FULLSTACK

CM #2 du 18/10/2022

Antonin ROSA-MARTIN - IDU7

Contenu du cours

01

"Quiz"

02

Quelques cas personnels

03

Présentation Fullstack

04

Présentation de Docker

05

Présentation du NoSQL

QUIZ

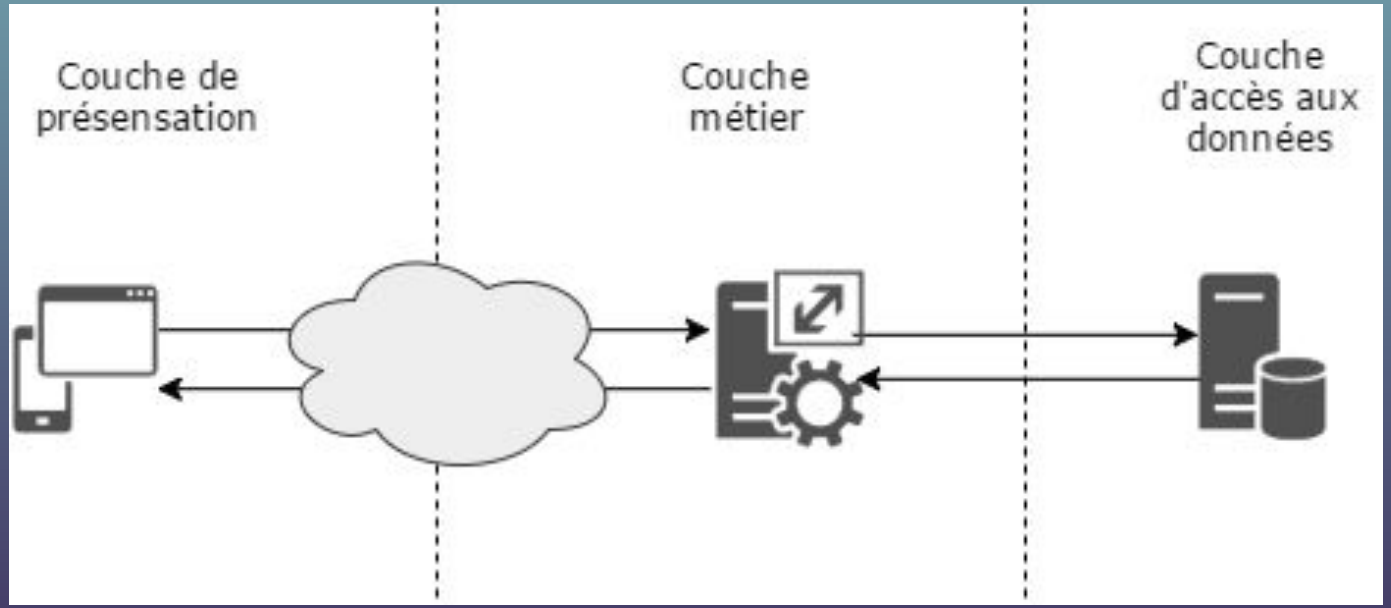

Qu'est-ce que vous évoque le Fullstack ?
Avec quels langages peut-on faire du Fullstack ?
Quelles sont les bases de données utilisées pour faire du Fullstack ?

Compétences et Qualité d'un développeur Fullstack

- **JavaScript** ou des **frameworks** Js pour le **frontend** : Angular, Ext.js, Ember.js, Vue.js, React.js, Next.js…
- Le **pack web classique** (HTML, CSS)
- Un ou plusieurs langages de développement **backend** : Java, Python, PHP, Go, C#, Ruby…)
- Certains **frameworks** associés aux langages de développement tels que Echo ou Gin pour **GoLang**, Spring ou Hibernate pour **Java**, Django ou Flask pour **Python** etc.
- Un système de gestion des **bases de données** : MySQL, Oracle, PostgreSQL, MongoDB…
- Un outil de **gestion de projet** en équipe comme github ou gitlab

Mais donc le Fullstack c'est quoi ?

! Architecture 3 tiers !



L'architecture 3 tiers

Frontend =>
Ce que l'utilisateur voit et ce avec quoi il interagit.
Il doit être le plus léger et fluide possible.

Backend =>
Ce que l'utilisateur ne voit pas, intéractions par le frontend.
Serveur de calculs et de logique métier. Interface avec la BDD.

BDD(s) =>
L'utilisateur n'y a pas accès, intéractions par le backend.
Il peut y avoir plusieurs bases de données dans un seul projet.

Attention à la sécurité !

Dos

Fermer l'accès à votre site.

Cross-site Scripting

Permet à l'attaquant d'injecter du code dans un script côté serveur.

SQL Injection

Le code malveillant est inséré dans des chaînes de caractères qui sont ensuite transmises au serveur SQL, interprétées et exécutées.

Logiciels non patchés

La plupart des menaces qui pèsent sur un serveur peuvent être évitées simplement en disposant de logiciels à jour et correctement corrigés.

Utilisateurs imprudents

La menace la plus courante pour la sécurité d'un serveur est la négligence des utilisateurs. Si vous ou vos utilisateurs ont des mots de passe faciles à deviner, un code mal écrit, un logiciel non corrigé ou un manque de mesures de sécurité...

CAS PERSONNELS

Geode Toolbox

- Frontends **Next.js**
- Backends **Express Js/Ts**
- Bases de données
  - **MongoBD**
  - **Redis**

NFTags

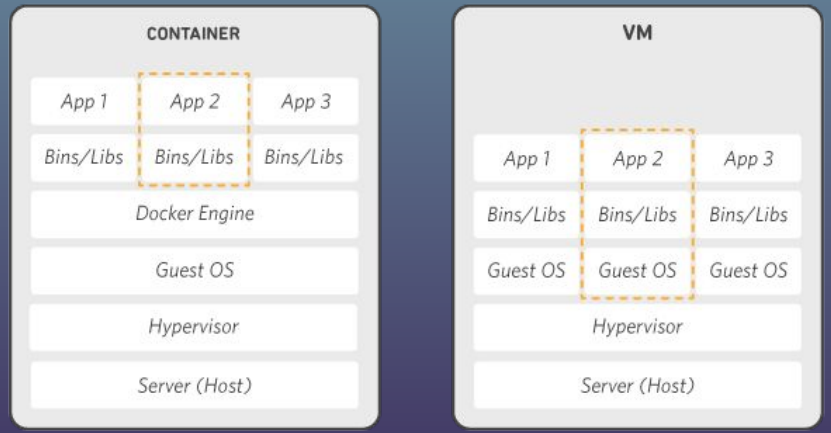
- Frontends **Next.js** (en Ts)
- Backends **Echo GoLang**
- Bases de données
  - **MongoBD**
  - **Avalanche**

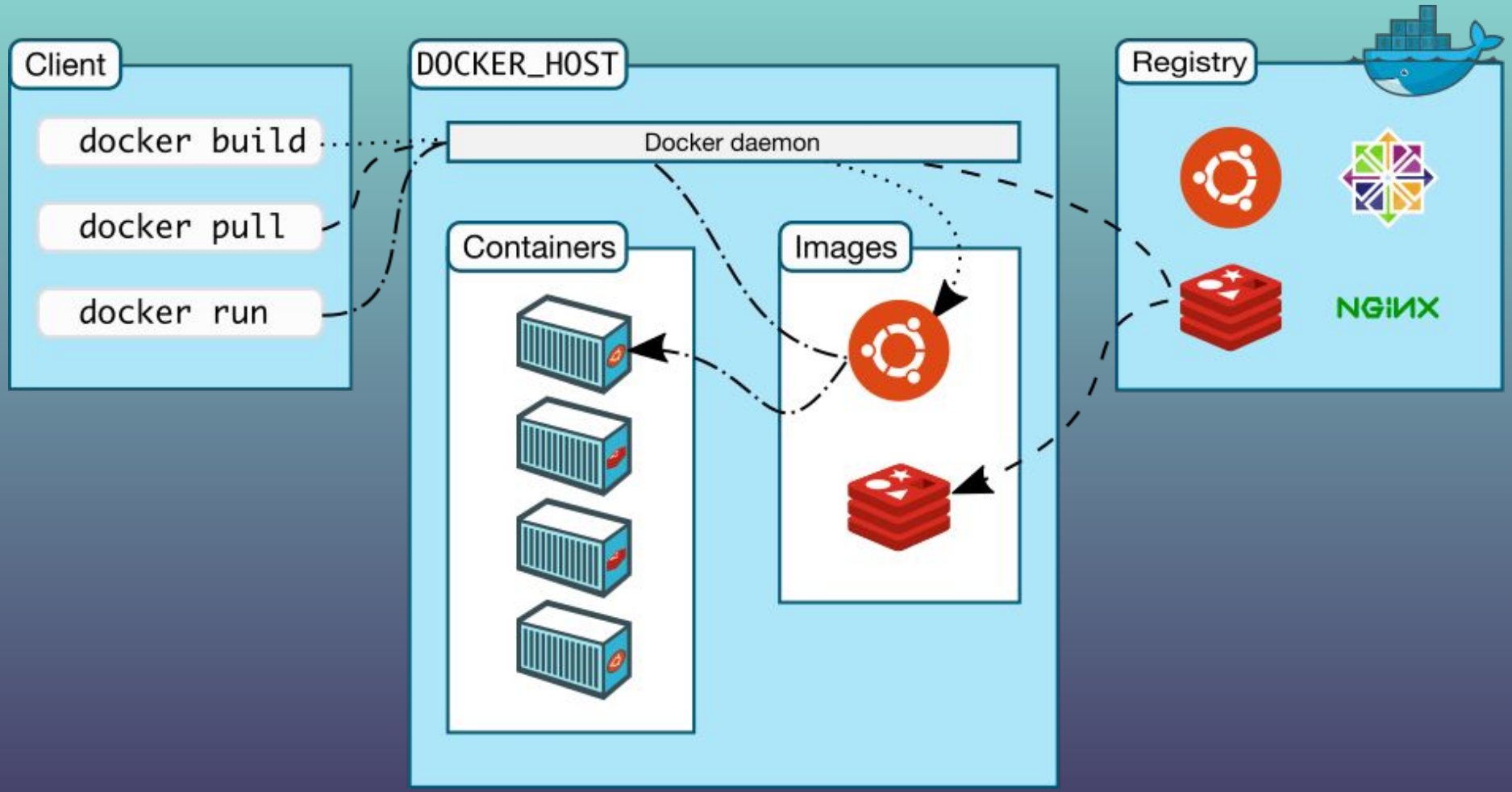
Votre nouveau meilleur ami: DOCKER

L'outil indispensable du développeur moderne

Docker en bref

Docker est une plate-forme logicielle qui vous permet de **concevoir**, **tester** et **déployer** des applications rapidement. Docker intègre les logiciels dans des **unités normalisées** appelées **conteneurs**, qui rassemblent tous les éléments nécessaires à leur fonctionnement, dont les **bibliothèques**, les **outils système**, le **code** et **l'environnement d'exécution**. Avec Docker, vous pouvez facilement **déployer** et **dimensionner** des applications dans **n'importe quel environnement**, avec l'assurance que **votre code s'exécutera correctement**.




Client
docker build
docker pull
docker run
DOCKER_HOST
Docker daemon
Containers
Images
Registry
NGINX

Quelques Dockerfiles

API NodeJs

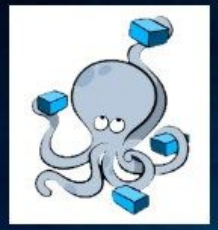
```
FROM node:latest

WORKDIR /usr/src/app

COPY package*.json ./

RUN npm install

COPY . .

RUN npm run build

EXPOSE 8080

CMD ["npm", "start"]
```

Frontend Next.js

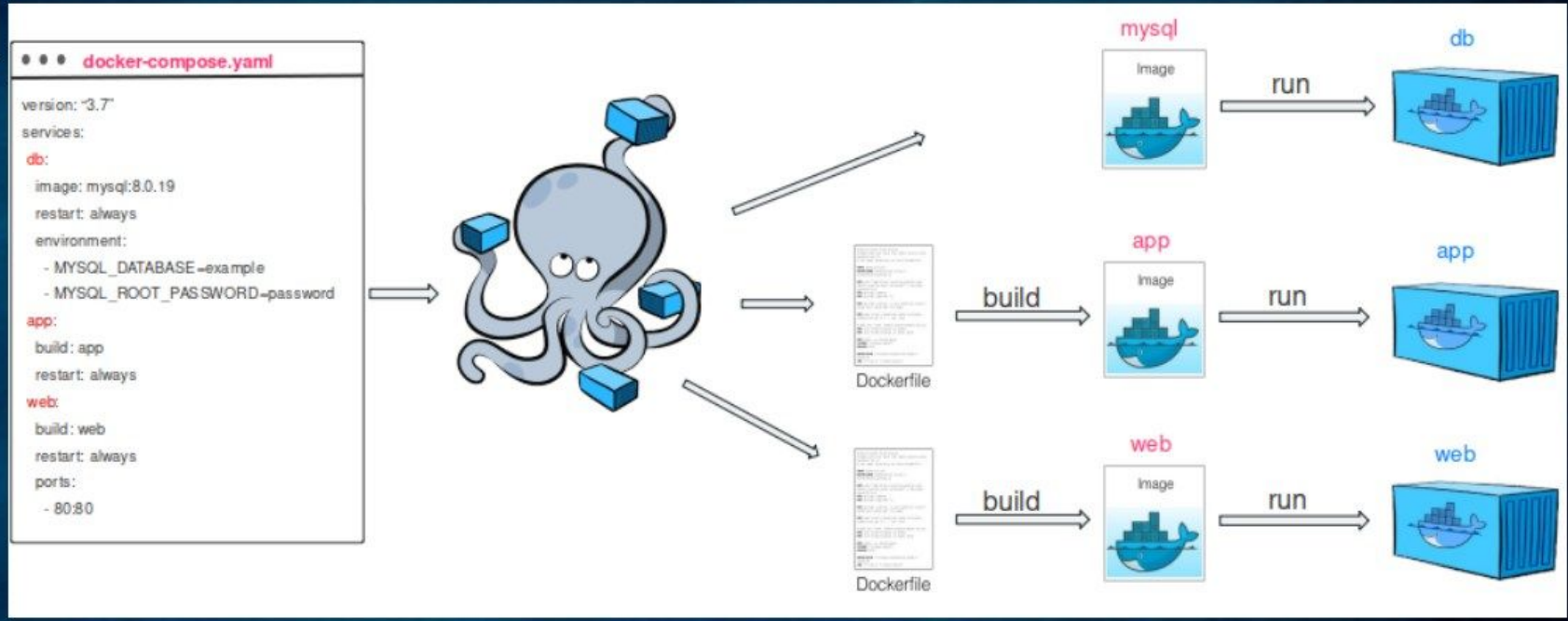
```
FROM node:latest

WORKDIR /usr/src/app

COPY package*.json ./

RUN npm install

COPY . .

RUN npx next build
--profile

EXPOSE 3000

CMD npx next start -p
3000
```

Backend GoLang

```
FROM golang@latest

WORKDIR /go/src/app

COPY . .

RUN go get -d -v

RUN go install -v

RUN go build
gitlab.com/nftags/nftagsb
ackend

CMD ["./nftagsbackend"]
```

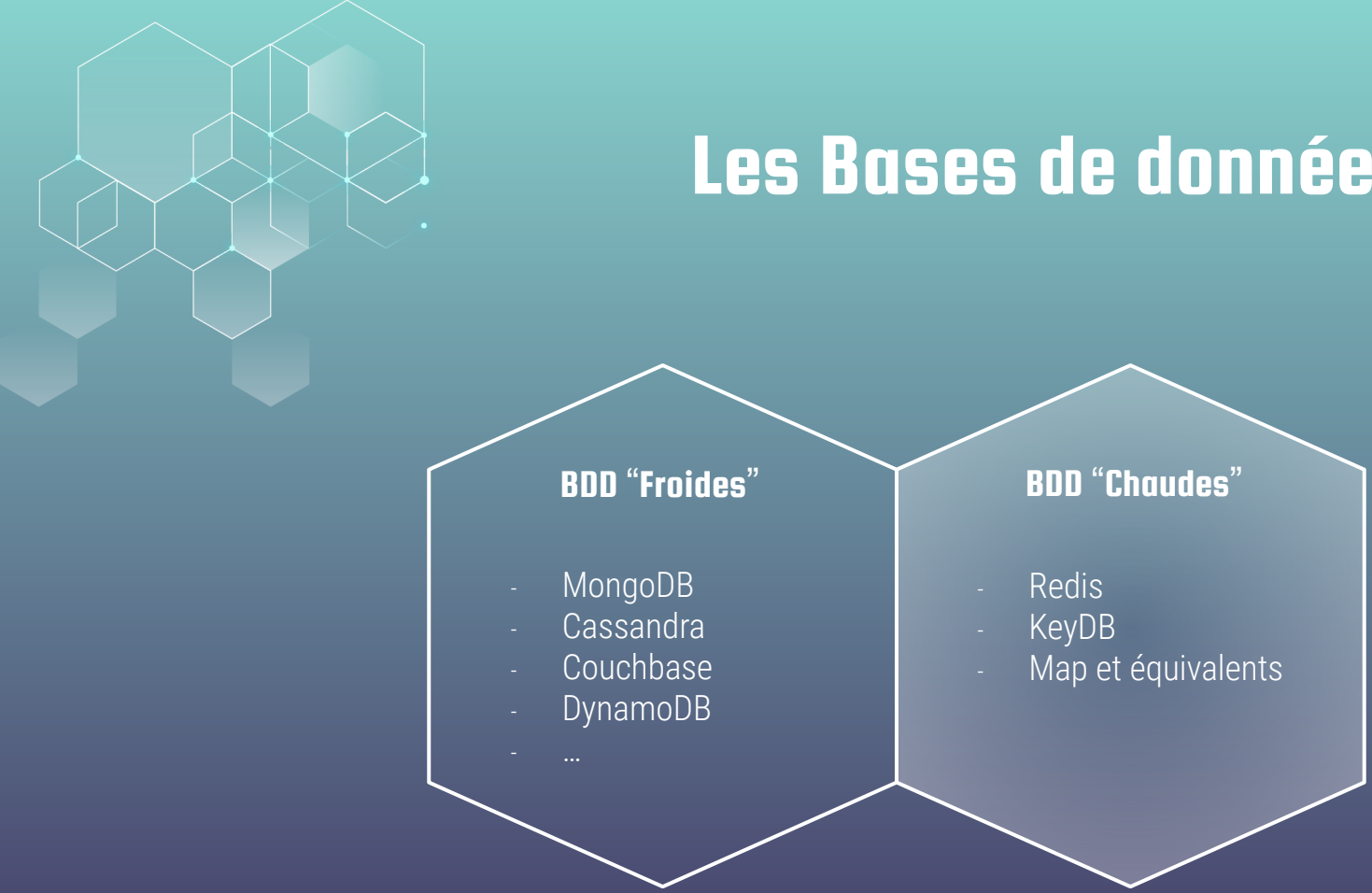


Docker Compose



Run multiple containers as a service from 1 .yaml file

Les Bases de données NoSQL



Le NoSQL ?

Le **NoSQL**, pour "not only SQL", désigne les bases de données qui ne sont **pas fondées sur l'architecture classique** des bases de données relationnelles. Développé à l'origine pour gérer du **"Big Data"**, l'utilisation de base de données NoSQL a explosée depuis quelques années.

Lorsque l'on parle de NoSQL, on regroupe des systèmes de base de données qui ne sont **pas relationnels**, mais il faut savoir qu'il existe **plusieurs types** de bases de données "NoSQL":

- Les **bases clef/valeur**, permettent de stocker des informations sous forme d'un **couple clef/valeur** où la valeur peut être une chaîne de caractère, un entier ou un objet sérialisé. Une base de données **Redis**. Ce type de base de données offre de **très bonnes performances** par sa simplicité et peut même être utilisé pour stocker les **sessions utilisateur** ou le **cache** de votre site par exemple.
- Les **bases orientées colonnes**, ressemblent aux bases de données relationnelles, car les données sont sauvegardées sous forme de ligne avec des colonnes, mais se distinguent par le fait que le nombre de colonnes peut varier d'une ligne à l'autre. Les solutions les plus connues sont **HBase** ou **Cassandra**.
- Les **bases orientées document**, représentent les informations **sous forme d'objet XML ou JSON**. L'avantage est de pouvoir récupérer simplement des **informations structurées de manière hiérarchique**. Les solutions les plus connues sont **CouchDB**, **RavenDB** et **MongoDB**.
- Les **bases orientées graphe**, présentent les données **sous forme de nœud et de relation**. Cette structure permet de récupérer simplement des **relations complexes**. Un exemple de base graphe est **Neo4J**.

Que choisir en BDD NoSQL ?

BDD Froides

Le stockage à **froid** est destiné aux données qui sont **rarement utilisées**. Il s'agit de données qui doivent être **conservées** pour une raison quelconque. Ces données sont également connues sous le nom de "**données dormantes**".

BDD Chaudes

Toutes les données auxquelles vous devez **accéder immédiatement** doivent être placées dans une base de données **chaude**:

- Connues pour être **modifiées très rapidement**
- **Session** utilisateur
- **Caching**
- **Chat**, **messagerie** et **files d'attente**
- Analyse en **temps réel**