**plan**

Introduction

Pourquoi virtualiser?

De la virtualisation des machines au conteneurisation.

docker

* définition
* comparaison entre docker et vms
* comment ça fonctionne (techniquement)
* avantages
* inconvénients

kubernetes

* définition
* mots clés
* avantages et inconvénients
* fonctionnement
* efficacite

**qlqs cas pratique d utilisation de docker et kuberenetes(on cite des entreprise**s )

presentation du docker

**c est quoi docker:**

## Comment fonctionne la technologie Docker ?

La technologie Docker utilise le noyau Linux et des fonctions de ce noyau, telles que les groupes de contrôle [cgroups](https://access.redhat.com/documentation/fr-fr/red_hat_enterprise_linux/6/html/resource_management_guide/index) et les [espaces de noms](https://lwn.net/Articles/528078/), pour séparer les processus afin qu'ils puissent s'exécuter de façon indépendante.

Docker permet aussi d'automatiser le déploiement des applications (ou d'ensembles de processus combinés qui forment une application) au sein d'un environnement de conteneurs.

**Define Docker terminology**

**Docker:**

A software container platform designed for developing, shipping, and running apps leveraging container technology. Docker comes in two versions: enterprise edition and community edition

**Container:**

Unlike a VM which provides hardware virtualization, a container provides lightweight, operating-system-level virtualization by abstracting the “user space.” Containers share the host system’s kernel with other containers. A container, which runs on the host operating system, is a standard software unit that packages code and all its dependencies, so applications can run quickly and reliably from one environment to another. Containers are nonpersistent and are spun up from images.

**Docker engine:**

The open source host software building and running the containers. Docker Engines act as the client-server application supporting containers on various Windows servers and Linux operating systems, including Oracle Linux, CentOS, Debian, Fedora, RHEL, SUSE, and Ubuntu.

D**ocker images:**

Collection of software to be run as a container that contains a set of instructions for creating a container that can run on the Docker platform. Images are immutable, and changes to an image require to build a new image.

**Docker Registry:**

Place to store and download images. The registry is a stateless and scalable server-side application that stores and distributes Docker images.

## Les avantages des conteneurs Docker

## limites à l'utilisation de Docker

https://www.redhat.com/fr/topics/containers/what-is-docker

https://www.oracle.com/dz/cloud-native/container-engine-kubernetes/what-is-kubernetes/

## Kubernetes 101 — Mots clés

Pour commencer, voici quelques termes clés liés à Kubernetes. Une liste plus exhaustive est disponible sur la page du glossaire normalisé de Kubernetes. Vous pouvez également tirer parti de la [Fiche de renseignements sur Kubernetes](https://kubernetes.io/docs/reference/kubectl/cheatsheet/), qui contient une liste de commandes et d’indicateurs kubectl couramment utilisés.

#### Cluster

Ensemble de machines, appelées individuellement « nœuds », utilisées pour exécuter des applications conteneurisées gérées par Kubernetes.

#### Nœud

Machine virtuelle ou physique. Un cluster se compose d’un nœud maître et d’un certain nombre de nœuds de travail.

#### Conteneur Cloud

Une image qui contient un logiciel et ses dépendances.

#### Pod

Conteneur unique ou ensemble de conteneurs s’exécutant sur votre cluster Kubernetes.

#### Déploiement

Objet qui gère les applications répliquées représentées par des pods. Les pods sont déployés sur les nœuds d’un cluster.

#### Jeu de répliques

Garantit qu’un nombre spécifié de répliques de pods s’exécutent en même temps.

#### Service

Décrit comment accéder aux applications représentées par un ensemble de pods. Les services décrivent généralement les ports et les équilibreurs de charge, et peuvent être utilisés pour contrôler l’accès interne et externe à un cluster.

