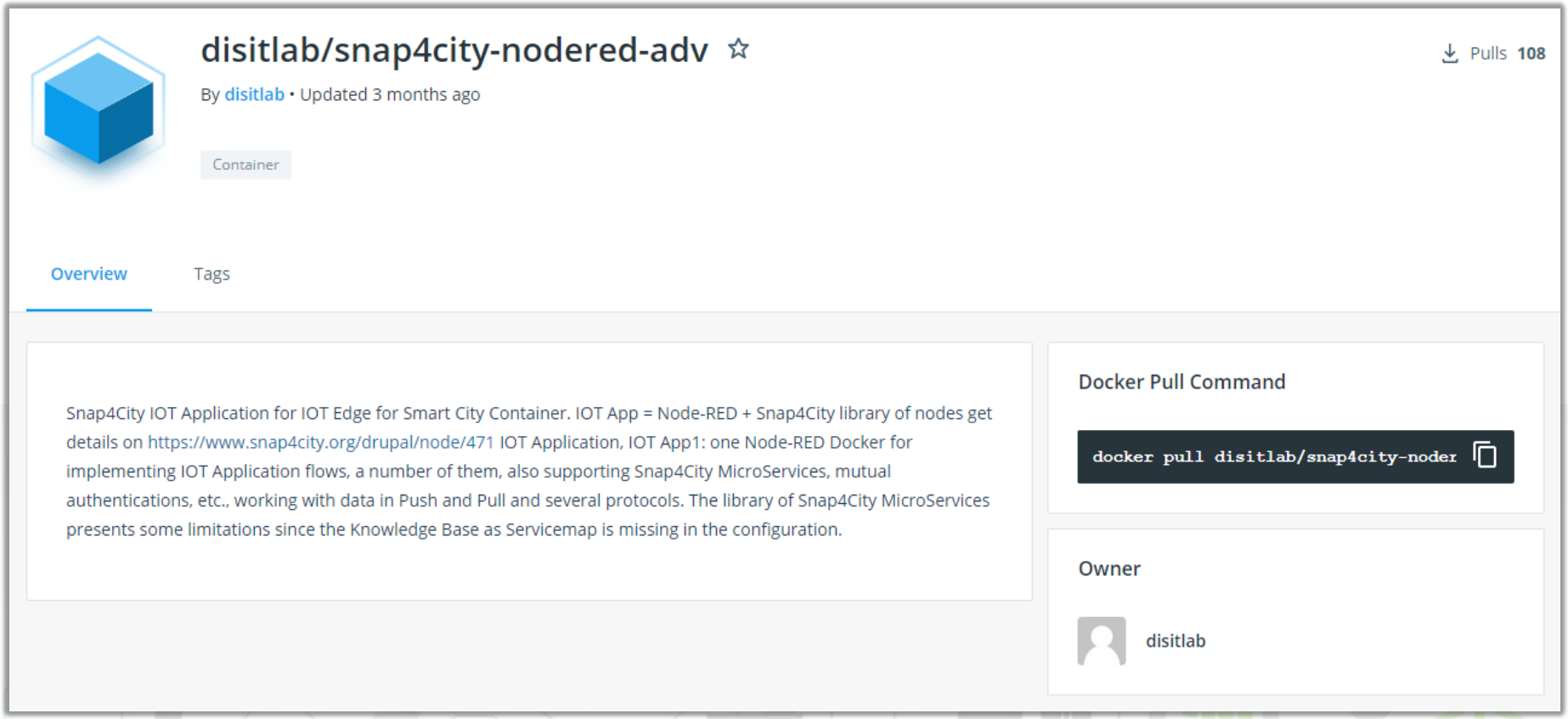
docker container ls --all

```
root@debian9:/home/badii# docker images
REPOSITORY              TAG                 IMAGE ID            CREATED             SIZE
ubuntu                  latest              9140108b62dc        2 weeks ago         72.9MB
snap4city-plumber       v8                  d30a16ab3a3f        4 months ago        2.84GB
hello-world             latest              bf756fb1ae65        9 months ago        13.3kB
trestletech/plumber     latest              f9aa6e6553fb        17 months ago       962MB
```

docker images

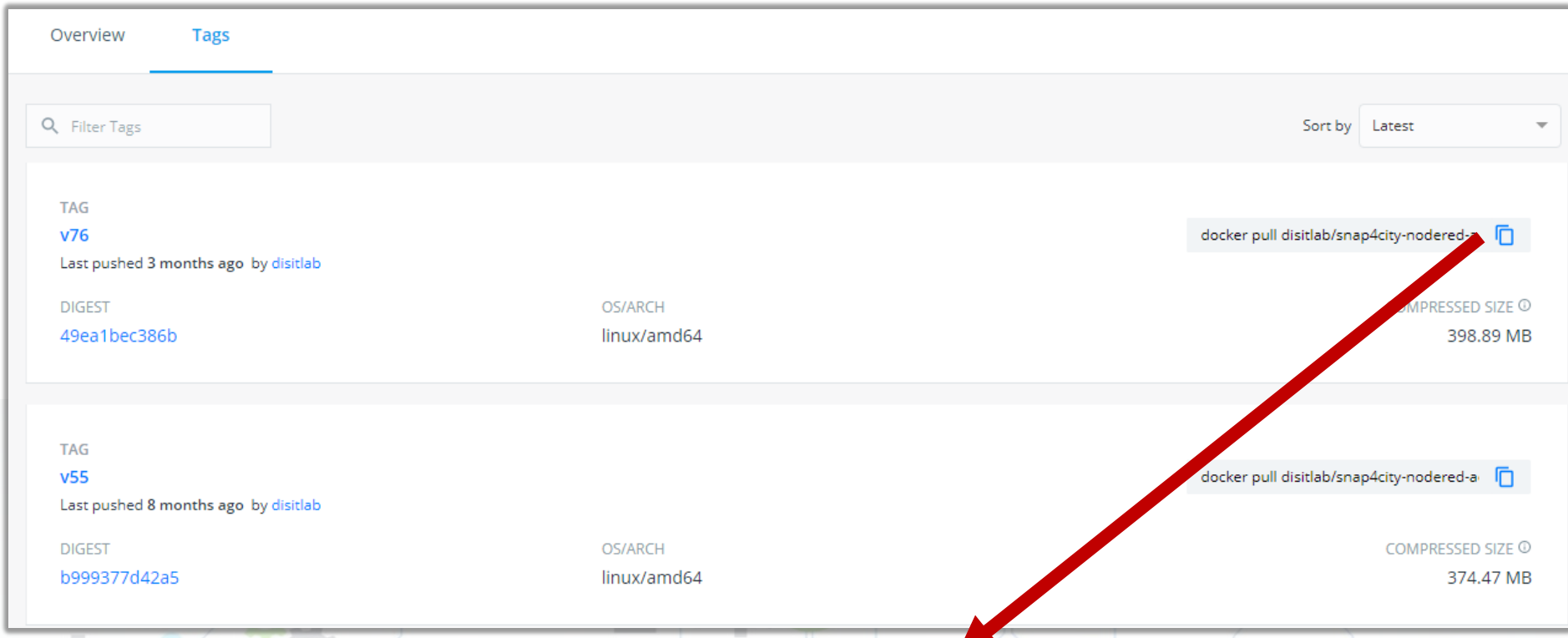
Make Little Changes

- *Let's see how to make small changes on an image we downloaded (for example the snap4city image)*



Make Little Changes

- *Let's see how to make small changes on an image we downloaded (for example the snap4city image)*



- *docker pull disitlab/snap4city-nodered-adv:v76*

Make Little Changes

- *Let's see how to make small changes on an image we downloaded (for example the snap4city image)*

```
root@debian9:/home/badii# docker pull disitlab/snap4city-nodered-adv:v76
v76: Pulling from disitlab/snap4city-nodered-adv
85b1f47fba49: Pull complete
ba6bd283713a: Pull complete
817c8cd48a09: Pull complete
47cc0ed96dc3: Pull complete
8888adcbd08b: Pull complete
6f2de60646b9: Pull complete
1dab1bd0d0d9: Pull complete
44ad4cf8b442: Pull complete
12fcc1c70dac: Pull complete
685330fe9c23: Pull complete
7d10c54dee0f: Pull complete
1fb8963ebd30: Pull complete
c451eb45c214: Pull complete
18397b7fb9e9: Pull complete
acd2fcc9f392: Pull complete
038f51de347c: Pull complete
Digest: sha256:49ea1bec386bfe93dbc4f7e10b91dd57314ea15c46d3aea455bb57245c8c0418
Status: Downloaded newer image for disitlab/snap4city-nodered-adv:v76
docker.io/disitlab/snap4city-nodered-adv:v76
```

- *docker pull disitlab/snap4city-nodered-adv:v76*

Make Little Changes

- *Let's see how to make small changes on an image we downloaded (for example the snap4city image)*

docker run imageName:imageVersion

```
root@debian9:/home/badii# docker run disitlab/snap4city-nodered-adv:v76

> node-red-docker@1.0.0 start /usr/src/node-red
> node $NODE_OPTIONS node_modules/node-red/red.js -v $FLOWS "--userDir" "/data"

26 Oct 14:25:43 - [info]

Welcome to Node-RED
===================

26 Oct 14:25:43 - [info] Node-RED version: v0.17.5
26 Oct 14:25:43 - [info] Node.js  version: v10.16.3
26 Oct 14:25:43 - [info] Linux 4.19.0-11-amd64 x64 LE
26 Oct 14:25:43 - [info] Loading palette nodes
26 Oct 14:25:45 - [info] Worldmap version 1.5.29
26 Oct 14:25:45 - [info] Dashboard version 2.14.0 started at /ui
26 Oct 14:25:45 - [warn] ---------------------------------------------------
26 Oct 14:25:45 - [warn] [rpi-gpio] Info : Ignoring Raspberry Pi specific node
26 Oct 14:25:45 - [warn] ---------------------------------------------------
26 Oct 14:25:45 - [info] Settings file  : /data/settings.js
26 Oct 14:25:45 - [info] User directory : /data
26 Oct 14:25:45 - [info] Flows file     : /data/flows.json
26 Oct 14:25:45 - [info] Creating new flow file
26 Oct 14:25:45 - [info] Starting flows
26 Oct 14:25:45 - [info] Started flows
26 Oct 14:25:45 - [info] Server now running at http://127.0.0.1:1880/
```

Make Little Changes

Directly inside the container with the command that executes instructions in the container

- *docker exec -it –user=root containerID command*

```
root@debian9:/home/badii# docker ps
CONTAINER ID        IMAGE                                COMMAND                  CREATED             STAT
US              PORTS               NAMES
b1528f7c72ca        disitlab/snap4city-nodered-adv:v76   "npm start -- --user…"   About a minute ago   Up 5
9 seconds       1880/tcp            unruffled_bartik
root@debian9:/home/badii# docker exec -it --user=root nodered-v1.1.3-adv /bin/bash
Error: No such container: nodered-v1.1.3-adv
root@debian9:/home/badii# docker exec -it --user=root b1528f7c72ca /bin/bash
root@b1528f7c72ca:/usr/src/node-red# apt-get update
Get:1 http://security.debian.org jessie/updates InRelease [44.9 kB]
Ign http://deb.debian.org jessie InRelease
Get:2 http://deb.debian.org jessie-updates InRelease [16.3 kB]
Get:3 http://deb.debian.org jessie Release.gpg [1652 B]
Get:4 http://deb.debian.org jessie Release [77.3 kB]
Get:5 http://security.debian.org jessie/updates/main amd64 Packages [992 kB]
Get:6 http://deb.debian.org jessie-updates/main amd64 Packages [20 B]
Get:7 http://deb.debian.org jessie/main amd64 Packages [9098 kB]
Fetched 10.2 MB in 22s (455 kB/s)
Reading package lists... Done
W: Size of file /var/lib/apt/lists/deb.debian.org_debian_dists_jessie-updates_main_binary-amd64_Packages.gz
 is not what the server reported 20 23046
root@b1528f7c72ca:/usr/src/node-red#
```

Make Little Changes

Using the command that copies files/folders from the external environment where the container is running inside the container

- *docker cp srcFolderPath containerID:dstFolderPath*

```
root@debian9:/home/badii# docker ps
CONTAINER ID        IMAGE                                  COMMAND                  CREATED             STAT
US              PORTS               NAMES
b1528f7c72ca        disitlab/snap4city-nodered-adv:v76   "npm start -- --user…"   About a minute ago   Up 5
9 seconds       1880/tcp            unruffled_bartik
root@debian9:/home/badii# docker exec -it --user=root nodered-v1.1.3-adv /bin/bash
Error: No such container: nodered-v1.1.3-adv
root@debian9:/home/badii# docker exec -it --user=root b1528f7c72ca /bin/bash
root@b1528f7c72ca:/usr/src/node-red# apt-get update
Get:1 http://security.debian.org jessie/updates InRelease [44.9 kB]
Ign http://deb.debian.org jessie InRelease
Get:2 http://deb.debian.org jessie-updates InRelease [16.3 kB]
Get:3 http://deb.debian.org jessie Release.gpg [1652 B]
Get:4 http://deb.debian.org jessie Release [77.3 kB]
Get:5 http://security.debian.org jessie/updates/main amd64 Packages [992 kB]
Get:6 http://deb.debian.org jessie-updates/main amd64 Packages [20 B]
Get:7 http://deb.debian.org jessie/main amd64 Packages [9098 kB]
Fetched 10.2 MB in 22s (455 kB/s)
Reading package lists... Done
W: Size of file /var/lib/apt/lists/deb.debian.org_debian_dists_jessie-updates_main_binary-amd64_Packages.gz
 is not what the server reported 20 23046
root@b1528f7c72ca:/usr/src/node-red#
```

Make Little Changes

- *Once the container has been modified, we can save it as a new image, In this way the new containers will have inside them the modifications made*

- *docker commit idContainer imageName:imageVersion*

- *docker save imageName:imageVersion | gzip > yourPath/filename.tgz*

Change Image With Docker File

Docker can build images automatically by reading the instructions from a Dockerfile.

A Dockerfile is a text document that contains all the commands a user could call on the command line to assemble an image.

Using docker build users can create an automated build that executes several command-line instructions in succession.

https://docs.docker.com/engine/reference/builder/

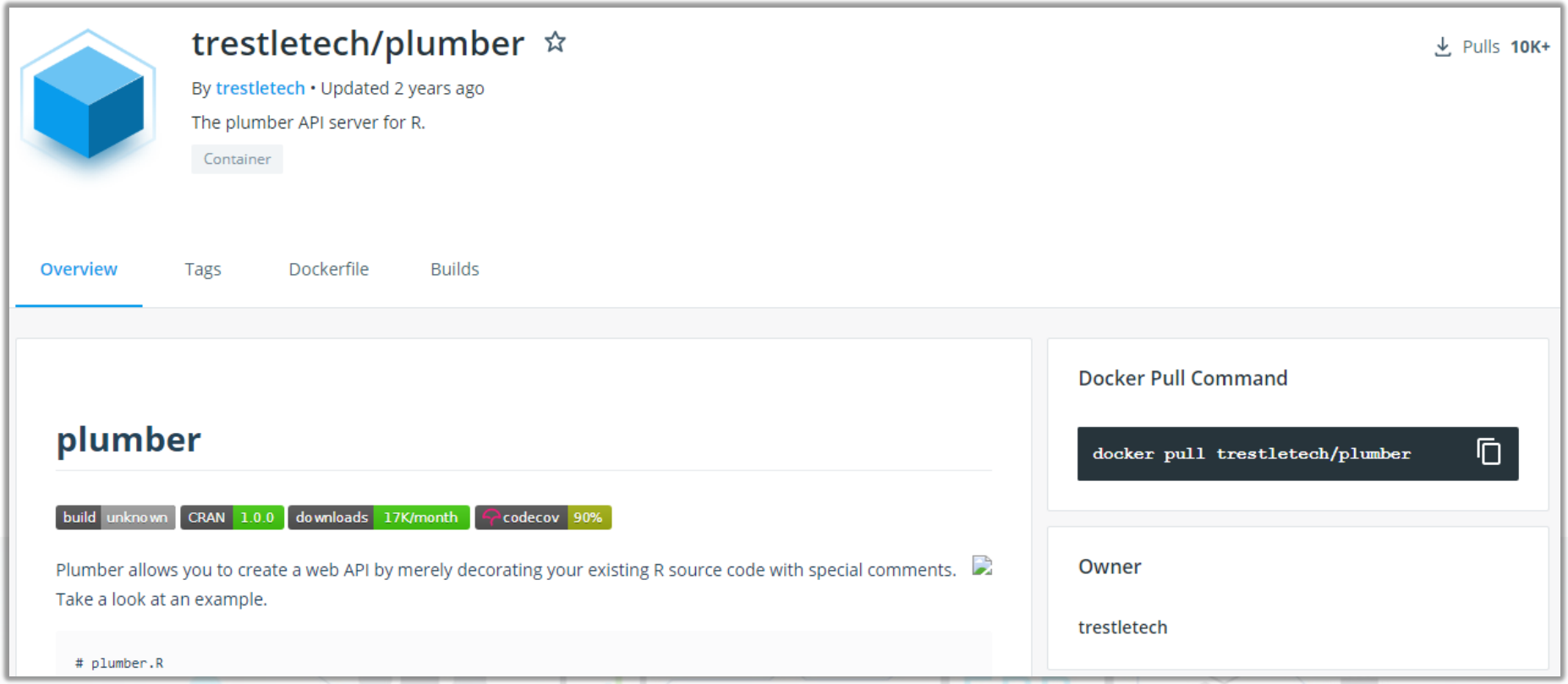
Change Image With Docker File

A docker file can be used when we need to create an image from another one and add packages or execute commands that we want to be already present in the first image we will use

For example, for the Snap4city platform we use an image based on the trestletech/plumber image to which we add the packages we are interested in for our scripts.

trestletech/plumber is a docker image that contains R Studio and creates an API on the script that is passed to it.

Change Image With Docker File



docker pull trestletech/plumber

Change Image With Docker File

```
FROM trestletech/plumber:latest

RUN echo 'deb http://deb.debian.org/debian bullseye main' > /etc/apt/sources.list
RUN apt-get update
RUN apt-get upgrade -y
RUN apt-get install -y apt-utils gdal-bin proj-bin libgdal-dev libproj-dev libxml2-dev
libproj-dev libnetcdf-dev libudunits2-dev libmariadbclient-dev unixodbc unixodbc-dev
libssl-dev libjemalloc-dev

ENV LD_PRELOAD /usr/lib/x86_64-linux-gnu/libjemalloc.so

RUN R -e "install.packages('RODBC')"
RUN R -e "install.packages('RMySQL')"
RUN R -e "install.packages('RCurl')"
RUN R -e "install.packages('RNetCDF')"

RUN R -e "install.packages('rjson')"
RUN R -e "install.packages('jsonlite')"
RUN R -e "install.packages('httr')"
RUN R -e "install.packages('xml2')"

RUN R -e "install.packages('ggplot2')"
RUN R -e "install.packages('dplyr')"
RUN R -e "install.packages('ggalt')"

RUN R -e "install.packages('reshape2')"
RUN R -e "install.packages('scales')"
RUN R -e "install.packages('stringr')"
```

Change Image With Docker File

- *With FROM you indicate which image you should start from*

```
FROM trestletech/plumber:latest
```

- *Let's update the image packages and add the libraries we need for the packages we will use on R Studio*

```
RUN echo 'deb http://deb.debian.org/debian bullseye main' > /etc/apt/sources.list
RUN apt-get update
RUN apt-get upgrade -y
RUN apt-get install -y apt-utils gdal-bin proj-bin libgdal-dev libproj-dev libxml2-dev
libproj-dev libnetcdf-dev libudunits2-dev libmariadbclient-dev unixodbc unixodbc-dev
libssl-dev libjemalloc-dev
```

- *We install the necessary packages within R*

```
RUN R -e "install.packages('RODBC')"
RUN R -e "install.packages('RMySQL')"
RUN R -e "install.packages('RCurl')"
RUN R -e "install.packages('RNetCDF')"
```

Change Image With Docker File

docker build --no-cache -t snap4city-plumber:v8 -f DockerFile .

docker save snap4city-plumber:v8 > snap4city-plumber-v8.tar

gzip < snap4city-plumber-v8.tar > snap4city-plumber-v8.tgz

Docker Compose

Compose is a tool for defining and running multi-container Docker applications.

With Compose, you use a YAML file to configure your application's services.

Then, with a single command, you create and start all the services from your configuration.

Using Compose is basically a three-step process:

1. *Define your app's environment with a Dockerfile so it can be reproduced anywhere.*
2. *Define the services that make up your app in docker-compose.yml so they can be run together in an isolated environment.*
3. *Run docker-compose up and Compose starts and runs your entire app.*

https://docs.docker.com/compose/

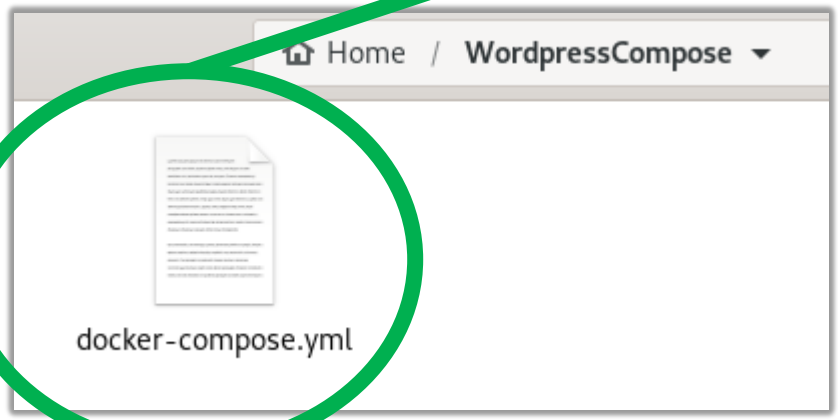
Docker Compose

A `docker-compose.yml` looks like this:

```
version: "3.8"
services:
  web:
    build: .
    ports:
      - "5000:5000"
    volumes:
      - .:/code
      - logvolume01:/var/log
    links:
      - redis
  redis:
    image: redis
volumes:
  logvolume01: {}
```

https://docs.docker.com/compose/

Docker Compose Example




```
version: '3.3'

services:
   db:
     image: mysql:5.7
     volumes:
       - db_data:/var/lib/mysql
     restart: always
     environment:
       MYSQL_ROOT_PASSWORD: somewordpress
       MYSQL_DATABASE: wordpress
       MYSQL_USER: wordpress
       MYSQL_PASSWORD: wordpress

   wordpress:
     depends_on:
       - db
     image: wordpress:latest
     ports:
       - "8000:80"
     restart: always
     environment:
       WORDPRESS_DB_HOST: db:3306
       WORDPRESS_DB_USER: wordpress
       WORDPRESS_DB_PASSWORD: wordpress
       WORDPRESS_DB_NAME: wordpress
volumes:
    db_data: {}
```

https://docs.docker.com/compose/wordpress

Docker Compose Example

- *docker-compose up -d*

```
badii@debian9:~/WordpressCompose$ docker-compose up -d
bash: docker-compose: comando non trovato
badii@debian9:~/WordpressCompose$ su
Password:
root@debian9:/home/badii/WordpressCompose# sudo curl -L "https://github.com/docker/compose/releases/
download/1.27.4/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   651  100   651    0     0   5046      0 --:--:-- --:--:-- --:--:--  5046
100 11.6M  100 11.6M    0     0  4151k      0  0:00:02  0:00:02 --:--:-- 4754k
root@debian9:/home/badii/WordpressCompose# sudo chmod +x /usr/local/bin/docker-compose
root@debian9:/home/badii/WordpressCompose#  docker-compose --version
docker-compose version 1.27.4, build 40524192
root@debian9:/home/badii/WordpressCompose#
```

- *https://docs.docker.com/compose/install/*

sudo curl -L "https://github.com/docker/compose/releases/download/1.27.4/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

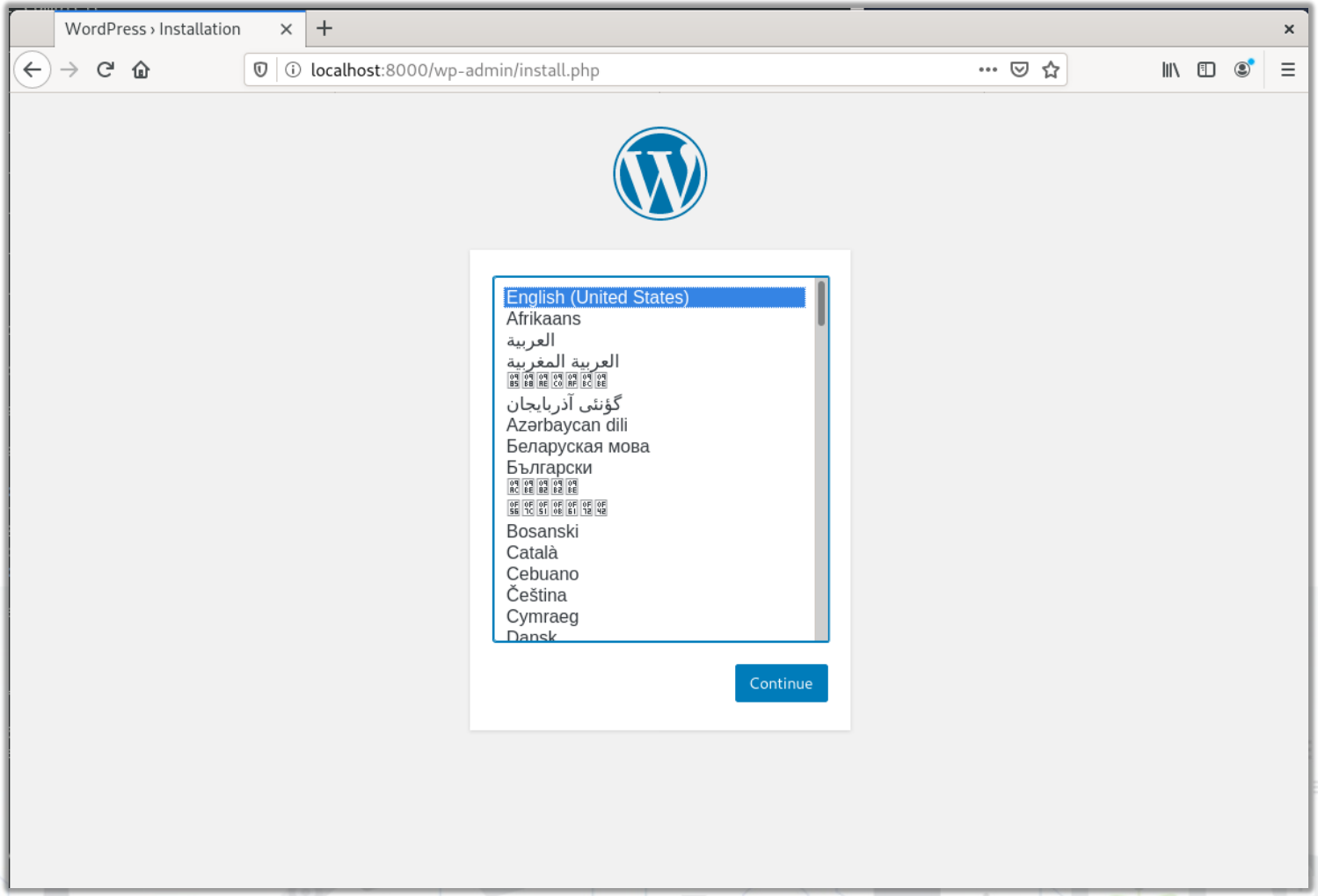
Docker Compose Example

```
badii@debian9:~/WordpressCompose$ sudo docker-compose up -d
[sudo] password di badii:
Creating network "wordpresscompose_default" with the default driver
Creating volume "wordpresscompose_db_data" with default driver
Pulling db (mysql:5.7)...
5.7: Pulling from library/mysql
bb79b6b2107f: Pull complete
49e22f6fb9f7: Pull complete
842b1255668c: Pull complete
Digest: sha256:4d2b34e99c14edb99cdd95ddad4d9aa7ea3f2c4405ff0c3509a29dc40bcb10ef
Status: Downloaded newer image for mysql:5.7
Pulling wordpress (wordpress:latest)...
latest: Pulling from library/wordpress
bb79b6b2107f: Already exists
80f7a64e4b25: Pull complete
da391f3e81f0: Pull complete
8199ae3052e1: Pull complete
284fd0f314b2: Pull complete
f38db365cd8a: Pull complete
Digest: sha256:20bffad04c9c3e696b3c6fbc48d769c5948718b57af8c9457d9a0f28b5066b4b
Status: Downloaded newer image for wordpress:latest
Creating wordpresscompose_db_1 ... done
Creating wordpresscompose_wordpress_1 ... done
badii@debian9:~/WordpressCompose$
```

Docker Compose Example

```
root@debian9:/home/badii/WordpressCompose# docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS
    PORTS                    NAMES
b232a6792b4f        wordpress:latest    "docker-entrypoint.s…"   19 minutes ago      Up 19 minutes
    0.0.0.0:8000->80/tcp   wordpresscompose_wordpress_1
66d4bc227553        mysql:5.7           "docker-entrypoint.s…"   19 minutes ago      Up 19 minutes
    3306/tcp, 33060/tcp    wordpresscompose_db_1
root@debian9:/home/badii/WordpressCompose# docker images
REPOSITORY                          TAG                 IMAGE ID            CREATED             SIZE
wordpress                           latest              6edecd0f5c75        6 days ago          546MB
mysql                               5.7                 1b12f2e9257b        2 weeks ago         448MB
ubuntu                              latest              9140108b62dc        6 weeks ago         72.9MB
disitlab/snap4city-nodered-adv      v76                 6ff49c05dc11        4 months ago        1.1GB
snap4city-plumber                   v8                  d30a16ab3a3f        5 months ago        2.84GB
hello-world                         latest              bf756fb1ae65        10 months ago       13.3kB
trestletech/plumber                 latest              f9aa6e6553fb        18 months ago       962MB
root@debian9:/home/badii/WordpressCompose#
```

Docker Compose Example



Docker Compose Example

```
root@debian9:/home/badii/WordpressCompose# docker-compose down
Stopping wordpresscompose_wordpress_1 ... done
Stopping wordpresscompose_db_1        ... done
Removing wordpresscompose_wordpress_1 ... done
Removing wordpresscompose_db_1        ... done
Removing network wordpresscompose_default
```

```
root@debian9:/home/badii/WordpressCompose# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              P
ORTS                NAMES
root@debian9:/home/badii/WordpressCompose# docker images
REPOSITORY                          TAG                 IMAGE ID            CREATED             SIZE
wordpress                           latest              6edecd0f5c75        6 days ago          546MB
mysql                               5.7                 1b12f2e9257b        2 weeks ago         448MB
ubuntu                              latest              9140108b62dc        6 weeks ago         72.9MB
disitlab/snap4city-nodered-adv      v76                 6ff49c05dc11        4 months ago        1.1GB
snap4city-plumber                   v8                  d30a16ab3a3f        5 months ago        2.84GB
hello-world                         latest              bf756fb1ae65        10 months ago       13.3kB
trestletech/plumber                 latest              f9aa6e6553fb        18 months ago       962MB
root@debian9:/home/badii/WordpressCompose#
```