

# Introduction à Ajax

---

CNAM le 1<sup>er</sup> mars 2008  
O. Pons    S. Rosmorduc

# Principe général de fonctionnement

- ▶ Faire des **requêtes http** sans avoir recharger une page entière.
- ▶ Communication de javascript avec les serveurs
- ▶ Ajout de l'objet (la classe) XMLHttpRequest a Javascript
- ▶ Asynchronous Javascript And XML

## creation d'un objet XMLHttpRequest

- ▶ Créer un objet pour chaque requete http.
- ▶ (historique) different selon le selon le navigateur
- ▶ Avec des if

```
if (window.XMLHttpRequest)
    //XMLHttpRequest est un objet
    //de la fenetre courant
{
    xhr = new XMLHttpRequest();
    // tous sauf IE<7, Firefox, Safari, ...
}
else
if (window.ActiveXObject)
    // Version Active
{
    xhr = new ActiveXObject("Microsoft.XMLHTTP");
    // IE
}
```

## creation d'un objet XMLHttpRequest

- ▶ Avec des try

```
try {
    xhr = new XMLHttpRequest("Microsoft.XMLHTTP");
        // essayer IE
    }
catch(e) // si Echech c'est pas IE
{
    xhr = new XMLHttpRequest()
}
```

# Utilisation d' XMLHttpRequest

Une fois l'object creer :

- ▶ la methode **open** permet d'ouvrir une connection
- ▶ la methode **send** permet d'envoyer une requete

apres la requete l'object contient (entre autre)

- ▶ le code de retour
- ▶ le document resultant de la requete

## Example trivial

```
<head><title> exemple</title>
  <script type="text/javascript">
    function exemple1(){
      var req = new XMLHttpRequest();
      alert(req);
      req.open('GET', 'test.php', false);//
      req.send(null);
      alert(req.status);
      if(req.status == 200)
        alert(req.responseText);
    }
  </script>
</head>
<body>
  <h1 onclick='exemple1()>clic moi pour tester</h1>
  <h1 onclick='alert("debloque")>bloquer</h1>
</body>
```

## Les méthodes

- ▶ open (Méthode , URL , type)
  - ▶ GET, POST
  - ▶ URL (un même domaine)
  - ▶ true ou false : asynchrone(true) ou synchrone(false)
- ▶ send (argument)
  - ▶ null
  - ▶ data :  
`unnom=unevaleur&autrenom=autre valeur&etenore=uneautre`
- ▶ abort
- ▶ `getResponseHeader(header)`
- ▶ `setRequestHeader(header, value)`  
si méthode POST changer le MIME :  

```
httpRequest.setRequestHeader('Content-Type',  
    'application/x-www-form-urlencoded');
```

## Les attributs

- ▶ `status` (200, 404 ...)
- ▶ `statusText` (OK, not found ...)
- ▶ `responseText` : du text
- ▶ `responseXML` : du xml
- ▶ `readyState`
- ▶ `onreadystatechange`



# Asynchrone

L'exemple était synchrone, donc bloquant

- ▶ si réponse immédiate et traitement court :

```
<?php
    echo "on a fini ...."
?>
```

- ▶ si traitement long cote serveur

```
<?php
    sleep(30); /attendre 30 secondes
    echo "on a fini ...."
?>
```

# Asynchrone

- ▶ m'attend pas la réponse pour continuer son exécution,
- ▶ doit pouvoir connaître l'état de traitement de la requête : `readyState`
- ▶ doit être prévenue des changements et décider quoi faire `onreadystatechange`

# readyState

Les états de `readyState` :

- ▶ 0 : non initialisé.
- ▶ 1 : connexion établie.
- ▶ 2 : requête reçue.
- ▶ 3 : réponse en cours.
- ▶ 4 : terminé. (le plus utile)

## onreadystatechange

- ▶ `onreadystatechange=functionDeTraitement`

## exemple 2

```
<head><title> exemple </title>
  <script type="text/javascript">
    function exemple2(){
      var req = new XMLHttpRequest();
      alert(req);
      req.onreadystatechange=function(){
        if(req.readyState == 4){
          alert(req.status);
          if(req.status == 200)
            alert(req.responseText);
        }
      }
      req.open('GET', 'test.php',true);//
      req.send(null);}
</script> </head> <body>
  <h1 onclick='exemple2()>clic moi pour tester</h1>
  <h1 onclick='alert("pas")>pas bloquer</h1>
</body>
```

# Javascript

- ▶ via le DOM
- ▶ via innerHTML

## Example

```
<head><title> exemple </title>
  <script type="text/javascript">
    function exemple2(){
      var req = new XMLHttpRequest();
      alert(req);
      req.onreadystatechange=function(){
        if(req.readyState == 4){
          alert(req.status);
          if(req.status == 200)
            document.getElementById("unid").innerHTML=
                req.responseText;}}
      req.open('GET', 'test.php',true);//
      req.send(null);}
</script> </head> <body>
  <h1 onclick='exemple2()>clic moi pour tester</h1>
  <div id="unid"></div>
</body>
```

# XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<catalog>
  <cd>
    <title>Empire burlesque</title>
    <artist>Bob Dylan</artist>
    <country>usa</country>
    <company>columbia</company>
    <year>1985</year>
  </cd>
  <cd>
    <title>Hide your heart</title>
    <artist>Bonnie Tyler</artist>
    <country>uk</country>
    <company>cbs records</company>
    <year>1988</year>
  </cd>
</catalog>
```



# XML

```
<html><head><title> exemple </title>
  <script type="text/javascript">
    function exemple3(){
      var req = new XMLHttpRequest();
      alert(req);
      req.onreadystatechange=function(){
        if(req.readyState == 4){
          if(req.status == 200)
            var xmlDoc = req.responseXML;
            var root_node =
              xmlDoc.getElementsByTagName('artist').item(0);
            alert(root_node.firstChild.data);}}
      req.open('GET', 'catalog.xml',true);//
      req.send(null);}
</script></head><body>
  <h1 onclick='exemple3()>clic moi pour tester</h1>
</body></html>
```