

SOA et Services Web (Cours 3)

Dr EL BOUHISSI Houda

Janvier 2021

Objectifs du cours

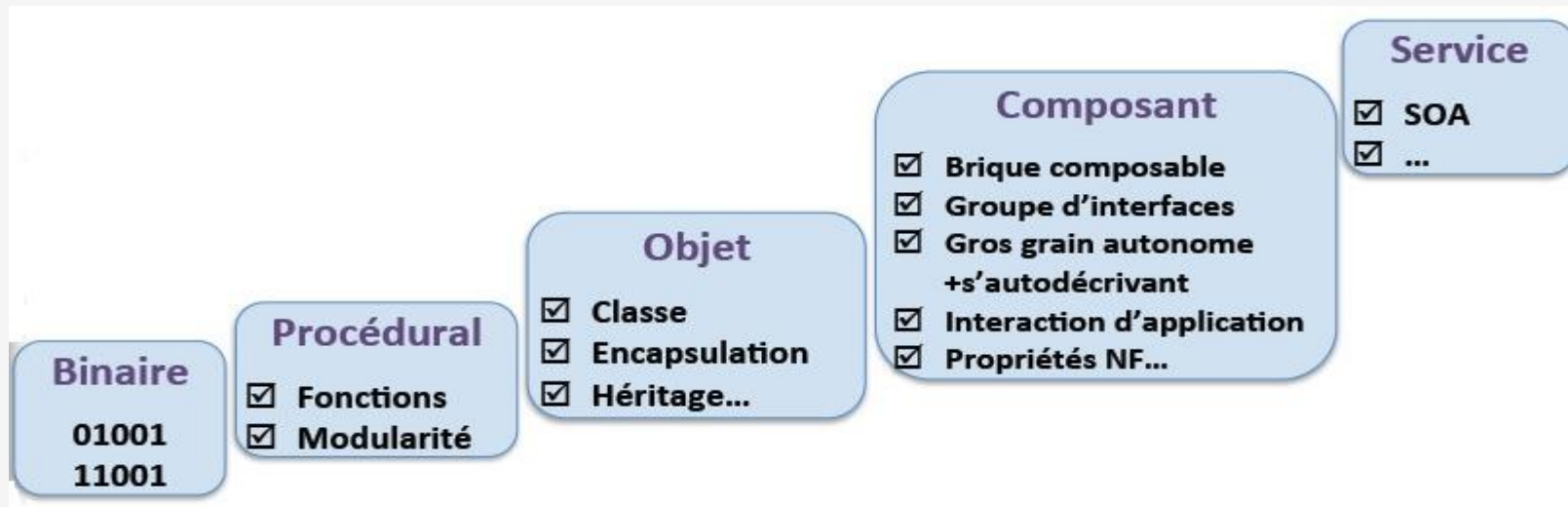
- 1 Comprendre le principe de l'architecture SOA
- 2 Comprendre l'intérêt et les enjeux de l'architecture SOA
- 3 Comprendre le concept de Services Web

Plan

- 1 Problématique
- 2 L'architecture SOA et le principe de service
- 3 Les Services Web

Evolution des paradigmes de programmation

- La conception d'un programme informatique s'effectue conformément à un paradigme de programmation (PP).
- Un PP définit un concept pour représenter le monde et des techniques pour traiter ce concept.
- Différents PP ont vu le jour et ont évolué du binaire, à différents modèles de programmation puis à **l'architecture SOA**.



Architecture distribuée : Problématique

Une architecture distribuée est une architecture où le traitement des données des applications est distribué sur plusieurs machines en réseau : Architectures client-serveur, N-Tiers, Web

Le SI d'une entreprise souvent confronté à des changements :

- Interopérabilité ?
- Réutilisabilité ?
- Communication entre applications hétérogènes ? (Java/C#)



L'interopérabilité exprime le besoin d'échanger des données (d'1 facture par ex) entre 2 systèmes distribués et **éventuellement hétérogènes** pour un besoin **d'intégration**

Le Paradigme SOA

- Style d'architecture distribuée qui permet de fournir ou consommer un processus métier en tant que service.
- Offre des services réutilisables et interopérables via des interfaces standards (construites autour de XML).
- Plusieurs partenaires peuvent communiquer et échanger des données dans le contexte de SOA indépendamment des Plateformes et langages.

Le paradigme SOA :

- Augmenter la flexibilité
- Faiblement couplé
- Basé sur des standards
- Hétérogène
- Plusieurs propriétaires
- Distribué

Service : composant logiciel qui exécute une action pour le compte d'un client , il traduit le niveau logique d'accès aux traitements, plutôt que le niveau physique d'implémentation.

Acteurs de la SOA

Fournisseur de service :

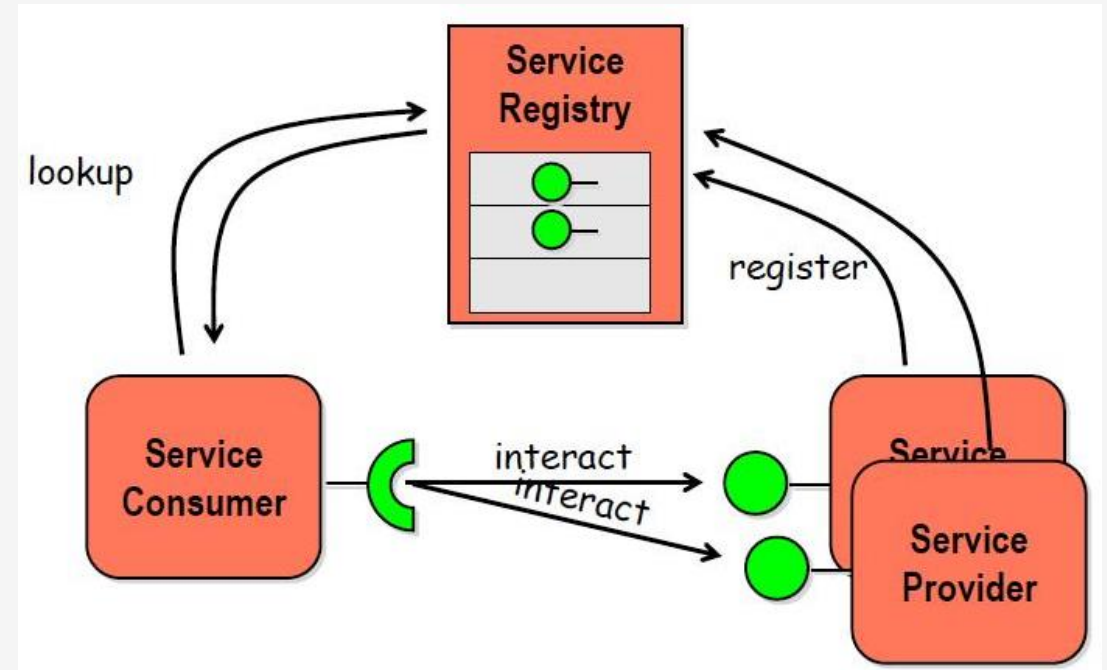
- Fournit un service accessible via une adresse
- Publie son contrat dans le registre de services
- Exécute les requêtes des consommateurs (un Proxy et un cache peuvent être utilisés du côté consommateur pour délocaliser le traitement et réduire le nombre d'appels réseau)

Consommateur de service

Application, service..., Cherche le service dans le registre (son adresse) - Se lie dynamiquement au service. Invoque le service via une requête conforme au contrat

Registre de services

Annuaire des contrats de services, Le Contrat décrit le format d'échange (format des requête/réponse, les pré et post conditions du service et sa QoS, ex: temps de réponse). Le contrat est renouvelable par demande de nouveau bail à partir du registre



Technologies d'implémentation de SOA

L'architecture SOA peut être implémentée par différentes Technologies :

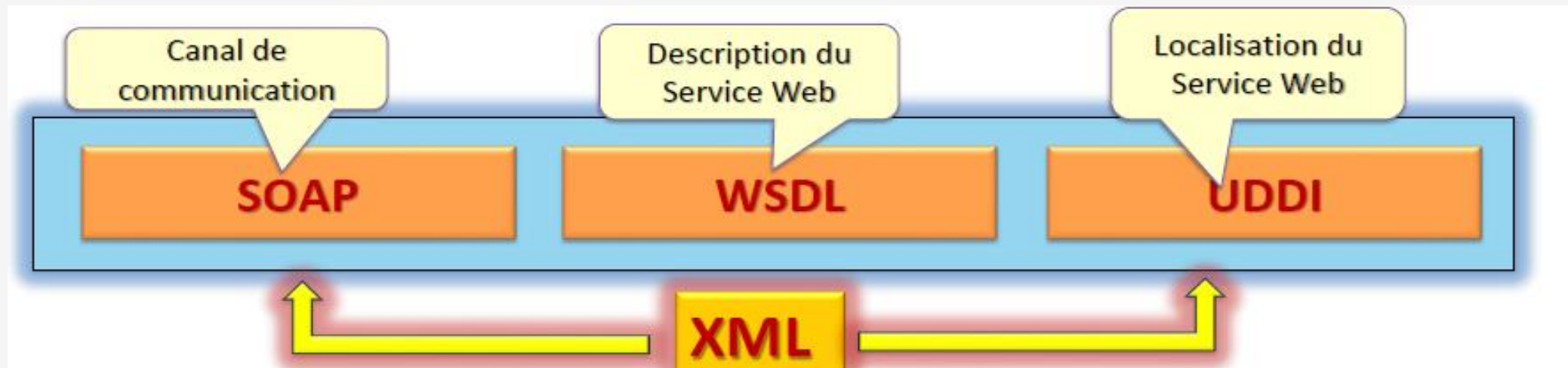
- **CORBA** : Multi-langage , Multiplateforme, installation coûteuse.
- **DCOM (Microsoft)** : Multi-langage , Mono-plateforme, faible diffusion (non disponible pour certains OS).
- **RMI** : Multi-langage , Multiplateforme, problème de performance et de sécurité.
- **Services Web**

Les **Services Web** demeurent la technologie émergente pour l'implémentation de l'architecture SOA :

- **Déployés** : sur n'importe quelle plateforme
- **Modulaires** : implémenté dans n'importe quel langage
- **Enveloppés** dans une couche de standards dérivés du XML
- **Ne nécessitant pas** une configuration réseau particulière
- **Publié, localisé et invoqué** de n'importe quel point du Web
- **Accessible** via des protocoles standards Internet

Services Web

- Les services Web sont une technologie d'implémentation de la SOA.
- Les services Web constituent la meilleure solution disponible (standardisée).
- Selon le W3C (2004) : **Un service web est un système logiciel identifié par un URI, dont les interfaces publiques et les « bindings » sont définies et décrites en XML. Sa définition peut être découverte dynamiquement par d'autres systèmes logiciels. Ces autres systèmes peuvent ensuite interagir avec le service web d'une façon décrite par sa définition, en utilisant des messages XML transportés par des protocoles Internet.**



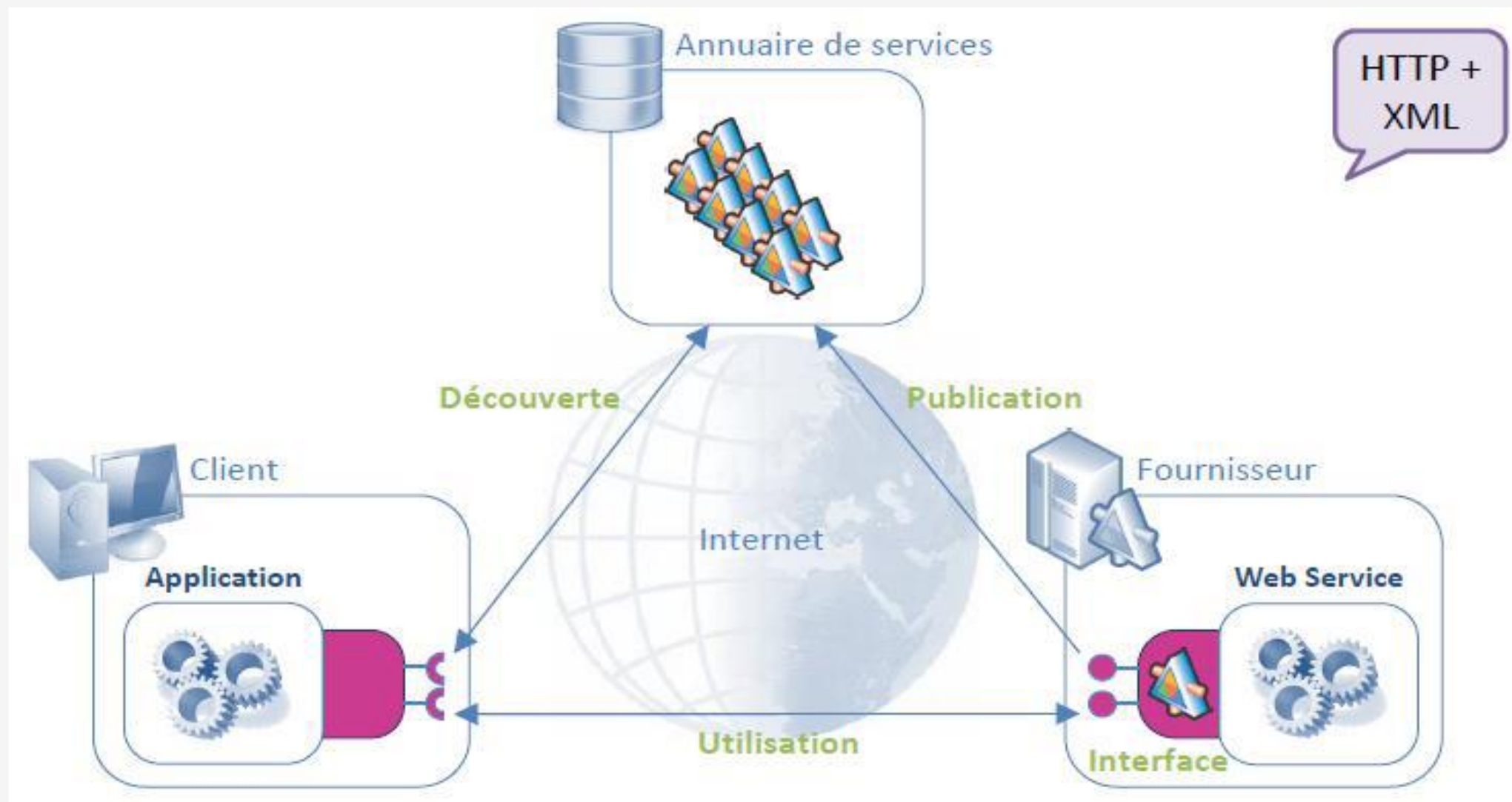
Infrastructure des Services Web

SOAP (Simple Object Access Protocol) : assure la communication avec et interServices Web.

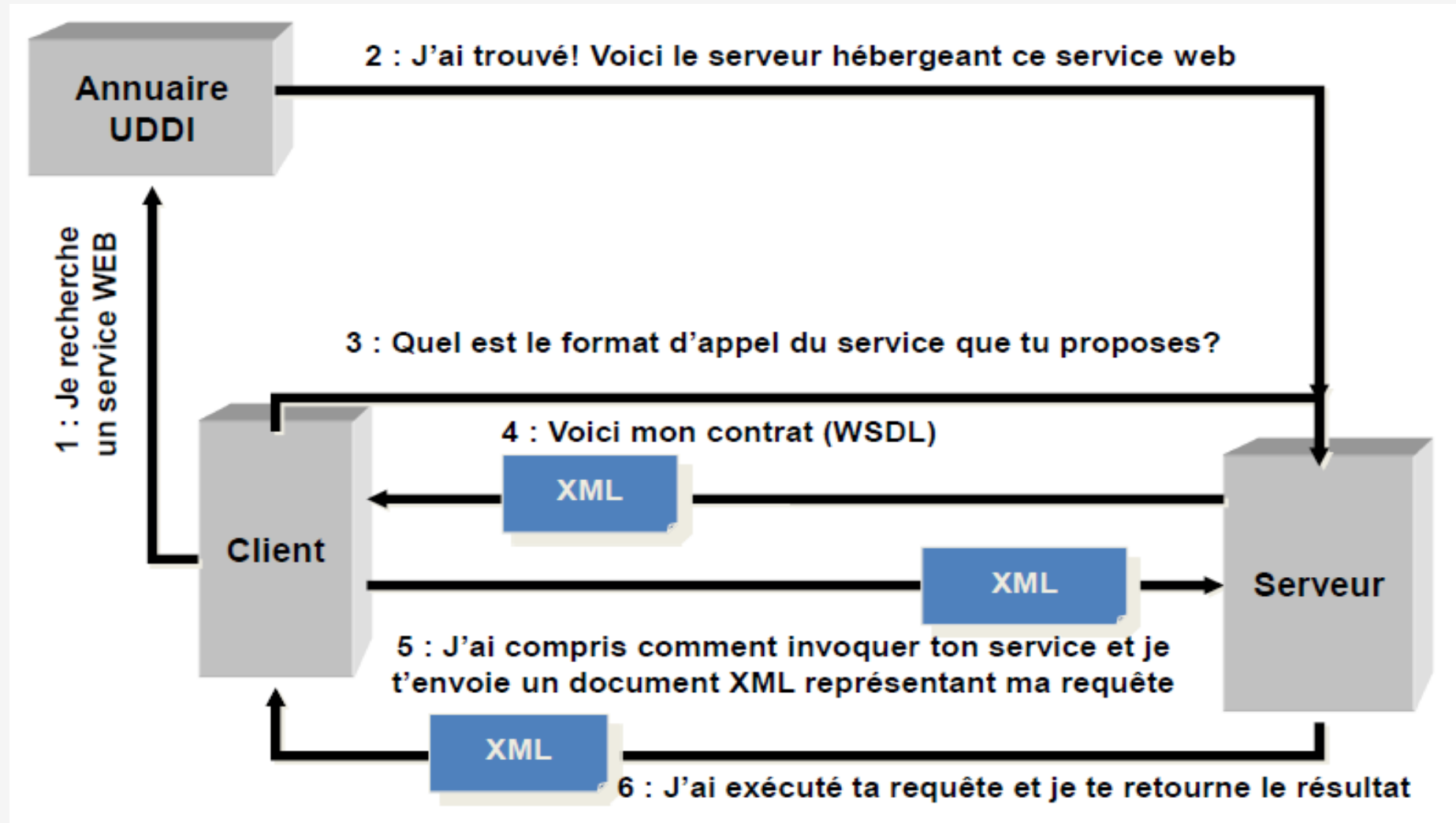
WSDL (Web Services Description Language) : offre un schéma formel de description des Services Web.

UDDI (Universal Description, Discovery and Integration) : offre une manière uniforme de définir des registres des Services Web et un schéma uniformément extensible de descriptions des Services Web.

Infrastructure des Services Web



Cycle de vie d'utilisation d'un Service Web



Types de Services Web

- **Services Web 1.0 (Première génération = Etendus) :**
utilisent les standards **UDDI** (annuaire) / **WSDL**(contrat) / **SOAP** (consommation)
- **Services Web 2.0 (Deuxième génération = REST(Representational State Transfer)) :**
utilisent :
 - Directement **HTTP** au lieu d'une enveloppe SOAP
 - Un **URI** pour nommer et identifier une ressource
 - Les méthodes HTTP (**POST**, **GET**, **PUT** et **DELETE**) pour effectuer les opérations de base CRUD
- REST n'est pas un remplacement générique de SOAP (ne couvre qu'une partie des besoins)
- REST utilise **WADL** (Web Application Description Language) pour décrire les Contrats, et qui est **non standardisé** : initiative isolée de SUN

HTTP (Hyper Text Transfer Protocol)

Protocole de communication dédié au web

Chaque ressource du web est identifiée par une URL

Mode de communication = requête / réponse



Exemples de Services Web existants

Google (<http://www.google.com/apis/>) :

- Accès gratuit mais limité (1000 requêtes par jour après enregistrement)

Amazon (<http://aws.amazon.com/fr/>) • accès gratuit mais limité (1 requête par seconde après enregistrement)



Bien d'autres (<http://webservicex.net> par exemple)

Implémentation des Services Web

Côté fournisseur

Pour créer un Web Service :

1. Définir le contrat du service
2. Développer le service
3. Développer la couche de traitement XML
4. Déployer sur le serveur
5. Publier dans l'annuaire



Suivant les technologies,
certaines tâches sont
automatisées

Implémentation des Services Web

Côté client

Pour créer une application cliente :

1. Rechercher le service dans l'annuaire
2. Récupérer le contrat du service
3. Développer la couche de traitement XML
4. Utiliser le service et présenter les résultats (rendu)



Suivant les technologies,
certaines tâches sont
automatisées

Tâches relatives aux Services Web

Invocation : vise à établir la communication entre le client et le fournisseur en décrivant la structure des messages échangés.

Découverte : permet de localiser un service web particulier dans un annuaire de services décrivant les fournisseurs ainsi les services fournis.

Composition : consiste à combiner des services pour former un nouveau service dit composé ou composite. Le but de la composition est avant tout la réutilisation de services (simples ou composés) et de préférence sans aucune modification de ces derniers.

Sélection : choisir parmi les services web découverts, ceux qui répondent au mieux aux exigences de l'utilisateur sur la base des besoins fonctionnels et/ou non fonctionnels.