**Séminaire Kubernetes**

Présenté par M. Damien Navez

1. Introduction

Présentation de sa société (carteSoft), offre une intégration et une migration d’application existante ou d’une nouvelle application.

Introduction aux containers Docker.

1. Contenu

* Les Containers :
  + Permet l’isolation entre containers tout en étant lié au même hôte) => Un OS pour tous les containers, mêmes *chroots, cgroups et namespaces*.
  + Un container = un package qui est scellé, ne contient pas de dépendances externes, pas de DLL non compatibles qui “s’entrechoquent”, permet de développer sur laptop et juste après le déployer dans le cloud.
  + Exemple:

“cat -> Dockerfile

FROM node:4.4

EXPOSE 8080

COPY server.js

CMD node server.js”

* CMD = “docker run -d -p 8080:8080 –name hello\_tutorial gcr.io/mohr-dev/hello-node:v1” => run le container come un daemon, prenant le port de l’interface privée (du container) 8080 en le redirigeant vers le port public (de la machine hôte) 8080.
* **Exemple de création et exécution d’un container docker.**
* Kubernetes :
  + Inspiré du framework “Board” de Google. Kubernetes est une version open source écrite en Go.
  + Permet le management de clusters de containers.
  + Paramétrer et utiliser un cluster :
    - Créer et paramétrer : Choisir un service de cloud, un os hôte, configurer les services, etc.
    - Utiliser : faire tourner des containers et leurs supersets (pods).
  + Pourquoi il est dur d’utiliser docker sans orchestrateur :
    - Si on a plusieurs containers avec des applications différentes qui s’interconnectent, il est possible qu’avec le rajout d’autres applications dans le futur, qu’il y’ait des conflits de ports, d’adresses IP (au niveau du NATage entre les différentes machines physiques).
  + Composants :
    - Pods = ensemble de containers qui partagent la même adresse privée.
    - Volumes = stockage partagé entre différents pods.
    - Labels & sélecteurs = Metadata arbitraires attachées à des containers, volumes, etc. Ces metadata peuvent être retrouvées via une requête faites par les sélecteurs. Permet de grouper des pods sous un même groupe appelée “Services”.
    - Réplicas = boucle de contrôle permettant d’assurer ‘n’ copies d’un pod.
    - Services internes/externes = Groupe de pods qui travaillent ensemble et qui sont groupés par un sélecteur.
* Démonstration :
  + Démonstration de la création d’un projet et d’un réplica avec OpenShift (superset de Kubernetes).

1. Conclusion

Monsieur Navez nous a présenté un séminaire très intéressant sur des technologies qui seront de plus en plus utilises. Même si la majorité de la présentation était de la théorie, il expliquait très bien et pouvait passer quelques minutes sur l’explication d’un terme afin de faciliter la compréhension. Il demandait après chaque gros point de théorie si nous avions questions. Il a mis un point d’honneur à interagir avec nous. Je le recommanderais personnellement à toutes les personnes voulant s’initier à la containerisation.