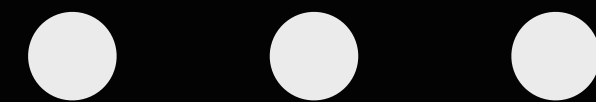




# Cours Webservice

CCI CAMPUS COLMAR



Paul LEPONT 2020-2021



# Webservice - Partie 1

Qu'est qu'une API ?



Paul LEPONT 2020-2021

# Webservice - Partie 1

Qu'est qu'une API ?

## API (Application Programming Interface)

Pour faire simple, les apis sont des passerelles qui permettent simplement à deux ou plusieurs applications de communiquer entre elles.

Ce sont des blocs de code déjà prêts qui permettent à un développeur d'implémenter des programmes qui seraient difficiles voire impossibles à implémenter sans elles.



# Webservice - Partie 1

Qu'est qu'une API ?

## API (Application Programming Interface)

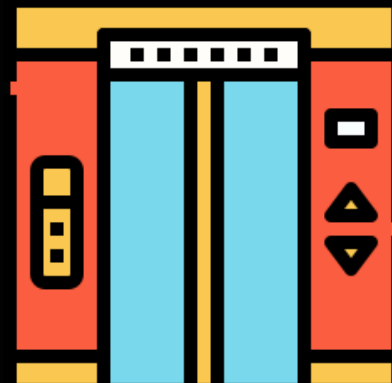
Pour être un peu plus parlant, une api serai une prise virtuelle délivrant un service  
pour un autre programme.



# Webservice - Partie 1

Qu'est qu'une API ?

## Des exemples dans la vie de tous les jours :



Un ascenseur



Une machine à café



Une télécommande



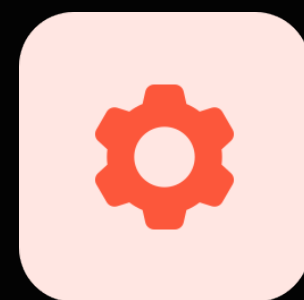
# Webservice - Partie 1

Qu'est qu'une API ?

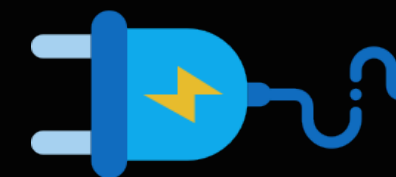
## Un exemple en programmation :



Équipe 1



Programme qui détermine  
les années bissextiles



Équipe 2



Programme qui gère  
un agenda



# Webservice - Partie 1

Qu'est qu'une API ?

## On retrouve les API un peu partout

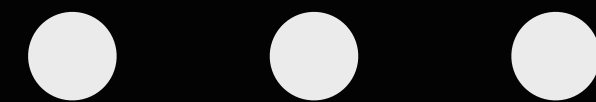
- Javascript (<https://developer.mozilla.org/fr/docs/Web/API>)
- JQuery (<https://api.jquery.com/>)
- Google (<https://developers.google.com/apis-explorer/>)
- Facebook (<https://developers.facebook.com>)
- Paypal (<https://developer.paypal.com/home>)





# Webservice - Partie 1

Qu'est ce qu'un webservice ?



Paul LEPONT 2020-2021



# Webservice - Partie 1

Qu'est qu'un webservice ?

## WEB SERVICE

Un Web Service est une application qui permet d'échanger des données avec d'autres applications web.

Ces dernières peuvent être construites dans des langages de programmation différents.

On communique en utilisant le protocole HTTP.

On peut dire que les web services sont des sous-parties des API.



Un service web s'utilise obligatoirement avec internet.

# Webservice - Partie 1

Qu'est qu'un webservice ?

## Un service web fonctionne de la manière suivante

- Le Client effectue une requête.
- Transmission de la requête via le protocole HTTP.
- Le client obtient une réponse sous format XML, JSON ou HTTP.



# Webservice - Partie 1

Qu'est qu'un webservice ?

## HTTP (HyperText Transfer Protocol)

Le HTTP est un protocole qui permet de communiquer avec un site internet. Il va permettre de charger nos pages HTML, des styles CSS, des polices, des images, etc ...

Ce protocole nous permet aussi d'envoyer des formulaires.



# Webservice - Partie 1

Qu'est qu'un webservice ?

## Les informations se trouvant dans une requête HTTP

- La méthode => L'action que l'on souhaite faire : récupérer une ressource, la modifier ...
- L'url => L'adresse du service web que vous souhaitez atteindre
- Les données

## Les informations se trouvant dans une réponse HTTP

- Le code HTTP => 200, 400, 401, 403, 404, 500 ...
- Les données





# Webservice - Partie 1

Les types de Web Service

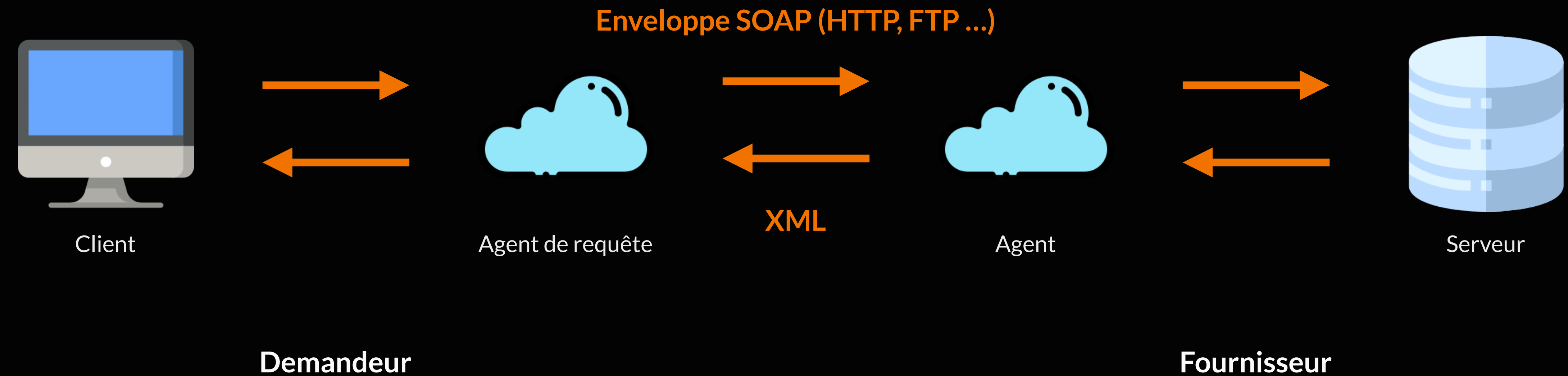


Paul LEPONT 2020-2021

# Webservice - Partie 1

Les types de Web Service

## Le protocole SOAP



# Webservice - Partie 1

Les types de Web Service

## Une enveloppe SOAP



```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" >
  <soap:Body>
    <po:purchaseOrder orderDate="2003-09-22"
      xmlns:po="http://www.Monson-Haefel.com/jwsbook/PO">
      <po:accountName>Amazon.com</po:accountName>
      <po:accountNumber>923</po:accountNumber>
      <po:address>
        <po:name>AMAZON.COM</po:name>
        <po:street>1850 Mercer Drive</po:street>
        <po:city>Lexington</po:city>
        <po:state>KY</po:state>
        <po:zip>40511</po:zip>
      </po:address>
      <po:book>
        <po:title>J2EE Web Services</po:title>
        <po:quantity>300</po:quantity>
        <po:wholesale-price>24.99</po:wholesale-price>
      </po:book>
    </po:purchaseOrder>
  </soap:Body>
</soap:Envelope>
```

# Webservice - Partie 1

Les types de Web Service

## Le protocole REST

- Aujourd'hui le protocole API le plus utilisé sur le web.
- Plus simple d'utilisation que SOAP.
- Utilise les méthodes de requête : GET, POST, PUT, DELETE.
- Renvoie une réponse le plus souvent sous format JSON.





# Webservice - Partie 1

Les types de Web Service

## Le protocole REST

[https://api.themoviedb.org/3/movie/550?api\\_key={{YOUR\\_API\\_KEY}}](https://api.themoviedb.org/3/movie/550?api_key={{YOUR_API_KEY}})



# Webservice - Partie 1

Les types de Web Service

Supporter par la  
plupart de vos CMS



# Webservice - Partie 1

Les types de Web Service

## Tester vos web services



Postman

<https://www.postman.com/>



SoapUi

<https://www.soapui.org/>





# Webservice - Partie 1

Les échanges de données



Paul LEPONT 2020-2021

# Webservice - Partie 1

Les échanges de données

## JSON (JavaScript Object Notation)

```
{  
  "squadName": "Super hero squad",  
  "homeTown": "Metro City",  
  "formed": 2016,  
  "secretBase": "Super tower",  
  "active": true,  
  "members": [  
    {  
      "name": "Molecule Man",  
      "age": 29,  
      "secretIdentity": "Dan Jukes",  
      "powers": [  
        "Radiation resistance",  
        "Turning tiny",  
        "Radiation blast"  
      ]  
    },  
  ]  
}
```



# Webservice - Partie 1

Les échanges de données

## XML (eXtensible Markup Language)

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

```
<CAT>
```

```
<NAME>Izzy</NAME>
```

```
<BREED>Siamese</BREED>
```

```
<AGE>6</AGE>
```

```
<ALTERED>yes</ALTERED>
```

```
<DECLAWED>no</DECLAWED>
```

```
<LICENSE>Izz138bod</LICENSE>
```

```
<OWNER>Colin Wilcox</OWNER>
```

```
</CAT>
```





# Webservice - Partie 1

Les appels externes en Javascript



Paul LEPONT 2020-2021

# Webservice - Partie 1

Les appels externes en Javascript

## Les objets XMLHttpRequest

Les objets **XMLHttpRequest** (XHR) permettent d'interagir avec des serveurs. On peut récupérer des données à partir d'une URL sans avoir à rafraîchir complètement la page. Cela permet à une page web d'être mise à jour sans perturber les actions de l'utilisateur.





# Webservice - Partie 1

Les appels externes en Javascript

## Les objets XMLHttpRequest

- Initialisation de l'objet : `var request = new XMLHttpRequest ()`
- Écoute lors du chargement : `request.onload = myCallbackFunction`
- Ouverture de la requête : `request.open('GET', 'www.urlofmyapi.fr')`
- Envoi de la requête : `request.send()`



# Webservice - Partie 1

Les appels externes en Javascript

## Les objets XMLHttpRequest

Dans ma fonction de callback, je récupère mon objet sous forme de 'this'.

On peut récupérer toutes sortes d'informations concernant la requête effectuée.



```
▼ XMLHttpRequest {onreadystatechange: null, abort: null, error: null, onload: f test(), onloadend: null, onloadstart: null, onprogress: null, onreadystatechange: null, ontimeout: null, readyState: 4, response: "{\\\"page\\\":1", responseText: "{\\\"page", responseType: "", responseURL: "https://", responseXML: null, status: 200, statusText: "", timeout: 0, upload: XMLHttpRequest, withCredentials: false, [[Prototype]]: XMLHttpRequest}
```

# Webservice - Partie 1

Les appels externes en Javascript

## La méthode JQuery \$.ajax()

- url => Url de la requête.
- method => Permet de préciser la méthode d'envoi de la requête (GET, POST, PUT, DELETE ...)
- dataType => Le type de données qu'on attend en réponse du serveur.
- data => Contient les données à envoyer au serveur.
- async => Permet de préciser si la requête doit être envoyée de manière asynchrone ou pas.



# Webservice - Partie 1

Les appels externes en Javascript

## La méthode JQuery \$.ajax() Exemple pour GET



```
$.ajax({  
  
    method: "GET",  
  
    url: "www.monapiexterne.com"  
  
})  
  
.done(function(response){  
  
    console.log(response)  
  
})  
  
.fail(function(error){  
  
    alert(error)  
  
})
```

# Webservice - Partie 1

Les appels externes en Javascript

## La méthode JQuery \$.ajax() Exemple pour **POST**



```
$.ajax({  
  
    method: "POST",  
  
    url: 'www.monapiexterne.com',  
  
    data : { id : '12' , name : 'Johanna' }  
  
})  
  
.done(function(response){  
  
    console.log(response)  
  
})  
  
.fail(function(error){  
  
    alert(error)  
  
})
```



# Webservice - Partie 1

Sécurisation des web service



Paul LEPONT 2020-2021

# Webservice - Partie 1

Cas pratique

## Sécurisation au niveau du transport



**HTTPS**

(Port 443)



**HTTP**

(Port 80)



# Webservice - Partie 1

Cas pratique

## Quelles bonnes pratiques pour sécurisé ?

- Utiliser des jetons (Token)
- Utiliser le chiffrement et des signatures
- Identifier les vulnérabilités
- Utiliser des quotas et la limitation de requêtes

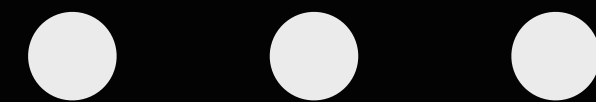






# Webservice - Partie 1

Cas pratique



Paul LEPONT 2020-2021

# Webservice - Partie 1

Cas pratique

## Récupération et utilisation d'api REST

- Se connecter sur <https://developers.themoviedb.org/>
- Voir les informations que l'on récupère à travers Postman.
- Afficher ces informations dans notre page web.
- Envoyer une note concernant un film
- Effectuer une recherche

