



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

Université Abderrahmane Mira de Béjaïa
Faculté des Sciences Exactes
Département d'Informatique

Module : Services Web (M2 – GL – ReSyD – IA (S3))



Cours 04 : WSDL

Présenté par **Dr BRAHAMI EL BOUHISSI H.**

2017-2018

WSDL (Web Service Description Language)

O Standard du W3C :

Version 1.1 en 2001

Version 2.0 en 2007, encore peu supporté par les outils

O **Objectif** : décrire l'interface publique d'un Web Service (contrat de service)

O **Grammaire** dérivée d'XML

O **Web Service** = ensemble de ports de connexions mettant à disposition des opérations qui reçoivent et envoient des messages

O **Deux types d'informations** :

Fonctionnelles : interface du service (signature des méthodes...)

Techniques : URL, protocole...

O **Fichier WSDL** : utilisable par des outils de génération de code

WSDL (Web Service Description Language)

Regroupe les informations nécessaires pour interagir avec le service

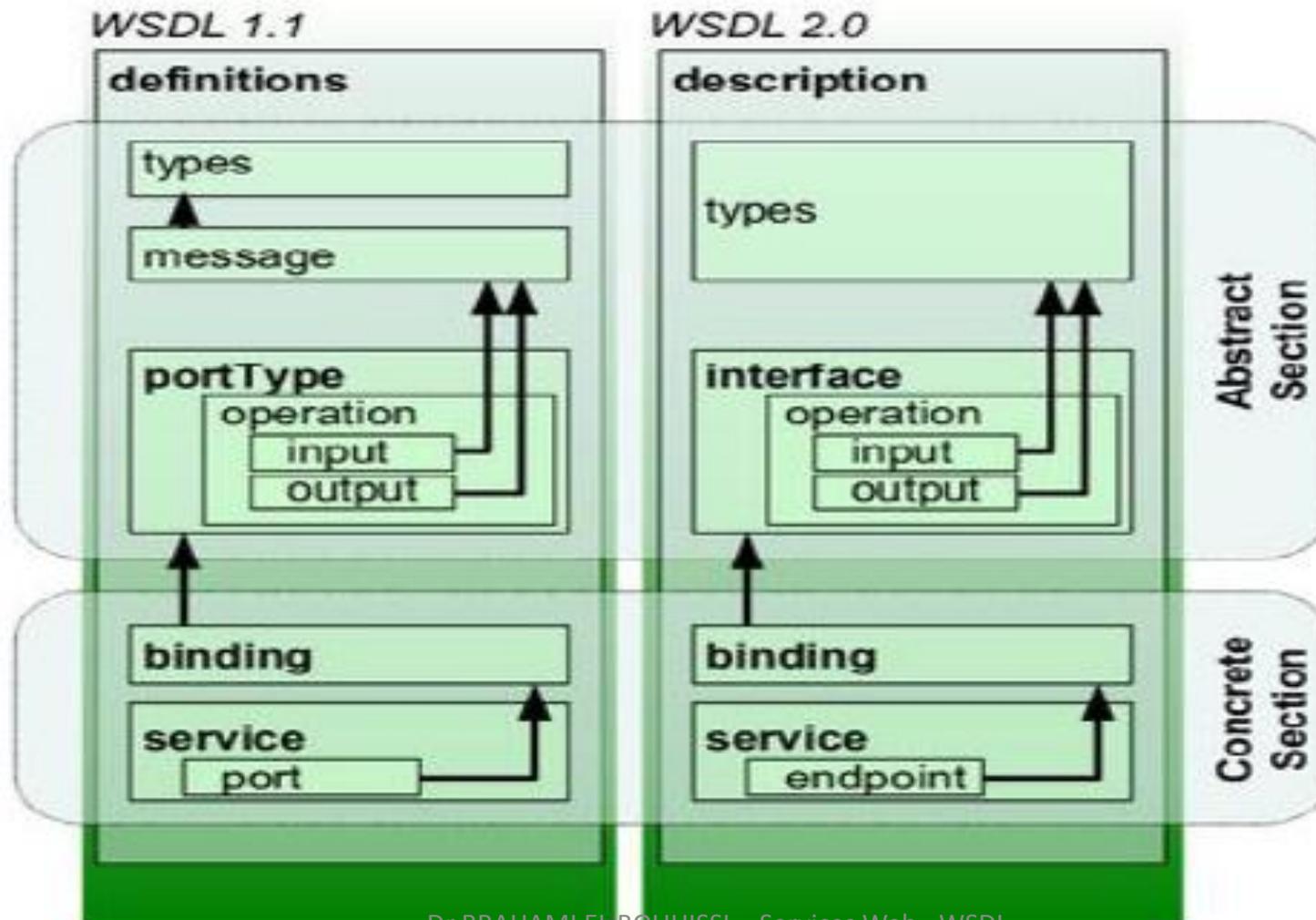
- Les méthodes, les paramètres et valeurs retournées, le protocole de transport utilisé, la localisation du service

Document indispensable au déploiement de Services Web

- Publication et recherche de services au sein de l'annuaire se font via les documents WSDL
- Pour l'accès à un service particulier, un client se voit retourné l'URL du fichier WSDL décrivant l'implémentation du service

WSDL : Version

Description à 2 niveaux: Séparation entre la partie abstraite et concrète



Structure d'un fichier WSDL 1.1

- **Élément Type**

Types des paramètres (schéma XML)

- **Élément Message**

Appel et retour d'opération

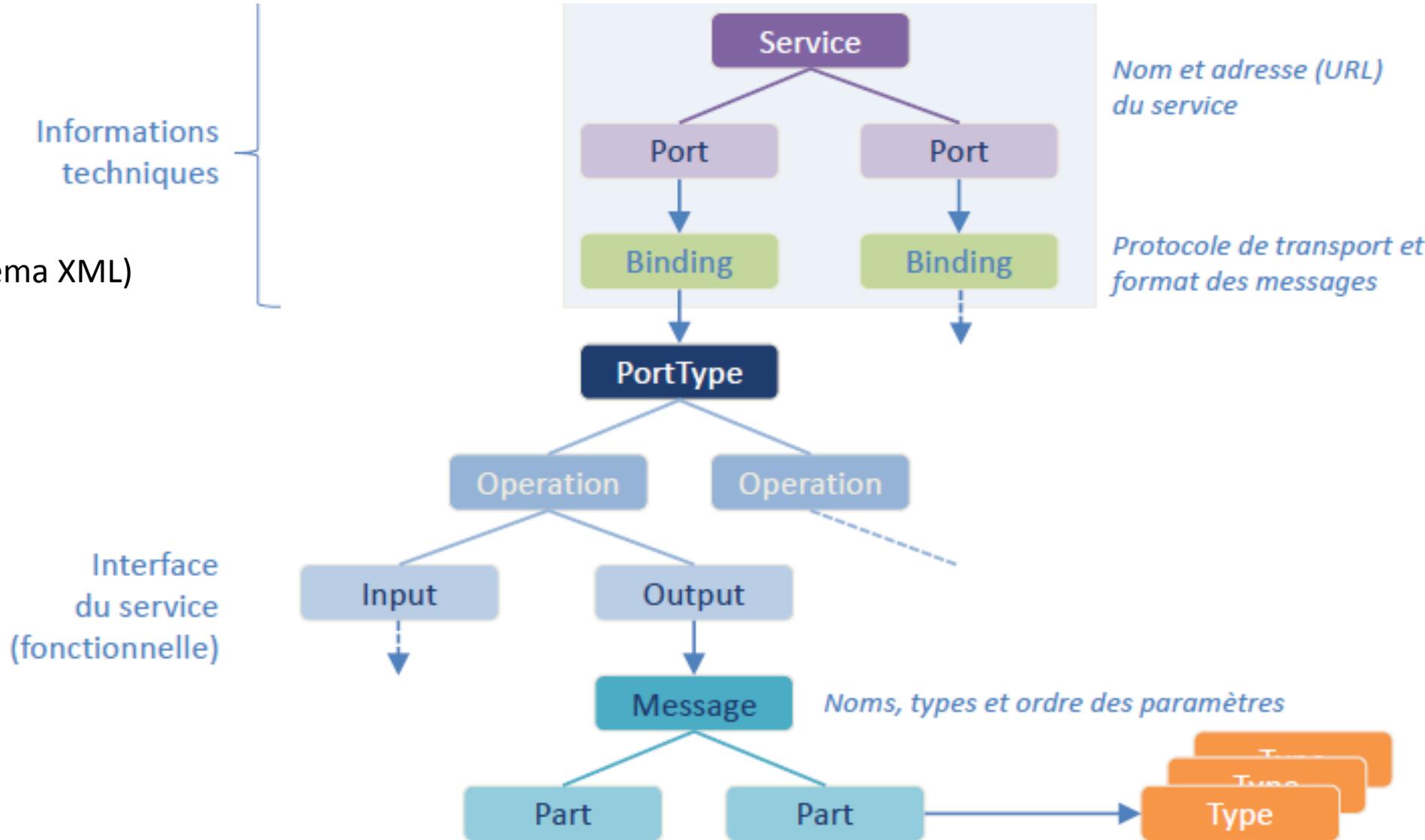
- **Élément Port type**

Groupe d'opération

- **Élément Binding**

URL de l'opération

Type de protocole



Structure d'un fichier WSDL 1.1

```
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/"  
             xmlns:xsd="http://www.w3.org/2001/XMLSchema" ...>
```

```
<types> [...] </types>
```

```
<message [...]><part [...] /></message>
```

```
<portType [...]>
```

```
  <operation [...]>
```

```
    <input [...] />
```

```
    <output [...] />
```

```
  </operation>
```

```
</portType>
```

Interface
du service
(fonctionnelle)

```
<binding [...]>[...]</binding>
```

```
<service [...]>
```

```
  <port [...]>[...]</port>
```

```
</service>
```

Informations
techniques

Exemple WSDL 1.1: sommer.wSDL

```
<wsdl:definitions targetNamespace="http://localhost:8080/axis/sommer.jws">
```

Racine du document

```
WSDL created by Apache Axis version: 1.4  
Built on Apr 22, 2006 (06:55:48 PDT)
```

Définition des types de données (facultatif)

```
- <wsdl:message name="getsommeResponse">  
  <wsdl:part name="getsommeReturn" type="xsd:int"/>  
</wsdl:message>  
- <wsdl:message name="getsommeRequest">  
  <wsdl:part name="a" type="xsd:int"/>  
  <wsdl:part name="b" type="xsd:int"/>  
</wsdl:message>
```

Message: Définition des messages échangeables

```
- <wsdl:portType name="sommer">  
  - <wsdl:operation name="getsomme" parameterOrder="a b">  
    <wsdl:input message="impl:getsommeRequest" name="getsommeRequest"/>  
    <wsdl:output message="impl:getsommeResponse" name="getsommeResponse"/>  
  </wsdl:operation>  
</wsdl:portType>
```

PortType: définition des ensembles d'opérations

```
- <wsdl:binding name="sommerSoapBinding" type="impl:sommer">  
  <wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>  
  - <wsdl:operation name="getsomme">  
    <wsdlsoap:operation soapAction="" />  
    - <wsdl:input name="getsommeRequest">  
      <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" namespace="http://DefaultNamespace" use="encoded"/>  
    </wsdl:input>  
    - <wsdl:output name="getsommeResponse">  
      <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" namespace="http://localhost:8080/axis/sommer.jws" use="encoded"/>  
    </wsdl:output>  
  </wsdl:operation>  
</wsdl:binding>
```

```
- <wsdl:service name="sommerService">  
  - <wsdl:port binding="impl:sommerSoapBinding" name="sommer">  
    <wsdlsoap:address location="http://localhost:8080/axis/sommer"/>  
  </wsdl:port>  
</wsdl:service>  
</wsdl:definitions>
```

service: localisation des services web

L'élément définition

L'élément racine dans un document WSDL est `<wsdl:definition>`. Il contient un attribut `targetNamespace` qui définit un certain nombre d'espaces de noms namespace auquel tous les noms déclarés dans un élément du document WSDL appartiennent, ce qui permet d'éviter les conflits de nommage.

```
<wsdl:definition name="customerExemple"  
targetNamespace="http://www.stevepotts.com/customer.wsdl"  
  xmlns:soap="http://www.schemas.xmlsoap.org/wsdl/soap/"  
  xmlns:wsdl="http://www.schemas.xmlsoap.org/wsdl/"  
  xmlns="http://www.stevepotts.com/customer.xsd"
```

Dans cet exemple, l'attribut `targetNamespace` a pour valeur l'URL : «`[..]www.stevepotts.com/customer.wsdl`». Cela signifie que tous les noms déclarés dans ce document WSDL appartiennent à cet espace de noms.

Les espaces de noms

- Un espace de noms est identifié par une **URI** (Uniform Resource Identifier) qui permet de l'identifier de manière unique.
- Schéma XML : <http://www.w3.org/2001/XMLSchema>
- La **déclaration d'un espace de noms par défaut** se fait dans le premier élément qui utilise le vocabulaire, grâce au mot clef **xmlns** comme **XML namespace**.
- Exemple : **xmlns="mon_uri"**

Structure d'un WSDL : <Types>

- Un type décrit la structure de données transmises dans un message.
- Contient les définitions de types utilisant un système de typage par défaut XML Schema (XSD).
- Pouvant contenir des types simples et complexes

Structure d'un WSDL : <message>

- Décrit les données associées à une opération (1 requête et 1 réponse HTTP par opération, 1 message d'erreur "fault" optionnel).
- Un document WSDL peut contenir zéro ou plusieurs messages.
- Chaque message peut être un composé de plusieurs parties

```
- <wsdl:message name="getsommeResponse">  
  <wsdl:part name="getsommeReturn" type="xsd:int"/>  
</wsdl:message>  
- <wsdl:message name="getsommeRequest">  
  <wsdl:part name="a" type="xsd:int"/>  
  <wsdl:part name="b" type="xsd:int"/>  
</wsdl:message>
```

Structure d'un WSDL : <portType>

- Un document WSDL peut contenir 0 à plusieurs portType
- L'élément portType contient un seul attribut name. La convention de nommage nameOfWebService PortType.
- Composé d'un ensemble d'opérations abstraites (i.e. signature de la méthode). Une opération est composée d'un message pour l'appel (Input) et un pour le retour (Output).

```
<wsdl:portType name="sommer">  
- <wsdl:operation name="getsomme" parameterOrder="a b">  
  <wsdl:input message="impl:getsommeRequest" name="getsommeRequest"/>  
  <wsdl:output message="impl:getsommeResponse" name="getsommeResponse"/>  
</wsdl:operation>  
</wsdl:portType>
```

Définition d'un seul type de port, avec les opérations abstraites, correspondant aux déclarations de méthodes dans l'interface Java

Structure d'un WSDL : <binding>

- Une liaison (ou binding) décrit la façon dont un portType (en d'autres termes l'abstraction du service, i.e. ses opérations abstraites) est mis en œuvre pour un protocole particulier (HTTP par exemple) et un mode d'invocation (RPC par exemple).
- Pour un portType, on peut avoir plusieurs liaisons, pour différencier les modes d'invocation (RPC ou autres) ou de transport (HTTP ou autre) des différentes opérations.

Structure d'un WSDL : <binding>

```
<wsdl:binding name="sommerSoapBinding" type="impl:sommer">  
  <wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>  
  <wsdl:operation name="getsomme">  
    <wsdlsoap:operation soapAction="">  
      <wsdl:input name="getsommeRequest">  
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" namespace="http://DefaultNamespace" use="encoded"/>  
      </wsdl:input>  
      <wsdl:output name="getsommeResponse">  
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" namespace="http://localhost:8080/axis/sommer.jws" use="encoded"/>  
      </wsdl:output>  
    </wsdl:operation>  
  </wsdl:binding>
```

Mode d'invocation

Protocole

Nom de l'opération dans le type de port

La représentation du message request

La représentation du message response

Structure d'un WSDL : <binding>

Une liaison WSDL décrit comment le service est lié à un protocole de messagerie, généralement le protocole de messagerie SOAP. Une liaison WSDL SOAP peut être une liaison de style Remote Procedure (RPC) ou une liaison de style document.

Le style RPC spécifie que le <soap: body> contient un élément avec le nom de la méthode Web appelée. Cet élément contient à son tour une entrée pour chaque paramètre et la valeur de retour de cette méthode.

Avec le style de document, les parties de message apparaissent directement sous l'élément <soap: body>. Il n'existe pas de règles de formatage SOAP pour ce que contient le <soap: body>. L'application serveur est responsable du mappage des objets serveur (paramètres, appels de méthode, etc.) et des valeurs des documents XML.

Structure d'un WSDL : <service>

- Un service est décrit comme un ensemble de points finaux du réseau appelés « ports »
- Un port spécifie une URL qui correspond à l'implémentation du service par un fournisseur.
- Le port est associé à un « binding » définissant ainsi un simple point de terminaison (endpoint:@ où se situe le WS)



WSDL : exemple d'interface de service

O Avec des types simples

```
<portType name="Hello">
  <operation name="sayHello">
    <input message="tns:sayHello" />
    <output message="tns:sayHelloResponse" />
  </operation>
</portType>

<message name="sayHello">
  <part name="n" type="xsd:string" />
</message>
<message name="sayHelloResponse">
  <part name="return" type="xsd:string" />
</message>
```

WSDL : exemple d'interface de service

O Avec des types complexes

Déclarés dans un fichier XSD (XML Schema) séparé
Ou déclarés dans le fichier WSDL

```
<part name="parameters" element="sayHello" />
```



```
<xsd:element name="sayHello" >  
  <xsd:complexType>  
    <xsd:sequence>  
      <xsd:element name="n" type="xs:string" minOccurs="0"/>  
    </xsd:sequence>  
  </xsd:complexType>  
</xsd:element>
```

Exercice

Annoter ce WSDL de vos commentaires :à quoi il sert, comment s'en servir d'un point de vue client, quelles données vont circuler quand on l'utilise, etc.

Cliquez [ici](#) pour visualiser le Fichier WSDL

Solution

Ce service web offre une unique opération : sur envoi d'un message de type `GetLastTradePrice`, l'opération nommée `GetLastTradePrice` va s'exécuter. L'URL pour déclencher cette opération depuis un navigateur web devrait être `http://example.com/stockquote/method=GetLastPrice?tickerSymbol=MonEntreprisePreferee`.

Le message sera véhiculé par une enveloppe SOAP (on a défini un soap binding). Le type de donnée en entrée est un `tickerSymbol` prenant une valeur de type chaîne de caractères. En réponse, on obtient un `price` que l'on peut interpréter comme un flottant.