

Découverte des Services Web étendus - SOAP

Module SOA
A.U 2025-2026

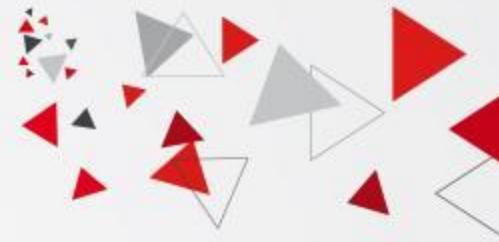


Délivrée par la
Commission
des Titres
d'Ingénieur





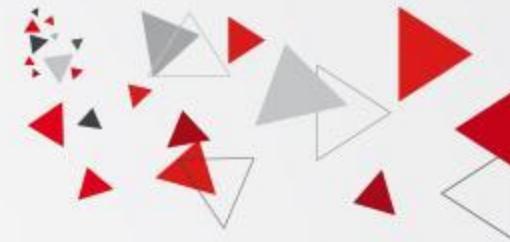
Objectif



- Comprendre le principe des services web étendus
- Comprendre la structure et les différents éléments d'un document WSDL

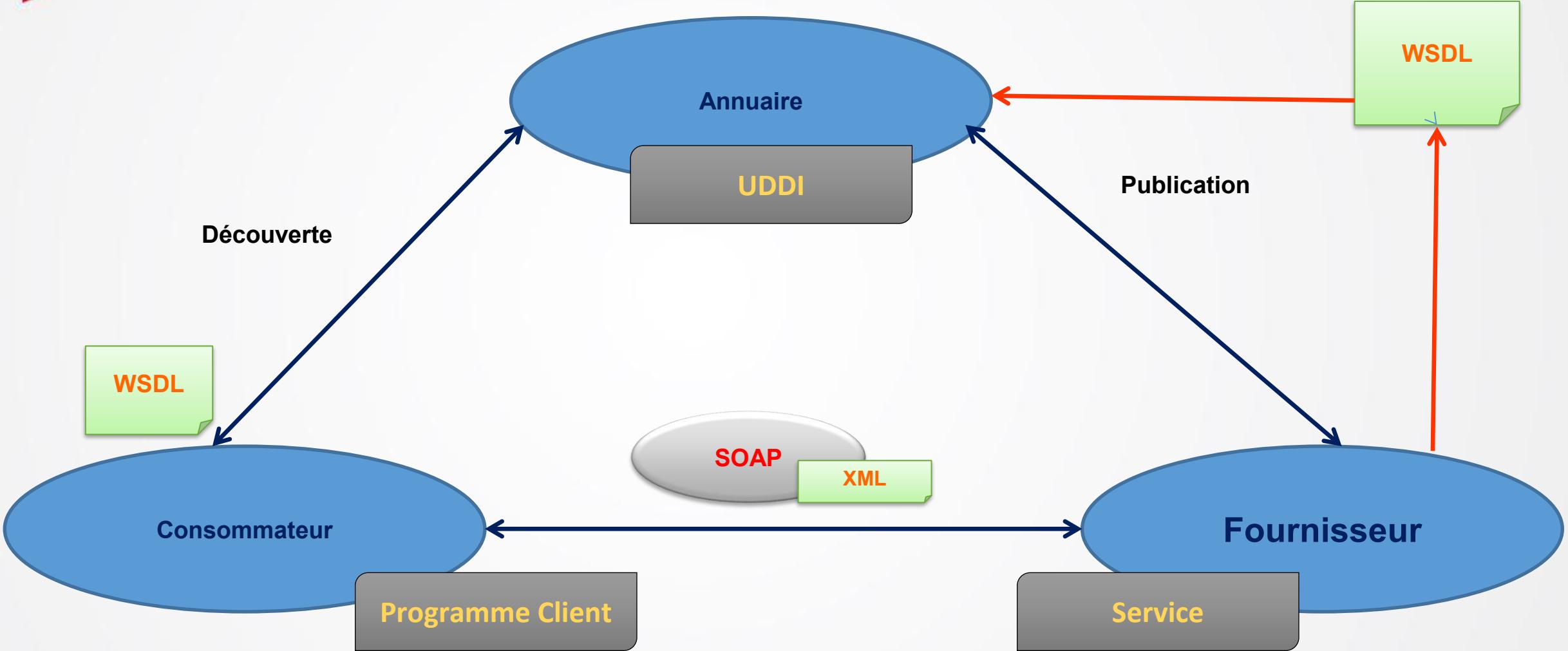


Plan



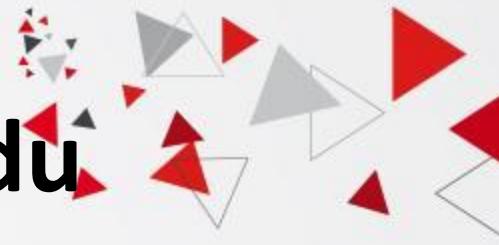
- Introduction
- Les standards des services Web étendus
- Structure du contrat WSDL et des messages SOAP

Introduction





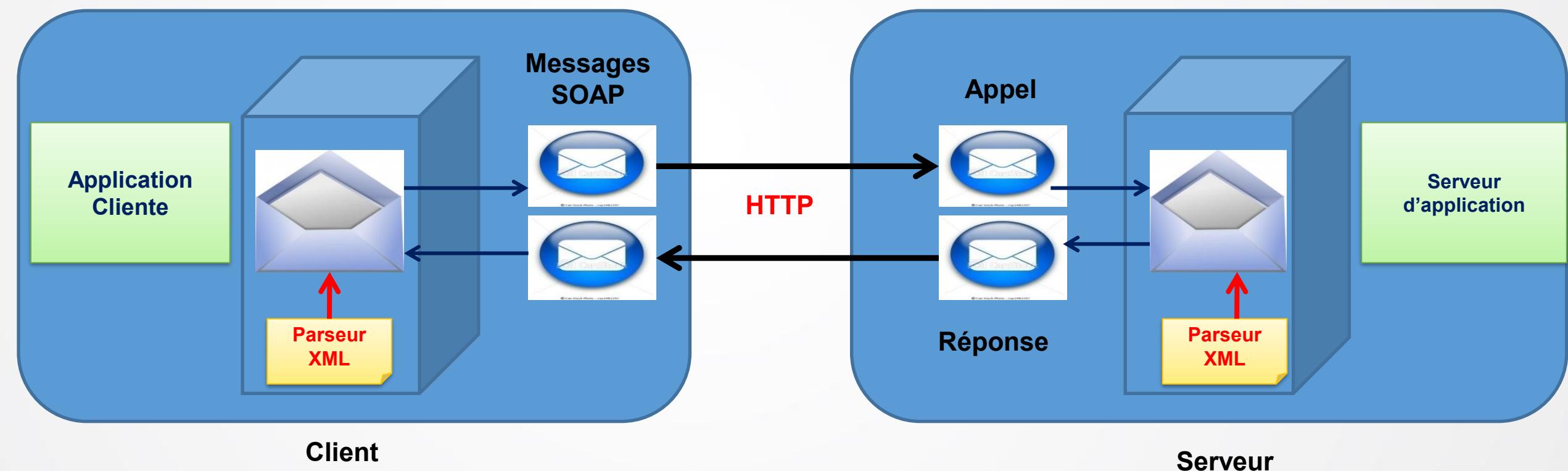
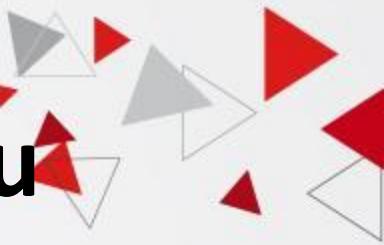
Les standards d'un service Web étendu



Afin de mettre en œuvre un service web, trois composants sont nécessaires :

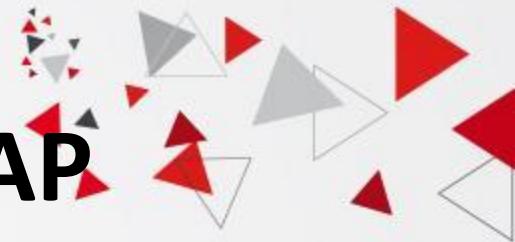
- Un langage de description de services WSDL(**Web Service Description Language**)
- Un protocole de communication SOAP (**S**imple **O**bject **A**cce~~s~~ **P**rotocol)
- Un annuaire UDDI (**U**niversal **D**escription, **D**iscovery and **I**ntegration) pour le référencement du Web Service

Fonctionnement d'un Service Web étendu





Développement de Service Web SOAP



- Deux façons pour développer un Service Web étendu:

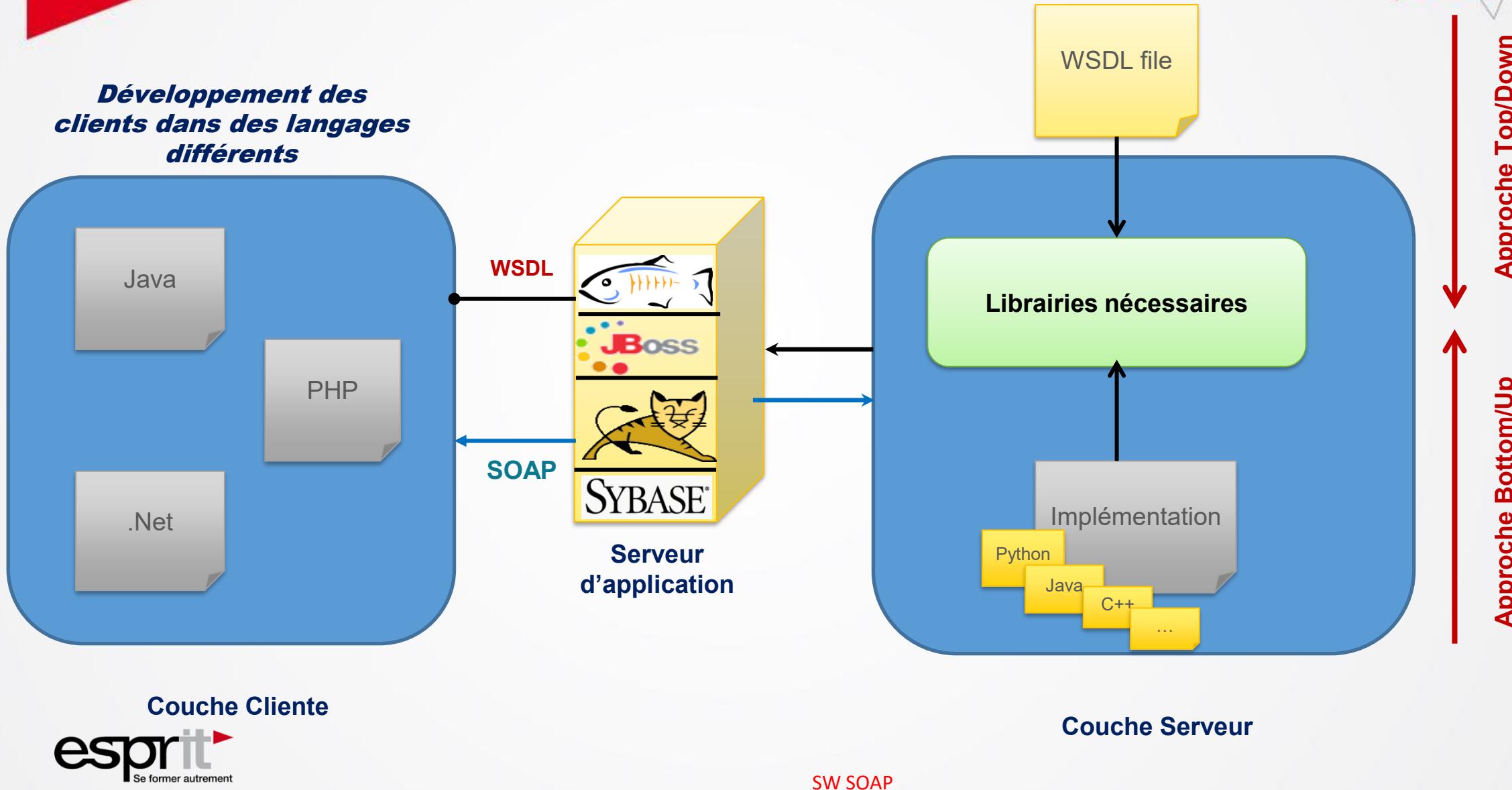
❖Approche Bottom / Up

- ✓ Implémenter l'application.
- ✓ Compiler, déployer et tester.
- ✓ Le document WSDL est automatiquement généré.

❖Approche Top / Down (à partir d'un document WSDL)

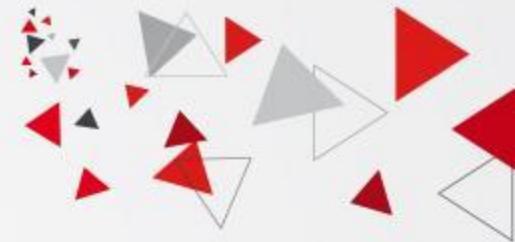
- ✓ Génération des différentes classes et squelette du Web Service en utilisant l'outil de parsing.
- ✓ Compléter le squelette de classe de l'implémentation.
- ✓ Compiler, déployer et tester.

Développement de Service Web SOAP





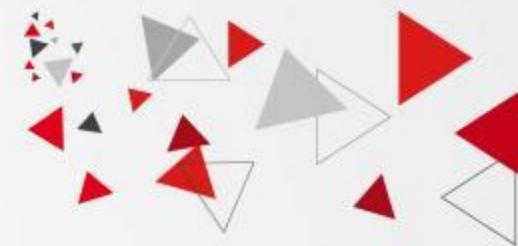
Présentation de WSDL



- **Web Service Description Language**
- Un langage de description des services web
- Une spécification W3C
- Basé sur XML
- Une description de type « boîte noire » du service web:
 - Quelles sont les opérations disponibles, leurs entrées, leur sorties, leur types?
 - Comment y accéder : adresse(s), protocole(s) utilisé(s),...
 - Quel est le format des échanges : les messages (requête/réponse)



Structure d'un contrat WSDL



Un document WSDL se structure en **2 parties** :

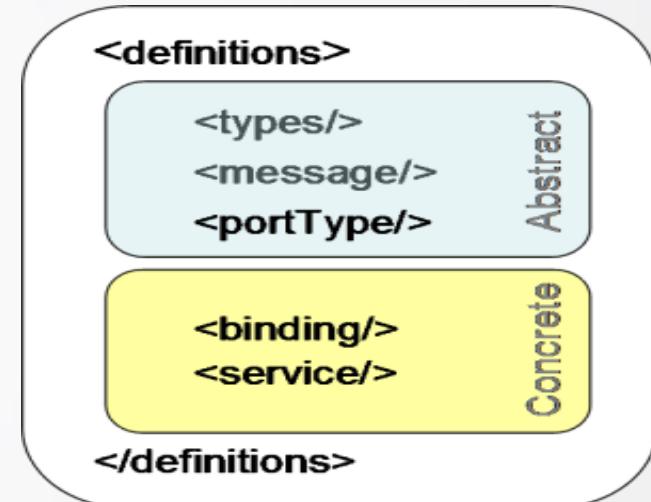
- **Une description concrète**

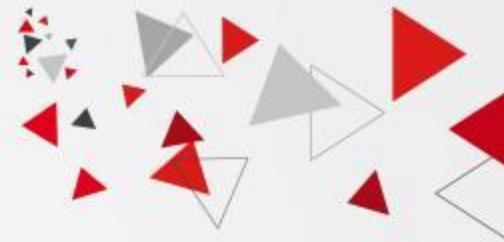
- Définition du protocole d'accès et de l'URI à partir de laquelle on peut accéder au service web

- **Une description abstraite**

- nom des opérations, paramètres d'entrée, de sortie, structure des messages

Contrat WSDL





Structure d'un contrat WSDL

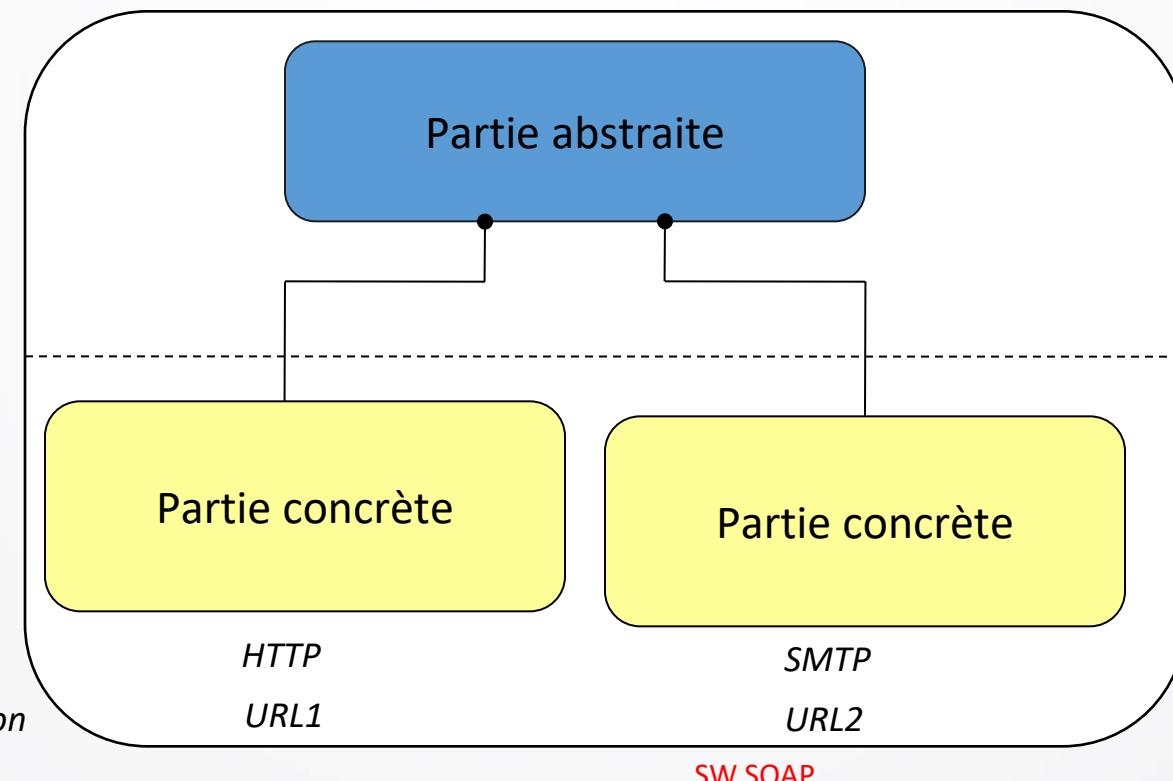


Avantages d'une structuration en 2 parties :

On peut associer la description abstraite avec plusieurs descriptions concrètes.

→ Réutilisabilité de la description abstraite

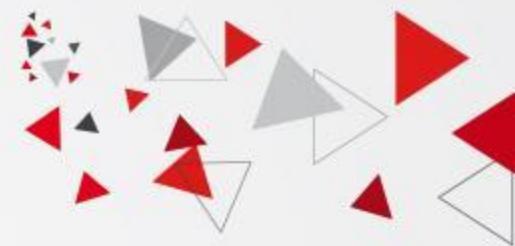
Contrat WSDL



Deux parties concrètes sont associées avec la même partie abstraite.



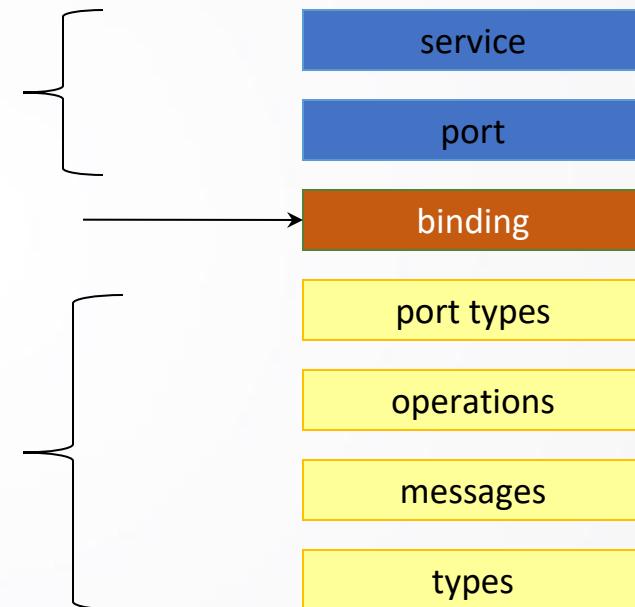
Les éléments WSDL



- Le document WSDL contient 7 éléments :

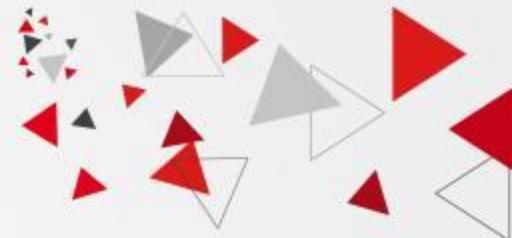
Description **concrète**
Elément de **liaison** entre la description abstraite et concrète

Description **abstraite**





Exemple d'un contrat WSDL



```
<?xml version="1.0" encoding="utf-8"?>
| | | | <!-- Structure d'un contrat WSDL -->
<wsdl:definitions >

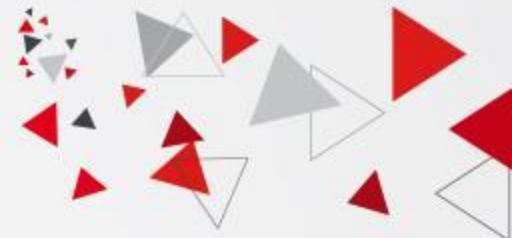
    <!-- Definition abstraite -->
    <wsdl:types> ..... </wsdl:types>
    <wsdl:message > ..... </wsdl:message>
    <wsdl:message > ..... </wsdl:message>
    <wsdl:message > ..... </wsdl:message>
    <wsdl:message > ..... </wsdl:message>
    <wsdl:portType>
        <wsdl:operation name="operation1"> .... </wsdl:operation>
        <wsdl:operation name="operation2"> .... </wsdl:operation>
    </wsdl:portType>

    <!-- Definition concrete -->
    <wsdl:binding > ..... </wsdl:binding>
    <wsdl:service name="monService">
        <wsdl:port>
            <soap:address location="http://swLocation" />
        </wsdl:port>
    </wsdl:service>

</wsdl:definitions>
```



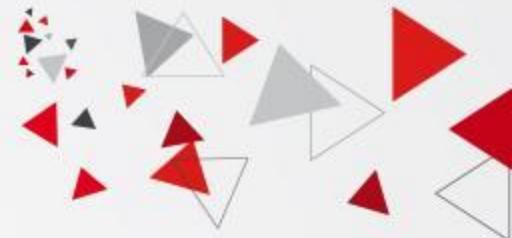
L'annuaire UDDI



- **UDDI: Universal Description, Discovery, and Integration)**
- *“A platform-independent, XML-based registry for businesses worldwide to list themselves on the Internet. UDDI is an open industry initiative (sponsored by OASIS) enabling businesses to discover each other and define how they interact over the Internet”*
- UDDI représente actuellement la couche de découverte dans le « web service protocol stack »



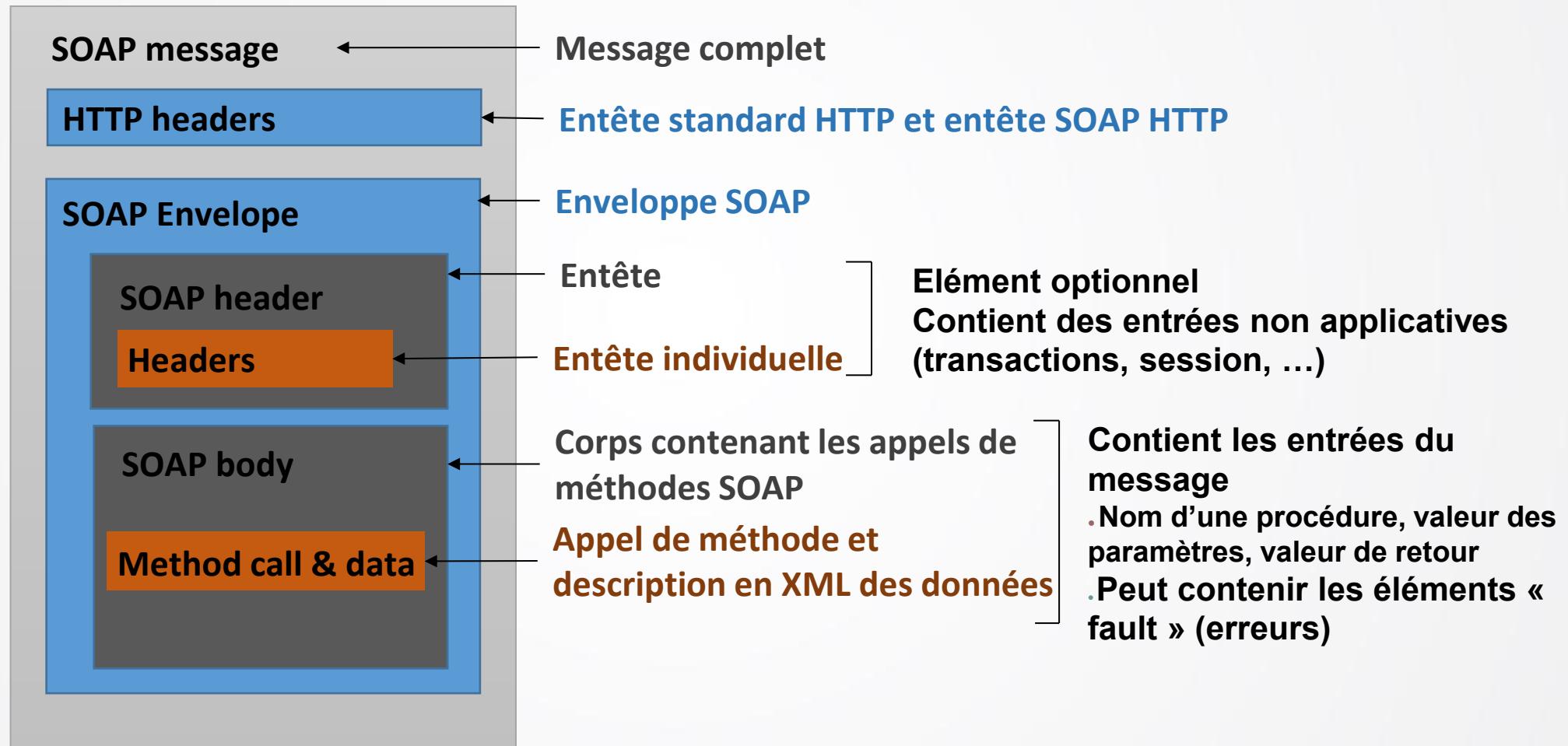
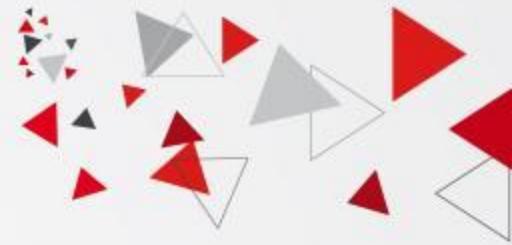
L'annuaire UDDI



- Il se compose de deux parties:
 - une spécification technique pour établir un annuaire distribué des entreprises et des services web.
 - L'UDDI registry permet à n'importe qui de rechercher des données dans l'UDDI. Il permet également à n'importe quelle compagnie de s'enregistrer ainsi que ses services.



Structure des messages SOAP (Simple Object Access Protocol)



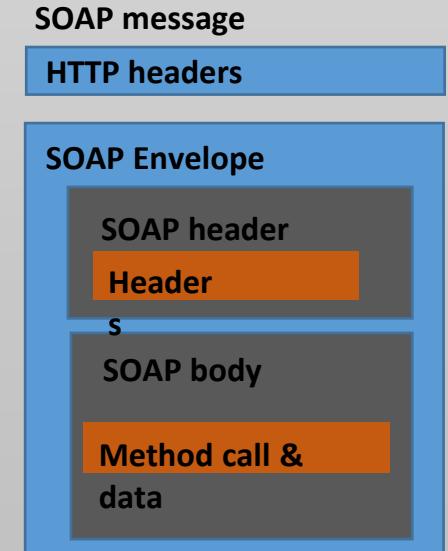


Les messages SOAP

L'enveloppe SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-  
ENV="http://schemas.xmlsoap.org/soap/envelope">  
  <SOAP-ENV:Body>  
    <!-- Contenu de la requête-->  
  </SOAP-ENV:Body>  
</SOAP-ENV:Envelope>
```

**Message SOAP
d'une requête**



```
<SOAP-ENV:Envelope xmlns:SOAP-  
ENV="http://schemas.xmlsoap.org/soap/envelope">  
  <SOAP-ENV:Body>  
    <!-- Contenu de la réponse -->  
  </SOAP-ENV:Body>  
</SOAP-ENV:Envelope>
```

**Message SOAP
d'une réponse**

Les messages SOAP

Exemple: Appeler les opérations du service HelloWorld

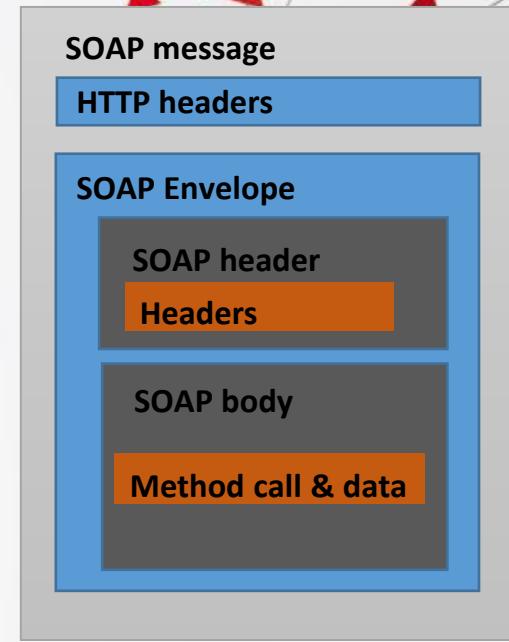
```
<soap-env:Envelope  
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope"  
    xmlns:hel="http://helloworldservice... .com/">  
    <soapenv:Header/>  
    <soapenv:Body>  
        <hel:makeHelloWorld>  
            <value> toto </value>  
        </hel:makeHelloWorld>  
    </soapenv:Body>  
</soapenv:Envelope>
```

Message SOAP pour appeler l'opération *makeHelloWorld* contenant un paramètre *value*

```
<soap-env:Envelope  
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope"  
    xmlns:hel="http://helloworldservice... .com/">  
    <soapenv:Header/>  
    <soapenv:Body>  
        <hel:simpleHelloWorld>  
        </hel:simpleHelloWorld>  
    </soapenv:Body>  
</soapenv:Envelope>
```

Message SOAP pour appeler l'opération *simpleHelloWorld* ne contenant pas de paramètres

Requête vers le service HelloWorld



Les messages SOAP

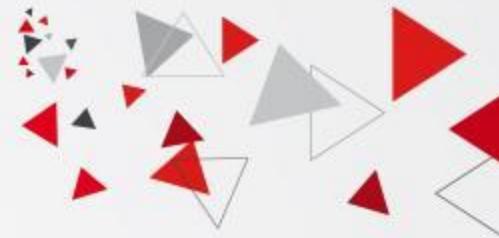
Reponse vers le service HelloWorld

Exemple (suite) : Messages retour de l'appel des opérations du service HelloWorld

```
<soap-env:Envelope  
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope">  
    <soapenv:Body>  
        <ns2:makeHelloWorldResponse xmlns:ns2="http://helloworldwebservice.... .com/">  
            <helloWorldResult>Hello World toto </helloWorldResult>  
        </ns2:makeHelloWorldResponse>  
    </soapenv:Body>  
</soapenv:Envelope>
```

```
<soap-env:Envelope  
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope">  
    <soapenv:Body>  
        <ns2:simpleHelloWorldResponse xmlns:ns2="http://helloworldwebservice.... .com/">  
            <helloWorldResult>Hello World Everybody </helloWorldResult>  
        </ns2:makeHelloWorldResponse>  
    </soapenv:Body>  
</soapenv:Envelope>
```

Les messages SOAP



Exemple : corps de messages SOAP pour appeler des opérations du service Web Notebook

```
<soap-env:Envelope  
    xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope  
    xmlns: not="http://notebookwebservice.... .com/">>  
    <soapenv:Header/>  
  
    <soapenv:Body>  
        <not:addPersonWithComplexType>  
            <newperson>  
                <name>toto</name>  
                <adress>France</adress>  
                <birthyear>1980</birthyear>  
            </newperson>  
        </not:addPersonWithComplexType>  
    </soapenv:Body>  
</soapenv:Envelope>
```

```
<soap-env:Envelope  
    xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope  
    xmlns: not="http://notebookwebservice.... .com/">>  
    <soapenv:Header/>  
  
    <soapenv:Body>  
        <not:addPersonWithSimpleType>  
            <name>toto</name>  
            <adress>France</adress>  
            <birthyear>1980</birthyear>  
        </not:addPersonWithSimpleType>  
    </soapenv:Body>  
</soapenv:Envelope>
```

Les messages SOAP



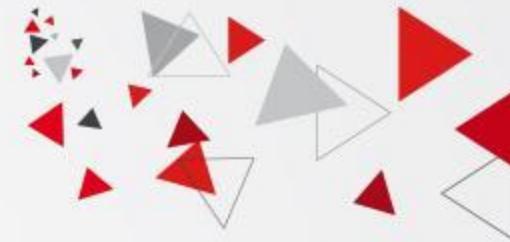
Exemple (suite) : corps de messages SOAP pour le résultat des opérations du service Web Notebook

```
<soap-env:Envelope  
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope">  
        <soapenv:Body>  
            <ns2:addPersonWithComplexTypeResponse xmlns:ns2="http://notebookwebservice.... .com/">  
                <addPersonWithComplexTypeResult> true </addPersonWithComplexTypeResult>  
            </ns2: addPersonWithComplexTypeResponse >  
        </soapenv:Body>  
</soapenv:Envelope>
```

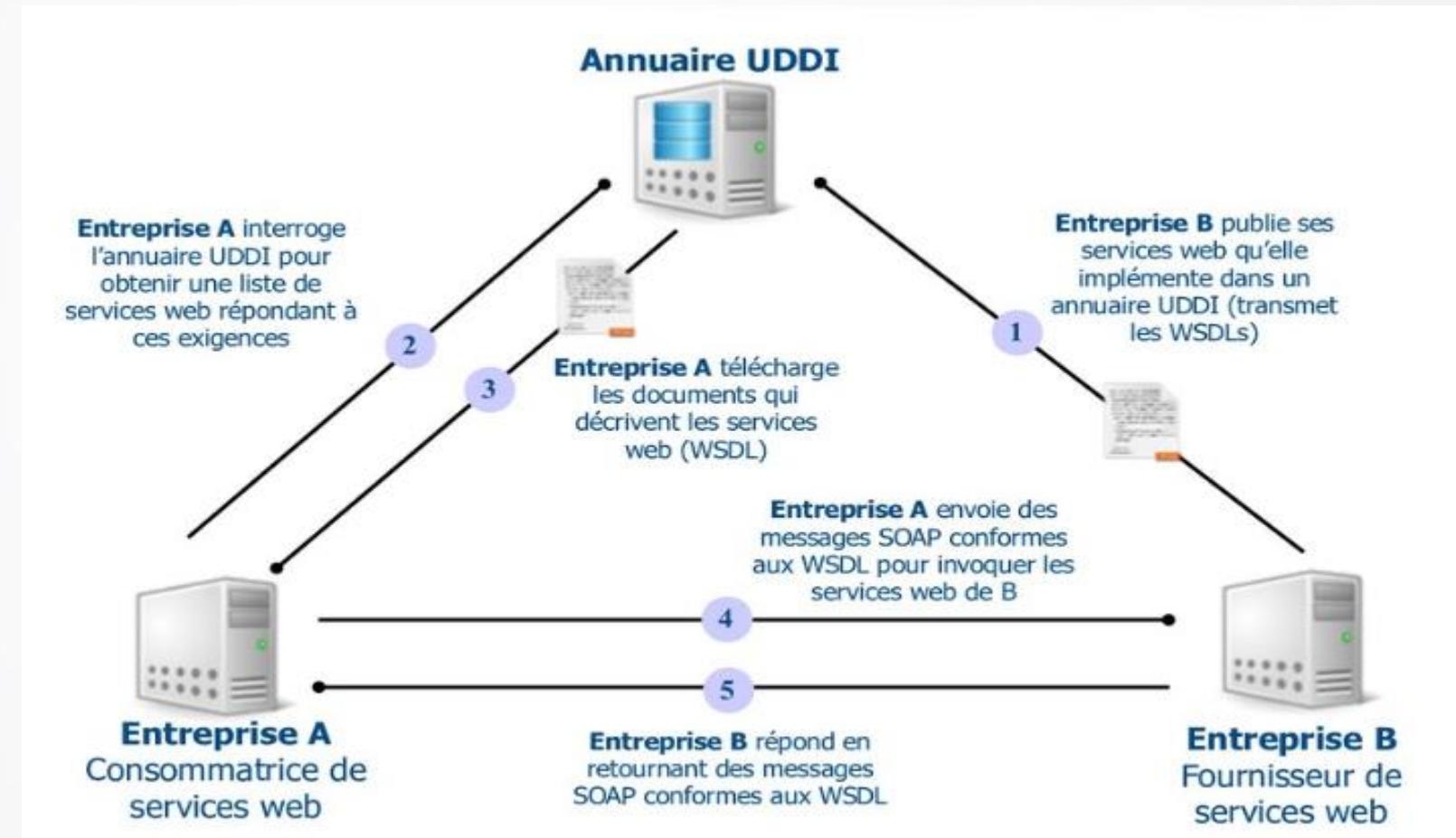
```
<soap-env:Envelope  
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope">  
        <soapenv:Body>  
            <ns2:addPersonWithSimpleTypeResponse xmlns:ns2="http://notebookwebservice.... .com/">  
                <addPersonWithSimpleTypeResult> true </addPersonWithSimpleTypeResult>  
            </ns2: addPersonWithSimpleTypeResponse >  
        </soapenv:Body>  
</soapenv:Envelope>
```

Les paramètres de sorties suivent la même convention que les paramètres d'entrées

Principe de fonctionnement

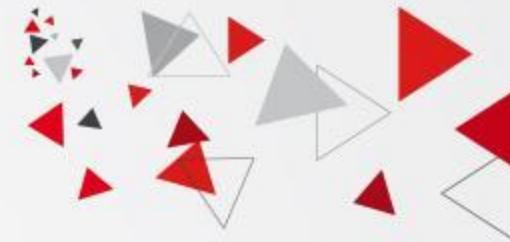


Un service web étendu en action





Références



- <http://www.w3.org/TR/wsdl>
- <http://www.webservicex.net/ws/WSDetails.aspx?CATID=12&WSID=63>
- <http://www.slideshare.net/baronm/wsdl-dcrire-et-configurer-4677823>
- <http://www.uddi.org/>
- <http://www.w3schools.com/>