

CSI 3540

Structures, techniques et normes du Web

Traitements asynchrones et applications réactives

Objectifs:

- Introduction à Ajax
- Développer des applications réactives

Lectures:

- Web Technologies (2007) § 7

Plan

1. Survol de Ajax

2. Bibliothèques de programmes

XHTML côté client (Ajax)

XPath côté-client

- **XPath** est-il supporté par les navigateurs?
- Bien sûr, puisque les feuilles de styles (**XSL**) sont supportées et qu'elles utilisent **XPath**!
- Est-ce **JavaScript** supporte **XPath**?

XPath et JavaScript

```
function init() {
```

```
    var nextElem;
```

```
    var nodes = document.evaluate( '//strong', document, null, XPathResult.ANY_TYPE, null );
```

```
    var elem = nodes.iterateNext();
```

```
    while ( elem ) {
```

```
        elem.style.color = "red";
```

```
        var parent = document.evaluate( '../..', elem, null, XPathResult.FIRST_ORDERED_NODE_TYPE, null );  
        parent.singleNodeValue.style.color = "orange";
```

```
        elem = nodes.iterateNext();
```

```
    }
```

```
    return;
```

```
}
```



```

<head>
<title>La passante ; Emile NELLIGAN (1879-1941)</title>
<meta http-equiv="Content-Type" content="text/html; charset=latin1" />
<script type="text/javascript" src="index.js">
</script>
</head>
<body style="font-size:large" onload="init();">
<h1>Émile NELLIGAN (1879-1941)</h1>
<h2>La passante</h2>
<pre>
(1) Hier, j'ai vu passer, comme une ombre qu'on plaint,
En un grand parc obscur, une femme voilée :
Funèbre et singulière, elle s'en est allée,
Recélant sa fierté sous son masque opalin.
</pre>
<pre>
(2) Et rien que d'un regard, par ce soir cristallin,
J'eus deviné bientôt <i>sa douleur <strong>refoulée</strong>
Puis elle disparut en quelque noire allée
Propice au deuil profond dont son coeur était plein.
</pre>
<ul>
<li>Grand</li>
<li><strong>Jacques</strong> 1964</li>
</ul>
<pre>
(3) Ma jeunesse est pareille à la pauvre passante :
Beaucoup la croiseront ici-bas dans la sente
Où la vie à la tombe âprement nous conduit;
</pre>
<pre>
(4) Tous la verront passer, feuille sèche à la brise
Qui tourbillonne, tombe et se fane en la nuit ;
Mais nul ne l'aimera, nul ne l'aura comprise.
</pre>
</body>
</html>

```

La passante ; Emile NELLIGAN (1879-1941)

index.html Google

Émile NELLIGAN (1879-1941)

La passante

(1) Hier, j'ai vu passer, comme une ombre qu'on plaint,
En un grand parc obscur, une femme voilée :
Funèbre et singulière, elle s'en est allée,
Recélant sa fierté sous son masque opalin.

(2) Et rien que d'un regard, par ce soir cristallin,
J'eus deviné bientôt *sa douleur **refoulée*** ;
Puis elle disparut en quelque noire allée
Propice au deuil profond dont son coeur était plein.

- Grand
- Jacques 1964

(3) Ma jeunesse est pareille à la pauvre passante :
Beaucoup la croiseront ici-bas dans la sente
Où la vie à la tombe âprement nous conduit;

(4) Tous la verront passer, feuille sèche à la brise
Qui tourbillonne, tombe et se fane en la nuit ;
Mais nul ne l'aimera, nul ne l'aura comprise.

Connaissez-vous Ajax?

- Une ville de l'Ontario



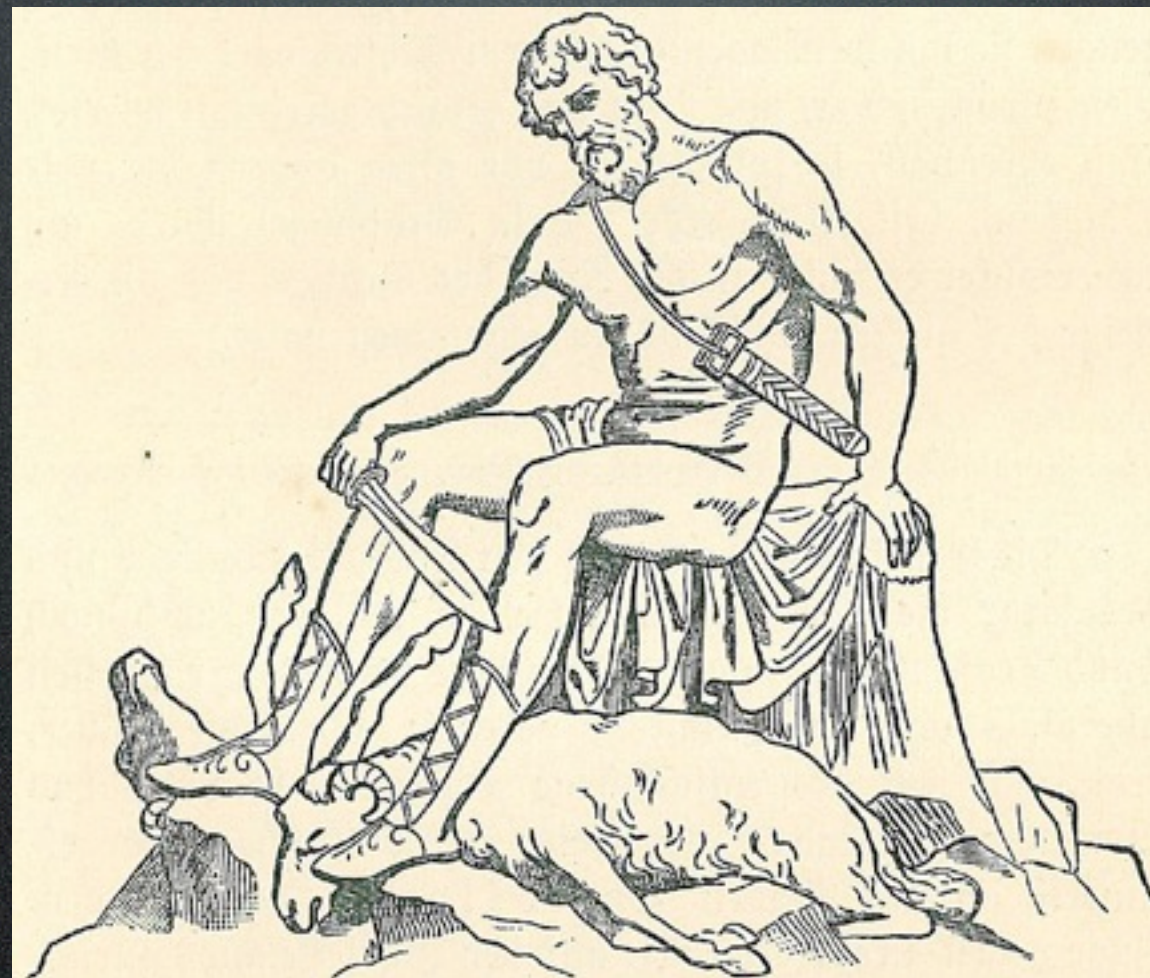
Connaissez-vous Ajax?

- Équipe de soccer d'Amsterdam



Connaissez-vous Ajax?

- Deux héros de la mythologie grèque ; fils de Télamon, fils d'Oïlée



Connaissez-vous Ajax?

- Connaissez-vous **GMail** et **Google Maps**?
- Si oui, vous avez utilisé la technologie Ajax
- **Asynchronous JavaScript + XML**


Ajax :

voyez par vous même

- **Google Suggest :**
<http://www.google.com/webhp?complete=1&hl=en>
- **Google Maps :**
<http://maps.google.ca>
- **Google Mail (Gmail)**
<http://gmail.com>
- Et probablement **Google Docs, Spreasheets, Calendar, etc.**

aircanada.com - The freedom to fly your own way.

http://www.aircanada.com/en/home.html

AIR CANADA  [Join now/Sign in](#)

[Canadian Edition](#) | [Français](#) | [Contact Us](#)

Book Travel | **Manage my Bookings** | **Special Offers** | **Information & Services**

Flights | Flight Passes | Hotels | Cars | Vacations

Travel: Round-trip [Multi-city](#)

Leaving from: ot

Going to: Enter the 3 first characters of:

- City Name
- Airport Name or Code
- Country Name
- Province or U.S. State


Departure date:

Return date:

Adult (16+): 1 [Children?](#) [More than 9 passengers](#)

Country of residence: Canada

Promotion Code: ? **SEARCH**



Can't get enough New York?

The New York Weekender Pass, starting at \$549 for two round trips. Offer ends May 31, 2008.

[Learn more](#)

Offers

- webSaver:** Daily Deals
- Great deals for early summer travel.** Book now!
- New York Weekender Pass:** only \$549!
- Up to 10,000 Bonus Aeroplan Miles:** on our new nonstop Ottawa-Frankfurt
- Paris Hotels:** One FREE night hotel stay in Paris
- Hertz car rentals:** 50% Off or save \$30 plus earn Aeroplan Miles


[Check-in...](#) and get your **boarding pass**

[Flight Status...](#) check **Arrivals & Departures**

[All offers...](#)

aircanada.com - The freedom to fly your own way.

http://www.aircanada.com/en/home.html

AIR CANADA  [Join now/Sign in](#)

[Canadian Edition](#) | [Français](#) | [Contact Us](#)

Book Travel | **Manage my Bookings** | **Special Offers** | **Information & Services**

Flights | **Flight Passes** | **Hotels** | **Cars** | **Vacations**

Travel: [Multi-city](#)

Leaving from:

Going to: [Map search](#)


Departure date:

Return date:

Adult (16+): [Children?](#) [More than 9 passengers](#)

Country of residence:

Promotion Code:



Can't get enough New York?

The New York Weekender Pass, starting at \$549 for two round trips. Offer ends May 31, 2008.

[Learn more](#)

Special Offers

- **webSaver:** Daily Deals
- **Great deals for early summer travel.** Book now!
- **New York Weekender Pass:** only \$549!
- **Up to 10,000 Bonus Aeroplan Miles:** on our new nonstop Ottawa-Frankfurt
- **Paris Hotels:** One FREE night hotel stay in Paris
- **Hertz car rentals:** 50% Off or save \$30 plus earn Aeroplan Miles


[Check-in...](#) and get your **boarding pass**

[Flight Status...](#) check **Arrivals & Departures**

[All offers...](#)

aircanada.com - The freedom to fly your own way.

http://www.aircanada.com/en/home.html

AIR CANADA  [Join now/Sign in](#)

[Canadian Edition](#) | [Français](#) | [Contact Us](#)

Book Travel | **Manage my Bookings** | **Special Offers** | **Information & Services**

Flights | Flight Passes | Hotels | Cars | Vacations

Travel: [Multi-city](#)

Leaving from:

Going to:

Departure date:

Return date:

Adult (16+):


Country of residence:

Promotion Code:

Enter the 3 first characters of:

- City Name
- Airport Name or Code
- Country Name
- Province or U.S. State

[Close](#)



Can't get enough New York?

The New York Weekender Pass, starting at \$549 for two round trips. Offer ends May 31, 2008.

[Learn more](#)

Offers

Category: Daily Deals

- Great deals for early summer travel.** Book now!
- New York Weekender Pass:** only \$549!
- Up to 10,000 Bonus Aeroplan Miles:** on our new nonstop Ottawa-Frankfurt
- Paris Hotels:** One FREE night hotel stay in Paris
- Hertz car rentals:** 50% Off or save \$30 plus earn Aeroplan Miles


[All offers...](#)

[Check-in...](#) and get your **boarding pass**

[Flight Status...](#) check **Arrivals & Departures**

aircanada.com - The freedom to fly your own way.

http://www.aircanada.com/en/home.html

AIR CANADA  [Join now/Sign in](#)

[Canadian Edition](#) | [Français](#) | [Contact Us](#)

Book Travel | **Manage my Bookings** | **Special Offers** | **Information & Services**

Flights | Flight Passes | Hotels | Cars | Vacations

Travel: [Multi-city](#)

Leaving from:

Going to:

Departure date:

Return date:

Adult (16+):

Country of residence:

Promotion Code:

[Check-in...](#) and get your **boarding pass**

[Flight Status...](#) check **Arrivals & Departures**

Can't get enough New York?

The New York Weekender Pass, starting at \$549 for two round trips. Offer ends May 31, 2008. [Learn more](#)

Offers

- Up to 10,000 Bonus Aeroplan Miles:** on our new nonstop Ottawa-Frankfurt
- Paris Hotels:** One FREE night hotel stay in Paris
- Hertz car rentals:** 50% Off or save \$30 plus earn Aeroplan Miles

[All offers...](#)

ADOBE® PHOTOSHOP® EXPRESS BETA

Join Now

Already a Member? [Sign In](#)



Amplify Your Image ✂

Crop, rotate, tweak, twirl, and show off photos here on Photoshop Express, on your blog or Facebook page. Give your friend a tragically oversized head or get rid of the mole on his cheek that's the size of a small asteroid. It's totally up to you.

2GB Free Rent

There's nothing sadder than a photo without a home. You hate to see that. Give your photos a free ride with Photoshop Express. Just sign up, then start uploading, polishing and showing off up to 2 GB of photos on our dime.

+ Your Gallery or Mine?

The world needs to know how very excellent your photos are, so you've got to show them off in a big way. Might want to do that in your Gallery. Display your images in 3D, or a grid. Or like an alien crop circle. The choice is yours. You control how everything looks in your Gallery.

Start here →

Join Now

or

take a

Test Drive

[Go Browse Galleries](#)

Upload Photos

- Library
 - All Photos (1)
 - Most Recent (1)
- Albums + x
- Other Sites
 - Facebook [Login](#)
 - Photobucket [Login](#)
 - Picasa [Login](#)

Most Recent

View (By Newest Date)



DSCN3267.jpg
 ★ ★ ★ ★ ★ 06/21/2006
 click to add caption

Search

- Create Album
- Edit Photo
- E-mail Photo
- Remove Photo

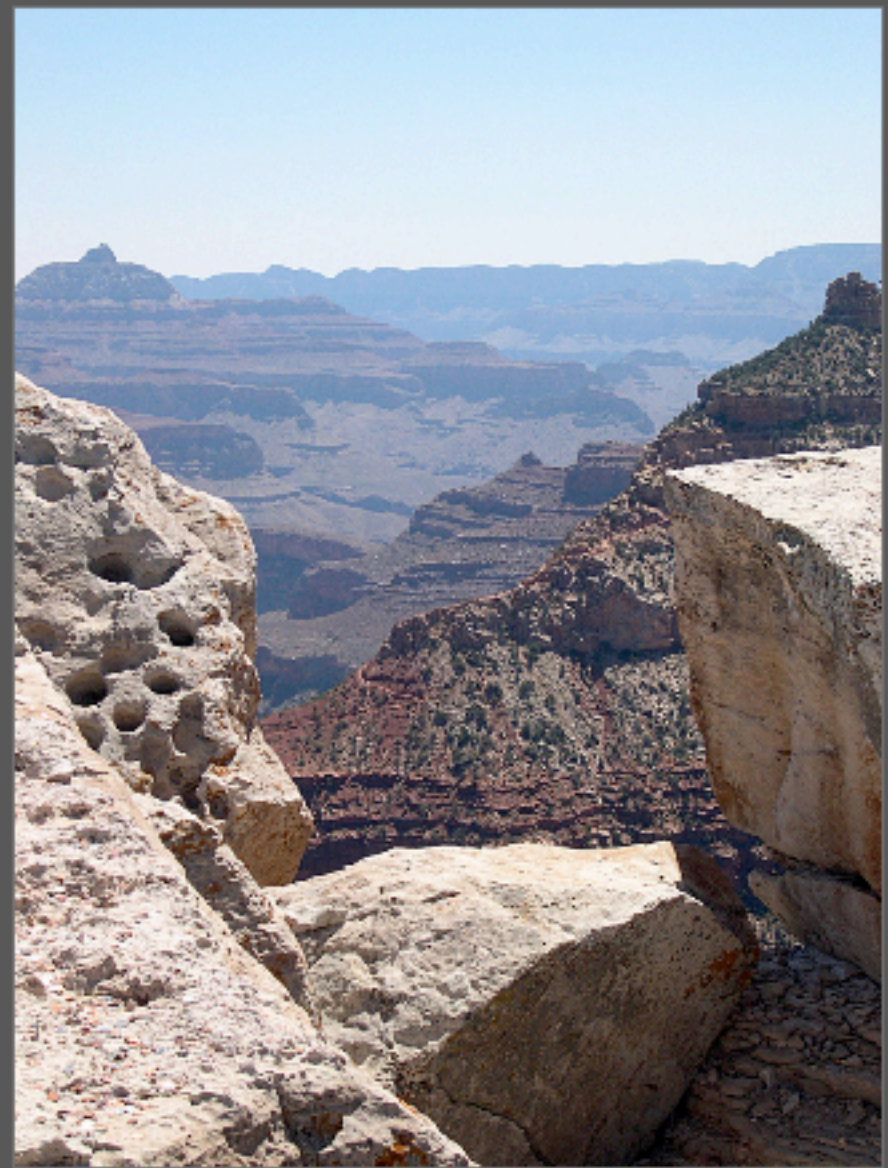
Upload Photos

Most Recent

View (By Newest Date)



- Library
 - All Photos (1)
 - Most Recent (1)
- Albums + x
- Other Sites
 - Facebook [Login](#)
 - Photobucket [Login](#)
 - Picasa [Login](#)




Search

Create Album
 Edit Photo
 E-mail Photo
 Remove Photo

Upload Photos

- Library
 - All Photos (1)
 - Most Recent (1)
- Albums + x
- Other Sites
 - Facebook [Login](#)
 - Photobucket [Login](#)
 - Picasa [Login](#)

Most Recent

Thumbnail	Filename	Caption	Albums	Taken	Posted	Modified	Wid	Ht	Rating
	DSCN3267.jpg		Most Recent	06/21/2006 01:39 PM	03/27/2008 04:11 PM	03/27/2008 04:16 PM	1536	2048	★ ★

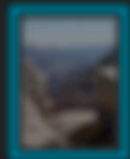
Search

 Create Album
  Edit Photo
  E-mail Photo
  Remove Photo

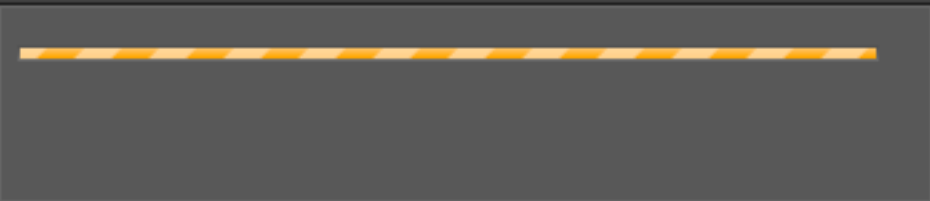
Uploaded Photos

- Library
 - All Photos (1)
 - Most Recent (1)
- Albums
- Other Sites
 - Facebook Login
 - Photobucket Login
 - Picasa Login

Most Recent

Thumbnail	Filename	Caption	Albums	Taken	Posted	Modified	Wid	Hgt	Ratio
	DISC0267.jpg		Most Recent	06/21/2008 04:20 PM	06/27/2008 04:11 PM	06/27/2008 04:16 PM	1536	2048	

Preparing Photo for Edit...

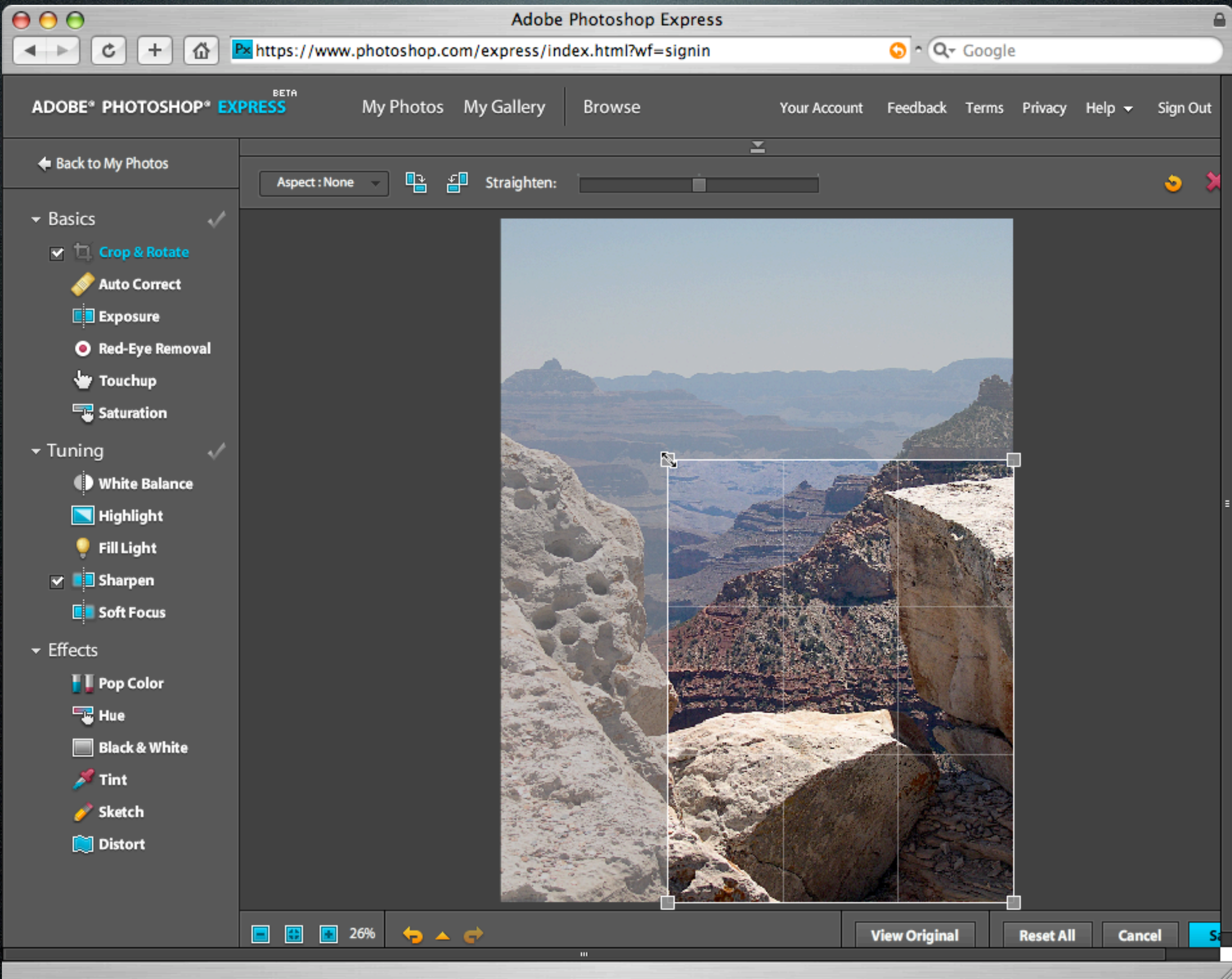


Cancel

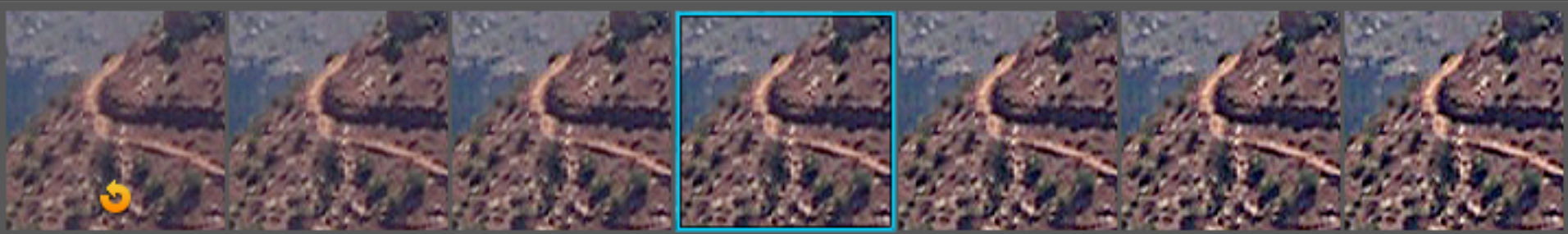
← Back to My Photos

- Basics
 - Crop & Rotate
 - Auto Correct
 - Exposure
 - Red-Eye Removal
 - Touchup
 - Saturation
- Tuning ✓
 - White Balance
 - Highlight
 - Fill Light
 - Sharpen
 - Soft Focus
- Effects
 - Pop Color
 - Hue
 - Black & White
 - Tint
 - Sketch
 - Distort





Back to My Photos



- Basics
 - Crop & Rotate
 - Auto Correct
 - Exposure
 - Red-Eye Removal
 - Touchup
 - Saturation
- Tuning
 - White Balance
 - Highlight
 - Fill Light
 - Sharpen
 - Soft Focus
- Effects
 - Pop Color
 - Hue
 - Black & White
 - Tint
 - Sketch
 - Distort



- **Ajax ce n'est pas une technologie, mais plutôt l'amalgame de plusieurs technologies :**
 - Présentation : **XHTML + CSS**
 - Affichage et interactions dynamiques à l'aide du **DOM**
 - **XML + XSLT** pour les échanges d'informations
 - **XMLHttpRequest** pour les accès au serveur
 - Le tout cimenté à l'aide de **JavaScript**

Ajax

- La bonne nouvelle est que vous connaissez déjà toutes ces technologies

Ajax : XMLHttpRequest, CSS, JavaScript, DOM

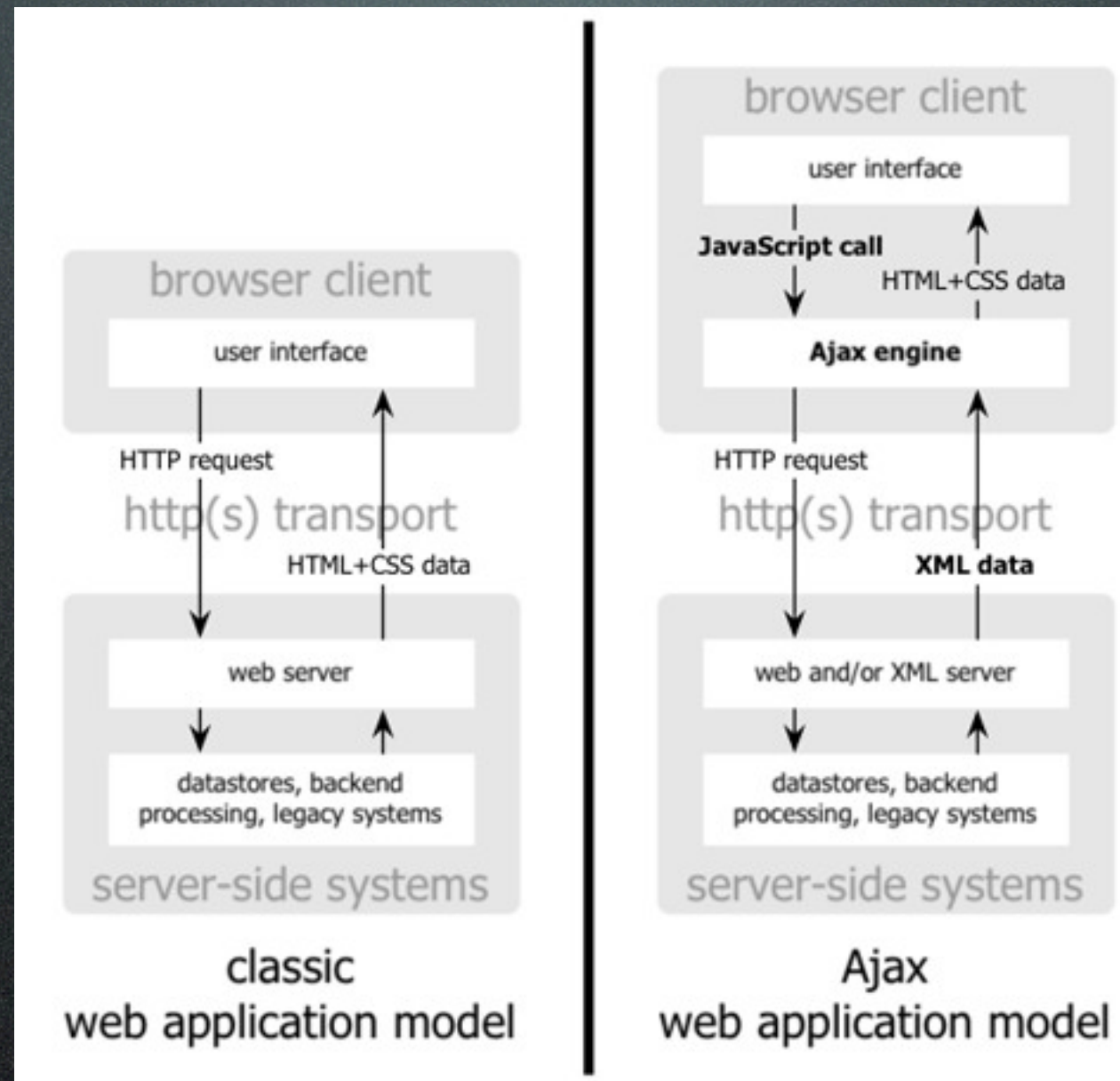


Faire glisser une image

Ajax

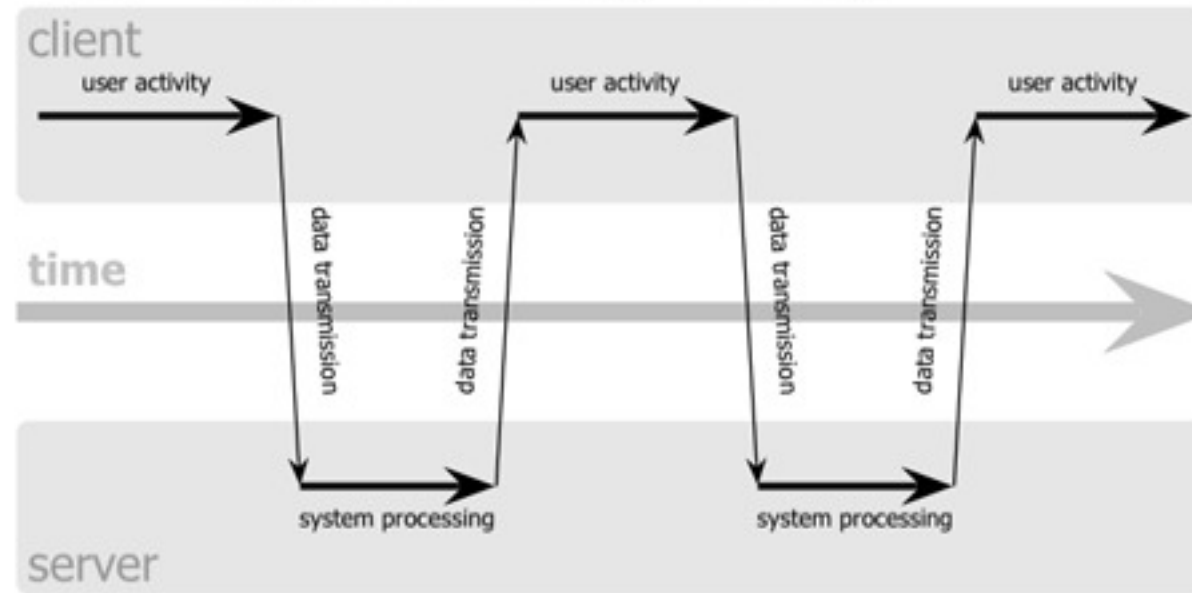
- La bonne nouvelle est vous connaissez déjà toutes ces technologies
- Possiblement à l'exception de de **XMLHttpRequest (XHR)**

Ajax

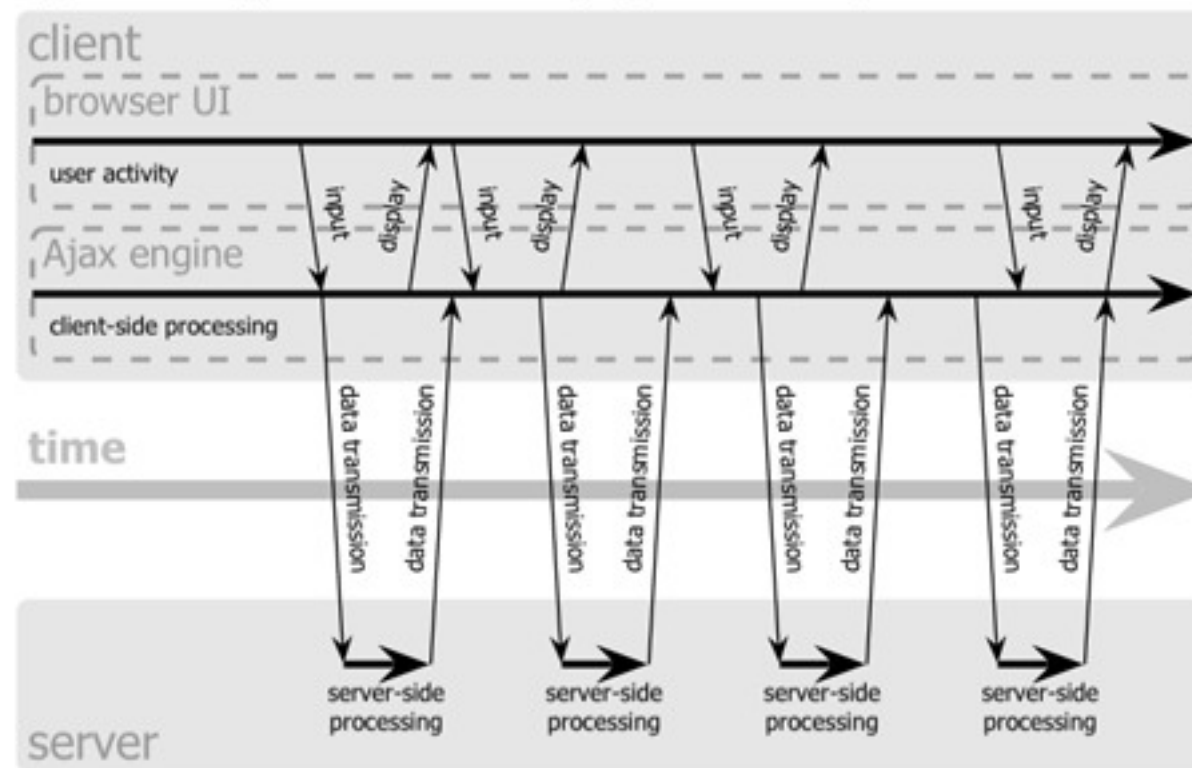


Jesse James Garrett (18 février 2005) Ajax: A New Approach to Web Applications
[<http://www.adaptivepath.com/publications/essays/archives/000385.php>]

classic web application model (synchronous)



Ajax web application model (asynchronous)

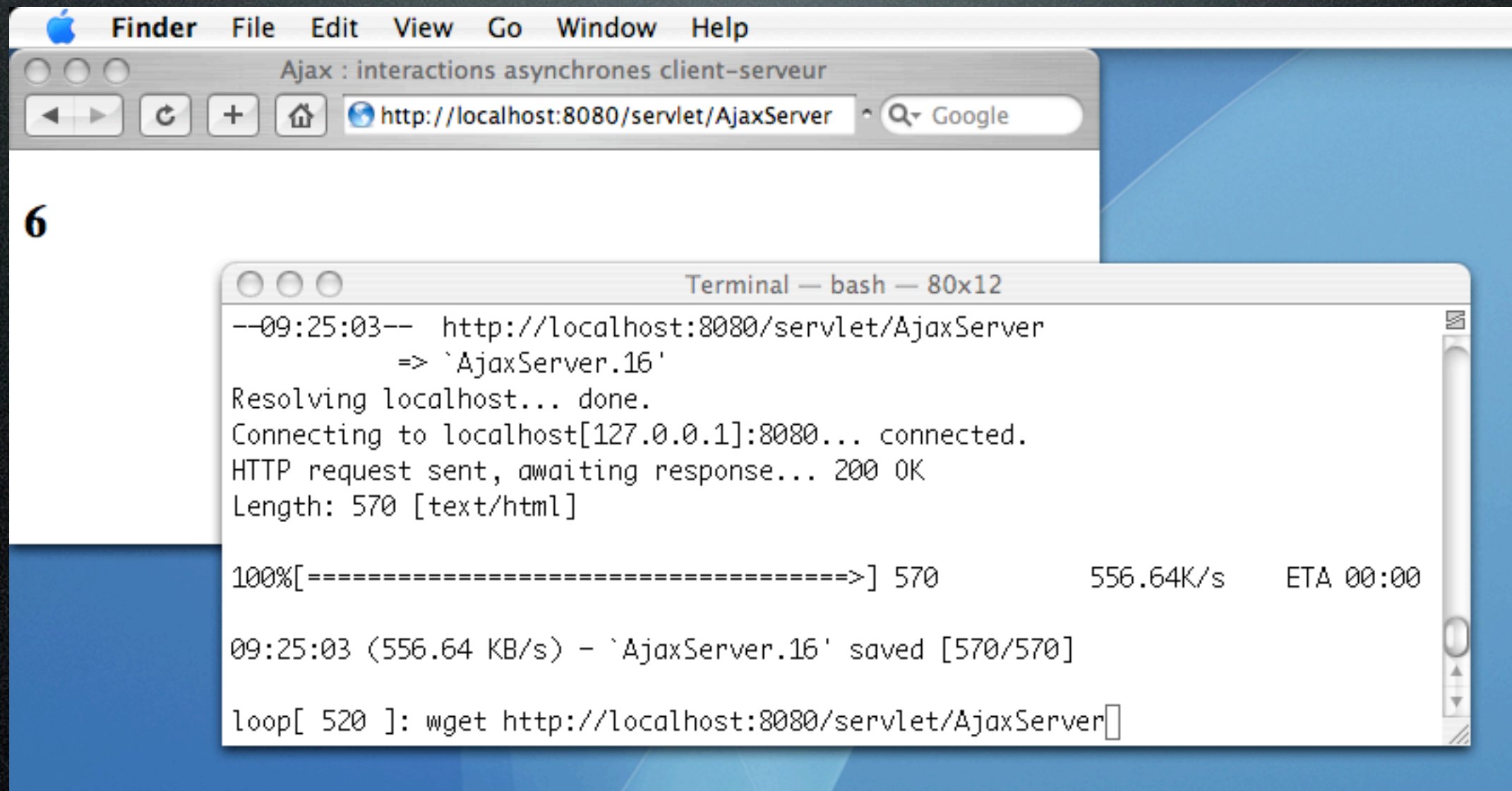


Jesse James Garrett (18 février 2005) Ajax: A New Approach to Web Applications [<http://www.adaptivepath.com/publications/essays/archives/000385.php>]

XMLHttpRequest (XHR)

- **Objet hôte JavaScript**
- Un programme JavaScript envoie une requête HTTP au serveur à l'aide de **XMLHttpRequest**
- Il reçoit un **document XML**
- **Okay**, pas forcément un document XML

Ajax : mise à jour asynchrone d'un compteur



Ajax : UI

```
public void doGet( HttpServletRequest request, HttpServletResponse response )  
    throws ServletException, IOException {
```

```
    response.setContentType( "text/html; charset=UTF-8" );
```

```
    ⇒ count++;
```

```
    PrintWriter doc = response.getWriter();
```

```
    ⇒ doc.println( "..." );
```

```
    doc.close();
```

```
}
```


Ajax : UI

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
  encoding="text/html; charset="UTF-8"" lang="fr-CA">
  <head>
    <title>Ajax : interactions asynchrone entre client et serveur</title>
    <script type='text/javascript' src='/AjaxClient.js'>
    </script>
    <meta http-equiv='Content-Script-Type' content='text/javascript' />
  </head>
  <body onload='init();' style="font-size:x-large">
    <p>
      <b id='dynamic'>1</b>
    </p>
  </body>
</html>
```


Ajax : côté client (engin Ajax)

```
function init() {  
    setInterval( "getDynamicContent()", 5000 )  
}
```

Au 5 secondes, faire un appel à
getDynamicContent()

Ajax : côté client (engin Ajax)

```
function getDynamicContent() {  
  var xhr = new XMLHttpRequest();  
  if ( xhr ) {
```

Communications
asynchrones



```
    xhr.open( "POST", "/servlet/AjaxServer", true );
```

```
    xhr.onreadystatechange = function () { updateDynamicContent( xhr ); };
```

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

```
    xhr.send( "" );
```

```
  }
```

```
  return;
```

```
}
```

subtilité (capture connection):

```
connection.onreadystatechange =  
nom d'une fonction
```


vs

```
connection.onreadystatechange =  
function()  
{updateDynamicContent  
( connection ) }
```


Ajax : côté client (engin Ajax)

```
function updateDynamicContent( connection ) {  
  if ( connection.readyState == 4 && connection.status == 200 ) {  
    var elem = document.getElementById( "dynamic" );  
    var info = connection.responseXML.documentElement;  
    elem.childNodes[ 0 ].data = info.childNodes[ 0 ].data;  
  }  
  return;  
}
```

Remplace le contenu
d'elem par celui d'info

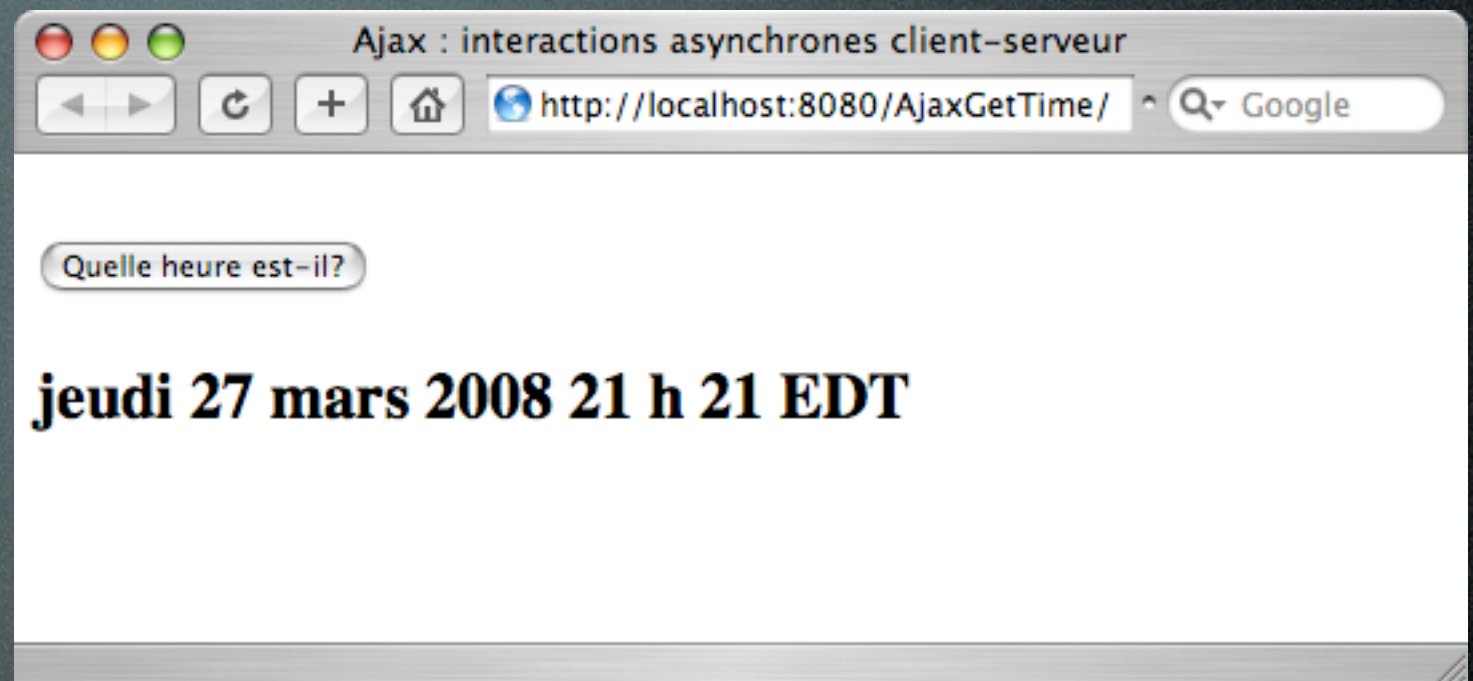


readyState = 4 -> tout reçu
status = 200 -> succès

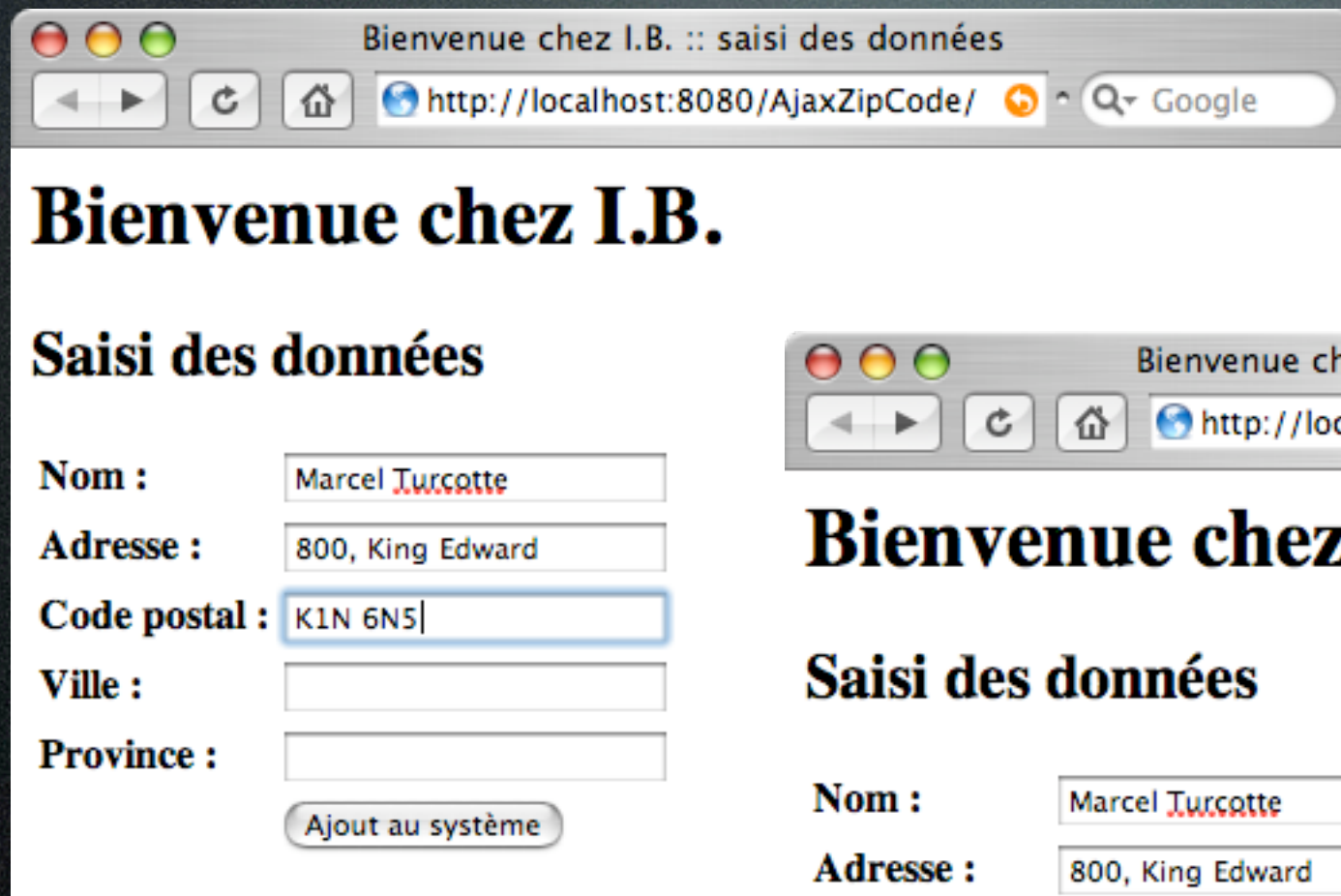
Ajax : côté serveur

```
public void doPost( HttpServletRequest requete, HttpServletResponse reponse )  
    throws ServletException, IOException {  
  
    reponse.setContentType( "application/xml; charset=\"UTF-8\"" );  
  
    PrintWriter doc = reponse.getWriter();  
  
    doc.println( "<?xml version='1.0' encoding='UTF-8'?>" );  
    doc.println( "<info>" + count + "</info>" );  
    doc.close();  
  
}
```

Réponse en XML



```
<html ...>
<head>
  <title>Ajax : interactions asynchrones client-serveur</title>
  <script type='text/javascript' src='AjaxClient.js'>
  </script>
  <meta http-equiv='Content-Script-Type' content='text/javascript' />
</head>
<body style="font-size:x-large">
  <p>
    <button onclick='getDynamicContent();'>Quelle heure est-il?</button> <p />
    <b id='dynamic'>vendredi 28 mars 2008 14 h 11 EDT</b>
  </p>
</body>
</html>
```


```

<html ...>...
<body>
  <h1>Corporate CRM System</h1>
  <h2>Enter Customer Data</h2>
  <table>
    <tr>
      <th>Customer Name:</th>
      <td><input type="text" name="name"/></td>
    </tr>
    <tr>
      ...
    <tr>
      <th>Zip:</th>
      <td><input onblur="getZipData( this.value )"
        type="text" name="zip"/></td>
    </tr>
    <tr>
      <th>City:</th>
      <td><input id="city" type="text" name="city"/></td>
    </tr>
    <tr>
      <th>State:</th>
      <td><input id="state" type="text" name="state"/></td>
    </tr>
    <tr>
      ...
    </tr>
  </table>
</body>
</html>

```



```
<script type="text/javascript">
  var xhr;

  function getZipData( zipCode ) {
    xhr = new XMLHttpRequest();
    xhr.onreadystatechange = processZipData;
    xhr.open( "GET", "/AjaxZipCode/getInfo?code=" + zipCode );
    xhr.send( null );
  }

  function processZipData() {

    if ( xhr.readyState == 4 ) {
      var elem;
      var info = xhr.responseXML.documentElement;

      var node = xhr.responseXML.evaluate( '/info/ville', info, null, XPathResult.FIRST_ORDERED_NODE_TYPE, null );
      elem = document.getElementById( "ville" );
      elem.value = node.singleNodeValue.firstChild.data;

      node = xhr.responseXML.evaluate( '/info/province', info, null, XPathResult.FIRST_ORDERED_NODE_TYPE, null );
      elem = document.getElementById( "province" );
      elem.value = node.singleNodeValue.firstChild.data;

    }
  }
</script>
```



```

public class GetInfo extends HttpServlet {

    public void doGet( HttpServletRequest requete, HttpServletResponse reponse )
        throws ServletException, IOException {

        // initialisation

        String ville, province, code = requete.getParameter( "code" );

        if ( code.startsWith( "J8" ) || code.startsWith( "J9" ) ) {
            ville = "Gatineau";
            province = "Quebec";
        } else if ( code.startsWith( "K1" ) ) {
            ville = "Ottawa";
            province = "Ontario";
        } else {
            ville = "Entrez une valeur";
            province = "Entrez une valeur";
        }

        reponse.setContentType( "application/xml; charset=\"UTF-8\"" );

        PrintWriter doc = reponse.getWriter();

        doc.println( "<?xml version='1.0' encoding='UTF-8'?>" );
        doc.println( "<info>" );
        doc.println( "  <ville>" + ville + "</ville>" );
        doc.println( "  <province>" + province + "</province>" );
        doc.println( "</info>" );
        doc.close();
    }
}

```


Définition

- **Ajax** est une technique pour la conception d'applications Web par laquelle le **client** obtient des **informations** de façon **asynchrone** (à l'aide de XMLHttpRequest, par exemple) afin de **mettre à jour** le document affiché (à l'aide de JavaScript, du DOM, de CSS...) **sans** avoir à charger un nouveau document (X)HTML.

XMLHttpRequest (XHR)

- **xhr.onreadystatechange**
- Le gestionnaire («callback») sera appelé plusieurs fois
- La propriété **xhr.readyState** indique la progression de la requête
 - 0: uninitialized, 1: loading, 2: loaded, 3: interactive, **4: completed**

Ajax : côté client (engin Ajax)

```
function updateDynamicContent( connection ) {  
    if ( connection.readyState == 4 && connection.status == 200 ) {  
        var elem = document.getElementById( "dynamic" );  
        var info = connection.responseXML.documentElement;  
        elem.childNodes[ 0 ].data = info.childNodes[ 0 ].data;  
    }  
    return;  
}
```


XMLHttpRequest (XHR)

- Attention les requêtes **Get** peuvent être cachées par le navigateur
- Solutions :
 - Utiliser **POST** plutôt que **GET**
 - Le serveur ajoute une en-tête «**Cache-control: no-cache**» à la réponse
 - Constuire des URLs qui varient à chaque requête :
/get?date=20080328142330

Ajax : histoire

- La préhistoire d'Ajax remonte à l'introduction de l'élément **iFrame**

```
<html>  
<head>Page extérieure</head>  
<body>  
<h1>...</h1>  
<iframe src= "page_intérieure.html">  
</iframe>
```

- Un élément **iFrame** de taille 0x0 dont le contenu est rafraîchi à intervalle régulier fournit un mécanisme primitif pour les communications asynchrones

```
...  
</html>
```


Ajax : histoire

- La popularité de **Google Map** et **GMail**, ainsi que la publication de de Jesse James Garrett de 2005 sont considérés comme les éléments déclencheurs
- Jesse James Garrett (18 février 2005) **Ajax: A New Approach to Web Applications**
[<http://www.adaptivepath.com/publications/essays/archives/000385.php>]

Ajax : les avantages

- Des interfaces riches
- Surtout, une réponse rapide, pas d'attentes
 - Des portions de pages Web sont mises à jour (sans télécharger toute la page)
- Universel

Ajax : les avantages

- Des portions de pages Web sont mises à jour (sans télécharger toute la page)
 - Affichage rapide
(seulement les parties qui changent)
 - Transfert rapide des informations
(plus petits documents)
 - Utilise moins de bande passante
(plus petits documents)

Ajax : les avantages

- Applications réactives («responsive»)
 - Puis que les communications sont asynchrones, l'application demeure interactive durant le transfert des informations
 - L'application peut aussi «cacher» les données afin de rendre l'application encore plus réactive

Ajax : les avantages

- Repose sur **JavaScript**, **HTML**, **XML**, **DOM** et **CSS**, qui sont des technologies bien connues des développeurs d'applications Web
- Ces technologies sont implémentées dans tous les navigateurs modernes
- Impose peu de contraintes côté serveur

Ajax : les désavantages

- Doit charger l'engin Ajax au début de la session
- Possiblement plus difficile à debugger
- Peut-être des problèmes de sécurité (injection de code malicieux, ...)

Ajax : désavantages

```
var req;

function loadXMLDoc( url )
{
    // branch for native XMLHttpRequest object
    if ( window.XMLHttpRequest ) {
        req = new XMLHttpRequest();
        req.onreadystatechange = processReqChange;
        req.open( "GET", url, true );
        req.send( null );
    }
    // branch for IE/Windows ActiveX version
    } else if ( window.ActiveXObject ) {
        req = new ActiveXObject( "Microsoft.XMLHTTP" );
        if ( req ) {
            req.onreadystatechange = processReqChange;
            req.open( "GET", url, true );
            req.send();
        }
    }
}
```


Ajax : alternatives

- Traditionnellement, Ajax est **modèle tiré** (le client envoie périodiquement des requêtes au serveur)
- **Comet** : implémente un **modèle poussé**
- Il y aura un API standard avec la version **3.0** des **Servlets**
- **GlassFish** supporte déjà **Comet**, mais rendre l'option disponible et l'API n'est pas encore un standard

Ajaxian Web Frameworks
Rails, Tapestry, WebWork...

UI Toolkit
Dojo, SmartClient, Backbase...

Remoting Toolkit
DWR, JSON-RPC, dojo.io.bind()

XMLHttpRequest

iframe

...

JavaScript
Utilities
& Tools

Adapté de J. Gehrtland, B. Galbraith et D. Almaer (2006) *Pragmatic Ajax: A Web 2.0 Primer*. The Pragmatic Programmers.

Engins Ajax

- **Dojo** : <http://dojotoolkit.com>
- **Prototype** : <http://prototype.conio.net>
- **Script.aculo.us** : <http://script.aculo.us>
- **DRW** : <https://dwr.dev.java.net>
- **Backbase** : <http://backbase.com>

Engins Ajax

- **SmartClient** : <http://www.isomorphic.com>
- **Ajax.NET** : <http://ajax.schwarz-interactive.de>
- **SAJAX** : <http://www.modernmethod.com/sajax>
- **JSON-RPC** : <http://json-rpc.org>
- **JMaki** : <https://ajax.dev.java.net>



Astuces

Fermetures, JavaScript et XSL

Remarque : fermetures

```
function getDynamicContent() {  
    var connection = new XMLHttpRequest();  
    if ( connection ) {  
        connection.open( "POST", "/servlet/AjaxServer", true );  
        connection.onreadystatechange =  
            function () { updateDynamicContent( connection ); };  
        connection.setRequestHeader( "Content-Type",  
            "application/x-www-form-urlencoded" );  
        connection.send( "" );  
    }  
    return;  
}
```


Remarque : fermetures

- Une **fermeture** capture des variables de son environnement lexical, ces dernières ne sont pas des variables globales !

```
function makeFunc() {  
  var name = "bar";  
  function getName() {  
    return name;  
  }  
  return getName();  
}
```

```
js> name = "autre chose";  
js> var myFunc = makeFunc();  
js> myFunc();  
bar
```


Remarque : fermetures

- Une **fermeture** capture des variables de son environnement lexical, ces dernières ne sont pas des variables globales

```
function makeAdder( op ) {  
    return function( value ) {  
        return op + value;  
    };  
}
```

```
js> var add1 = makeAdder( 1 );  
js> var add5 = makeAdder( 5 );  
js> add1( 1 );  
2  
js> add5( 1 );  
6
```


Remarque : fermetures

- Ici, la capture d'une variable (connexion) lors de la création d'un gestionnaire («callback») évite la création de plusieurs variables globales, dans le cas où il y a plusieurs éléments dynamiques
- On évite ainsi des erreurs où un gestionnaire utiliserait l'objet connexion d'un autre gestionnaire !


```
<script type="text/javascript">
  var xhr;

  function getZipData( zipCode ) {
    xhr = new XMLHttpRequest();
    xhr.onreadystatechange = processZipData;
    xhr.open( "GET", "/AjaxZipCode/getInfo?code=" + zipCode );
    xhr.send( null );
  }

  function processZipData() {

    if ( xhr.readyState == 4 ) {
      var elem;
      var info = xhr.responseXML.documentElement;

      var node = xhr.responseXML.evaluate( '/info/ville', info, null, XPathResult.FIRST_ORDERED_NODE_TYPE, null );
      elem = document.getElementById( "ville" );
      elem.value = node.singleNodeValue.firstChild.data;

      node = xhr.responseXML.evaluate( '/info/province', info, null, XPathResult.FIRST_ORDERED_NODE_TYPE, null );
      elem = document.getElementById( "province" );
      elem.value = node.singleNodeValue.firstChild.data;

    }
  }
</script>
```


Remarque : fermetures

```
function getDynamicContent() {  
  
    var xhr = new XMLHttpRequest();  
  
    if ( xhr ) {  
        xhr.open( "POST", "/servlet/AjaxServer", true );  
        xhr.onreadystatechange = function () { updateDynamicContent( xhr ); };  
        xhr.setRequestHeader( "Content-Type", "application/x-www-form-urlencoded" );  
        xhr.send( "" );  
    }  
    return;  
}  
  
function updateDynamicContent( connection ) {  
    if ( connection.readyState == 4 && connection.status == 200 ) {  
        var elem = document.getElementById( "dynamic" );  
        var info = connection.responseXML.documentElement;  
        elem.childNodes[ 0 ].data = info.childNodes[ 0 ].data;  
    }  
    return;  
}
```


JavaScript et XSL

- Voici un exemple où une **transformation XSL** est exécutée à partir d'un programme **JavaScript**
 - **index.html** (JavaScript)
 - **MusicLibrary.xml**
 - **TracksAndAlbum.xsl**

MusicLibrary.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<catalog>  
  <album>  
    <name>OK Computer</name>  
    <track>  
      <name>Karma Police</name>  
      <artist>Radiohead</artist>  
      <genre>Alternative</genre>  
    </track>  
    <track>  
      <name>Exit Music (For A Film)</name>  
      <artist>Radiohead</artist>  
      <genre>Alternative</genre>  
    </track>  
  </album>  
  <album>  
    <name>Dark Side Of The Moon</name>  
    <track>  
      <name>Us And Them</name>  
      <artist>Pink Floyd</artist>  
      <genre>Progressive</genre>  
    </track>  
    <track>  
      <name>Brain Damage</name>  
      <artist>Pink Floyd</artist>  
      <genre>Progressive</genre>  
    </track>  
  </album>
```

...

...

```
<album>  
  <name>Requiem</name>  
  <track>  
    <name>Kyrie</name>  
    <artist>Wiener Philharmoniker - Herbert von Karajan</artist>  
    <composer>Wolfgang Amadeus Mozart</composer>  
    <genre>Classical</genre>  
  </track>  
</album>  
<album>  
  <name>African Guitar Summit</name>  
  <track>  
    <name>Obaa Y Ewa</name>  
    <artist>Pa Joe</artist>  
    <genre>World</genre>  
  </track>  
  <track>  
    <name>Mwajuma</name>  
    <artist>Adam Solomon</artist>  
    <genre>World</genre>  
  </track>  
</album>  
</catalog>
```


TracksAndAlbum.xsl

```
<xsl:template match="/catalog">
  <html>
    <head>
      <title>Liste des pièces</title>
    </head>
    <body>
      <ul>
        <xsl:apply-templates select="album/track/name"/>
      </ul>
    </body>
  </html>
</xsl:template>
```

```
<xsl:template match="track/name">
  <li>
    <xsl:value-of select="."/>
    <xsl:apply-templates select="ancestor::track/ancestor::album/name"/>
  </li>
</xsl:template>
```

```
<xsl:template match="album/name">
  <br/>
  ( <em> <xsl:value-of select="."/> </em> )
</xsl:template>
```


index.html

```
<html>
<head>
  <script>
    function loadXMLDoc( fname ) {
      var doc;
      doc = document.implementation.createDocument( "", null, null );
      doc.async=false;
      doc.load( fname );
      return( doc );
    }
    function displayResult() {
      xml=loadXMLDoc( "MusicLibrary.xml" );
      xsl=loadXMLDoc( "TracksAndAlbum.xsl" );

      xsltProcessor = new XSLTProcessor();
      xsltProcessor.importStylesheet( xsl );

      resultDocument = xsltProcessor.transformToFragment( xml, document );

      document.getElementById( "example" ).appendChild( resultDocument );
    }
  </script>
</head>
<body id="example" onLoad="displayResult()">
</body>
</html>
```


JavaScript et XSL



Applications

- Serveur transmet les données à l'aide d'un document XML (liste de villes)
- Le client applique une transformation afin de créer des éléments XHTML (les items d'un menu)
- ...

XMLHttpRequestExample

developer.apple.com/internet/webcontent/XMLHttpRequestExample/example.html

XMLHttpRequest Object Demo

Category:

Items:

XMLHttpRequest Object Demo

Category:

Top 10 Just Added

Items:

A Snow Capped Romance - 36 Crazyfists
Conspiracy - 3kStatic
50 Foot Wave - EP - 50 Foot Wave
De La Isla 'el Caiman - 90 Millas
Mer de Noms - A Perfect Circle
Jazz Workshop, Vol. 3 - Ada Moore
Dreams of Water Themes - Adventure Time
Honkin' On Bobo - Aerosmith
Road Runner - Single - Aerosmith
Music Is a Virus - Air Liquide

XMLHttpRequest Object Demo

Category:

Top 10 Just Added

Items:

- A Snow Capped Romance - 36 Crazyfists
- Conspiracy - 3kStatic**
- 50 Foot Wave - EP - 50 Foot Wave
- De La Isla 'el Caiman - 90 Millas
- Mer de Noms - A Perfect Circle
- Jazz Workshop, Vol. 3 - Ada Moore
- Dreams of Water Themes - Adventure Time
- Honkin' On Bobo - Aerosmith
- Road Runner - Single - Aerosmith
- Music Is a Virus - Air Liquide



[Conspiracy](#)
[3kStatic](#)

Release Date: May 23, 2003
Total Songs: 13
Genre: [Electronic](#)
Price: \$9.99
Copyright 2003 INgrooves


```
<html ...>
<head>...</head>
<body>
<h1>XMLHttpRequest Object Demo</h1>
<hr />

<form>
<p>Category:<br />
<select onchange="loadDoc(event)">
  <option value="">Choose One</option>
  <option value="songs.xml">Top 10 Songs</option>
  <option value="albums.xml">Top 10 Albums</option>
  <option value="newreleases.xml">Top 10 New Releases</option>
  <option value="justadded.xml">Top 10 Just Added</option>
</select>
</p>
<p>Items:<br />
<select size="10" id="topics" onchange="showDetail(event)">
  <option value="">Choose a Category First</option>
</select>
</p>
</form>
<div id="details"><span></span></div>
</body>
</html>
```



```

// invoked by "Category" select element change;
// loads chosen XML document, clears Topics select
// element, loads new items into Topics select element
function loadDoc( evt ) {
    // equalize W3C/IE event models to get event object
    evt = (evt) ? evt : ((window.event) ? window.event : null);
    if ( evt ) {
        // equalize W3C/IE models to get event target reference
        var elem = (evt.target) ? evt.target : ((evt.srcElement) ? evt.srcElement : null);
        if ( elem ) {
            try {
                if ( elem.selectedIndex > 0 ) {
                    loadXMLDoc( elem.options[ elem.selectedIndex ].value );
                }
            }
            catch( e ) {
                var msg = (typeof e == "string") ? e : ((e.message) ? e.message : "Unknown Error");
                alert("Unable to get XML data:\n" + msg);
                return;
            }
        }
    }
}

```



```

// global request and XML document objects
var req;

// retrieve XML document (reusable generic function);
// parameter is URL string (relative or complete) to
// an .xml file whose Content-Type is a valid XML
// type, such as text/xml; XML source must be from
// same domain as HTML file

function loadXMLDoc( url ) {
    // branch for native XMLHttpRequest object
    if ( window.XMLHttpRequest ) {
        req = new XMLHttpRequest();
        req.onreadystatechange = processReqChange;
        req.open( "GET", url, true );
        req.send( null );
    }
    // branch for IE/Windows ActiveX version
    } else if ( window.ActiveXObject ) {
        isIE = true;
        req = new ActiveXObject("Microsoft.XMLHTTP");
        if (req) {
            req.onreadystatechange = processReqChange;
            req.open("GET", url, true);
            req.send();
        }
    }
}

```



```
// handle onreadystatechange event of req object
```

```
function processReqChange() {
```

```
    // only if req shows "loaded"
```

```
    if ( req.readyState == 4 ) {
```

```
        // only if "OK"
```

```
        if ( req.status == 200 ) {
```

```
            clearTopicList();
```

```
            buildTopicList();
```

```
        } else {
```

```
            alert("There was a problem retrieving the XML data:\n" +  
                req.statusText);
```

```
        }
```

```
    }
```

```
}
```



```
// empty Topics select list content
```

```
function clearTopicList() {  
    var select = document.getElementById( "topics" );  
    while ( select.length > 0 ) {  
        select.remove( 0 );  
    }  
}
```

```
// add item to select element the less  
// elegant, but compatible way.
```

```
function appendToSelect( select, value, content ) {  
    var opt;  
    opt = document.createElement( "option" );  
    opt.value = value;  
    opt.appendChild( content );  
    select.appendChild( opt );  
}
```



```
// fill Topics select list with items from  
// the current XML document
```

```
function buildTopicList() {
```

```
    var select = document.getElementById( "topics" );  
    var items = req.responseXML.getElementsByTagName( "item" );
```

```
    // loop through <item> elements, and add each nested  
    // <title> element to Topics select element
```

```
    for ( var i = 0; i < items.length; i++ ) {  
        appendToSelect( select, i,  
            document.createTextNode( getElementTextNS( "", "title", items[ i ], 0 ) ) );  
    }
```

```
    // clear detail display  
    document.getElementById( "details" ).innerHTML = "";  
}
```



```

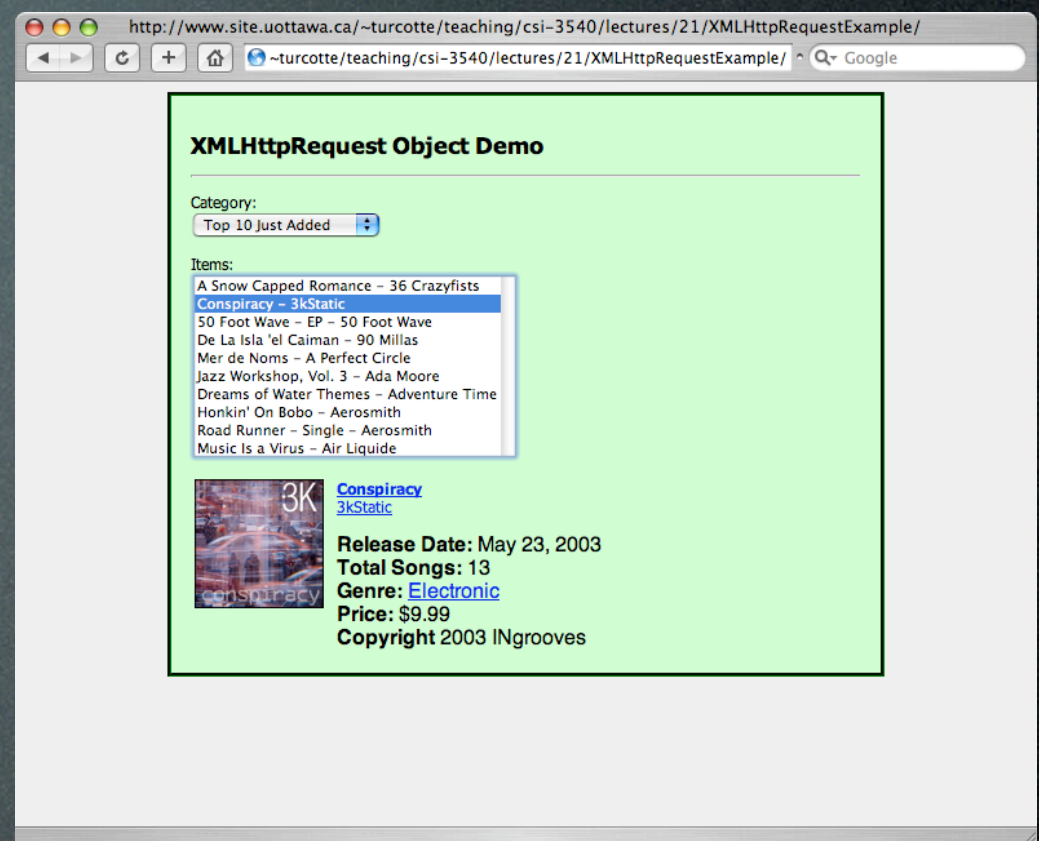
<html ...>
<head>...</head>
<body>
<h1>XMLHttpRequest Object Demo</h1>
<hr />

```

```

<form>
<p>Category:<br />
<select onchange="loadDoc(event)">
  <option value="">Choose One</option>
  <option value="songs.xml">Top 10 Songs</option>
  <option value="albums.xml">Top 10 Albums</option>
  <option value="newreleases.xml">Top 10 New Releases</option>
  <option value="justadded.xml">Top 10 Just Added</option>
</select>
</p>
<p>Items:<br />
<select size="10" id="topics" onchange="showDetail(event)">
  <option value="">Choose a Category First</option>
</select>
</p>
</form>
<div id="details"><span></span></div>
</body>
</html>

```




```
// display details retrieved from XML document
```

```
function showDetail( evt ) {  
    evt = (evt) ? evt : ((window.event) ? window.event : null);  
    var item, content, div;  
    if ( evt ) {  
        var select = (evt.target) ? evt.target : ((evt.srcElement) ? evt.srcElement : null);  
        if ( select && select.options.length > 1 ) {  
            // copy <content:encoded> element text for  
            // the selected item  
            item = req.responseXML.getElementsByTagName( "item" )[ select.value ];  
            content = getElementTextNS( "content", "encoded", item, 0 );  
            div = document.getElementById( "details" );  
            div.innerHTML = "";  
            // blast new HTML content into "details" <div>  
            div.innerHTML = content;  
        }  
    }  
}
```


Documentation chez Google

- <http://code.google.com/edu/client/ajax-tutorial.html>
- <http://code.google.com/edu/client/intro-to-js.html>
- <http://code.google.com/edu/client/ajax-search-api.html>
- <http://code.google.com/edu/videlectures.html>

Ajax : les alternatives

- Adobe Flash/Macromedia
- Microsoft Silverlight
- Sun microsystems JavaFX

Ressources (suite)

- OpenAjax Alliance [<http://www.openajax.org/>] 2007
- SAX [<http://www.saxproject.org>] 2007
- The XMLHttpRequest Object [<http://www.w3.org/TR/XMLHttpRequest>]
2008-03-28