



WSDL



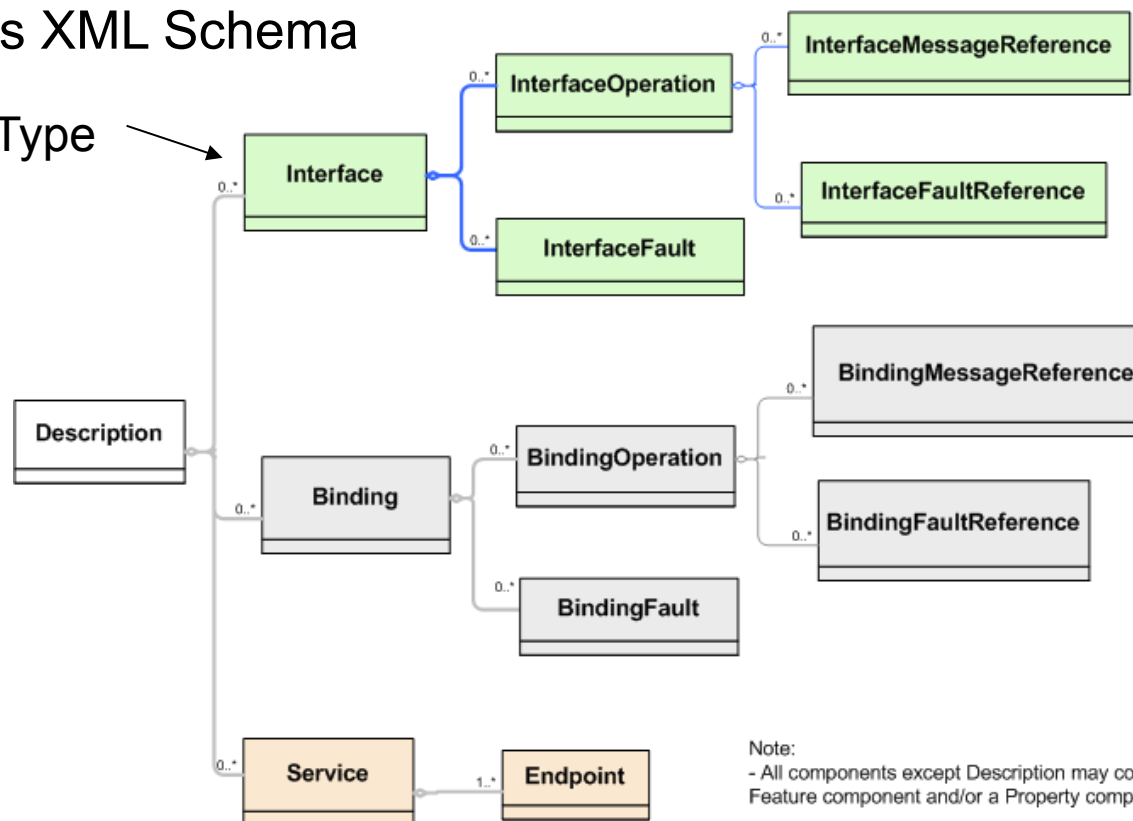
WSLD

- Web Services Description Language
- Format XML
- Spécification:
 - Des messages envoyés ou reçus et des erreurs possibles
 - Des types XML des messages (via XML Schema)
 - Du mode de transmission
 - Du protocole utilisé pour le transfert
 - De l'adresse du service

Structure

+ Types XML Schema

WSDL 1.1: portType



Note:
- All components except Description may contain a Feature component and/or a Property component

(Source: W3C)

Prologue du fichier

<description

xmlns="http://www.w3.org/ns/wsd1"

targetNamespace= "http://www.glaces.com/wsd1/commande"

xmlns:tns= "http://www.glaces.com/wsd1/commande"

xmlns:glaces = "http://www.glaces.com/schema/commande"

xmlns:wsoap= "http://www.w3.org/ns/wsd1/soap"

xmlns:soap="http://www.w3.org/2003/05/soap-envelope"

xmlns:wsd1x= "http://www.w3.org/ns/wsd1-extensions">

<documentation>Bla bla</documentation>



Types en WSDL

- Basés sur XML Schema (bien que d' autres langages de spécifications soient utilisables)
- Spécifiés dans un élément <types>
 - directement dans le fichier WSDL
 - via une référence à un fichier xsd externe

Exemple

```
<types>
  <schema xmlns="http://www.w3.org/2001/XMLSchema" targetNamespace="http://
    www.glaces.com/schema/commande">
    <element name="commande" type="glaces:commandeContent"/>
    <element name="confirmation" type="glaces:confirmationContent"/>
    <element name="erreurLivraison" type="glaces:erreurLivraisonContent"/>
    ...
  </schema>
</types>
```

Ou

```
<types>
  <import xmlns="http://www.w3.org/2001/XMLSchema" namespace="http://
    www.glaces.com/schema/commande" schemaLocation="commandes.xsd"/>
</types>
```



Interfaces

- Ensemble d'opérations
 - Groupes de messages à envoyer/recevoir
 - Références à des erreurs (Faults) pouvant survenir
- Définitions des erreurs
 - En particulier, les éléments qu'elles peuvent contenir
- Fonctionnalité proche des interfaces Java



Opérations

- En général, au plus 1 entrée et 1 sortie
- Pattern: schéma d'interaction utilisé
 - <http://www.w3.org/ns/wsdli/xxxx>
 - in-only, robust-in-only, in-out
- wsdlx:safe
 - true si l'opération n'a pas de conséquence côté serveur: à utiliser pour les demandes d'informations (similaire à GET en HTTP)

Exemple

```
<interface name="commandeItf">
  <fault name="commandeFault »
    element="glaces:erreurLivraison" />
  <operation name="commandeOp » pattern="http://www.w3.org/ns/
    wsdl/in-out" wsdlx:safe="false">
    <input messageLabel="In" element="glaces:commande" />

    <ouput messageLabel="Out" element="glaces:confirmation" />

    <outfault ref="tns:commandeFault" messageLabel="Out" />

  </operation>
</interface>
```



Binding

- Spécifie la manière dont seront transmis les messages
- 1 ou plusieurs bindings par interface
- En général:
 - SOAP
 - HTTP

Exemple

```
<binding name="commandeSOAPBinding"  
  interface="tns:commandeltf"  
    type="http://www.w3.org/ns/wsdl/soap"  
  wsoap:protocol="http://www.w3.org/2003/05/soap/bindings/  
  HTTP/">
```

```
<operation ref="tns:commandeOp"  
  wsoap:mep="http://www.w3.org/2003/05/soap/mep/  
  request-response/" />
```

```
<fault ref="tns:commandeFault" wsoap:code="soap:Receiver" />  
</binding>
```



Service

- Correspond à une certaine interface
- Spécifie un ensemble de « points d'entrées » pour le service:
 - Le binding utilisé
 - L'adresse à utiliser pour accéder au service



Exemple

```
<service name="commandesService"  
  interface="tns:commandeItf">  
  <endpoint name="commandeEndPoint"  
binding="tns:commandeSOAPBinding"  
address="http://www.glaces.com/commande/sw-impl" />  
</service>
```



Imports et inclusions

- `<include>`
 - les définitions sont incluses dans le document courant (recopie)
 - L' espace de nommage cible (`targetNameSpace`) doit être identique dans le document inclus et dans le document courant
- `<import>`
 - Les définitions sont rendues accessibles et utilisables
 - L' espace de nommage cible est différent dans les deux documents

Exemple d' import (W3C)

```
<import namespace="http://finance.example.com/CreditCards/wsd1"
  location="credit-card-faults.wsd1"/>
  ...
<interface name="reservation" extends="cc:creditCardFaults">
  ...
<operation name="makeReservation" pattern="http://www.w3.org/ns/
  wsdl/in-out">
  <input messageLabel="In" element="ghns:makeReservation" />
  <output messageLabel="Out"
    element="ghns:makeReservationResponse" />
  <outfault ref="invalidDataFault" messageLabel="Out" />
  <outfault ref="cc:cancelledCreditCard" messageLabel="Out" />
  <outfault ref="cc:expiredCreditCard" messageLabel="Out" />
</operation>
</interface>
```

WSDL 1.1

- <schema> obligatoire dans <types>
- Pas <interface>, mais:
 - <message>
 - <portType>
 - <operation>
 - <input>, <output>, <fault>
- Exemple en version 1.1