

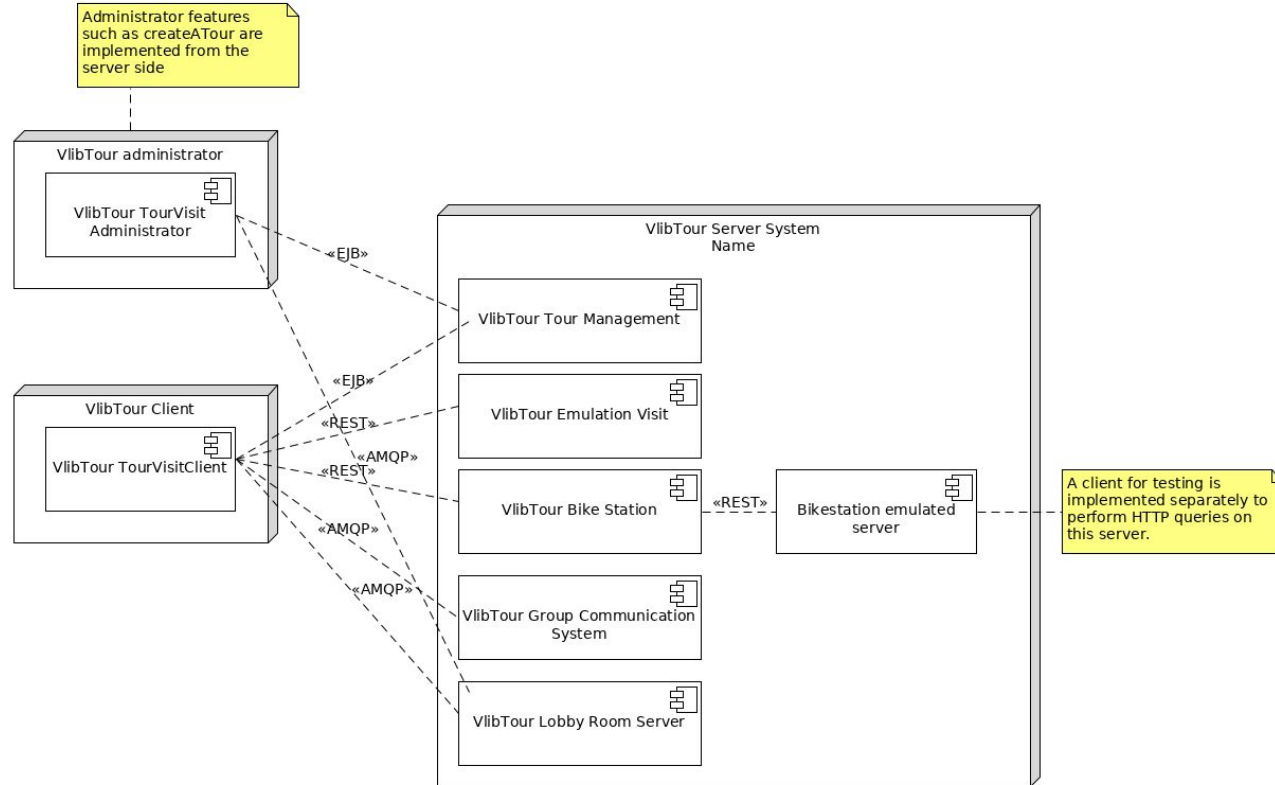
POC VLIBTOUR



Réalisé par :
Farah KILANI
Nada Touil

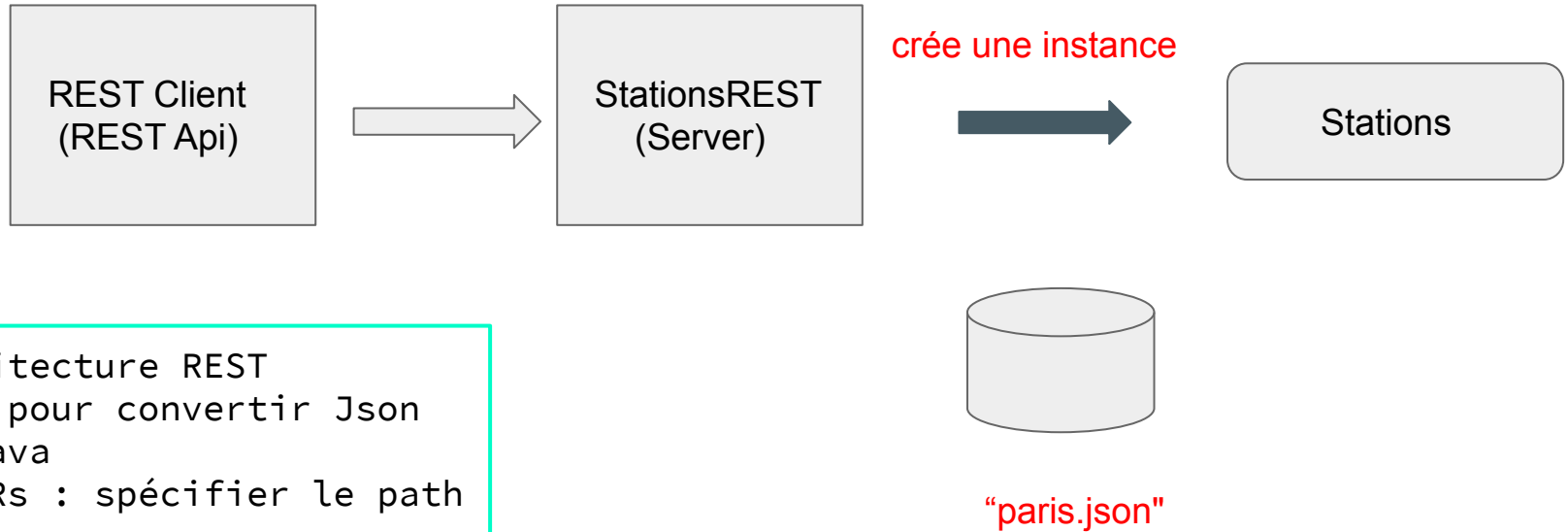
INTRODUCTION

- Contexte :
Middlewares pour les systèmes répartis
- Implique de prendre en considération une large échelle d'application clients
- Technos variées utilisées : EJB, REST, RabbitMq.



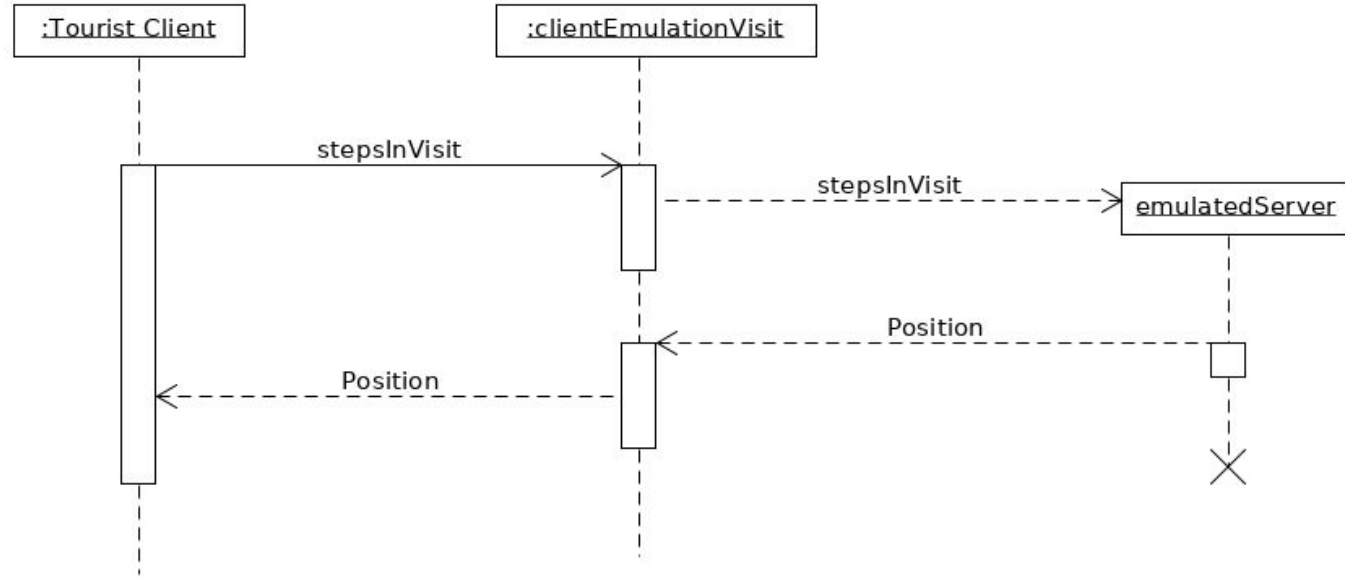
ARCHITECTURE

API REST: VlibTour Bike Station



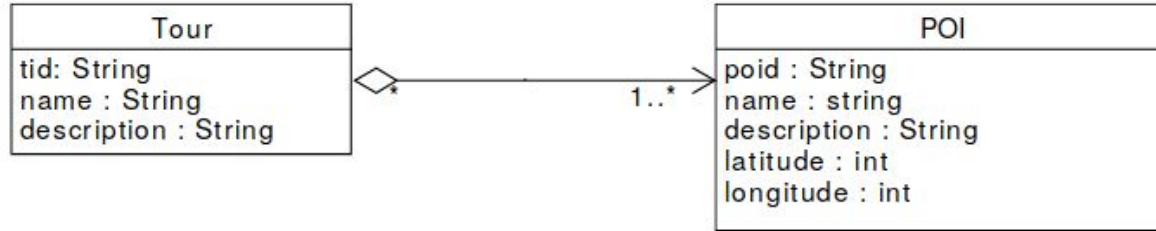
VLIBTOUR EMULATIONVISIT

- des requêtes HTTP à travers le proxy
- exposer une différente interface pour alléger le code



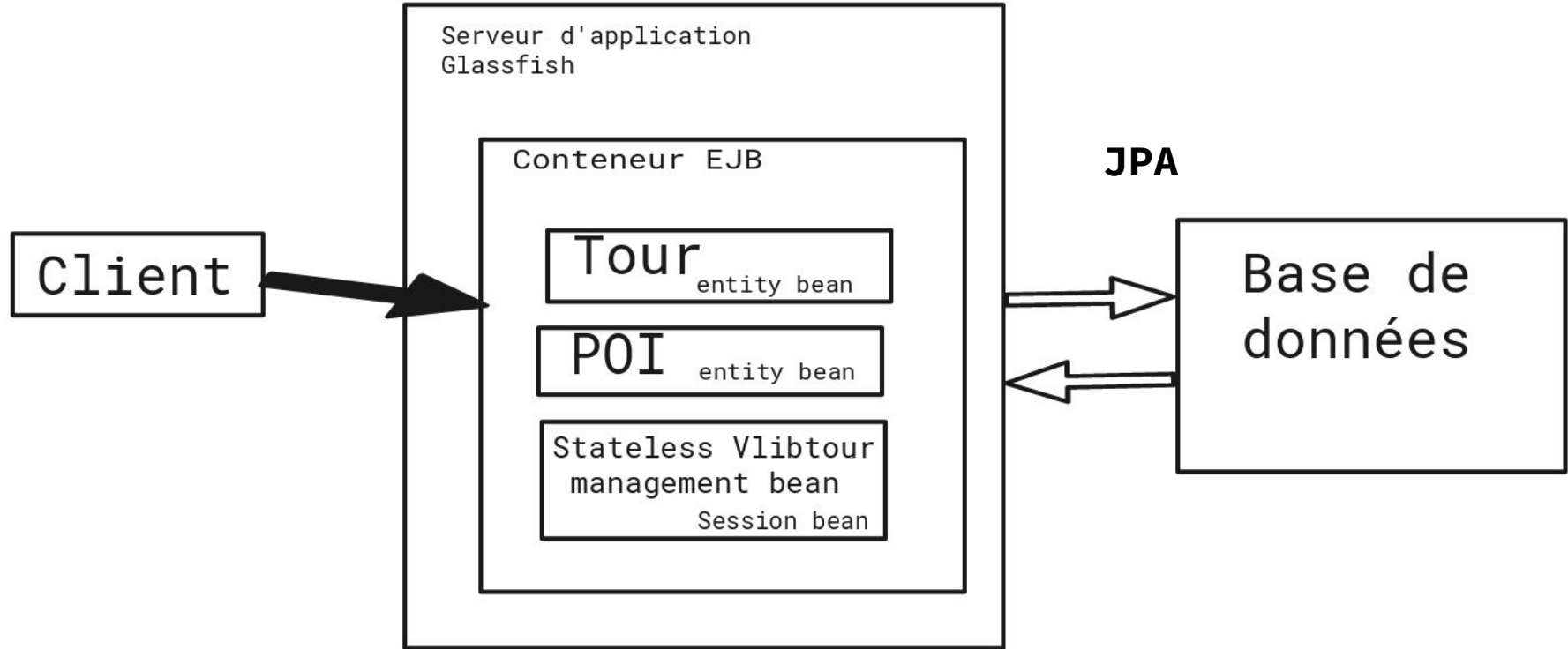
VLIBTOUR TOUR MANAGEMENT

- Architecture utilisée : EJB.



Relation Many to Many entre le Tour et le POI

VLIBTOUR TOUR MANAGEMENT



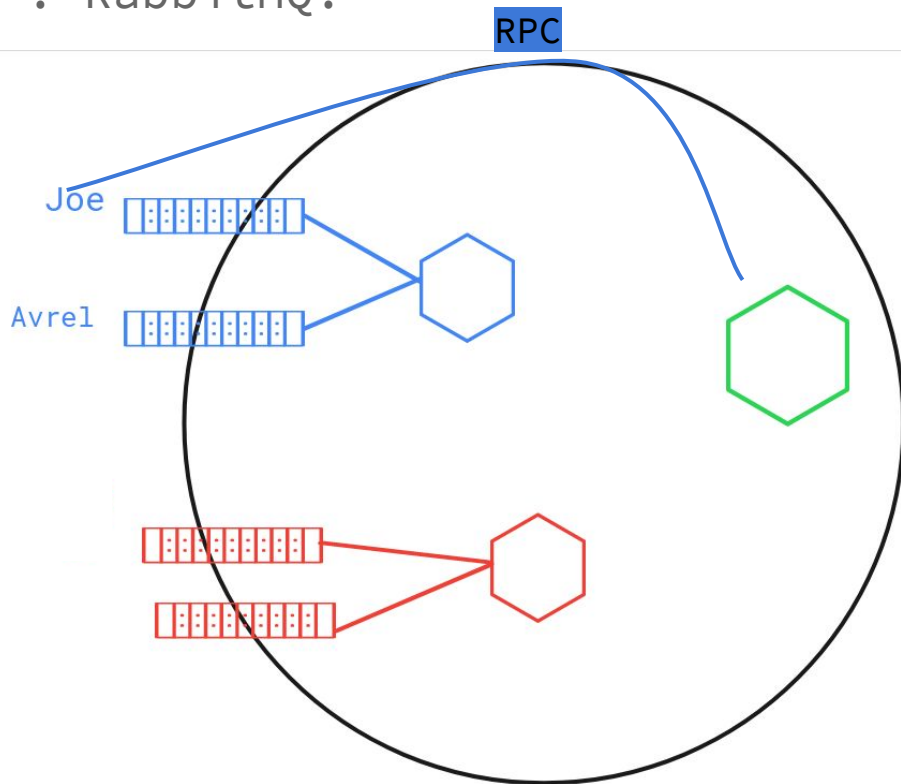
VLIBTOUR TOUR MANAGEMENT

Pourquoi EJB?

- Gestion des transactions
- Concurrency sans utiliser les threads
- Sécurité d'accès

LOBBY ROOM & GROUP COMMUNICATION

- Architecture utilisée : RabbitMQ.



LOBBY ROOM & GROUP COMMUNICATION

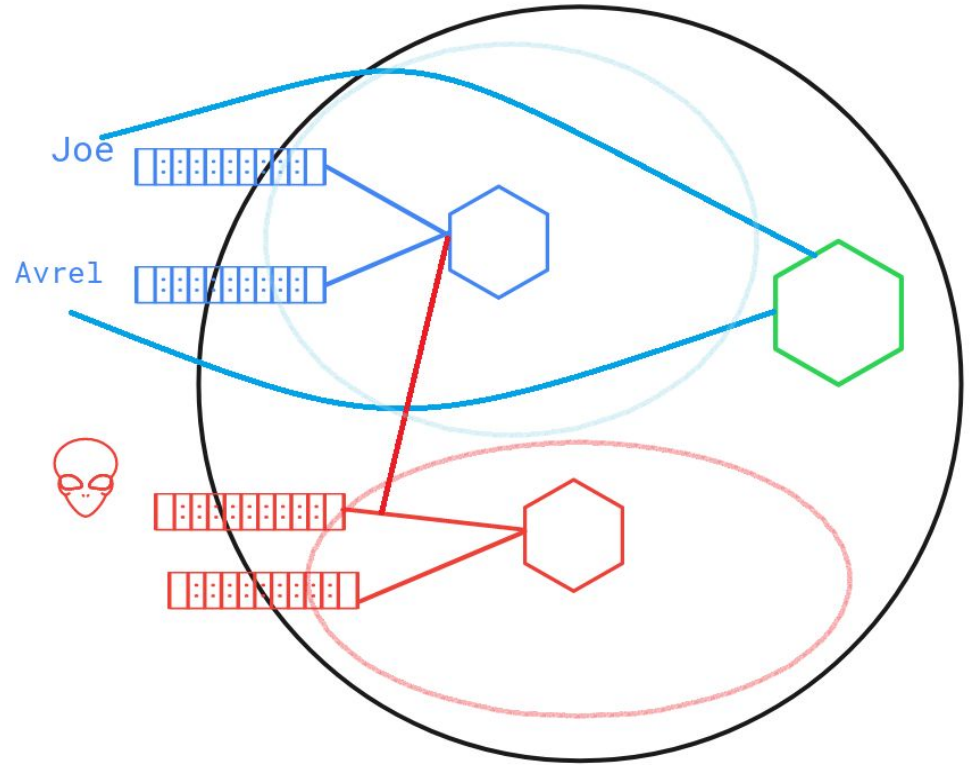
Pourquoi rabbitmq?

- Utilise un protocole AMQP qui assure l'interopérabilité.
- Pas besoin de garder les messages échangés.
- Robuste.
- Système de clustering pour la haute disponibilité et la scalabilité.
- Les vhost permettent de cloisonner des environnements.
- Une qualité de service qui permet de prioriser les messages

EXTRA FONCTIONNALITÉS

- Sécurité

même si quelqu'un intercepte l'URL, l'accès à la communication est impossible.



EXTRA FONCTIONNALITÉS

- Scalabilité

Sessions Beans + REST server are
stateless => scalabilité
possible car pas de dépendance
de session

