

















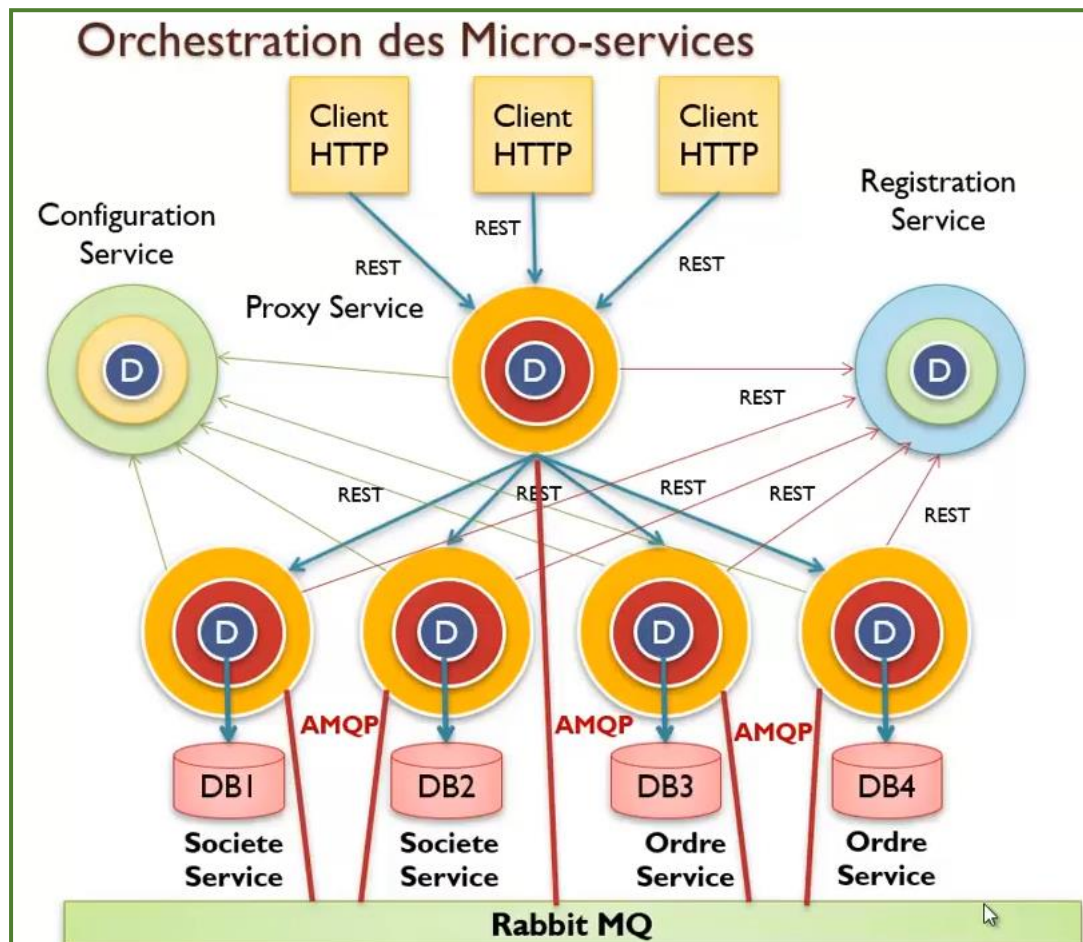
Création d'une application qui permet de gérer des sociétés qui sont cotées en bourse.

- On va créer deux micro services :
 - Un micro service qui gère les sociétés
 - Un micro service qui gère les transactions de ses sociétés.

- L'application sera composée de :
 - D'un micro service nommé service-company. Il utilise les dépendances :
 -  Base de données H2
 -  Lombok
 -  spring-cloud-starter-netflix-eureka-client
 -  spring-cloud-starter-config
 -  spring-boot-starter-data-rest
 -  spring-boot-starter-actuator
 -  spring-boot-starter-data-jpa
 -  Autres (cf pom.xml)
 - D'un service d'enregistrement nommé service-register. Il utilise les dépendances :
 -  spring-cloud-starter-config
 -  spring-cloud-starter-netflix-eureka-server
 -  Autres (cf pom.xml)
 - D'un service de configuration nommé service-config. Il utilise les dépendances :
 -  spring-cloud-config-server
 -  Autres (cf pom.xml)

Remarque : Le service de configuration a besoin de connaître le chemin du dossier qui va contenir l'ensemble des fichiers de configuration. Il faut créer un repository local nommé cloud-config, et le donner le chemin de ce repository.

- D'un service de proxy nommé service-proxy. Il utilise les dépendances :
 -  spring-cloud-starter-netflix-zuul
 -  spring-cloud-starter-netflix-eureka-client
 -  Autres (cf pom.xml)



- On utilise trois services :
 - **Configuration Service** : Il permet de centraliser la configuration. Permet de ne pas créer un fichier de configuration pour chaque micro service. On mettra dans ce service la configuration nécessaire pour l'ensemble des micro services.
 - **Registration Service** : Il permet d'enregistrer la localisation de chaque micro service. Les micro services, lorsqu'ils démarrent se connectent au service de registration pour publier trois informations (son adresse IP, numéro de port, son nom).
 - **Proxy Service** : Il permet d'orchestrer l'ensemble des micro services. Toutes les requêtes http venant du client (front end) passe par le proxy. S'il connaît le nom du micro service il peut contacter le service de registration en lui donnant son nom, et ce dernier lui donne en retour l'adresse IP et le port du micro service. Le nom du micro service se trouve dans l'url envoyé vers le proxy. Ainsi il peut envoyer la requête

ver le micro service en question, afin qu'il fasse le traitement. Ensuite le micro service retourne le résultat au service proxy, et le service proxy le retourne ensuite au client.

