

## Partie 3 : Évaluation et Comparaison

### 1. Comparaison des deux services : REST vs SOAP

Critères	API RESTful	API SOAP
<b>Performance</b>	REST est généralement plus léger et rapide, car il utilise des messages au format JSON ou XML, qui sont moins volumineux que SOAP. REST utilise HTTP directement pour la communication, ce qui réduit la surcharge par rapport à SOAP.	SOAP est plus lent en raison de son format XML verbeux et de la nécessité de respecter un cadre strict pour les requêtes/réponses. La surcharge du protocole SOAP, les en-têtes et le traitement XML peuvent entraîner des latences.
<b>Facilité d'implémentation</b>	REST est plus facile à mettre en œuvre. Il repose sur des méthodes HTTP standard (GET, POST, PUT, DELETE) et s'intègre facilement à des applications web modernes. Il nécessite peu de configurations, avec un développement rapide	SOAP est plus complexe à implémenter. Il nécessite la gestion des WSDL, l'encapsulation XML, et une configuration plus lourde. SOAP a également besoin de bibliothèques spécifiques pour la sérialisation/désérialisation des messages
<b>Interfaçage</b>	REST s'interface facilement avec des clients web, des appareils mobiles, et des services tiers, car il utilise des formats de données courants comme JSON. Il est plus flexible pour des systèmes distribués	SOAP est souvent utilisé pour les systèmes d'entreprise avec des exigences de sécurité plus strictes. SOAP fonctionne bien dans des environnements fermés ou pour des intégrations point à point.

### 2. Avantages et inconvénients de chaque approche dans une banque en ligne

**API RESTful :**

- **Avantages :**
  - **Légèreté** : REST est plus léger et mieux adapté aux applications mobiles et aux environnements où la bande passante est limitée.
  - **Facilité d'intégration** : La plupart des services modernes (clients web, mobiles, etc.) préfèrent les API REST car elles utilisent des formats de données légers comme JSON.
  - **Scalabilité** : REST est idéal pour les services web distribués et les microservices en raison de sa simplicité et de sa flexibilité.
  - **Performance** : Les requêtes REST sont généralement plus rapides grâce à l'utilisation des méthodes HTTP standards et des messages plus petits.
- **Inconvénients :**
  - **Sécurité** : Bien que les API REST puissent utiliser HTTPS, elles ne disposent pas des mécanismes de sécurité intégrés de SOAP, comme WS-Security.
  - **Transactions complexes** : REST n'est pas idéal pour des transactions très complexes nécessitant des messages formels et rigides.

## API SOAP :

- **Avantages :**
  - **Sécurité** : SOAP dispose de normes de sécurité bien définies telles que WS-Security, ce qui le rend adapté aux environnements bancaires où la sécurité est critique.
  - **Fiabilité** : SOAP garantit des communications fiables avec des fonctionnalités comme les accusés de réception et la gestion des erreurs. SOAP peut être utilisé sur des protocoles autres que HTTP (ex : JMS), ce qui est utile pour certaines infrastructures.
  - **Support pour des transactions complexes** : SOAP est mieux adapté pour des cas d'usage exigeants, comme les transactions bancaires où un schéma rigide et des fonctionnalités avancées sont nécessaires.
- **Inconvénients :**
  - **Complexité** : Le développement d'un service SOAP est plus complexe et demande plus de configuration (WSDL, enveloppes XML).
  - **Surcharge** : Les messages SOAP sont lourds, avec une structure XML détaillée, ce qui entraîne une augmentation de la latence et de l'utilisation de la bande passante.
  - **Moins adapté aux applications modernes** : SOAP n'est pas aussi populaire que REST dans les applications modernes, notamment pour les environnements mobiles et les API ouvertes.

## Conclusion :

Pour une banque en ligne, l'utilisation de REST serait plus appropriée pour des applications destinées aux clients finaux (applications mobiles et web), en raison de sa légèreté, de sa rapidité, et de sa facilité d'implémentation. Cependant, SOAP pourrait être préféré pour des systèmes internes ou des intégrations entre partenaires nécessitant des transactions complexes et une sécurité renforcée.



