



Rapport du mini projet

Sysytemes repartis

Profs:

-Mr.Abbes Heithem

-Mr.Denden Islem

Réalisé par :

Abid Nouha IGL3

INTRODUCTION :

Ce rapport présente une comparaison entre trois technologies de communication distribuée en Java : Java RMI, gRPC et les sockets. L'objectif est d'évaluer leurs performances, leur facilité d'utilisation, leur flexibilité et leurs avantages respectifs dans des contextes réels.

JAVA RMI :

Java RMI (Java Remote Method Invocation) permet d'appeler des méthodes sur des objets distants comme s'ils étaient locaux, simplifiant ainsi les applications distribuées en Java.

Avantages	Limitations
<ul style="list-style-type: none">-Intégration transparente avec Java.-Facilité de développement pour les applications Java.-Utilisation simple des objets distants.	<ul style="list-style-type: none">-Limité à l'écosystème Java, ce qui peut entraîner des difficultés d'interopérabilité.-La configuration du registre RMI peut être fastidieuse.

GRPC :

GRPC est un framework de communication entre services, développé par Google. Il offre des performances élevées et une interopérabilité multi-langage grâce à HTTP/2 et à la sérialisation protobuf. En bref, gRPC simplifie la communication dans les architectures distribuées.

Avantages	Limitations
<ul style="list-style-type: none"> -Performances élevées grâce à HTTP/2 et à la sérialisation protobuf. -Support multi-langage. 	<ul style="list-style-type: none"> -Courbe d'apprentissage plus raide en raison de la complexité des concepts RPC.

Sockets :

Les sockets sont une API de communication permettant aux applications d'échanger des données sur un réseau informatique

Avantages	Limitations
<ul style="list-style-type: none"> -Contrôle maximal sur la communication réseau. -Flexibilité pour implémenter des fonctionnalités avancées. -Pas de dépendance à une plateforme spécifique. 	<ul style="list-style-type: none"> -Complexité accrue en raison de la gestion manuelle de la communication réseau. -Moins adapté aux applications nécessitant une communication complexe entre les composants distants.

COMPARAISON :

- **Facilité d'utilisation :**

Java RMI est simple pour les applications Java, gRPC offre une facilité d'utilisation élevée une fois maîtrisé, tandis que les sockets demandent une gestion manuelle complexe.

- **Performances :**

gRPC offre généralement des performances élevées, tandis que Java RMI et les sockets dépendent plus de l'implémentation spécifique.

- **Flexibilité :**

gRPC est flexible avec un support multi-langage, Java RMI est limité à Java, et les sockets offrent une flexibilité maximale.

CONCLUSION :

Le choix de la technologie dépend des besoins spécifiques du projet. Pour des performances élevées et une facilité de développement, gRPC est recommandé. Java RMI convient aux applications Java existantes, tandis que les sockets offrent une flexibilité maximale mais nécessitent une gestion manuelle plus complexe.