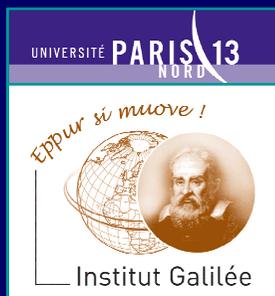


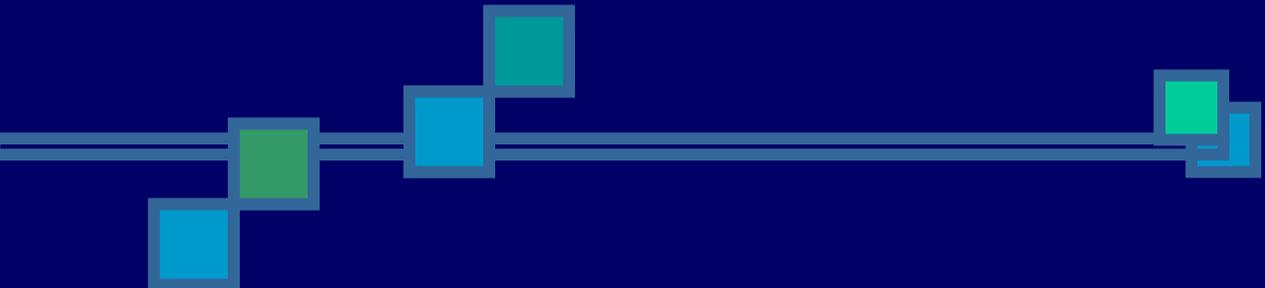
XML-Family

Web Services Description Language W.S.D.L.

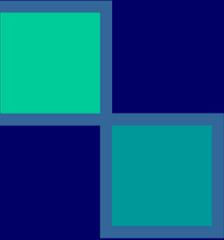
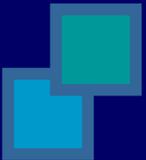


Master Informatique – Cours WSIA
COTTIN Nicolas – FRENDI Agour
2007





Sommaire

- 
- **Introduction**
 - Présentation de la technologie
 - Description d'un fichier WSDL
 - Exemples d'utilisation
 - Conclusion
- 



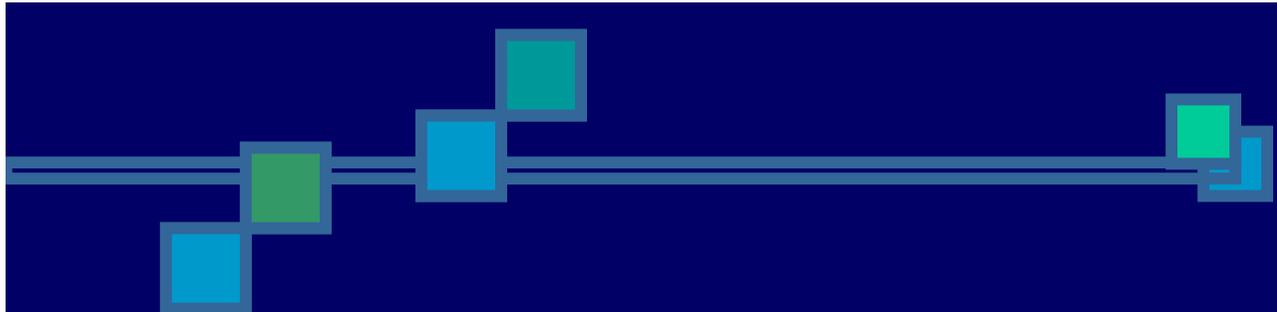
Introduction

■ Contexte

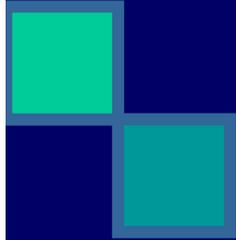
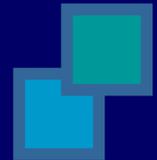
- De plus en plus de services WEB
- Des services WEB hétérogènes
- Difficultés de communication inter services

■ Solution

- Apportée par Microsoft, IBM et Ariba
- Création d'un langage commun pour décrire les services WEB et une plateforme pour les intégrer automatiquement



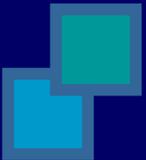
Sommaire

- 
- Introduction
 - **Présentation de la technologie**
 - Description d'un fichier WSDL
 - Exemples d'utilisation
 - Conclusion
- 



Présentation de la technologie

Historique

- Appartient à la famille XML
 - Première version en 2000
 - La version 1.2 (rebaptisée 2.0) est sur le point d'être validée par le W3C
 - Cette dernière version est déjà bien implantée dans les entreprises
- 



Présentation de la technologie

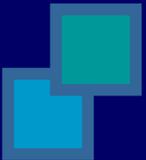
Définition

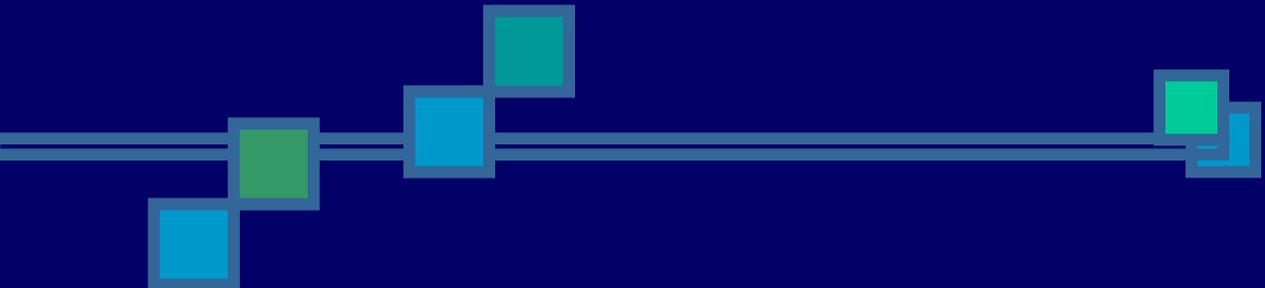
- Il décrit les caractéristiques des services WEB en précisant :
 - Les méthodes invoquées
 - Leur signature
 - Le point d'accès
- Cette description est complètement abstraite et indépendante du langage utilisé (D'un point de vue du développeur)



Présentation de la technologie

Caractéristiques

- Le langage s'appuie sur la technologie XML
 - La grammaire du langage est définie par un XML schéma :
<http://schemas.xmlsoap.org/wsdl/>
 - Elle définit 12 éléments
- 



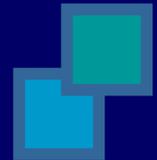
Présentation de la technologie

Le principe

- Le WSDL décrit quatre ensembles de données importants
 - Information d'interface décrivant toutes les fonctions disponibles publiquement
 - Information de type de données pour toutes les requêtes de message et requêtes de réponse
 - Information de liaison sur le protocole de transport utilisé
 - Information d'adresse pour localiser le service spécifié
- 



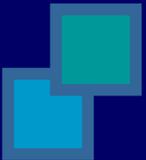
Présentation de la technologie *Intérêt*

- Facilite l'accès aux applications entre entreprises
 - Permet de décharger les utilisateurs des détails techniques de réalisation d'un appel
 - Homogénéise la description des interfaces des services WEB
 - Apporte une certaine transparence à la définition de ces services
 - Il permet l'appel dynamique de WEB services
- 



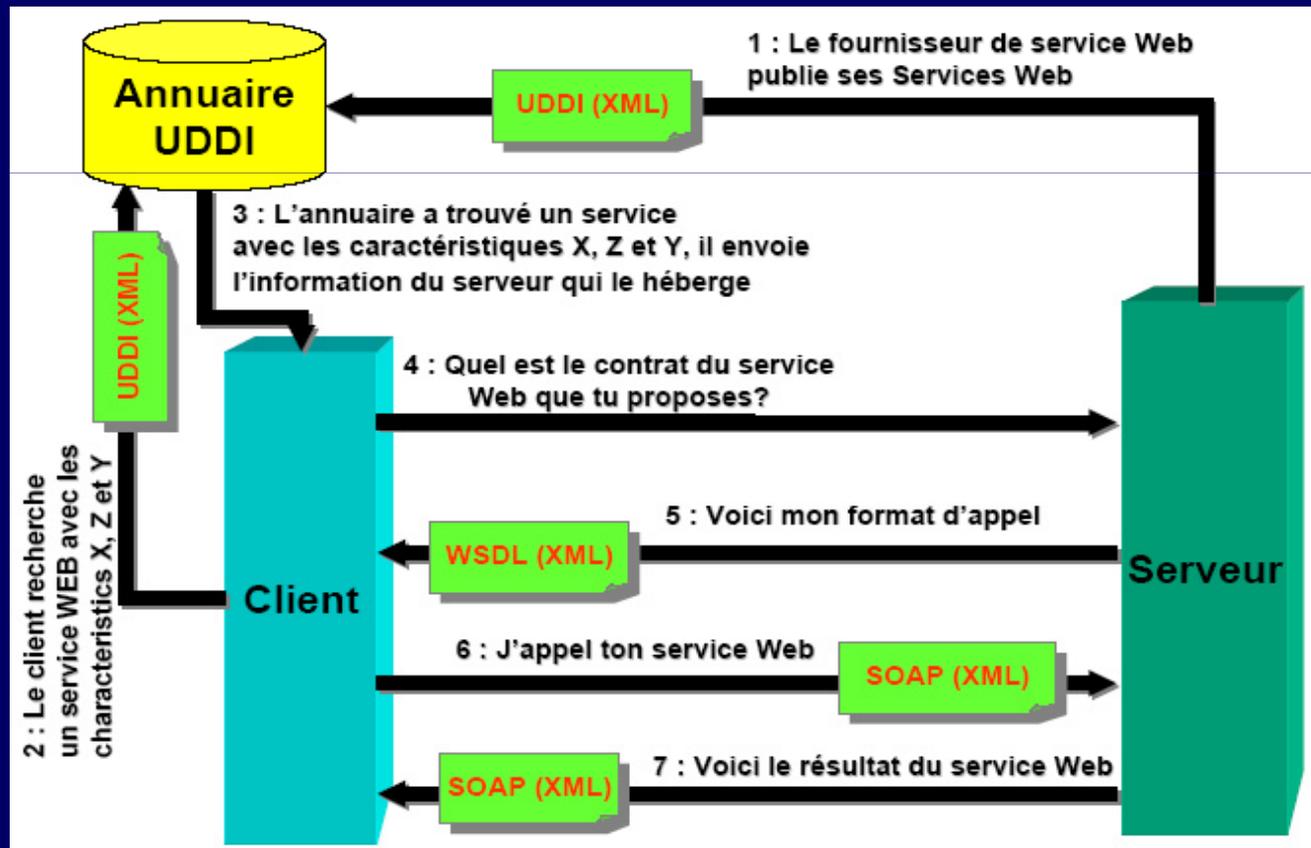
Présentation de la technologie

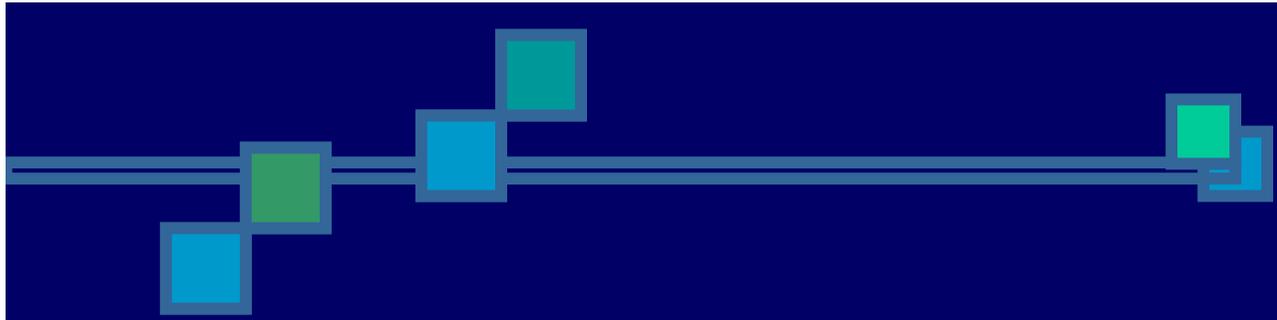
Mise en place

- Les services WEB sont recensés au sein d'un annuaire l'UDDI (Universal Description, Discovery and Integration)
 - L'utilisateur récupère le fichier WSDL à partir de cet annuaire et connaît ainsi la description du service pour communiquer avec celui-ci
 - Très souvent cette communication est effectuée à l'aide de SOAP
- 

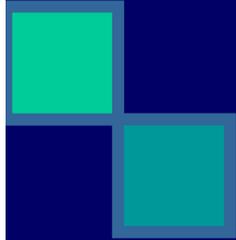
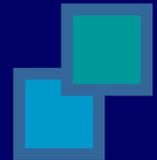
Présentation de la technologie

Mise en place



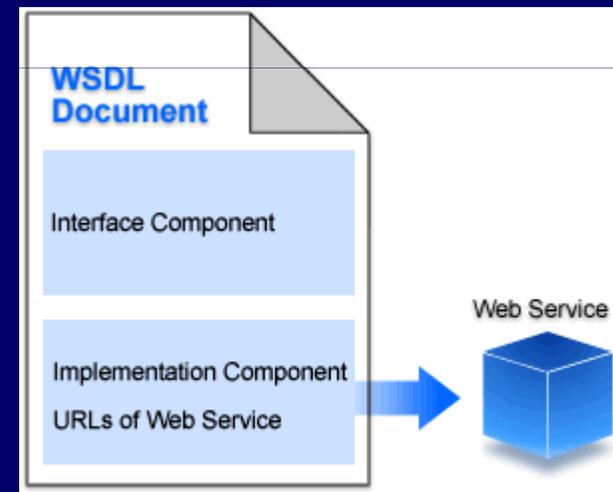
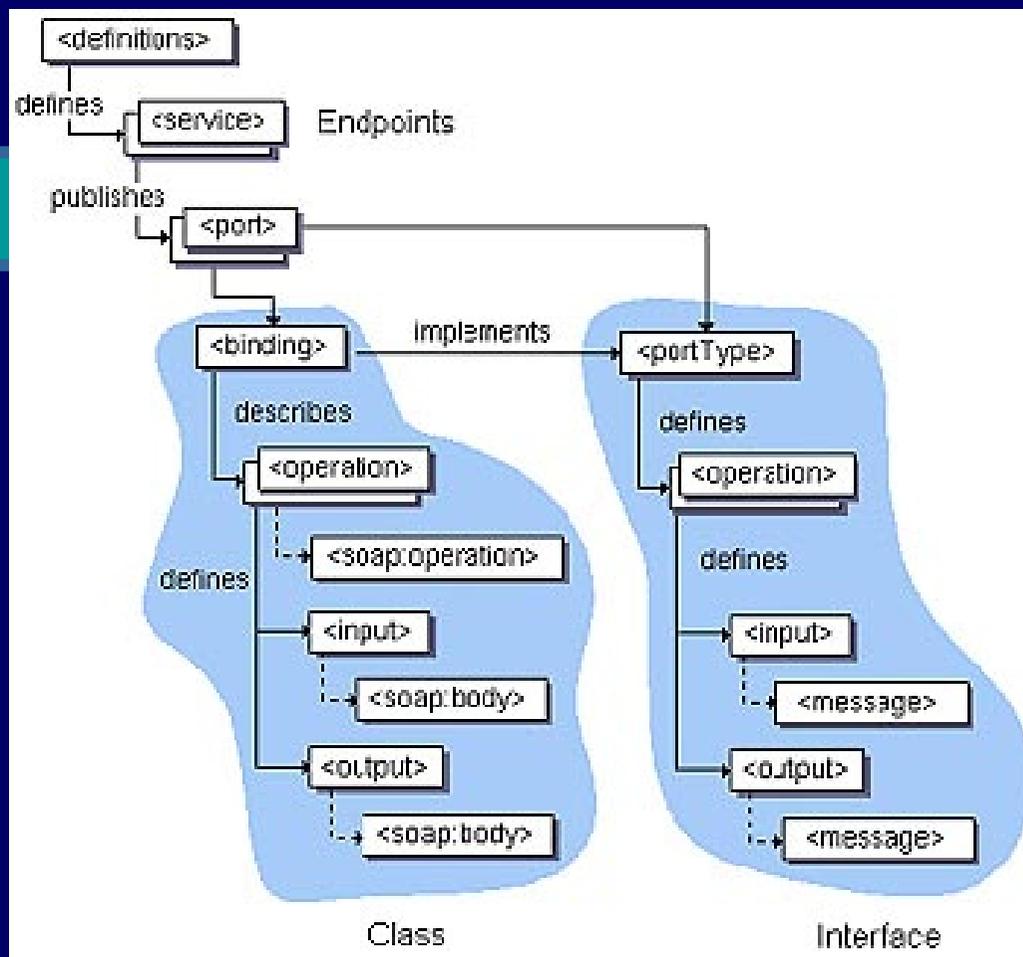


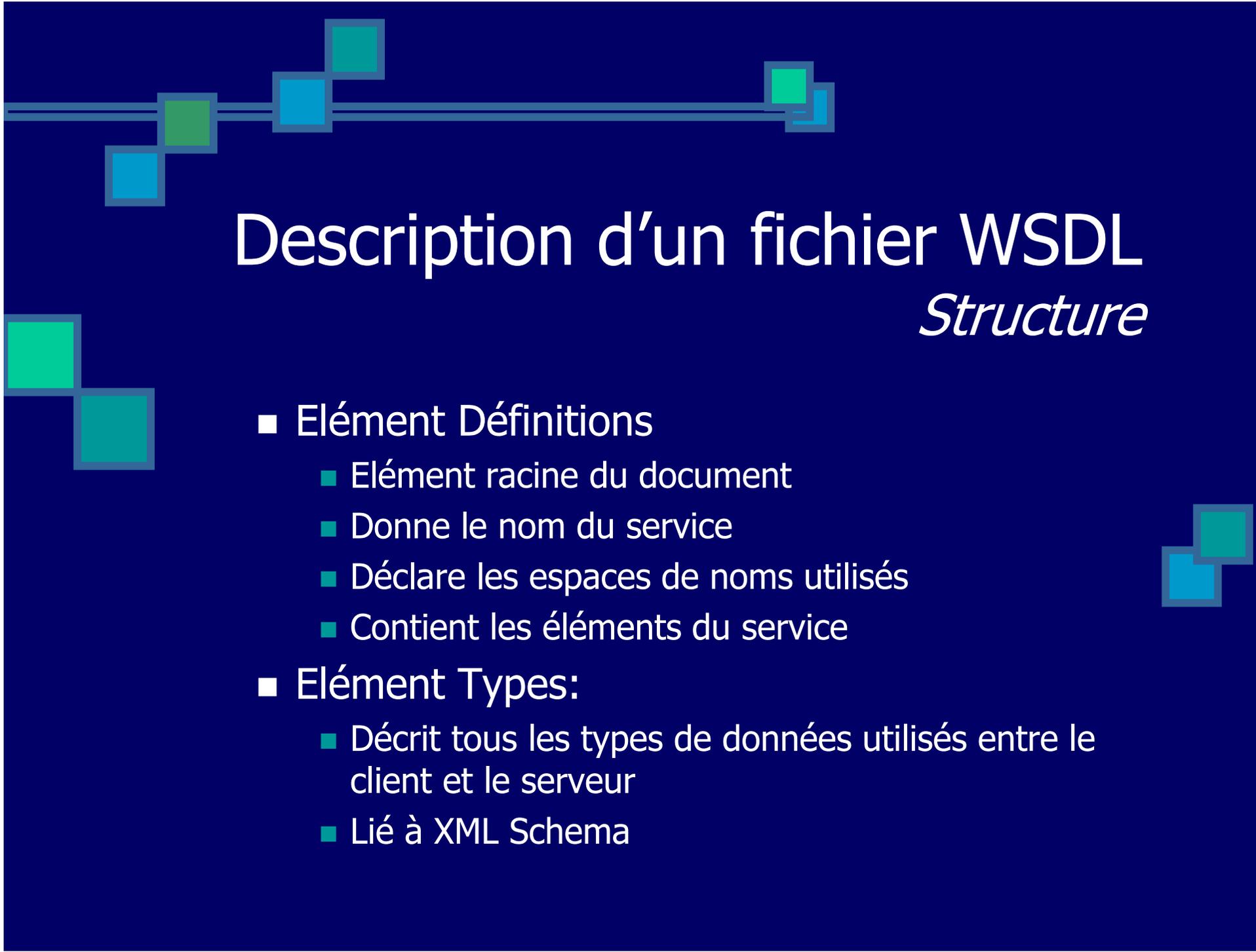
Sommaire

- 
- Introduction
 - Présentation de la technologie
 - **Description d'un fichier WSDL**
 - Exemples d'utilisation
 - Conclusion
- 

Description d'un fichier WSDL

Structure

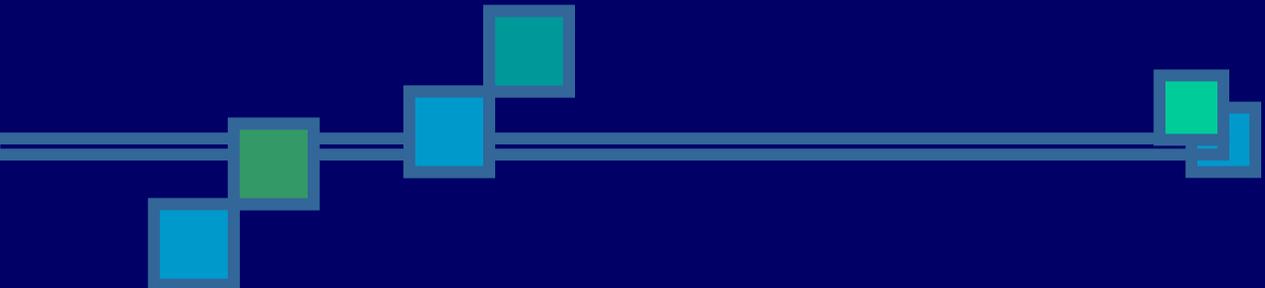




Description d'un fichier WSDL

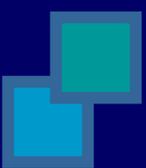
Structure

- **Élément Définitions**
 - Élément racine du document
 - Donne le nom du service
 - Déclare les espaces de noms utilisés
 - Contient les éléments du service
- **Élément Types:**
 - Décrit tous les types de données utilisés entre le client et le serveur
 - Lié à XML Schema



Description d'un fichier WSDL

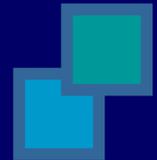
Structure

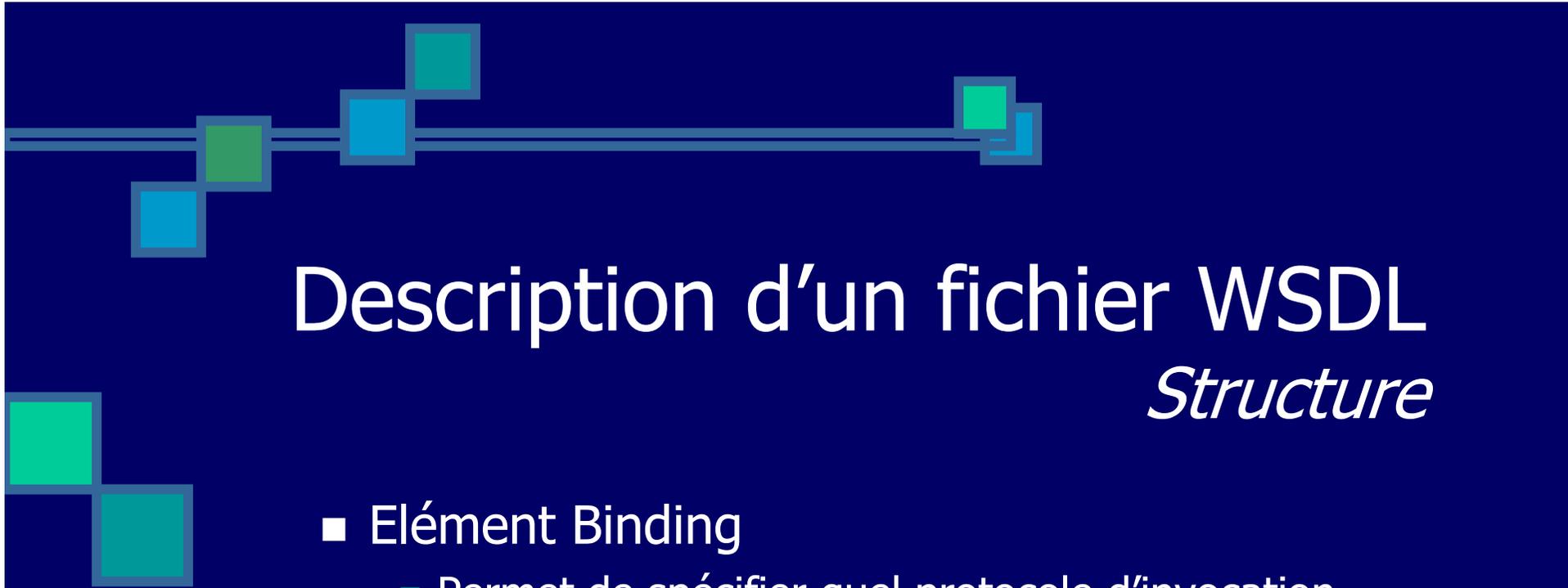
- **Élément Message**
 - Deux types de message IN et OUT
 - Définition abstraite des messages échangés entre deux nœuds
 - Peut être composé de plusieurs parties (Parts)
 - **Élément Part**
 - Il peut être défini comme un type (simple ou complexe) ou un élément
- 



Description d'un fichier WSDL

Structure

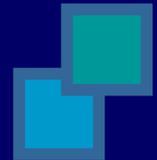
- Élément PortType
 - Correspond à une interface. Il contient les classes accessibles
 - A chaque portType sont associées des opérations, correspondant aux méthodes. Pour chaque méthode on définit le message d'entrée et de sortie.
 - Les opérations peuvent être de natures différentes: unidirectionnelle, requête/réponse, sollicitation/réponse et notification.
- 

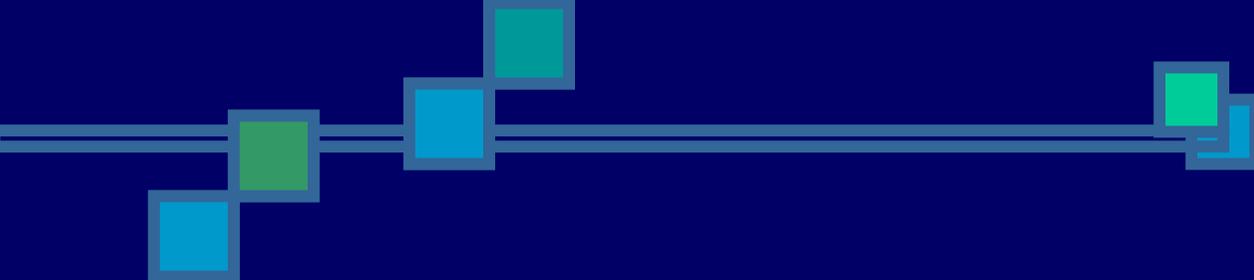


Description d'un fichier WSDL

Structure

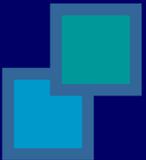
- **Élément Binding**

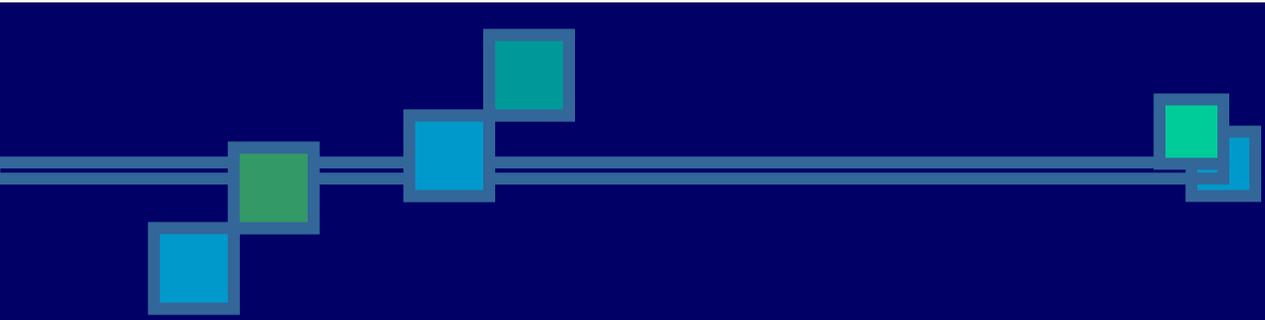
- Permet de spécifier quel protocole d'invocation utiliser HTTP GET/POST, SOAP, SMTP, FTP
 - Définit le format des données pour les opérations et messages définis par un type de port particulier.
 - WSDL possède des extensions internes pour définir des services SOAP, les informations spécifiques à SOAP se retrouvent dans cet élément.
- 



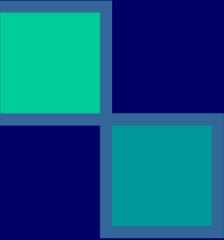
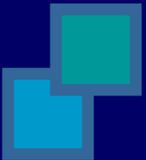
Description d'un fichier WSDL

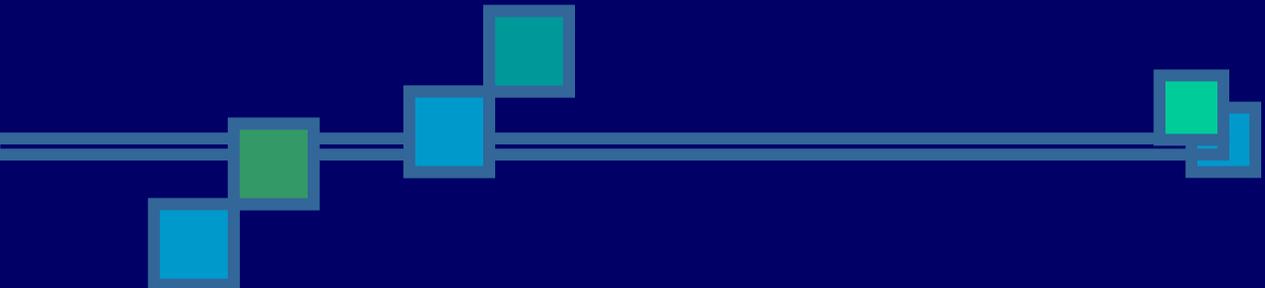
Structure

- **Élément Service**
 - Défini les adresses permettant d'invoquer le service donné
 - Regroupe un ensemble de ports reliés
 - Généralement c'est une URL qui invoque le service
 - **Élément Port**
 - Définition d'un point d'entrée pour un service
 - Référence le composant binding permettant son traitement
 - **Élément Documentation**
- 



Sommaire

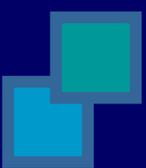
- 
- Introduction
 - Présentation de la technologie
 - Description d'un fichier WSDL
 - **Exemples d'utilisation**
 - Conclusion
- 

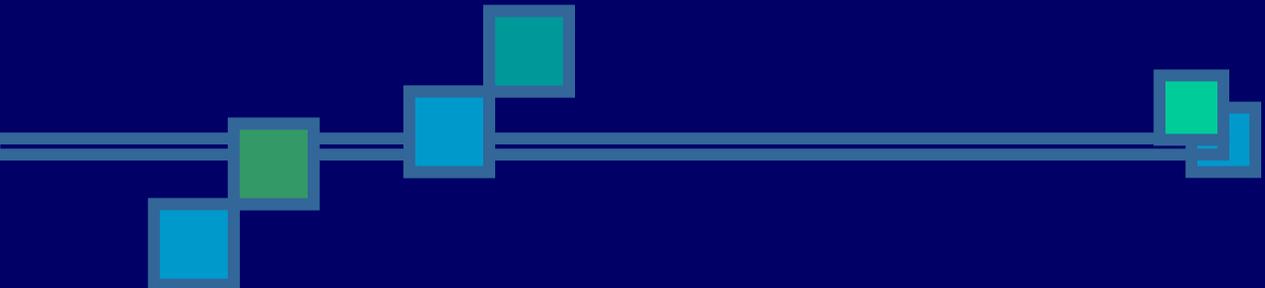


Exemple d'utilisation

Hello World !

```
<?xml version="1.0" encoding="UTF-8"?>
  <definitions name="HelloWorld"
    targetNamespace="http://hello.jaxrpc.samples/"
    xmlns:tns="http://hello.jaxrpc.samples/"
    xmlns="http://schemas.xmlsoap.org/wsdl/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
    <types />
    <message name="sayHello">
      <part name="String_1" type="xsd:string" />
    </message>
    <message name="sayHelloResponse">
      <part name="result" type="xsd:string" />
    </message>
```

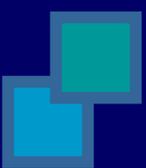


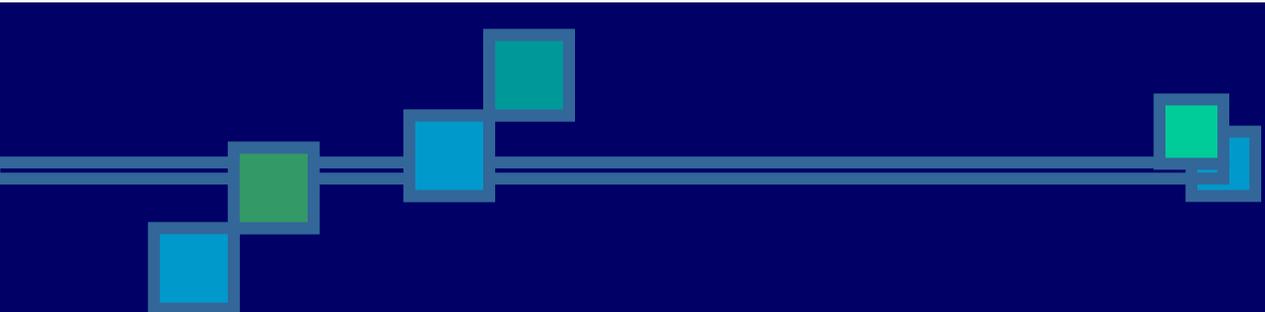


Exemple d'utilisation

Hello World !

```
<portType name="Hello">
  <operation name="sayHello" parameterOrder="String_1">
    <input message="tns:sayHello" />
    <output message="tns:sayHelloResponse" />
  </operation>
</portType>
<binding name="HelloBinding" type="tns:Hello">
  <operation name="sayHello">
    <input>
      <soap:body
encodingStyle=http://schemas.xmlsoap.org/soap/encoding/
use="encoded" namespace="http://hello.jaxrpc.samples/" />
    </input>
```

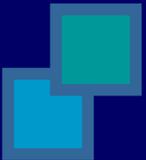


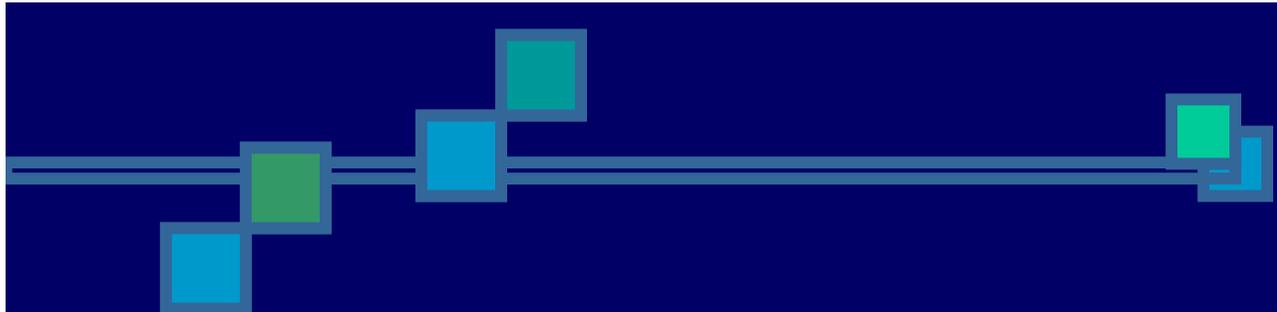


Exemple d'utilisation

Hello World !

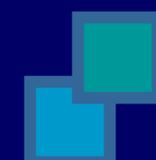
```
<output>
  <soap:body
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    use="encoded" namespace="http://hello.jaxrpc.samples/" />
</output>
<soap:operation soapAction="" />
</operation>
<soap:binding transport="http://schemas.xmlsoap.org/soap/http"
  style="rpc" />
</binding>
<service name="HelloWorld">
  <port name="HelloPort" binding="tns:HelloBinding">
    <soap:address location="http://localhost:8080/axis/Hello" />
  </port>
</service>
</definitions>
```

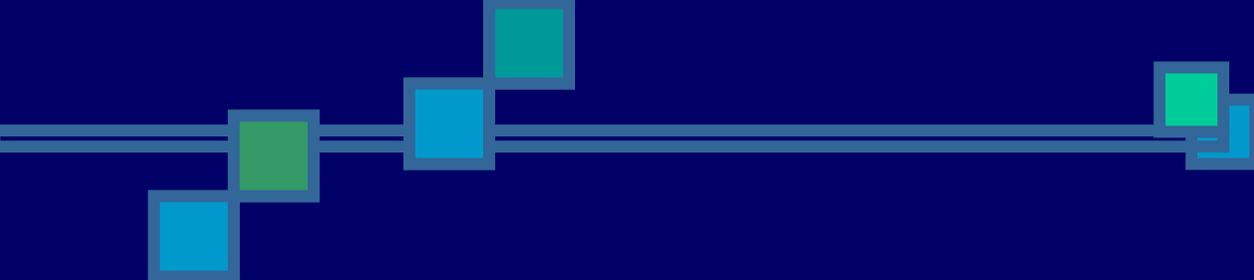




Exemple d'utilisation

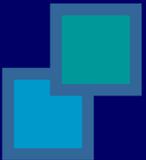
Explications

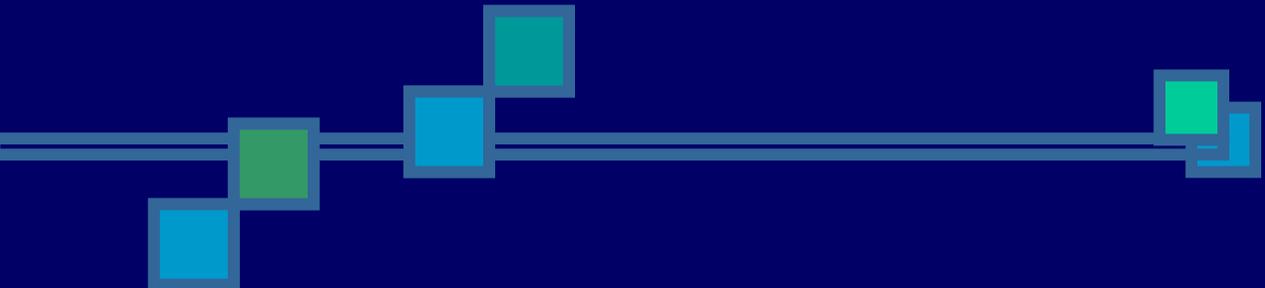
- Partie abstraite du document :
 - les types, les messages, les types de port
 - Partie concrète :
 - les bindings, les services
 - la partie concrète propose une ou plusieurs réalisations de la partie abstraite, Exemple
 - SOAP+RPC+HTTP et/ou
 - SOAP+RPC+STMP
- 



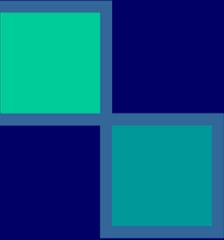
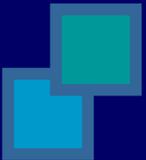
Exemple d'utilisation

API Google

- Google fournit une API permettant de faire appel à ses services de recherches
 - L'API est défini par une interface SOAP / WSDL
 - Le document WSDL comporte 2 types de ports :
 - Un pour la recherche
 - Un pour le résultat
 - Peut être implémenter dans tout langage
 - Java, PHP, VB, etc.
- 



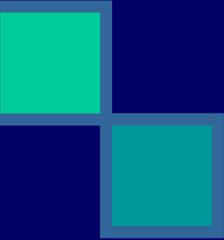
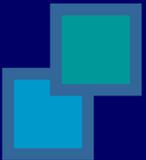
Sommaire

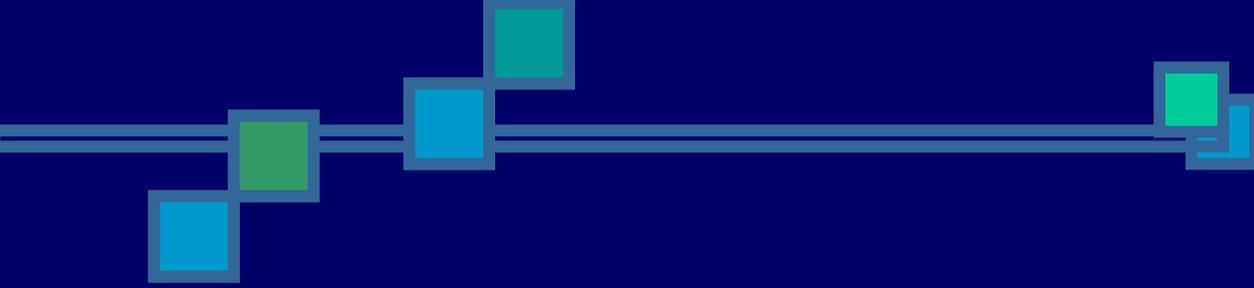
- 
- Introduction
 - Présentation de la technologie
 - Exemples d'utilisation
 - **Conclusion**
- 



Conclusion

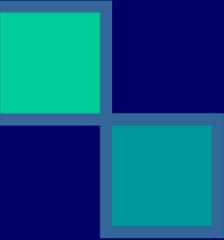
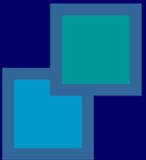
Avantages

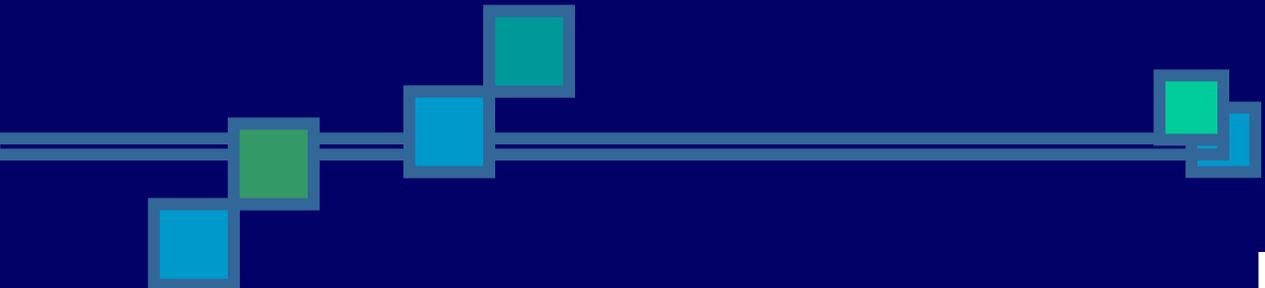
- 
- Les ressources informatiques pourraient inter opérer à travers un réseau, indépendamment de leurs plates-formes d'origine
 - WSDL offre beaucoup de souplesse :
 - le choix de la représentation des messages
 - Le choix des protocoles
 - Implémentations concrètes et différentes d'un même service
 - Très adaptées aux problèmes des communications entre applications WEB
 - Il utilise les standards de W3C
- 



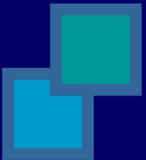
Conclusion

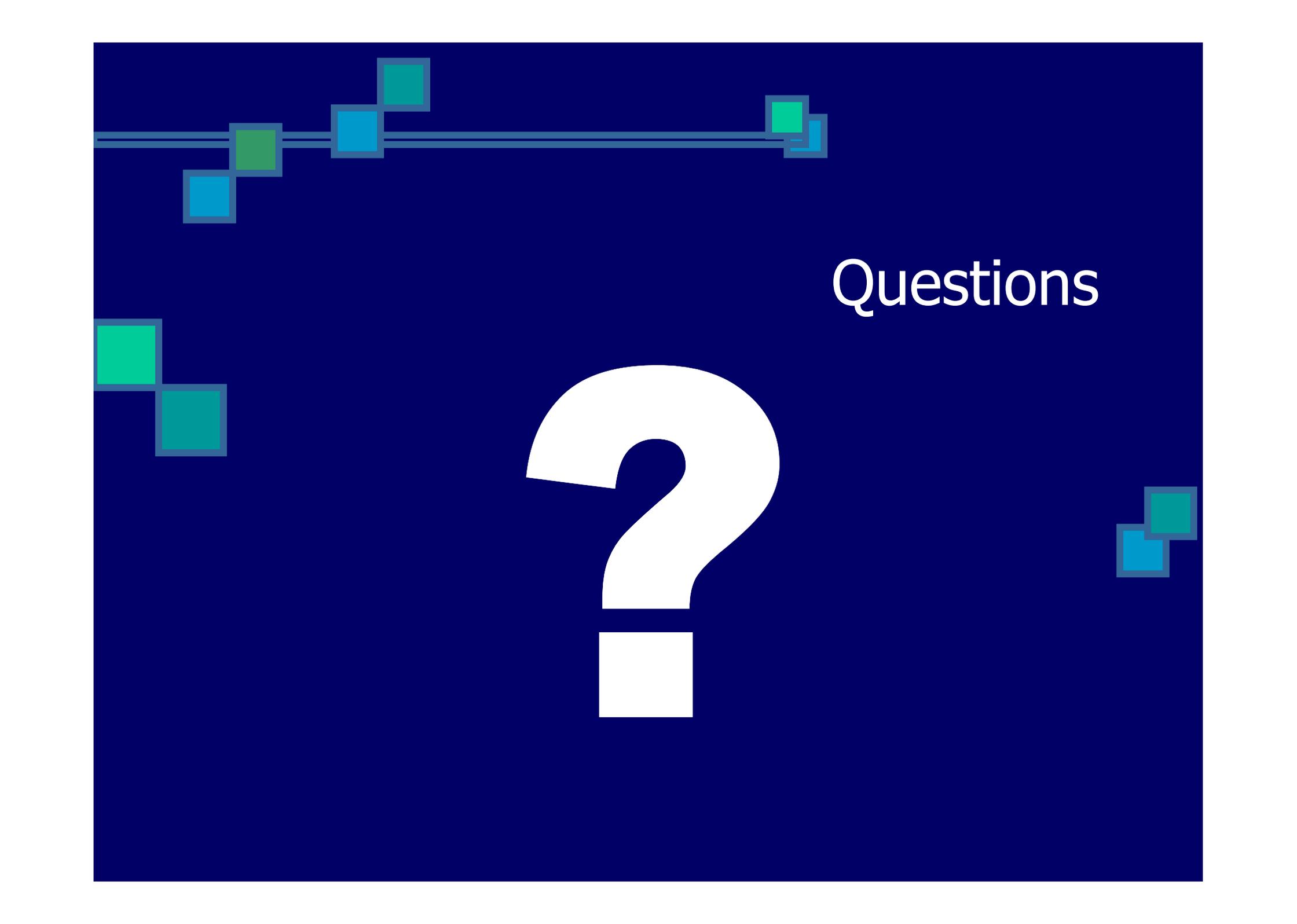
Inconvénients

- 
- L'utilisation de WSDL avec SOAP et UDDI, ne règle pas tout, il reste quelques problèmes de :
 - vocabulaire
 - sémantique pour faire dialoguer des applications
 - Problèmes de performances
 - Problèmes de sécurité ?
- 



Références

- <http://www.w3.org/2002/ws>
 - <http://encyclopedie.journaldunet.com>
 - <http://www.planetexml.com>
 - <http://www.esup-portail.org/consortium/espace>
 - <http://xmlfr.org>
 - <http://ife.developpez.com/WebService>
 - http://www.softteam.fr/technologies_web_services.php
 - <http://www.djefer.com/articles/soap/index.htm>
 - <http://www.w3.org/TR/wsdl>
- 



Questions