



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 05 : SOAP**

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

**2017-2018**

# SOAP (Simple Object Access Protocol)

- Norme W3C
- SOAP 1.0: 1999, basé sur HTTP
- SOAP 1.1: 2000, plus générique, autres protocoles
- SOAP 1.2: recommandation W3C, 2007: <https://www.w3.org/TR/soap/>
- Est un protocole de transmission de messages
- Permet des appels de procédures à distance (RPC) s'appuyant principalement sur le protocole HTTP et sur XML, mais aussi sur SMTP et POP.

# SOAP (Simple Object Access Protocol)

S'appuie sur le protocole HTTP: Tim Berners-Lee, fin 89 - début 90

Protocole de communication client/serveur basé sur TCP/IP

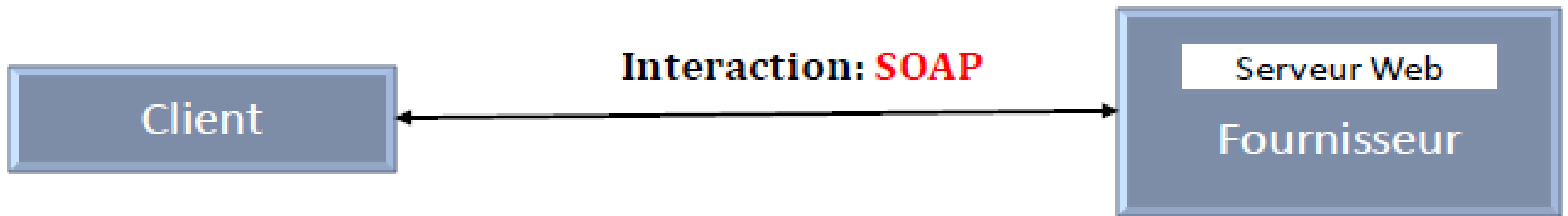
Simple pour la récupération de documents

Simple pour la transmission de données (GET, POST)

Le message SOAP est destiné au fournisseur du Service après avoir contacté l'annuaire pour chercher le service correspondant au besoin du client, les informations obtenues permettent au client de connaître la localisation du service pour pouvoir l'invoquer à l'aide de messages SOAP.

# SOAP (Simple Object Access Protocol)

- La requête SOAP intervient sur le réseau entre le client et le serveur



## 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

# SOAP : Structure

Un message SOAP est un document XML contenant les éléments suivants:

- Un élément « **Envelope** » qui identifie le document XML en tant que message SOAP
- Un élément « **Header** » d'en-tête qui contient des informations d'en-tête
- Un élément « **Body** » contenant des informations d'appel et de réponse
- Un élément « **fault** » contenant des erreurs et des informations d'état

Tous les éléments ci-dessus sont déclarés dans l'espace de noms par défaut de l'enveloppe SOAP:

<http://www.w3.org/2003/05/soap-envelope/>

Et l'espace de noms par défaut pour le codage SOAP et les types de données est:

<http://www.w3.org/2003/05/soap-encoding>

# SOAP : Structure

**Envelope:** c'est lui qui contient le message et ses différentes sous-blocs. Il s'agit du bloc racine XML. Il peut contenir un attribut *encodingStyle* dont la valeur est une URL vers un fichier de typage XML qui décrira les types applicables au message SOAP.

- **Header:** c'est un bloc optionnel qui contient des informations d'en-têtes sur le message. Si il est présent, ce bloc doit toujours se trouver avant le bloc **Body** à l'intérieur du bloc **Envelope**.
- **Body:** c'est le bloc qui contient le corps du message. Il doit absolument être présent de manière unique dans chaque message et être contenu dans le bloc **Envelope**.

SOAP ne définit pas comment est structuré le contenu de ce bloc. Cependant, il définit le bloc **Fault** qui peut s'y trouver.

- **Fault:** ce bloc est la seule structure définie par SOAP dans le bloc Body. Il sert à reporter des erreurs lors du traitement du message, ou lors de son transport. Il ne peut apparaître qu'une seule fois par message. Sa présence n'est pas obligatoire.

# Squelette d'un Message SOAP

```
<?xml version="1.0"?>
```

```
<soap:Envelope
```

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

```
soap:encodingStyle="http://www.w3.org/2003/05/soap-encoding">
```

```
<soap:Header>
```

```
...
```

```
</soap:Header>
```

```
<soap:Body>
```

```
...
```

```
<soap:Fault>
```

```
...
```

```
</soap:Fault>
```

```
</soap:Body>
```

```
</soap:Envelope>
```

Ce qui nous intéresse , l'élément Body car il contient la méthode à invoquer, les paramètres et les données (parfois on trouve aussi les types de données des paramètres)

# SOAP: Exemple (1)

## Requête SOAP getsomme envoyée à un service Sommer

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:def="http://DefaultNamespace">
  <soapenv:Header/>
  <soapenv:Body>
    <def:getsomme soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <a xsi:type="xsd:int">1</a>
      <b xsi:type="xsd:int">3</b>
    </def:getsomme>
  </soapenv:Body>
</soapenv:Envelope>
```



# SOAP: Exemple1 (suite)

## Réponse SOAP

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"  
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  
  <soapenv:Body>  
    <ns1:getSommeResponse soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:ns1="http://DefaultNamespace">  
      <getSommeReturn xsi:type="xsd:int">4</getSommeReturn>  
    </ns1:getSommeResponse>  
  </soapenv:Body>  
</soapenv:Envelope>
```

# SOAP Exemple2

## SOAP REQUEST

```
<?xml version="1.0"?>

<soap:Envelope
xmlns:soap="http://www.w3.org/2003/05/soap-envelope/"
soap:encodingStyle="http://www.w3.org/2003/05/soap-encoding">

  <soap:Body>
    <m:GetPrice xmlns:m="http://www.w3schools.com/prices">
      <m:Item>Apples</m:Item>
    </m:GetPrice>
  </soap:Body>

</soap:Envelope>
```

L'exemple ci-contre demande le prix des pommes. Notez que les éléments m: GetPrice et Item sont des éléments spécifiques à l'application. Ils ne font pas partie de l'espace de noms SOAP.

# SOAP Example2 : Suite

## SOAP RESPONSE

```
<?xml version="1.0"?>

<soap:Envelope
xmlns:soap="http://www.w3.org/2003/05/soap-envelope/"
soap:encodingStyle="http://www.w3.org/2003/05/soap-encoding">

  <soap:Body>
    <m:GetPriceResponse xmlns:m="http://www.w3schools.com/prices">
      <m:Price>1.90</m:Price>
    </m:GetPriceResponse>
  </soap:Body>

</soap:Envelope>
```