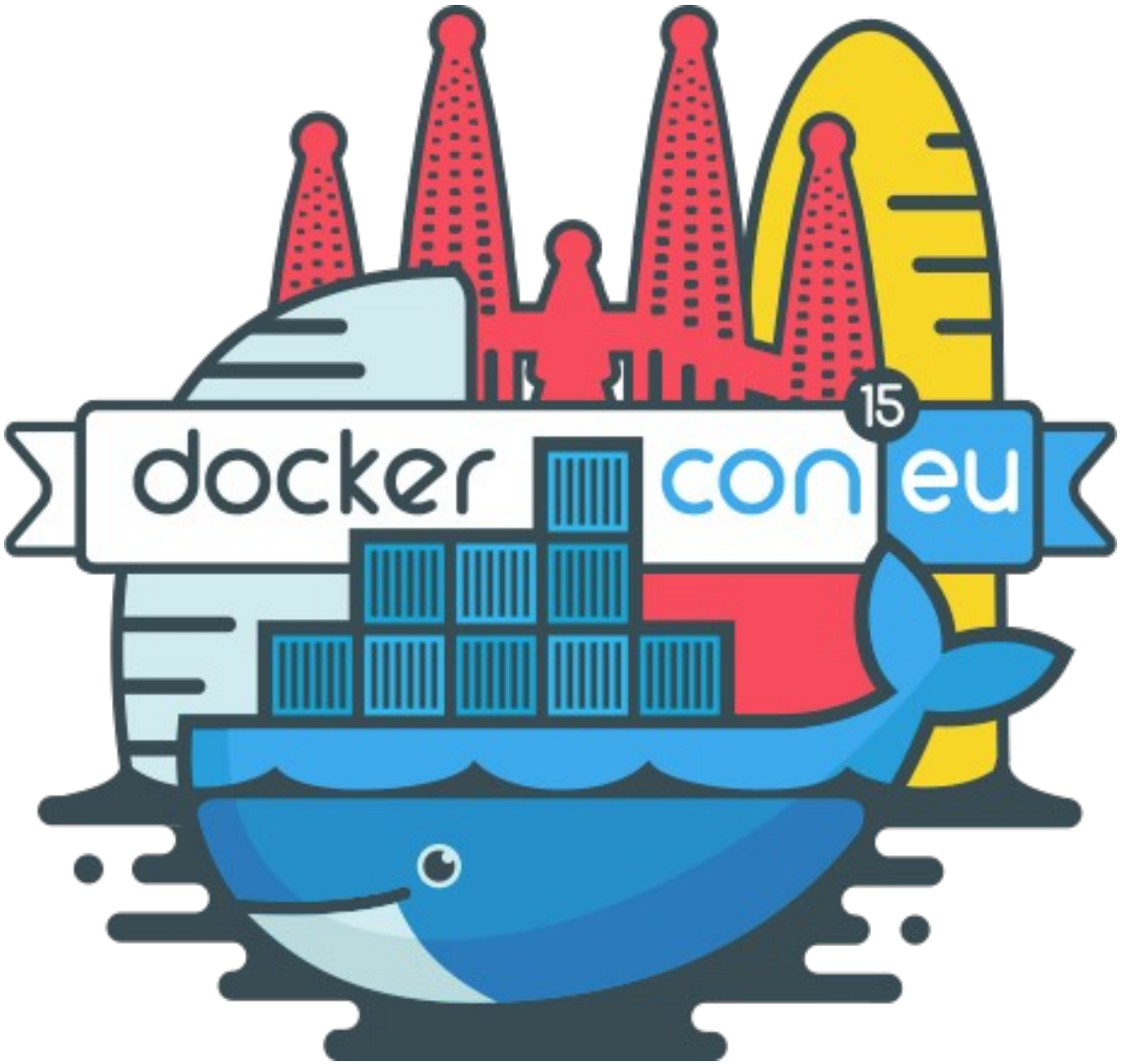
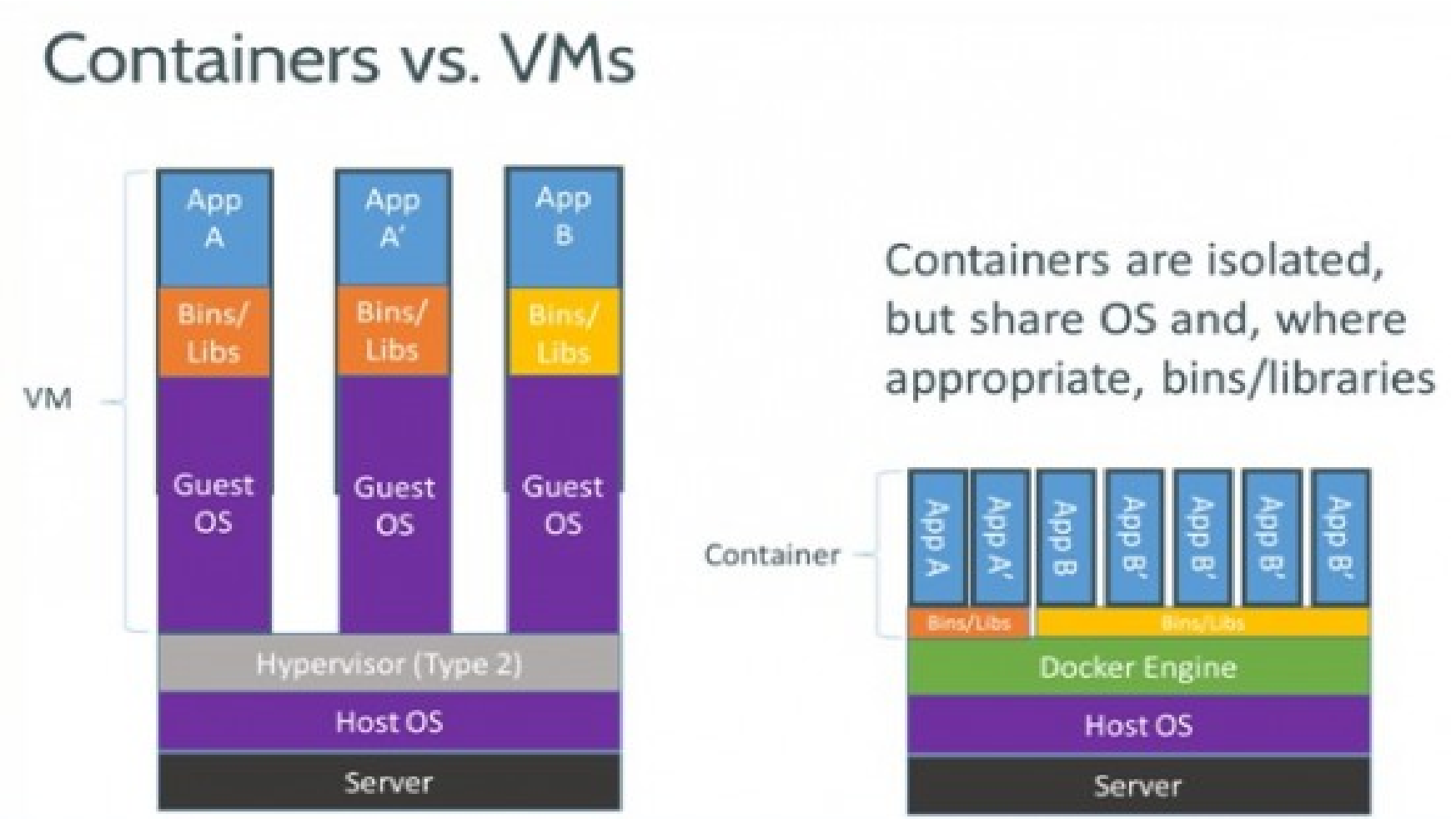

docker
con
15
eu

Qu'est ce Docker ?

- Gestionnaire de containers
- Basé sur LXC (control groups , kernel namespace chroot)
- Ecrit en Go

La philosophie :
On part d'un OS et on y ajoute des applications containerisées.

Différence avec la virtualisation



Avantages par rapport à la virtualisation

* Pas d'OS supplémentaire

* Bien plus léger

* Lancement beaucoup plus rapide

* Migration plus simple car un conteneur a un faible poids

* Très implémenté sur les clouds (Amazon, Google, Microsoft, OVH…)

Avantages côté développement

→ Accélération des déploiements

→ Bascule facile de dév/test en prod

Avantages côté production

→ Facilité de montée en charge

→ Boot très rapide du container

Système de build

- Rapide (cache)
- Fiable et reproductible
- Facile à apprendre (proche du shell)

Les limites de docker

- Un container créé sous linux ne pourra pas fonctionner sous windows
- La sécurité, LXC n'isole pas bien les ressources utilisées par le container. Risque qu'un container empiète sur les ressources d'un autre.
- Pas d'interface de gestion graphique éprouvée

Docker permet de: créer, éditer, publier et exécuter des containers.

Dockerfile : Fichier source qui contient les instructions, éléments à installer, c'est un fichier de configuration.

Image : Compilation d'un fichier Dockerfile pour former une image portable, prête à être déployée (classe en POO)

Container : Exécution d'une image, mise en container d'une image (objet en POO)

Les images de dockers (http://hub.docker.com)

~100 images officielles

→ Distrib. Linux : debian, Ubuntu, CentOS, Fedora…

→ composants : MySQL, PostgreSQL, MongoDB, NGINX…

→ langages : python, ruby, java, go,…

→ applications : wordpress…

~ 150 000 images contribuées

Modèle git (

- ~100 000 dépôts Github avec un Dockerfile
- ~1000 contributeurs du code de Docker
- Des milliers de projets s'intégrant avec Docker
- Quelques outils « officiels » :
  - Machine (déploiement de serveurs Docker)
  - Compose (gestion d'applications multi containers)
  - Swarm (agrégation de plusieurs hôtes Docker)

Démo

Webographie

Docker

http://www.journaldunet.com/solutions/cloud-computing/docker-definition-avantages-inconvenients.shtml

http://training.docker.com

Vidéos de docker des jdev

https://webcast.in2p3.fr/videos-introduction_a_docker

https://webcast.in2p3.fr/videos-jdev2015_docker_et_les_architectures_orientees_service

Survivor Kit 1/2

- création simple d'un conteneur
  - docker run -it ubuntu bash
- lister les conteneurs
  - docker ps -a
- supprimer un conteneur
  - docker rm n°conteneur
- build un dockerfile (dans le rép du Dockerfile)
  - docker build -t test .
- supprimer tous les conteneurs
  - docker rm $(docker ps -a -q)
- supprimer des images
  - docker rmi n°image
- rechercher une image
  - docker search nom
- télécharger une image
  - docker pull httpd

Survivor Kit 2/2

- Persistance - créer une images volume pour la persistance
  - docker build -t vol .
- Persistance - lancer le volume
  - docker run --name volume vol bash
- Persistance - lancer un conteneur lié au volume
  - docker run -it --volumes-from volume ubuntu bash
- utiliser un volume
  - docker run -it --volumes-from vol ubuntu bash
- lancer un conteneur en monde démont
  - docker run -dit ubuntu bash
- lancer le conteneur nginx
  - docker run -dti -p 80:80 --name ng5 nginx_ubuntu
- import / export
  - docker export 9d0045d5fa7a > container-nginx.tgz
  - cat example.tgz | sudo docker import - example