

Langages de description d'application (services) et catalogue

WSDL / OWL-S
UDDI

Tatiana Aubonnet

Plan

- ❖ Fonctionnement des services web
- ❖ Description syntactique des service :
 - **WSDL** (Web Services Description Language)
 - ❖ Annuaire de services :
 - **UDDI** (Universal Description, Discovery and Integration)
 - Dragon
 - SemRe utilise SAWSDL pour intégrer la partie sémantique dans l'annuaire
 - Seedka
 - Fusion
- ❖ Description sémantique des service :
 - **OWL-S** (Ontology Web Langage for web Services)
 - **SAWSDL** (Semantic annotation for *WSDL*)
- ❖ Communication : **SOAP** (Simple Object Access Protocol)

Qu'est-ce qu'un service web ?

Né du besoin de communiquer de programme à programme en utilisant comme support la toile mondiale.

Un service Web est un service offert par une application à d'autres applications via le Web.

Une application informatique quelconque qui rend ses services disponibles sur un réseau en utilisant des technologies et protocoles standards largement utilisés comme XML et HTTP

Scénario pour les services Web

Une application métier envoie une requête à un service situé à une URL donnée en utilisant le protocole SOAP sur HTTP.

Le service reçoit la requête, la traite, et retourne une réponse.

Exemple : un service propose de calculer une route optimisée pour des livraisons

Une requête contenant la destination finale de la livraison.

Cette requête est traitée par le service web qui détermine la route la moins coûteuse

La réponse est, en général, retournée comme une opération indépendante de la requête

Exemples d'applications

Commerce électronique : paiement sécurisé

Commerce électronique : calcul de frais de port

Commerce électronique : suivi de commandes

Offre de services : conversion de devises

Offre de services : tout service comportant un accès à des données (température, trafic, ...) ou une algorithmique plus ou moins complexe.

Commerce électronique : simulation de crédits

2 types d'utilisation

Proposer des composants réutilisables :

- Bon nombre de composants sont souvent utiles dans diverses applications
- Plutôt que les réécrire, ils sont offerts en tant que service sur le Web

Résoudre les problèmes d'interopérabilité entre applications :

- échange de données entre différentes applications et différentes plate-formes

Pour fonctionner

XML + HTTP

Producteurs et consommateurs de services Web sont capables de communiquer même s'ils utilisent des SI (Systèmes d'Informations) différents.

Ils peuvent aussi utiliser des plate-formes d'exécution différentes

Une application utilisant les services Web peut publier aussi bien une fonction qu'un message vers le reste du monde

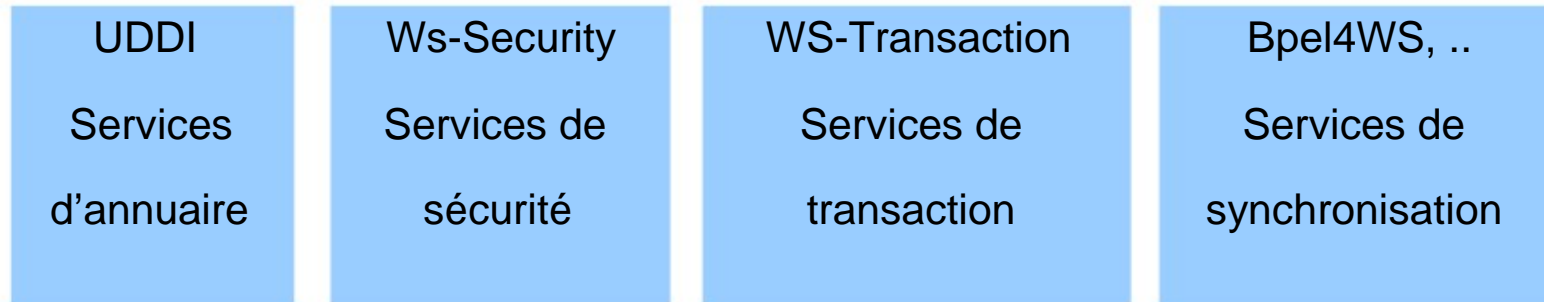
XML rend les données portables.

XML est un langage capable d'exprimer des messages complexes et qui peut être utilisé entre différentes plate-formes et langages de programmation

Les services Web utilisent XML pour coder puis décoder des données

HTTP est le protocole le plus largement répandu sur l'Internet

Organisation actuelle des standards



SOAP, WSDL : Interactions de communication

HTTP, SMTP, MOM(JMS) : Transmission effective

SOAP (Simple Object Access Protocol)

UDDI (Universal Description, Discovery and Integration)

WSDL (Web Services Description Language)

SOAP (Simple Object Access Protocol)

SOAP est un protocole de communication

SOAP définit un format pour l'envoi de messages

SOAP est conçu pour des communications via Internet

SOAP est indépendant de toute plate-forme d'exécution

SOAP est indépendant de tout langage de programmation

SOAP est fondé sur XML

SOAP utilise des protocoles applicatifs Internet pour acheminer ses messages (HTTP ou SMTP et MOM pour le mode asynchrone)

SOAP n'est pas bloqué par les murs anti-feux (« firewalls »)

SOAP est un standard W3C

WSDL (Web Services Definition Language)

WSDL est un standard W3C de description d'interfaces de services

WSDL permet de décharger les utilisateurs des détails techniques de réalisation d'un appel

WSDL est fondé sur XML et XML Schéma

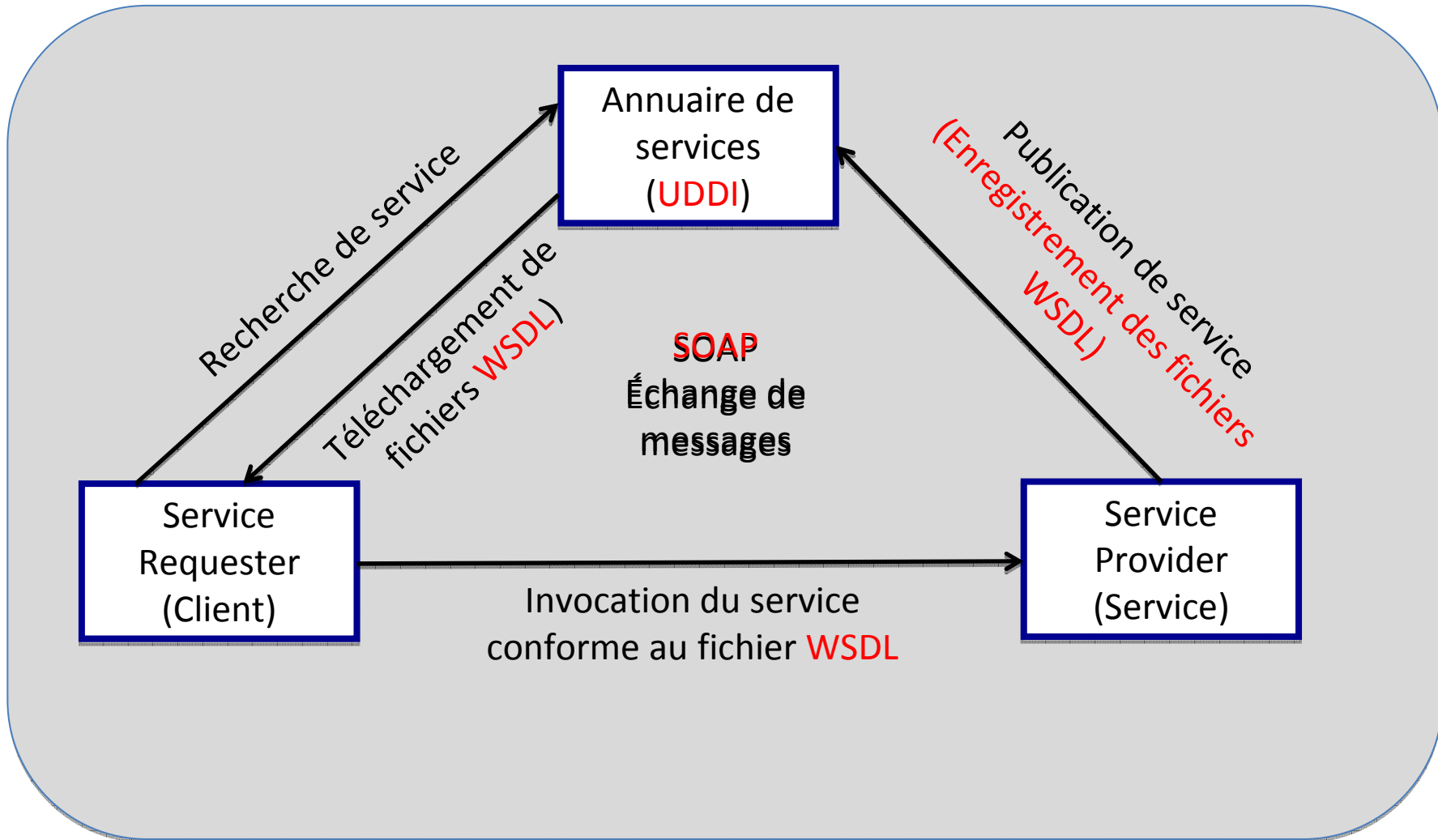
UDDI (Universal Description, Discovery and Integration)

UDDI est un annuaire réparti pour la description de services Web

Dans cet annuaire, les services Web sont décrits avec WSDL

UDDI est accessible au moyen de SOAP

Fonctionnement des services web



Fonctionnement des Services web

- **La publication du service** : le fournisseur diffuse les descriptions de ses services Web dans le catalogue (envoi le fichier WSDL vers le catalogue).
- **La recherche du service** : le client cherche un service particulier dans un annuaire qui va lui fournir les descriptions et les URL des services demandés (Récupère le fichier WSDL à partir du catalogue).
- **L'invocation du service** : une fois que le client récupère l'URL et la description du service, il peut l'invoquer auprès du fournisseur de services.

L'échange de messages se fait selon la spécification SOAP (basé sur xml)

le cnam

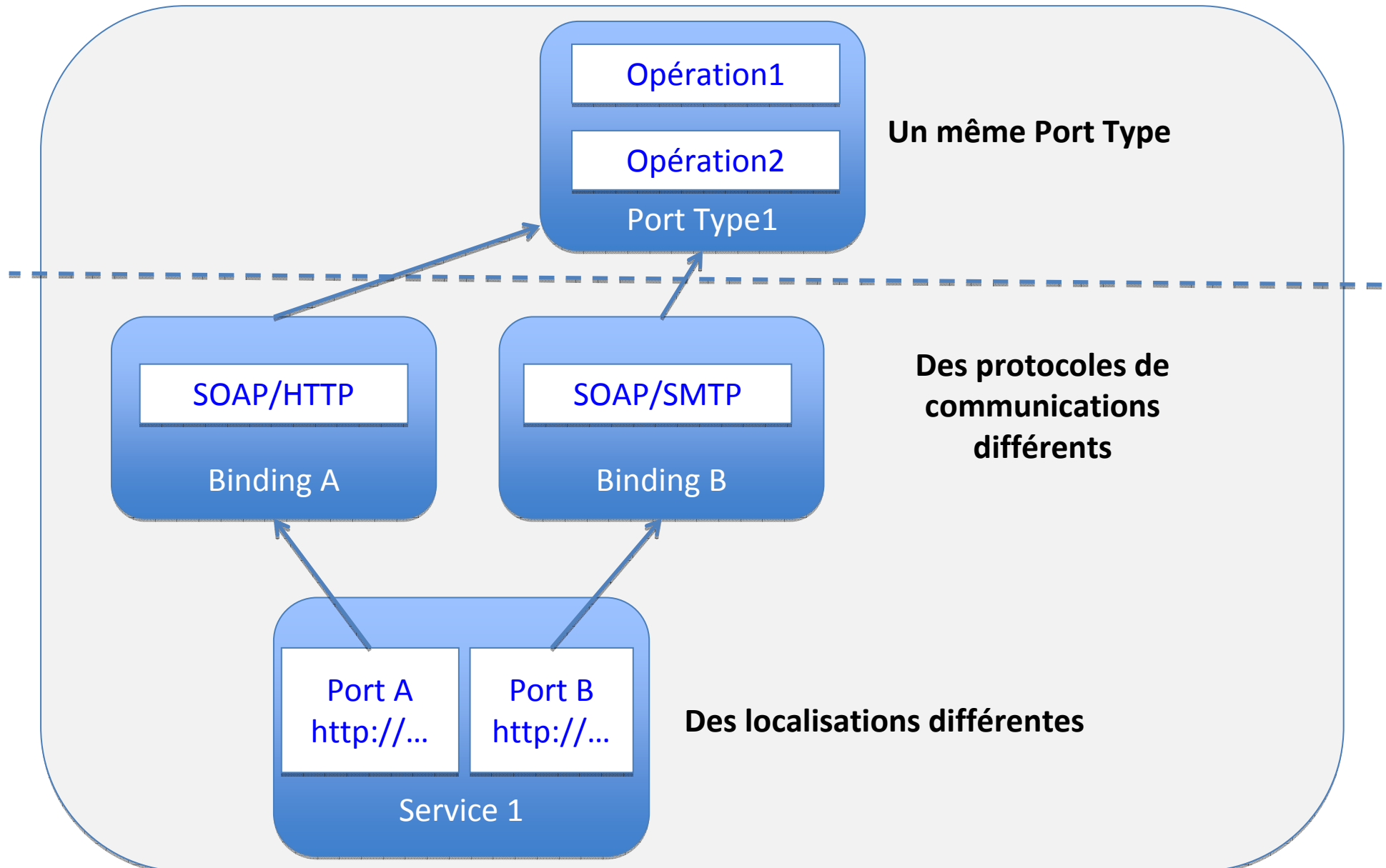
Le langage WSDL

- décrit l'interface du service au format XML
 - les méthodes, les paramètres et valeurs retournées,
 - les protocoles de transport possibles, et la localisation du service
- est généré par les outils de développement
- reste indépendant des plateformes et langages d'implémentation des services
 - Faible couplage

Éléments d'un fichier WSDL

- **Définitions** : Racine d'un document WSDL
- **Types** (optionnel) : définitions de types de données échangées sous forme de xml schema.
- **Message** : définition abstraite des données transmises.
- **portType** : ensemble d'opérations correspondant chacune à un message entrant ou sortant ou de fautes.
- **binding** : protocole de communication et format des données échangées pour un port. Spécifie une liaison d'un <porttype> à un protocole concret (HTTP, SOAP).
- **port** : spécifie un point d'entrée (endpoint) comme la combinaison d'un <binding> et d'une URI.
- **service** : regroupe un ensemble de ports (endpoint).

Organisation d'un document WSDL



Exemple d'un document WSDL

```
<definitions name="HelloService"
  targetNamespace="http://www.examples.com/wsdl/HelloService.wsdl"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://www.examples.com/wsdl/HelloService.wsdl"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <message name="SayHelloRequest">
    <part name="firstName" type="xsd:string"/>
  </message>

  <message name="SayHelloResponse">
    <part name="greeting" type="xsd:string"/>
  </message>
```

Suite de l'exemple

```
<binding name="Hello_Binding" type="tns:Hello_PortType">
  <soap:binding style="rpc"
    transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="sayHello">
    <soap:operation soapAction="sayHello"/>
    <input>
      <soap:body
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        namespace="urn:examples:helloservice"
        use="encoded"/>
    </input>
    <output>
      <soap:body
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        namespace="urn:examples:helloservice"
        use="encoded"/>
    </output>
  </operation>
</binding>
```

Suite de l'exemple

```
<portType name="Hello_PortType">
  <operation name="sayHello">
    <input message="tns:SayHelloRequest"/>
    <output message="tns:SayHelloResponse"/>
  </operation>
</portType>

<service name="Hello_Service">
  <documentation>WSDL File for HelloService</documentation>
  <port binding="tns:Hello_Binding" name="Hello_Port">
    <soap:address
      location="http://www.examples.com/SayHello/" />
  </port>
</service>
</definitions>
```

Exemple : HelloService

- **Definitions** : nom du service web : HelloService
- **Type** : définits en XML Schema (ici non requis car un String (paramètre World)).
- **Message** :
 - sayHelloRequest : firstName parameter
 - sayHelloresponse: greeting return value
- **Port Type** : **opération** sayHello qui consiste en un service de requête/réponse.
- **Binding** : utiliser SOAP HTTP transport protocol pour transporter l'opération (PortType) sayHello. (on peut avoir plusieurs binding pour un portType).
- **Service** : Service disponible à <http://www.examples.com/SayHello/>
- **Port** : Associe le binding avec l'URI <http://www.examples.com/SayHello/>.

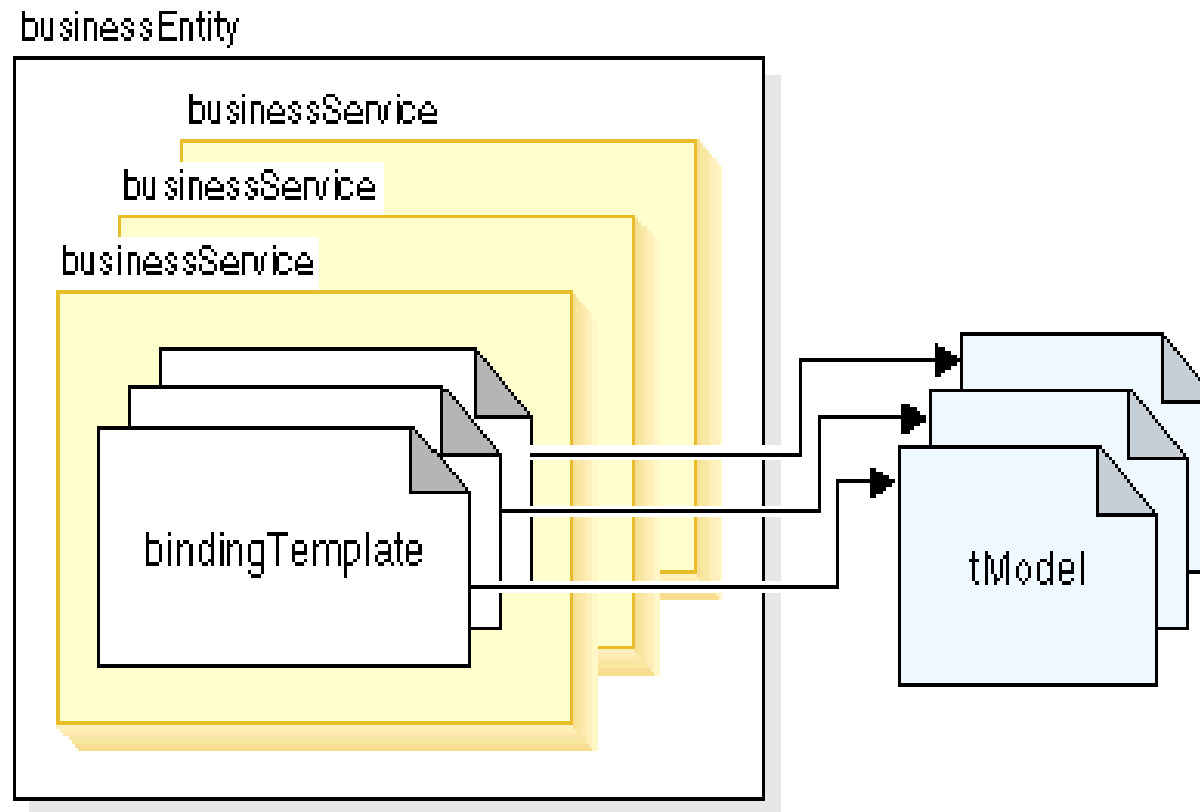
UDDI

- ❑ UDDI est une norme d'annuaire de services Web.
- ❑ Grammaire XML (schéma XML)
Publication/ recherche basée sur SOAP et WSDL
- ❑ Pour publier un nouveau service Web, il faut générer un document appelé **Business Registry**, il sera enregistré sur un UDDI Business Registry Node (IBM et Microsoft en hébergent chacun un).
- ❑ Les nodes sont répliqués entre eux suivant un mécanisme analogue aux DNS.
- ❑ Le Business Registry comprend 3 parties : **pages blanches**, **pages jaunes** et **pages vertes**.

Structures de données UDDI

- ❑ **BusinessEntity** (pages blanches) : décrivent les organisations ayant publié des services dans le catalogue (nom, adresses physiques et Web, identifiants des entreprises, contacts).
- ❑ **BusinessService** (pages jaunes) : décrivent la partie non technique des services proposés (description textuelle des services, référence à l'organisation proposant le service et un ou plusieurs « bindingTemplate »).
- ❑ **BindingTemplate** (pages vertes) : Donnent les coordonnées des services (point d'accès (URL) et les éventuels « tModels » associés).
- ❑ **tModel (index)** : contient les descriptions techniques des services (fichier WSDL).

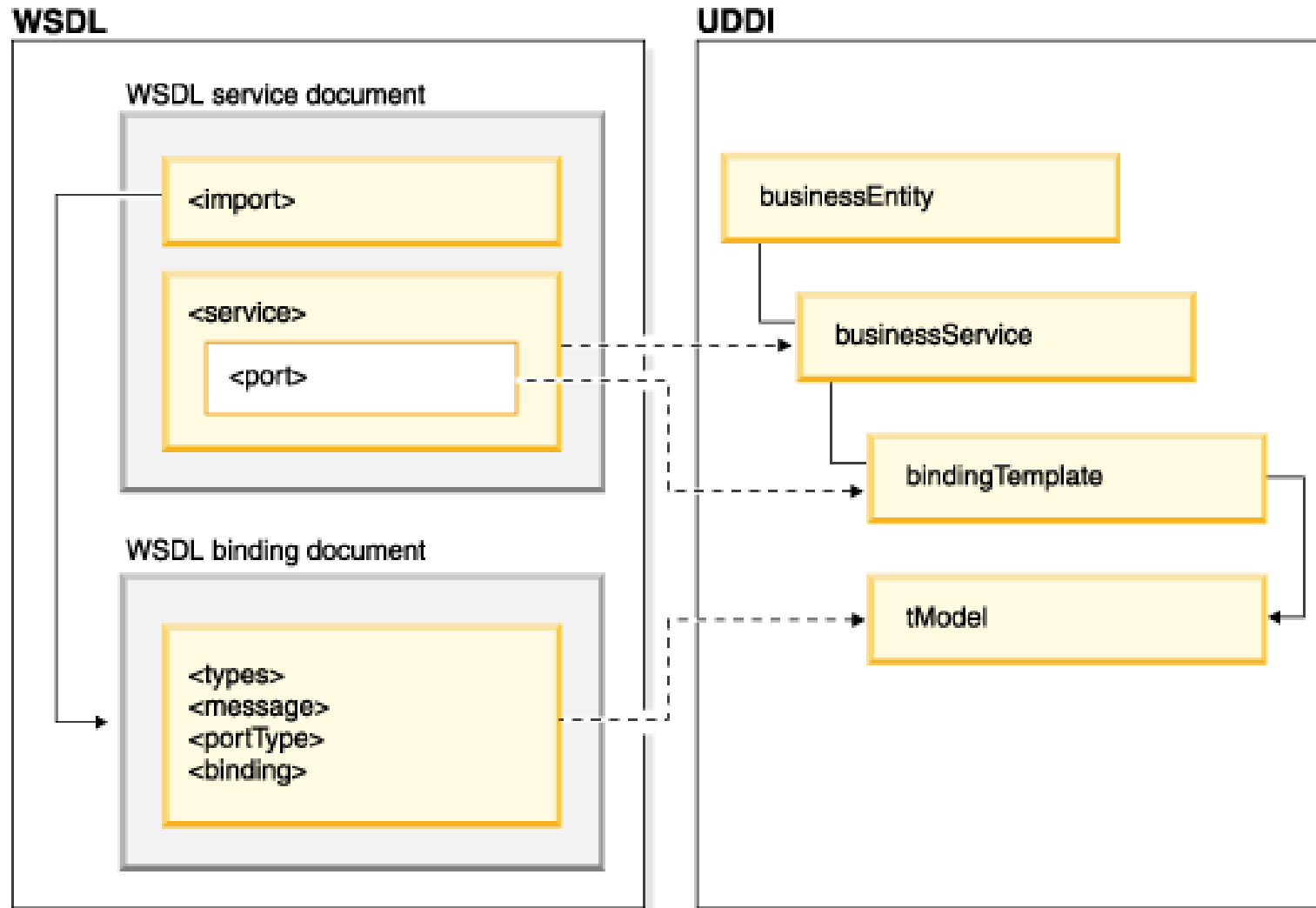
Structure de données UDDI



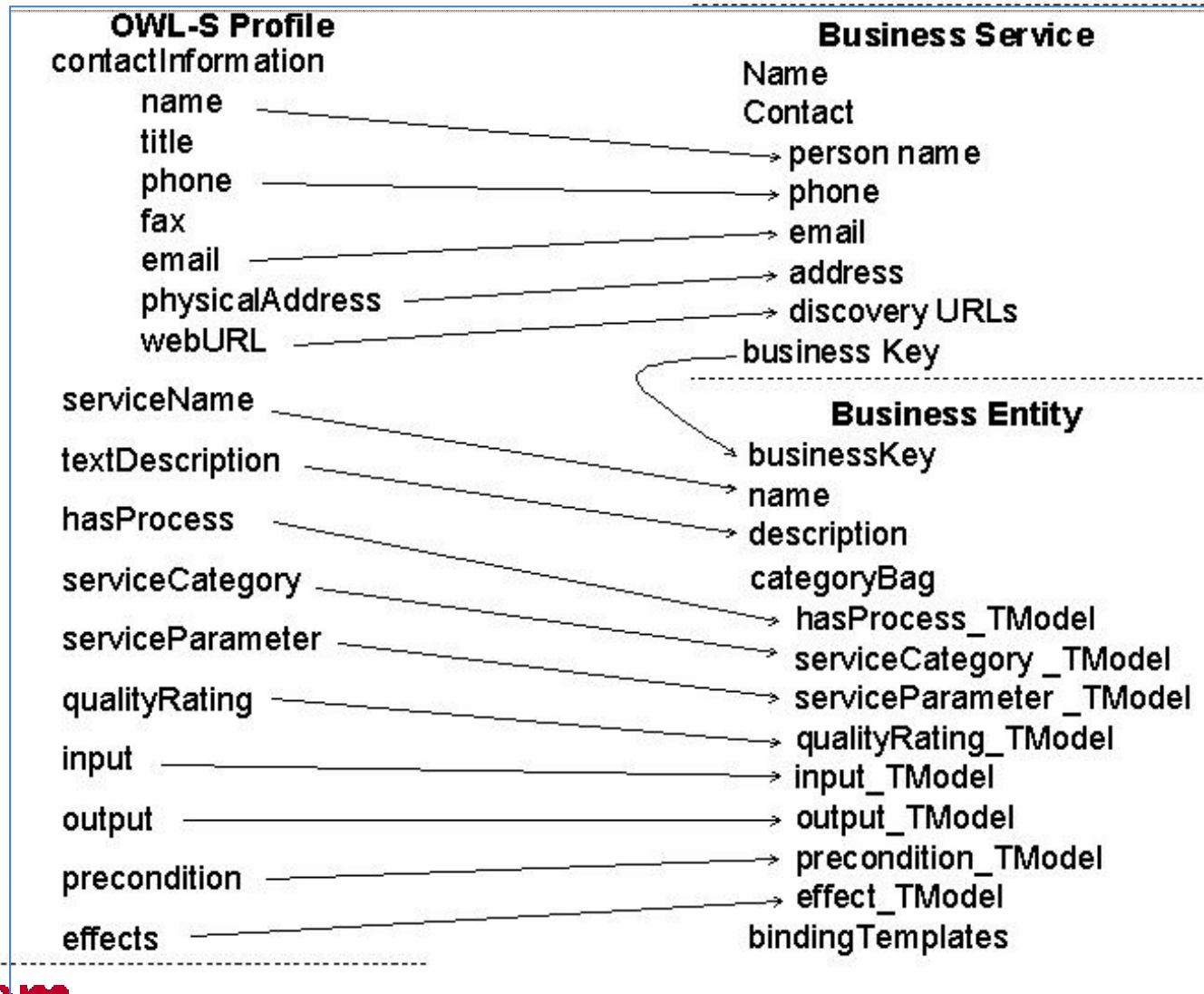
Outils UDDI

- Utilise 3 APIs de base :
 - **publish** pour enregistrer un nouveau service,
 - **find** pour interroger l'annuaire,
 - **bind** pour effectuer la connexion entre l'application cliente et le service.
- Outils :
 - jUDDI
 - UDDI4j (open source)

Relation entre WSDL et UDDI



Relation entre OWL-S et UDDI



Annuaire de services

DRAGON

FUSION

Seekda

SemRe

Dragon (Annuaire distribué open source)

- Dragon est basé sur UDDI
- A la conception des services Web, il offre un annuaire pour :
 - stocker des informations sur les services, les contrats SLA et d'autres méta-données telles que les propriétés sémantiques,
 - permettre une recherche de service et une découverte basées sur les méta-données, ainsi que la gestion du cycle de vie du service,
 - fournir des fonctionnalités facilitant les mises à jour des processus et du service.
- - A l'exécution des services, il offre une interface de plateforme de services pour :
 - appliquer les politiques des attributs de qualité de service (QoS),
 - appliquer les SLA aux contrats des clients/fournisseurs,
 - gérer les versions des services,
 - suivre les indicateurs de SOA : QoS, la (ré)utilisation de service, le temps de développement,
- A la publication, la catégorisation des services est automatique.
 - Les fournisseurs n'interviennent pas pour catégoriser les services Web qu'ils souhaitent publier.
- A la découverte, un appariement est effectué entre des mots-clés précisés par l'utilisateur et des mots-clés associés aux services publiés dans l'annuaire.

FUSION Semantic Registry

- Est basé sur UDDI. Le projet FUSION est un projet européen de recherche avec des partenaires académiques et industriels.
- Il a pour objectif de construire une solution basée sur la sémantique pour l'interopérabilité des applications métiers orientées services.
- Enrichit avec des informations de descriptions sémantiques.
- Pendant la phase de publication, la catégorisation des services est automatique.
- Il utilise ces mécanismes sémantiques afin d'améliorer les capacités de publication et de découverte. Pour ce faire, il utilise la spécification SAWSDL afin de créer des descriptions annotées sémantiquement pour les interfaces de services.
- FUSION Semantic Registry utilise plusieurs bibliothèques de logiciels open source (jUDDI, UDDI4J, WSDL4J, SAWSDL4J, OWL API) et lui-même publiée sous forme de logiciel libre sous la licence Apache.

SemRe (Semantic Registry)

- est une plateforme qui permet essentiellement de stocker et de sélectionner des services en fonction de critères fonctionnels et non-fonctionnels.
- intègre les descriptions sémantiques des propriétés fonctionnelles des services en utilisant SAWSDL.

Seekda (pas basé sur UDDI)

- Pendant la phase de publication, la catégorisation des services est manuelle. Les fournisseurs interviennent pour catégoriser les services Web qu'ils souhaitent publier.
- Seekda demande de fournir l'URL du WSDL du service ensuite il essaie d'extraire automatiquement des informations techniques et fonctionnelles à partir du WSDL. Ensuite, il propose au fournisseur d'ajouter des mots-clés (tags) et/ou une description compréhensibles par les utilisateurs potentiels du service.
- Une fois le service Web enregistré, le fournisseur peut ajouter d'autres informations qui concernent son identité et/ou l'utilisation du service.
- Chaque service possède une fiche composée de quatre rubriques : *Overview* pour des informations générales telle que l'adresse URL, *Use Now* pour des détails sur les interfaces et les opérations du service, *Availability* pour des statistiques sur la disponibilité et *Comments* pour les commentaires des utilisateurs.
- Le portail Seekda propose une recherche basique de services par mot-clé et une recherche avancée qui permet de découvrir les services à partir d'un nuage de tags, découvrir les fournisseurs par pays, découvrir les services Web les plus utilisés ou naviguer à travers les services

Exercice:

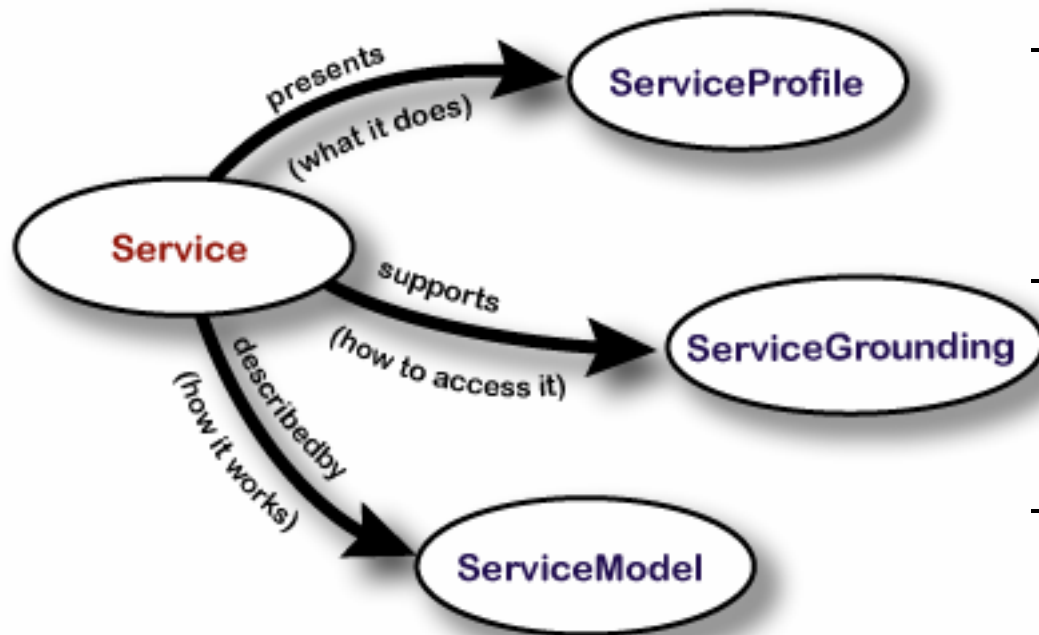
Réaliser un comparatif des annuaires de services

UDDI						
FUSION						
SemRe						
DRAGON						

Langage OWL-S

- Description sémantique des services.
- Référence à des ontologies (OWL).
- Intègre le langage WSDL.
- Description, découverte et Composition de services.
- Décrit avec précision la partie fonctionnelle d'un service mais limité pour décrire la partie non fonctionnelle.
- Des extensions de OWL-S sont proposées pour décrire la QoS par exemple ou d'autres caractéristiques.

Structuration de l'ontologie de OWL-S



Service profile

- Répond à: "Que fait le service?"
Description de haut niveau, utile pour les registres.
- Permet de sélectionner les services selon les propriétés fonctionnelles et non fonctionnelles.
- Les recherches de services peuvent se baser sur n'importe quel élément de *Profile* comme critère .
- utilisée à la fois par les fournisseurs pour publier leurs services et par les clients pour spécifier leurs besoins.

Service model

- Répond à: "Comment fonctionne le service?"
- Décrit les interactions du service.
- modélise les services en tant que processus.
 - (atomique, simple ou composite) process

Utilisation

- Invocation, composition, monitoring.

Service Grounding

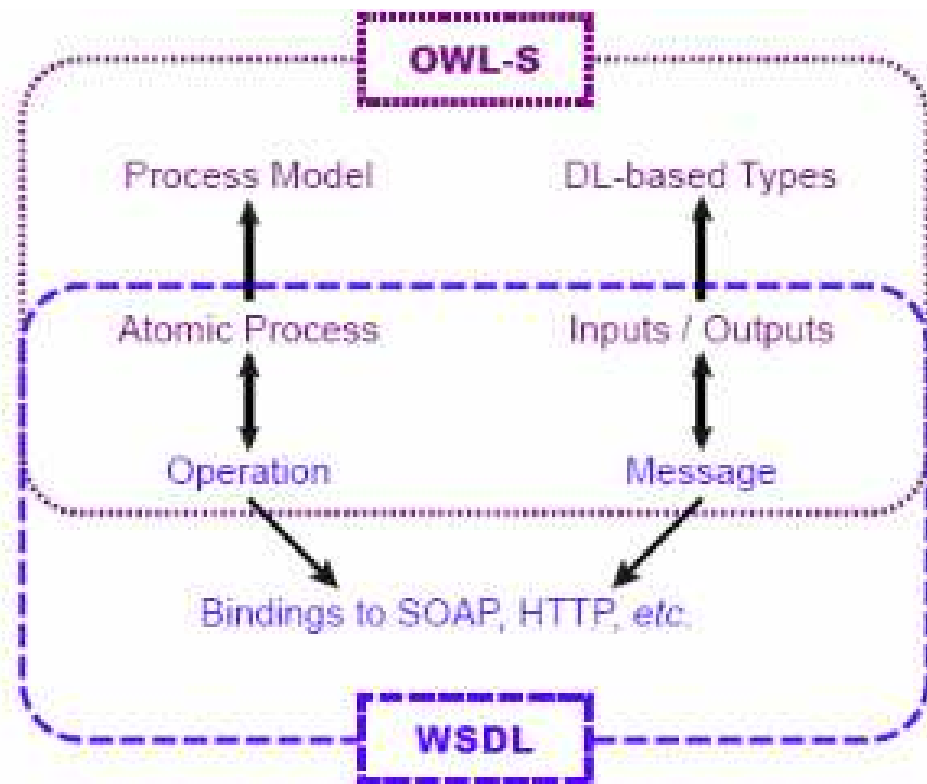
- Répond à: "Comment accéder au service?"
- Spécifique à l'implémentation.
- Formats des messages, protocoles de transport, adresses physiques.

Utilisation

- pour la découverte de service.
- On peut invoquer un service avec son "Model" et son "Grounding", construits autour de WSDL.

OWL-S et WSDL

OWL-S utilise le WSDL pour une description complète du Service Grounding



- Un processus atomique **OWL-S** correspond à une **opération WSDL**;
- Les Inputs et Outputs d'un processus atomique **OWL-S** correspondent aux messages **WSDL**;
- Les types des Inputs/Outputs d'un processus atomique **OWL-S** correspondent aux types abstraits **WSDL**.

Langages de description de services

- **SWORD (Software Ontology For Resource Description)** est un système expert basé sur des règles pour la composition des services web. SWORD peut composer des services automatiquement, mais il identifie des services uniquement par syntaxe - c'est-à-dire par leurs entrées et sorties.
- **WSMO (Web Service Modeling Ontology)** : est une ontologie qui décrit les différents aspects relatifs à l'invocation, la composition dynamique, la découverte et la médiation (ajout par rapport à OWL-S) des services Web. La médiation permet d'établir la correspondance entre la demande et l'offre.
- **SAWSDL (Semantic Annotation for WSDL)**: fournit un mécanisme permettant d'annoter sémantiquement les types de données, les opérations, les entrées et les sorties de *WSDL* et d'autre part, il ajoute des éléments pour spécifier les pré-conditions et les catégories des services Web. Les aspects relatifs à la qualité et l'orchestration des services ne sont pas traités dans *SAWSDL*.

SOAP

- Le protocole SOAP Assure les appels de procédure.
- Au dessus d'un protocole de transport (HTTP)
- Simple enveloppe autour des données à transmettre via HTTP.

- SOAP Côté client
 - Ouverture d'une connexion HTTP
 - Requête SOAP: document XML décrivant
 - la méthode à invoquer sur la machine distante
 - les paramètres de la méthode

- SOAP Côté Serveur
 - Récupère la requête
 - Exécution de la méthode avec les paramètres
 - Renvoie une réponse SOAP (document XML) au client

Structure d'un message SOAP

