

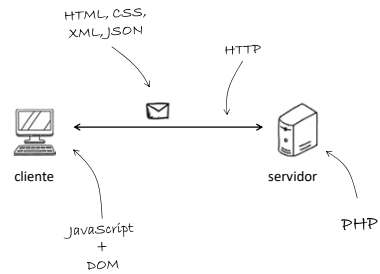
## Ingeniería de Aplicaciones Web

Diego C. Martínez

Departamento de Ciencias e Ingeniería de la Computación  
Universidad Nacional del Sur

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Tecnologías web



Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### AJAX



modelo de interacción para aplicaciones web,  
basado primordialmente en dos tecnologías:  
JavaScript y XML

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

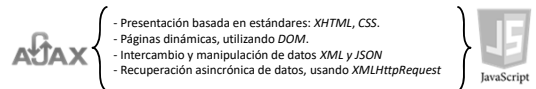
### AJAX Asynchronous JavaScript and XML

Es básicamente una técnica para implementar ciertos comportamientos en aplicaciones web.

La idea general es solicitar en background información a un servidor mientras el cliente opera sobre el navegador (por eso asincrónica).

Antes "inner-Browsing"

No es una tecnología, sino una agrupación de varias técnicas y tecnologías unidas para lograr un mismo efecto.



Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### El corazón de AJAX

Chrome, Explorer, Edge, Opera, Safari, Firefox, Konqueror y otros proveen todos un método para que los scripts JavaScript puedan realizar requerimientos HTTP.

Se realiza por medio de un objeto denominado XMLHttpRequest ("XHR")  
Aunque XML aparece en el nombre, no se limita sólo a ese formato.

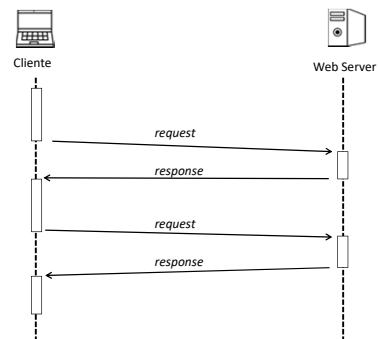
```
var req = new XMLHttpRequest()
var req = new XMLHttpRequest()
var req = window.createRequest()
```

Una vez obtenido el objeto, realizar un pedido HTTP es fácil.

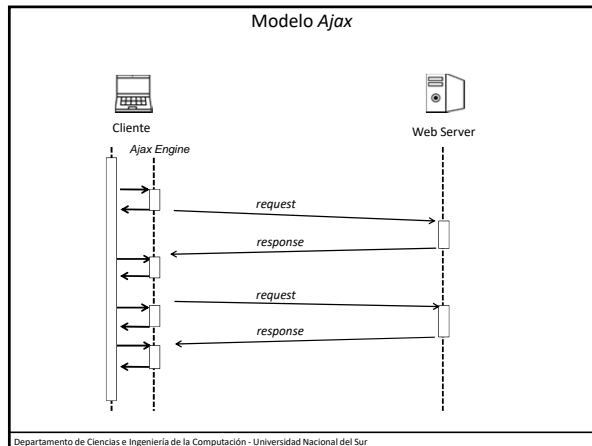
Hay que tener en cuenta que el pedido se realiza en background, en forma asincrónica, por lo que no "esperamos" por una respuesta....

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Modelo tradicional



Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur



### XMLHttpRequest

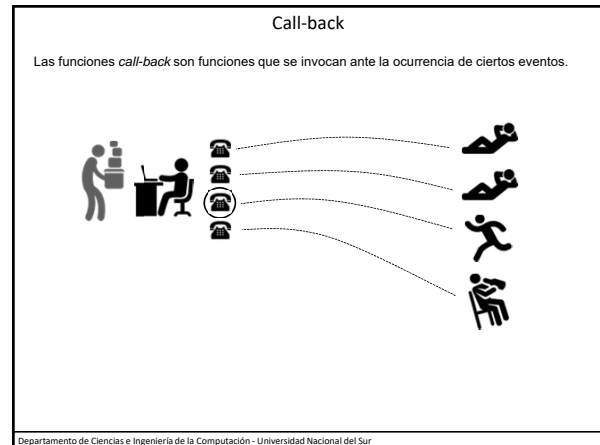
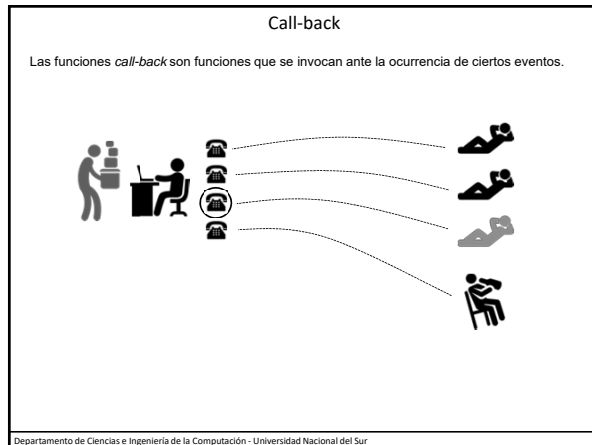
```

if (window.XMLHttpRequest){
    // IE7+, Firefox, Chrome, Opera, Safari, el mundo civilizado
    xmlhttp=new XMLHttpRequest();
} else {
    // IE6, IE5
    xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}

xmlhttp.open(metodo,URL,asincronico);
xmlhttp.send();
    
```

Si el request es asíncronico, hay que indicar previamente qué sucederá cuando se complete la recepción de la respuesta del servidor...

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur



### XMLHttpRequest en acción

```

req.onreadystatechange = processReqChange;
req.open("GET", url, true);
req.send();
    
```

**onreadystatechange** es una propiedad del objeto *XMLHttpRequest* que almacena la función que procesa la respuesta del servidor.

Es el oyente de cambio de estado: la función se invoca cada vez que cambia la propiedad **readyState**.

- después de invocar a **open**, **readyState** = 1.
- después de invocar a **send**, **readyState** = 2.
- mientras se recibe la respuesta, **readyState** = 3
- al finalizar la respuesta, **readyState** = 4.

¡La respuesta está lista para procesar!

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### XMLHttpRequest en acción

```

req.onreadystatechange = processReqChange;
req.open("GET", url, true);
req.send();
    
```

La respuesta del servidor se almacena finalmente en un atributo del objeto *XMLHttpRequest*.

Si la respuesta esperada es *texto plano*, se almacena en **responseText**.

```

function processReqChange(){
    if(req.readyState == 4){
        document.getElementById(id).innerHTML = req.responseText;
    }
}
    
```

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

## XMLHttpRequest en acción

```
req.onreadystatechange = processReqChange;
req.open("GET", url, true);
req.send();
```

El atributo **responseXML** es similar, pero nos permite encapsular la información para ser interpretada como XML.

*Al ser un XML, tiene una representación DOM y podemos usar operaciones para obtener los datos contenidos en ese documento!*

```
function processReqChange(){
  if(req.readyState == 4){
    var xmlDoc = req.responseXML;
    var nodo = xmlDoc.getElementsByTagName('root').item(0);
  }
}
```

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

## XMLHttpRequest en acción

```
req.onreadystatechange = processReqChange;
req.open("GET", url, true);
req.send();
```

Las operaciones **open** y **send** inician el proceso de pedido al servidor.

recibe como parámetros:

el método (GET, POST, HEAD, etc)

+

el URL (que puede ser absoluto o relativo) y

+

un indicativo de si el pedido es asíncrono o no.  
El valor "true" como último argumento indica que el script no esperará la respuesta luego de la invocación a la operación **send**, la cual comienza el pedido antes configurado

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

## Ejemplo Ajax

Con lo visto hasta el momento es fácil crear una simple aplicación que utilice esta técnica.



*Del lado cliente*, haremos pedidos HTTP a un servidor por medio del objeto **XMLHttpRequest**. Recibiremos la información y la procesamos.



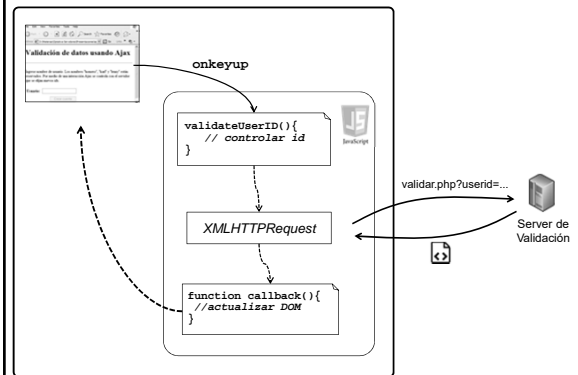
*Del lado servidor*, una aplicación (por ejemplo Java) o script (por ejemplo PHP) generará la respuesta solicitada por el cliente.

Los dos extremos colaboran creando el efecto deseado.

Es importante destacar que las tecnologías usadas tanto del lado cliente como del lado servidor son independientes y por lo tanto pueden cambiar.

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

## Ejemplo Ajax – validación de datos



Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

## JSON

La tendencia es utilizar JSON como estructura de datos, en lugar de XML:

<pre>&lt;albums&gt;   &lt;album&gt;     &lt;titulo&gt;       Master of Puppets     &lt;/titulo&gt;     &lt;artista&gt;       Metallica     &lt;/artista&gt;   &lt;/album&gt;   &lt;album&gt;     &lt;titulo&gt;       The Division Bell     &lt;/titulo&gt;     &lt;artista&gt;       Pink Floyd     &lt;/artista&gt;   &lt;/album&gt; &lt;/albums&gt;</pre> <p>171 bytes</p>	<pre>{   'album':   [     {       'titulo': 'Master of Puppets',       'artista': 'Metallica',     },     {       'titulo': 'The Division Bell',       'artista': 'Pink Floyd',     }   ] }</pre> <p>120 bytes</p>
---	--

La principal ventaja es que al ser un formato de datos propio de JavaScript, es fácil interpretarlo. No hace falta parsing como en XML.

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

## Algunos usos de Ajax

Validaciones de datos de formularios en tiempo real.  
Identificaciones de usuario, números de serie, códigos postales, y otros datos pueden ser validados en el formulario antes del envío de los datos.

Autocompletamiento  
Algunos datos pueden ser autocompletados mientras el usuario escribe.

Controles de interfaz de usuario  
Arboles, menús, barras de progreso se implementan sin necesidad de refresh de página completa.

Refresh de datos en la página.  
Algunos datos son obtenidos del servidor y mostrados en la página.

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Algunas desventajas de Ajax

#### Complejidad

Se necesita lógica de presentación en el cliente y generación y envío de documentos XML en el servidor. Requiere conocimientos de JavaScript.

#### Depuración

El proceso es complicado porque la lógica se distribuye en el cliente y en el servidor.

#### Código expuesto

Parte del código de la aplicación queda expuesto al ejecutarse en el cliente.

#### Indexado

Muchos web crawlers no encontrarán el contenido obtenido via AJAX.

#### Bookmarks

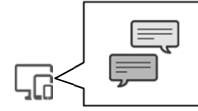
Probablemente los *bookmarks* no registren el último estado de la página

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Chat



¿Como podríamos implementar un chat con esta técnica?

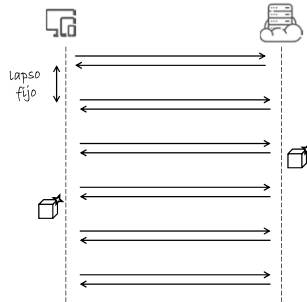


Polling  
Long Polling  
Server-Sent Events

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Polling, Long-Polling, Server-Sent Events

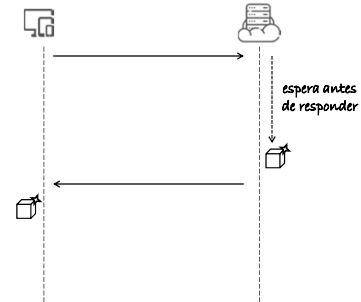
#### Polling



Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Polling, Long-Polling, Server-Sent Events

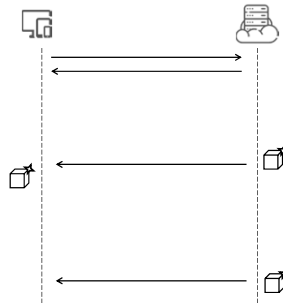
#### Long-Polling



Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Polling, Long-Polling, Server-Sent Events

#### Server-Sent Event



Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Polling, Long-Polling, Server-Sent Events

#### Server-Sent Event

```
var source = new EventSource("server.php");
source.onmessage = function(event) {
    document.getElementById("result").innerHTML += event.data + "<br>";
};

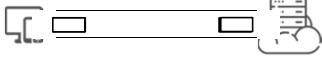
<?php
header('Content-Type: text/event-stream');
header('Cache-Control: no-cache');

$stime = date('r');
echo "data: The server time is: {$stime}\n\n";
flush();
?>
```

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Websockets

Tecnología que permite una comunicación bidireccional entre el navegador y el servidor

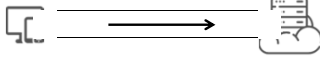


Representa una conexión aparte, con otro protocolo de comunicación

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Websockets

El primer paso es acordar la conexión websocket "handshake"

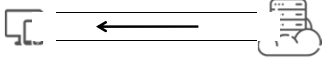


```
GET /chat HTTP/1.1
Host: server.example.com
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: dfhJGg634gDGFH442jdUIK==
Origin: http://unsitio.com
Sec-WebSocket-Protocol: chat, superchat
Sec-WebSocket-Version: 13
```

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Websockets

El primer paso es acordar la conexión websocket "handshake"

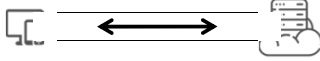


```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: sBiYGzTxaQ93pPLMkzhZRbK+XOa=
Sec-WebSocket-Protocol: chat
```

Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur

### Websockets

A partir del handshake se inicia la conexión full-duplex con websockets



Departamento de Ciencias e Ingeniería de la Computación - Universidad Nacional del Sur