Breizh C@amp

Jean-Tiare Le Bigot

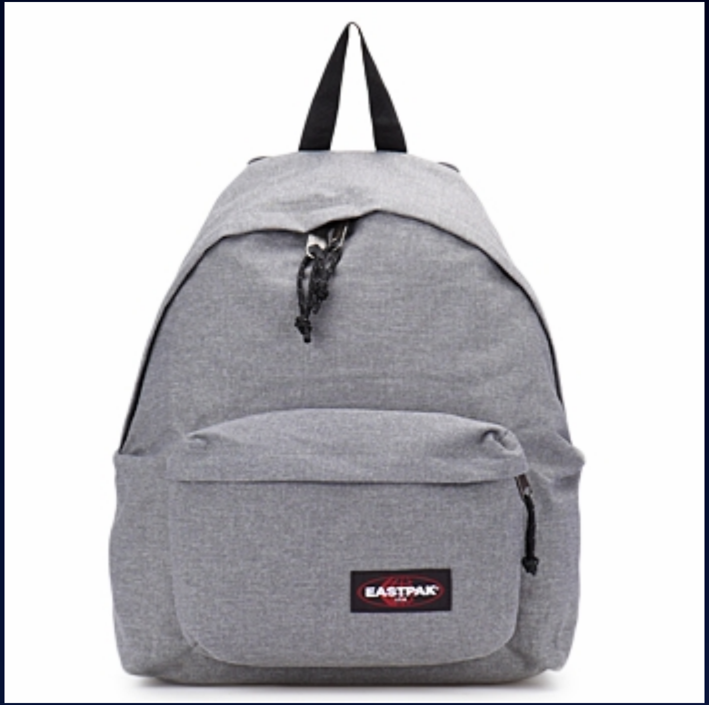
Premiers pas avec Docker

docker

OVH.com
Innovation is Freedom

Les containers sont partout

En voici un.



On y trouve parfois des serveurs…

…des étudiants…

…ou même toute une vie.

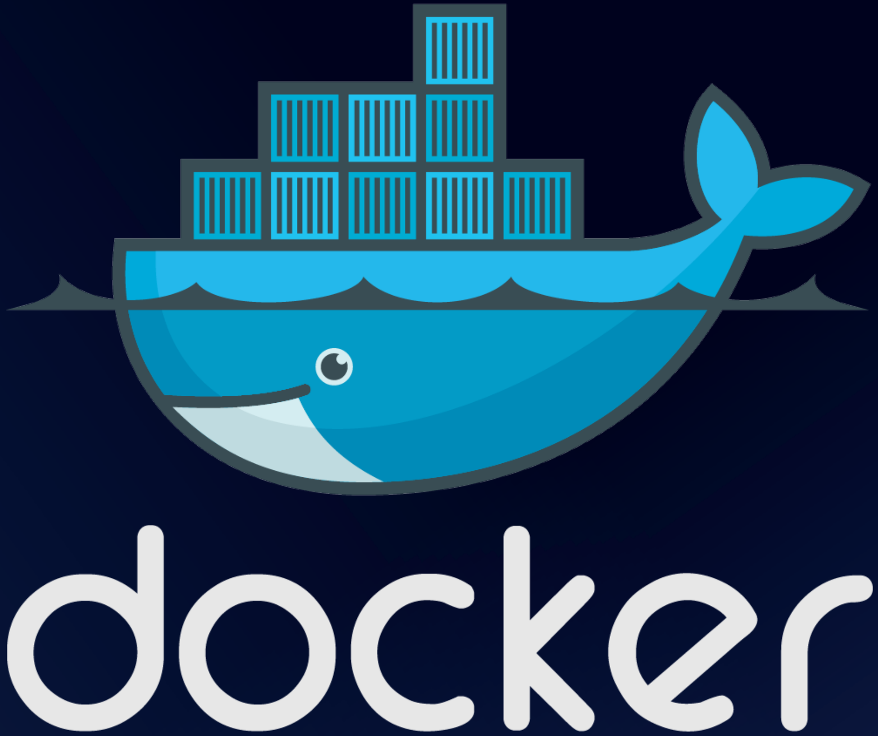
En fait, un container, c’est :

« quelque chose » où mettre *« n’importe quoi »*,
et qui se comportera toujours pareil, partout.

Quid de
Linux / Docker?

Quid de Linux / Docker?

- Kernel: Namespaces
- Kernel: Cgroups
- User-Land: LXC
- User-Land: **Docker** ← sans doctorat
- ...


docker

Les Dockers sont

Des briques

petites,

standards,

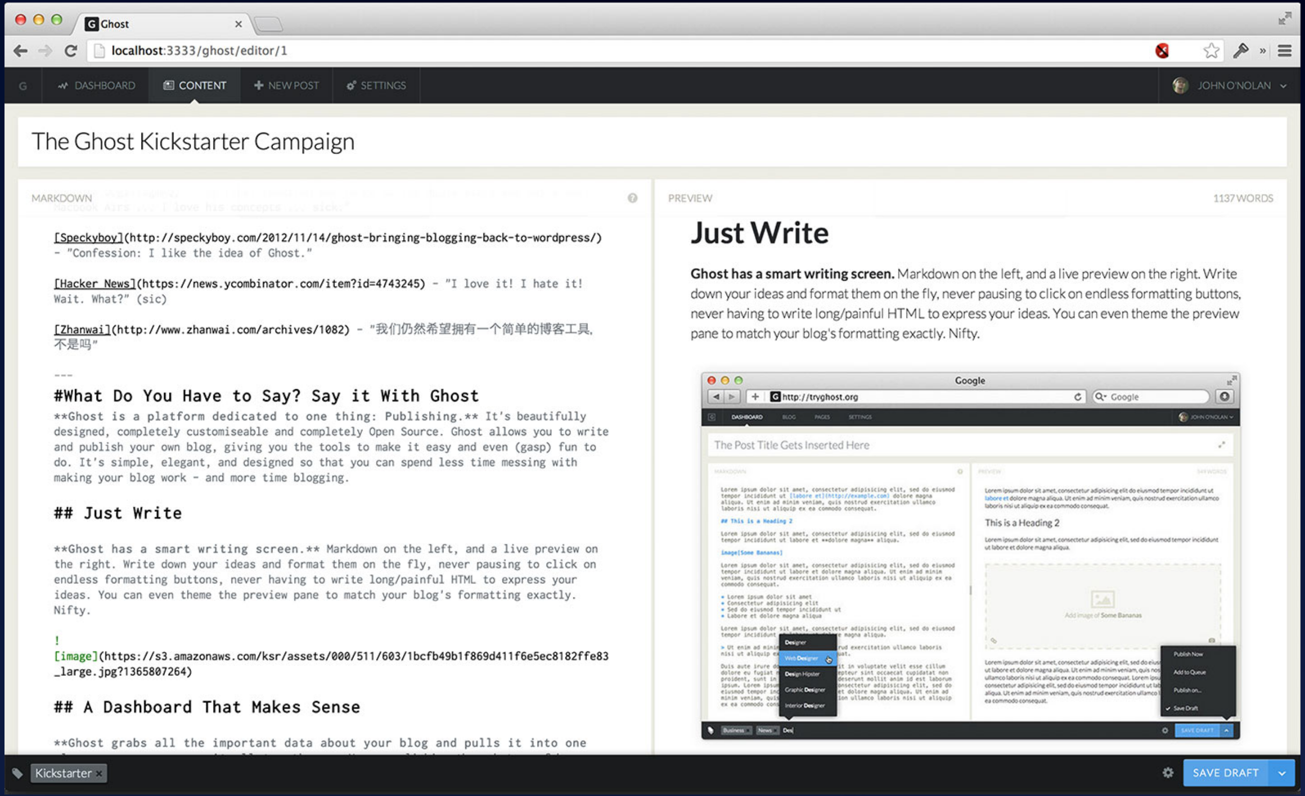
réutilisables,

Docker est parfait pour

Développement
Partage de briques
Composition d’applications

Si ça fonctionne en dev...
... ça fonctionne en prod !

Exemple : Ghost



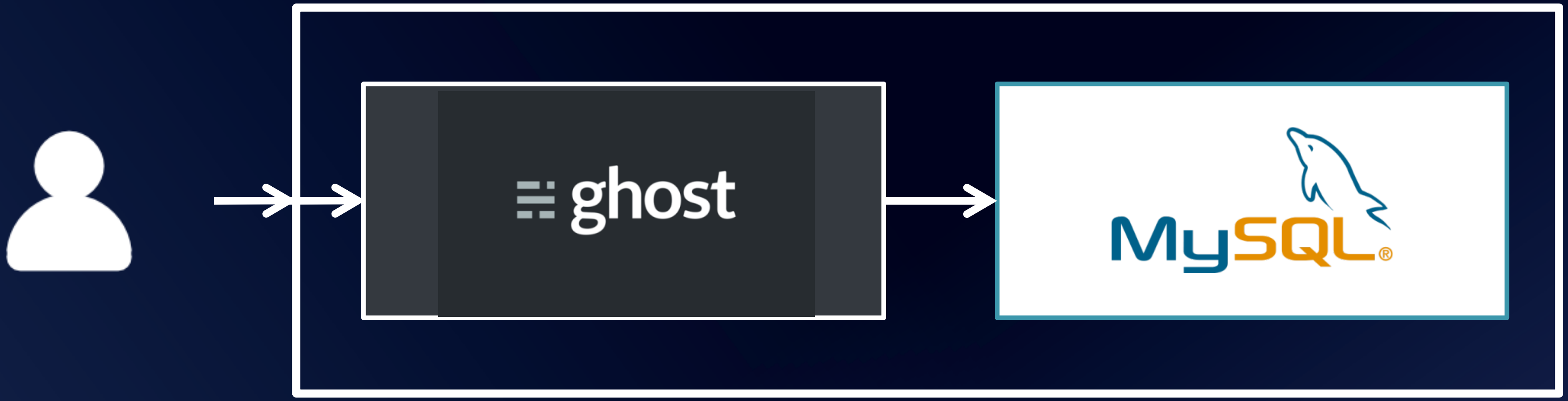
The Ghost Kickstarter Campaign

```
[Speckyboy](http://speckyboy.com/2012/11/14/ghost-bringing-blogging-back-to-wordpress/)
- "Confession: I like the idea of Ghost."

[Hacker News](https://news.ycombinator.com/item?id=4743245) - "I love it! I hate it!
Wait. What?" (sic)

[Zhanwai](http://www.zhanwai.com/archives/1082) - "我们仍然希望拥有一个简单的博客工具,
不是吗"

---

#What Do You Have to Say? Say it With Ghost
**Ghost is a platform dedicated to one thing: Publishing.** It's beautifully
designed, completely customiseable and completely Open Source. Ghost allows you to write
and publish your own blog, giving you the tools to make it easy and even (gasp) fun to
do. It's simple, elegant, and designed so that you can spend less time messing with
making your blog work - and more time blogging.

## Just Write

**Ghost has a smart writing screen.** Markdown on the left, and a live preview on
the right. Write down your ideas and format them on the fly, never pausing to click on
endless formatting buttons, never having to write long/painful HTML to express your
ideas. You can even theme the preview pane to match your blog's formatting exactly.
Nifty.

!
[image](https://s3.amazonaws.com/ksr/assets/000/511/603/1bcfb49b1f869d411f6e5ec8182ffe83
_large.jpg?1365807264)

## A Dashboard That Makes Sense

**Ghost grabs all the important data about your blog and pulls it into one
```

Just Write

Ghost has a smart writing screen. Markdown on the left, and a live preview on the right. Write down your ideas and format them on the fly, never pausing to click on endless formatting buttons, never having to write long/painful HTML to express your ideas. You can even theme the preview pane to match your blog's formatting exactly. Nifty.

Exemple : Ghost



VM / VPS / SD / …

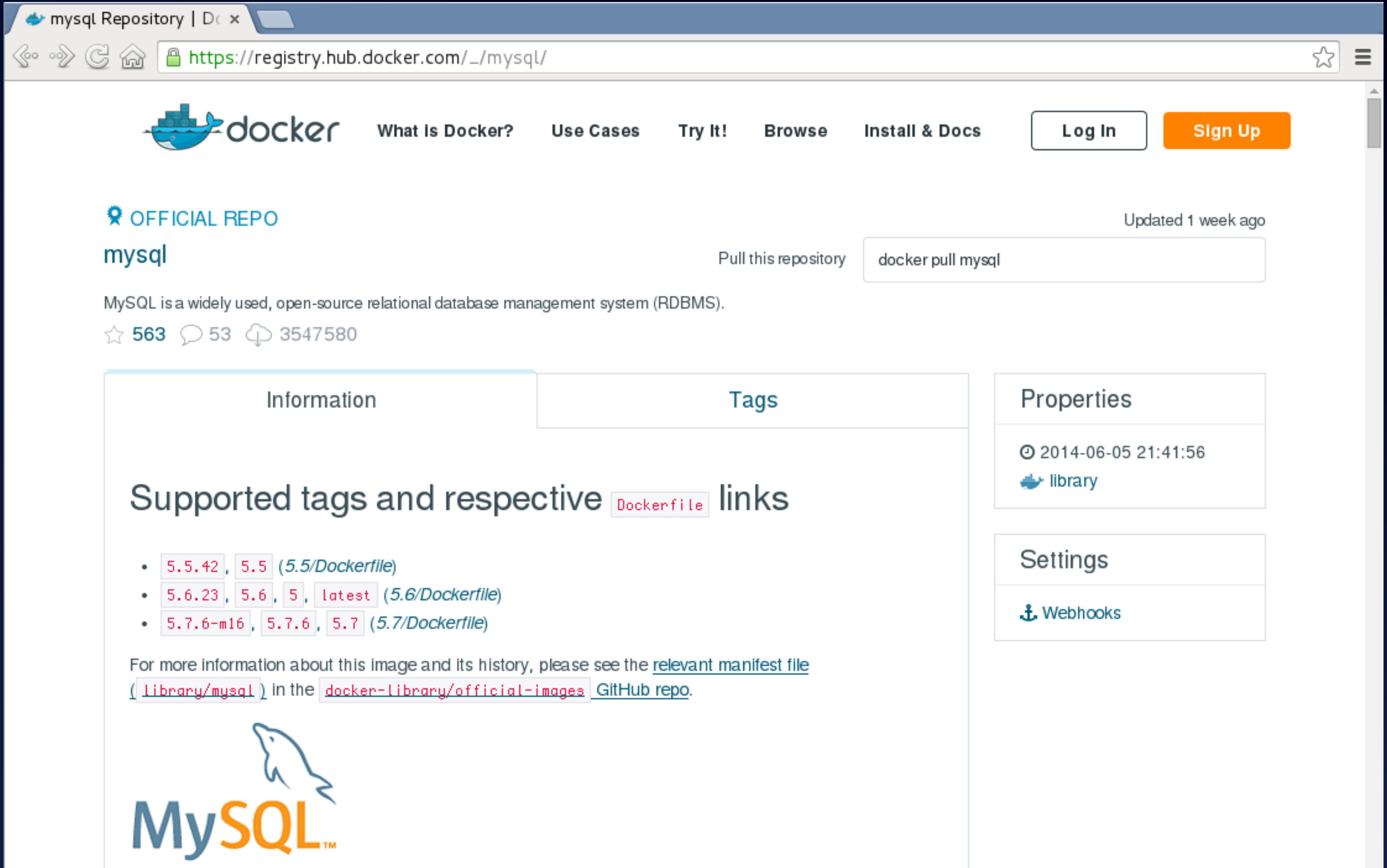
Installation de Docker

Avec Ubuntu 14.04 LTS:

```
# apt-get install docker.io
```

https://docs.docker.com/installation

MySQL



OFFICIAL REPO

mysql

Updated 1 week ago

Pull this repository

```
docker pull mysql
```

MySQL is a widely used, open-source relational database management system (RDBMS).

☆ 563 53 3547580

Supported tags and respective `Dockerfile` links

- `5.5.42`, `5.5` (*5.5/Dockerfile*)
- `5.6.23`, `5.6`, `5`, `latest` (*5.6/Dockerfile*)
- `5.7.6-m16`, `5.7.6`, `5.7` (*5.7/Dockerfile*)

For more information about this image and its history, please see the relevant manifest file (`library/mysql`) in the `docker-library/official-images` GitHub repo.

Properties

2014-06-05 21:41:56

library

Settings

Webhooks

Container MySQL

```
# docker run -d \
    --name ghostmysql \
    --env MYSQL_ROOT_PASSWORD=motdepasse \
    --env MYSQL_DATABASE=ghost \
    --volume /host-data-dir:/var/lib/mysql \
    mysql:latest
```

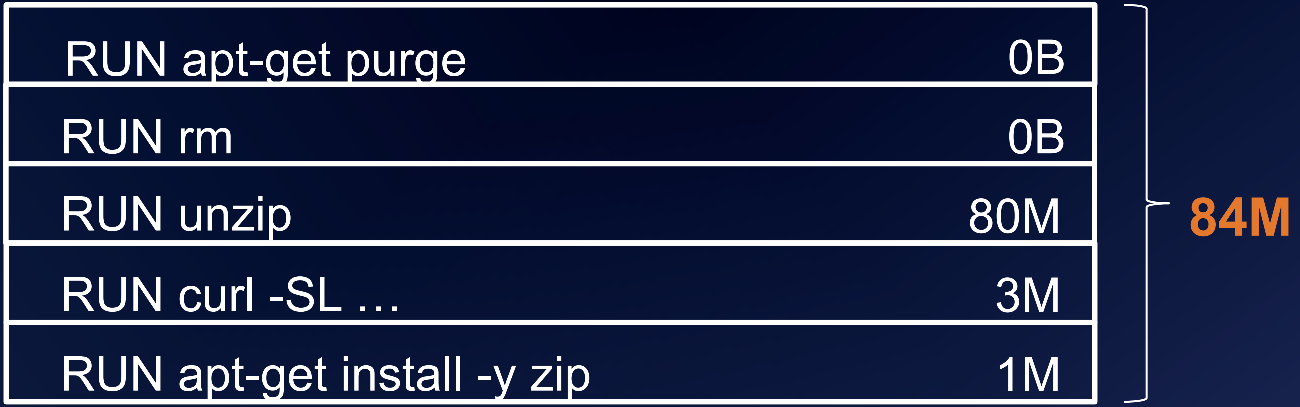
1 minute… Et les Dockerfiles ?

- Jeu d’instructions :
  FROM, RUN, CMD, ENV, VOLUME…
- 1instruction ↔ 1 “layer”:

RUN *[Ghost installation]*	81M
ENV GHOST_VERSION=0.5.10	0B
RUN *[Node.js installation]*	22M
ADD *[Base image – Debian]*	120M

Bonne pratique #1

```
RUN apt-get install -y unzip
RUN curl -SL https://ghost.org/zip/ghost-latest.zip
RUN unzip ghost-latest.zip -d /ghost
RUN rm -f ghost-latest.zip
RUN apt-get purge zip
```



RUN apt-get purge	0B
RUN rm	0B
RUN unzip	80M
RUN curl -SL …	3M
RUN apt-get install -y zip	1M

84M

Dockerfile best practice #1

```
RUN apt-get install –y unzip && \
    curl –SL https://ghost.org/zip/ghost-latest.zip && \
    unzip ghost-latest.zip –d /ghost && \
    rm –f ghost-latest.zip && \
    apt-get purge zip
```

RUN 80M

Dockerisation de Ghost

- Ne pas réinventer la roue (pas le temps)
- Lien MySQL standard
- Aucune donnée dans le container
  - Container : application, thème(s), etc.
  - Volume : images
  - Base de données : textes

... Dockerisation...

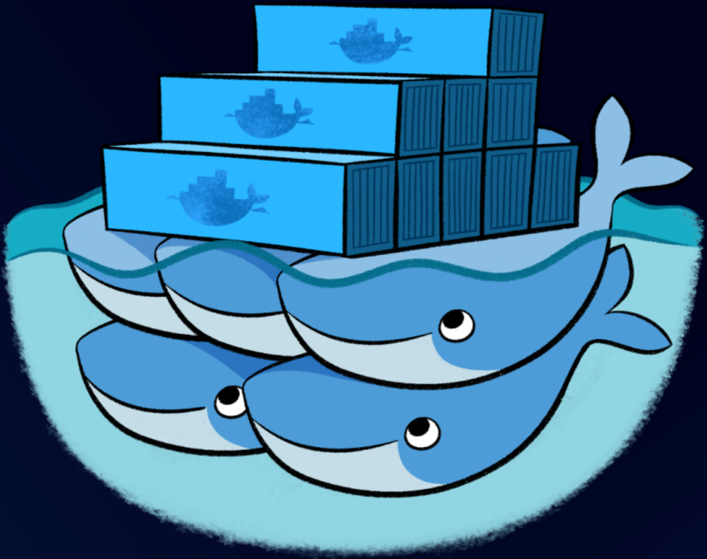
```
FROM node:0.12-slim
MAINTAINER Jean-Tiare Le Bigot <jean-tiare.le-bigot@ovh.net>

RUN useradd ghost --home /ghost --create-home
WORKDIR /ghost


RUN DEBIAN_FRONTEND=noninteractive apt-get update && \
    apt-get install -y unzip curl && \
    curl -L https://ghost.org/zip/ghost-latest.zip -o /ghost.zip && \
    unzip /ghost.zip -d /ghost && rm /ghost.zip && \
    apt-get clean && apt-get purge -y unzip curl ca-certificates && \
    apt-get autoremove -y && \
    npm install --production && \
    chown -R ghost:ghost /ghost

USER ghost
```

... Configuration...

```
module.exports = {
    production: {
        url: process.env.URL,
        server: {
            host: ‘0.0.0.0’,
            port: ‘2369’
        },
        database: {
            client: ‘mysql’,
            connection: {
                host: ‘mysql’,
                user: ‘root’,
                password:
process.env.MYSQL_ENV_MYSQL_ROOT_PASSWORD,
                database: process.env.MYSQL_ENV_MYSQL_DATABASE,
                charset: ‘utf8’
            }
```

... Compilation, exécution.

```
# docker build -t ghost .

# docker run -d \
    --name ghost \
    --link ghostmysql:mysql \
    --env URL=http://owt.example.com \
    --volume /host-images-dir:/ghost/content/images \
    --publish 80:2369 \
    ghost
```



Docker / Écosystème

Docker Compose

La même chose en 1 seule commande

```
$ pip install docker-compose
$ docker-compose up
```

https://docs.docker.com/compose

Docker Compose

La même chose en 1 seule commande

```
mysql:
    image: mysql
    volumes:
        - /host-mysql-dir:/var/lib/mysql
    environment:
        MYSQL_ROOT_PASSWORD: password
        MYSQL_DATABASE: ghost
ghost:
    build: .
    links:
        - mysql:mysql
    volumes:
        - /host-images-dir:/ghost/content/images
    environment:
        URL: http://owt.example.com
    ports:
        “80:2369”
```

Docker Machine

Des machines Docker, partout

```
$ wget -O docker-machine \
https://github.com/docker/machine/releases/download/v0.3.0-rc1/docker-machine_linux-amd64
$ chmod +x docker-machine

$ docker-machine create breizhcamp \
                 --driver virtualbox
$ eval $(docker-machine env breizhcamp)
```

https://docs.docker.com/machine

Docker Swarm

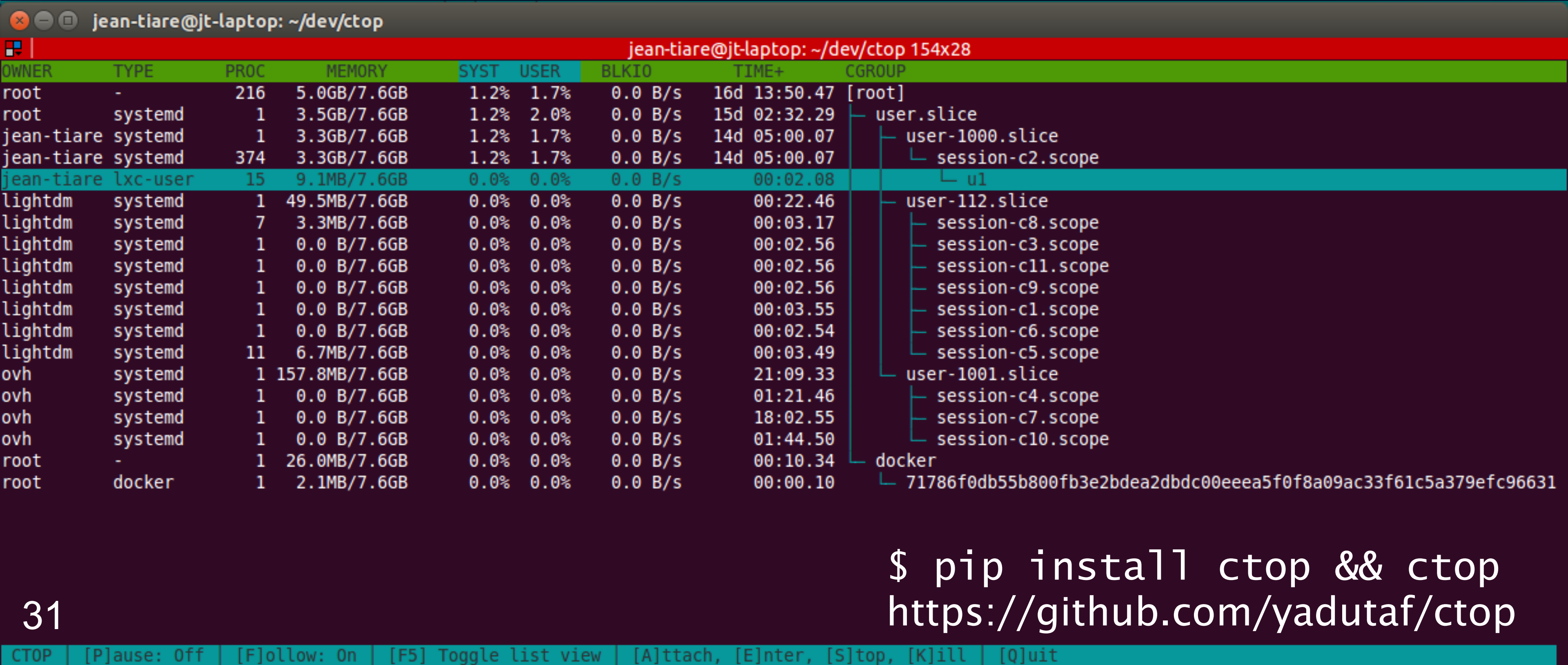
Plein de machines, ensemble

```
$ docker run swarm create
$ docker-machine create breizhcamp-master \
                 --driver virtualbox \
                 --swarm --swarm-master \
                 --swarm-discovery token://<token>
$ eval $(docker-machine env --swarm breizhcamp-master)
```

https://docs.docker.com/swarm

~~Docker~~ Ctop

Mais quel est le c** qui prend tout ?



```
jean-tiare@jt-laptop: ~/dev/ctop 154x28
OWNER      TYPE        PROC    MEMORY       SYST  USER   BLKIO      TIME+        CGROUP
root       -            216   5.0GB/7.6GB   1.2%  1.7%   0.0 B/s  16d 13:50.47 [root]
root       systemd        1   3.5GB/7.6GB   1.2%  2.0%   0.0 B/s  15d 02:32.29  ├─ user.slice
jean-tiare systemd        1   3.3GB/7.6GB   1.2%  1.7%   0.0 B/s  14d 05:00.07  │  ├─ user-1000.slice
jean-tiare systemd      374   3.3GB/7.6GB   1.2%  1.7%   0.0 B/s  14d 05:00.07  │  │  └─ session-c2.scope
jean-tiare lxc-user      15   9.1MB/7.6GB   0.0%  0.0%   0.0 B/s      00:02.08  │  │     └─ u1
lightdm    systemd        1  49.5MB/7.6GB   0.0%  0.0%   0.0 B/s      00:22.46  │  ├─ user-112.slice
lightdm    systemd        7   3.3MB/7.6GB   0.0%  0.0%   0.0 B/s      00:03.17  │  │  ├─ session-c8.scope
lightdm    systemd        1   0.0 B/7.6GB   0.0%  0.0%   0.0 B/s      00:02.56  │  │  ├─ session-c3.scope
lightdm    systemd        1   0.0 B/7.6GB   0.0%  0.0%   0.0 B/s      00:02.56  │  │  ├─ session-c11.scope
lightdm    systemd        1   0.0 B/7.6GB   0.0%  0.0%   0.0 B/s      00:02.56  │  │  ├─ session-c9.scope
lightdm    systemd        1   0.0 B/7.6GB   0.0%  0.0%   0.0 B/s      00:03.55  │  │  ├─ session-c1.scope
lightdm    systemd        1   0.0 B/7.6GB   0.0%  0.0%   0.0 B/s      00:02.54  │  │  ├─ session-c6.scope
lightdm    systemd       11   6.7MB/7.6GB   0.0%  0.0%   0.0 B/s      00:03.49  │  │  └─ session-c5.scope
ovh        systemd        1 157.8MB/7.6GB   0.0%  0.0%   0.0 B/s      21:09.33  │  └─ user-1001.slice
ovh        systemd        1   0.0 B/7.6GB   0.0%  0.0%   0.0 B/s      01:21.46  │     ├─ session-c4.scope
ovh        systemd        1   0.0 B/7.6GB   0.0%  0.0%   0.0 B/s      18:02.55  │     ├─ session-c7.scope
ovh        systemd        1   0.0 B/7.6GB   0.0%  0.0%   0.0 B/s      01:44.50  │     └─ session-c10.scope
root       -              1  26.0MB/7.6GB   0.0%  0.0%   0.0 B/s      00:10.34  └─ docker
root       docker         1   2.1MB/7.6GB   0.0%  0.0%   0.0 B/s      00:00.10     └─ 71786f0db55b800fb3e2bdea2dbdc00eeea5f0f8a09ac33f61c5a379efc96631

 CTOP | [P]ause: Off | [F]ollow: On | [F5] Toggle list view | [A]ttach, [E]nter, [S]top, [K]ill | [Q]uit
```

```
$ pip install ctop && ctop
```

https://github.com/yadutaf/ctop

Docker at OVH / RunAbove

Sailabove: Docker Hosting

1-click apps

1-click data discovery

https://www.runabove.com

Breizh C@amp

Jean-Tiare Le Bigot

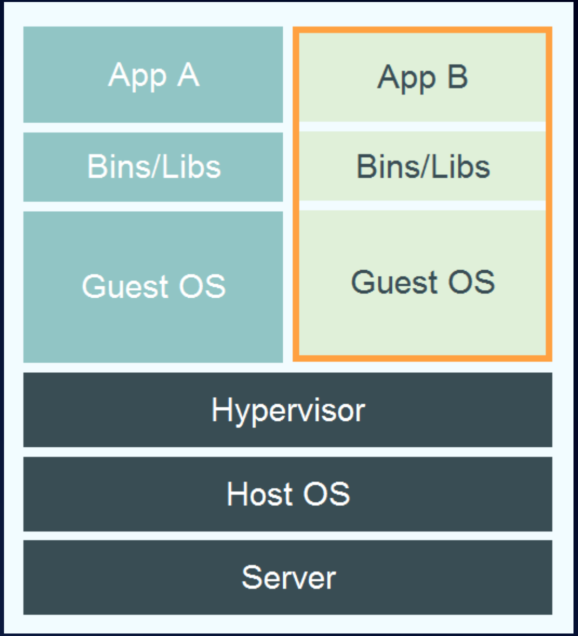
E-mail : jean-tiare.le-bigot@ovh.net
ML docker : docker-subscribe@ml.ovh.net

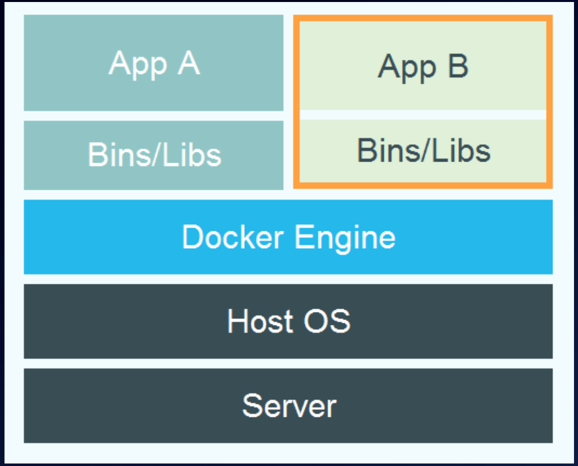
@oyadutaf

docker

OVH.com
Innovation is Freedom

Docker / VM


App A
Bins/Libs
Guest OS
App B
Bins/Libs
Guest OS
Hypervisor
Host OS
Server


App A
Bins/Libs
App B
Bins/Libs
Docker Engine
Host OS
Server