

# Deployment di servizi con **Docker**

**Linux Day 2016**

22 Ottobre

**Gaetano Carlucci**

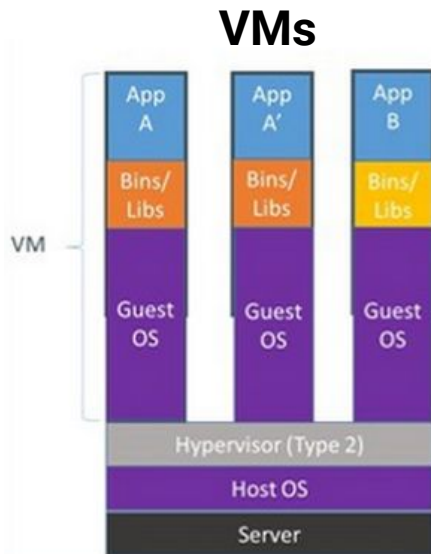
gaetano.carlucci@poliba.it

# DEVOPS

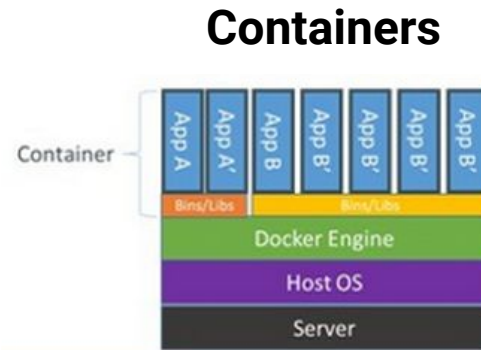


Riduzione costi e time to market ad alta efficienza

# COSA E' UN CONTAINER?



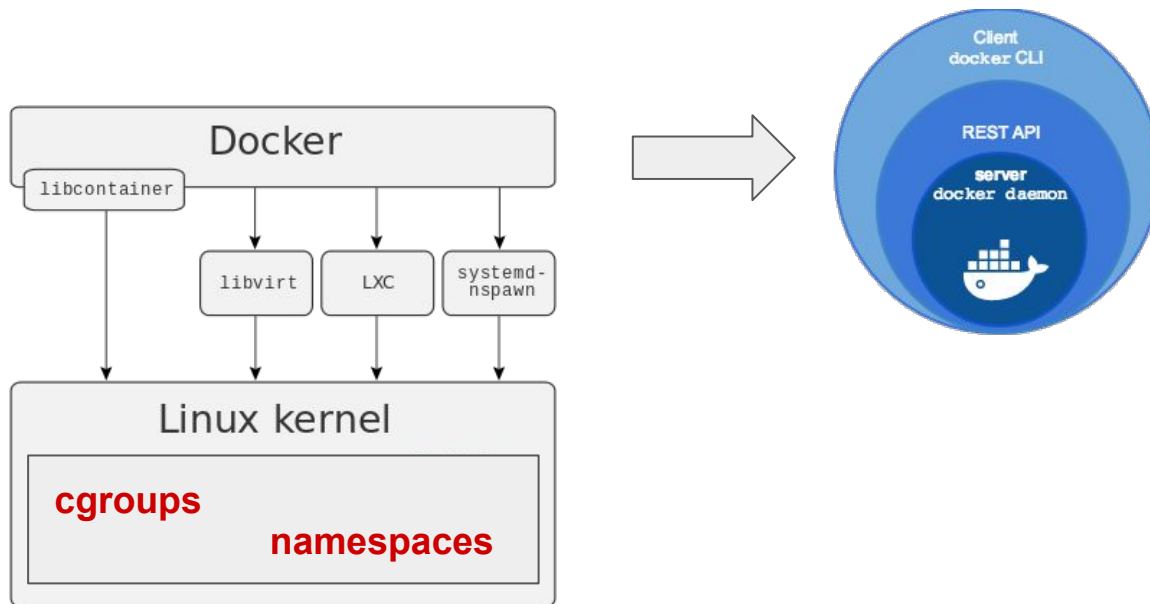
**Fornisce una astrazione dell' HW e isolamento degli OS**



**Fornisce una astrazione del SO e isolamento dei processi (lightweight VMs)**

**Soluzioni complementari: un container può essere creato in una VM**

# COSA E' DOCKER?



**Docker Engine è un demone che utilizza le funzionalità del kernel Linux come namespaces e cgroups per creare container al di sopra del SO.**

# PREREQUISITI (UBUNTU)

## Distribuzione

- Ubuntu Xenial 16.04
- Ubuntu Trusty 14.04
- Ubuntu Precise 12.04

64-bit installation

## Kernel

Almeno 3.10

## APT sources

Set APT to use packages from the Docker repository

<https://docs.docker.com/engine/installation/linux/ubuntu/linux/>

```
$sudo apt-get install docker-engine
```

```
$sudo service docker start
```

**DEMO**

# WORKFLOW

BUILD AN IMAGE



STORE THE IMAGE



PULL THE IMAGE



RUN THE CONTAINER

```
$docker build -t gaetano/nginx .
```



```
$docker push gaetano/nginx
```



```
$docker pull gaetano/nginx
```

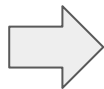


```
$docker run gaetano/nginx
```



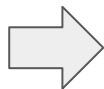
# BUILD AN IMAGE

CREARE DIRECTORY

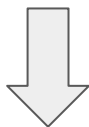


```
.  
├── Dockerfile  
└── index.html
```

CREARE DOCKERFILE



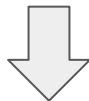
```
FROM ubuntu:14.04  
  
RUN apt-get update \  
    && apt-get -y -q install nginx \  
    && apt-get clean \  
    && rm -rf /var/lib/apt/lists/*  
  
COPY index.html /usr/share/nginx/html/index.html  
  
ENTRYPOINT ["/usr/sbin/nginx"]  
CMD ["-g", "daemon off;"]
```



```
$docker build -t gaetano/nginx .
```

## Lists all local images

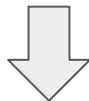
```
$docker images
```



REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
gaetano/nginx	latest	245547114011	2 minutes ago	206.1 MB
Ubuntu	14.04	f2d8ce9fa988	3 weeks ago	187.9 MB

## Removes local images

```
$docker rmi gaetano/nginx
```



REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
Ubuntu	14.04	f2d8ce9fa988	3 weeks ago	187.9 MB

## Analyzes build history of an image

```
$docker history gaetano/nginx
```

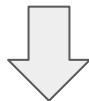
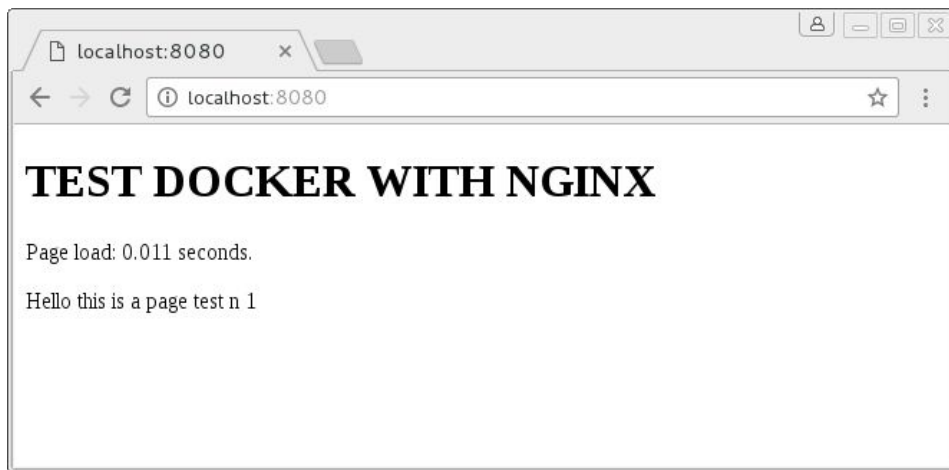
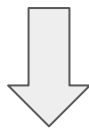


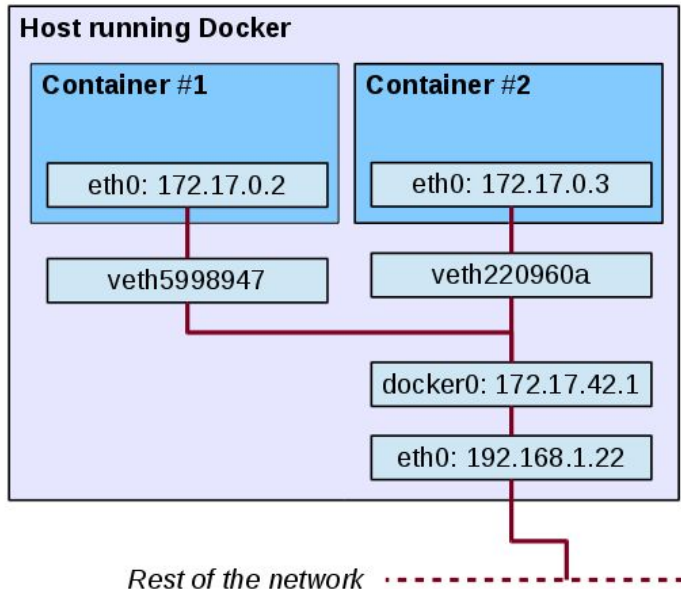
IMAGE	CREATED	CREATED BY	SIZE
Bc96272b5a15	About an hour ago	/bin/sh -c #(nop) ENTRYPOINT ["/usr/sbin/ngi	0 B
8e5b59579841	About an hour ago	/bin/sh -c #(nop) COPY file:e1671ab5ff12effdb	508 B
e71a81e1f0d0	About an hour ago	/bin/sh -c apt-get update && apt-get -y -	18.15 MB

# RUN CONTAINER

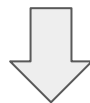
```
$docker run -d -p 8080:80 --name=my_web_server gaetano/nginx
```



# DOCKER NETWORK



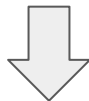
```
$iptables -t nat -L -n
```



```
Chain DOCKER (2 references)
target    prot opt source      destination
RETURN    all  --  0.0.0.0/0   0.0.0.0/0
DNAT      tcp  --  0.0.0.0/0   0.0.0.0/0   tcp dpt:8080 to:172.17.0.2:80
```

## Display the running processes of a container

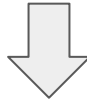
```
$docker top
```



UID	PID	PPID	C	STIME	TTY	TIME	CMD
Root	24084	24069	0	10:54	?	00:00:00	nginx: master process /usr/sbin/nginx
Www-data	24109	24084	0	10:54	?	00:00:00	nginx: worker process

Lists all running containers

```
$docker ps [-a]
```

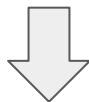


CONTAINER ID	IMAGE	COMMAND	STATUS	PORTS	NAMES
9e59303600e9	gaetano/nginx	"/usr/sbin/nginx"	Up About an hour	80/tc	my_web_server



## Enters into a container

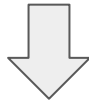
```
$docker exec -it my_web_server /bin/bash
```



```
PID  TTY  STAT      TIME COMMAND
1    ?    Ss        0:00 nginx: master process /usr/sbin/nginx -g daemon off;
7    ?    S         0:00 nginx: worker process
53   ?    Ss        0:00 /bin/bash
72   ?    R+        0:00 ps -ax
```

## Inspects a container

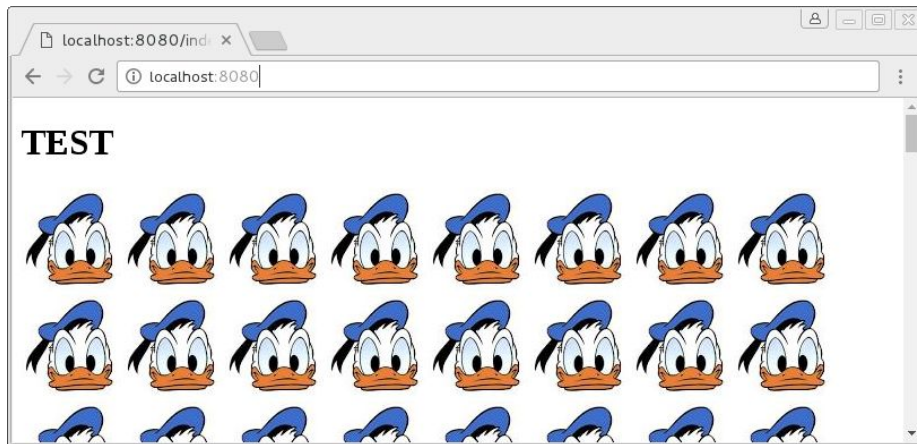
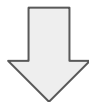
```
$ docker inspect -f "{{ .State.Status }}" my_web_server
```



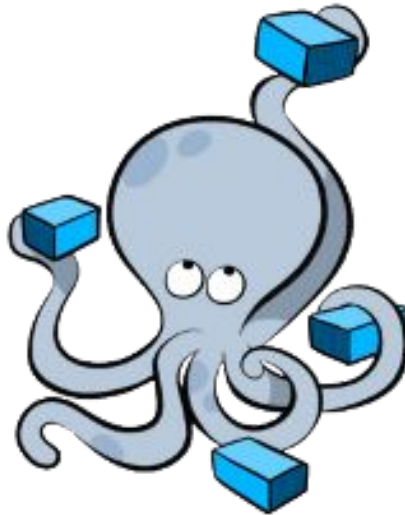
```
running
```

## External volumes:

```
$docker run -d -p 8080:80 --name=my_web_server \  
-v my_web_site_dir:/usr/share/nginx/html/ \  
gaetano/nginx
```

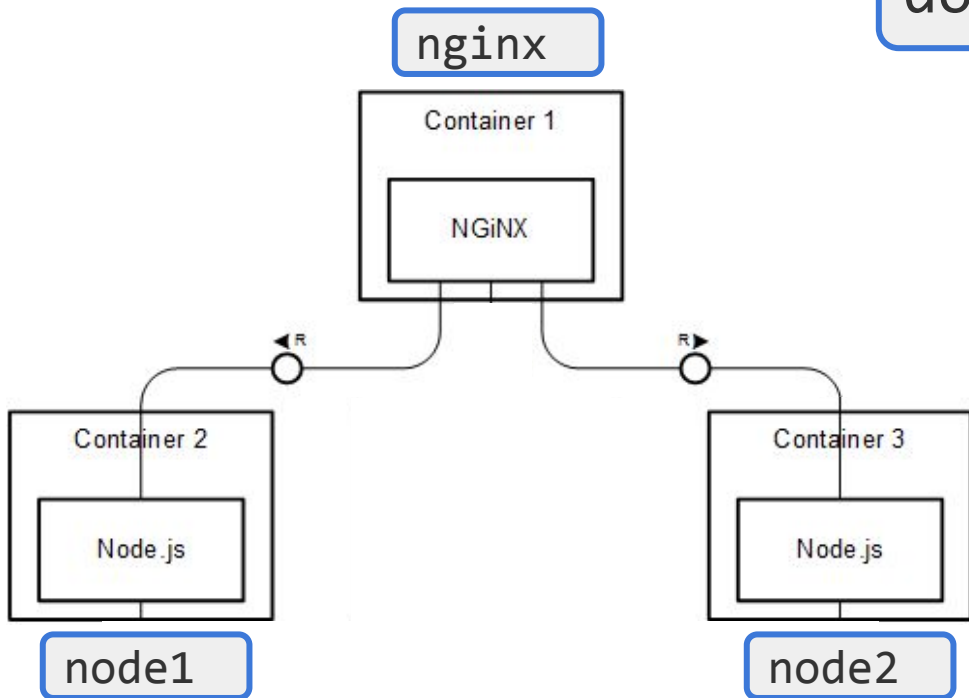
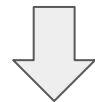


# DOCKER COMPOSE



# DOCKER COMPOSE

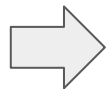
docker-compose.yml



```
nginx:
  image: gaetano/nginx2
  links:
    - node1
    - node2
  ports:
    - "5555:80"
node1:
  image: gaetano/node1
  ports:
    - "80"
node2:
  image: gaetano/node2
  ports:
    - "80"
```

# RUN COMPOSE

```
.  
├── docker-compose.yml  
└── other files you need
```



```
$docker-compose up -d
```

## DEMO



[docs.docker.com](https://docs.docker.com)