

Desarrollo de Aplicaciones Web

AJAX.

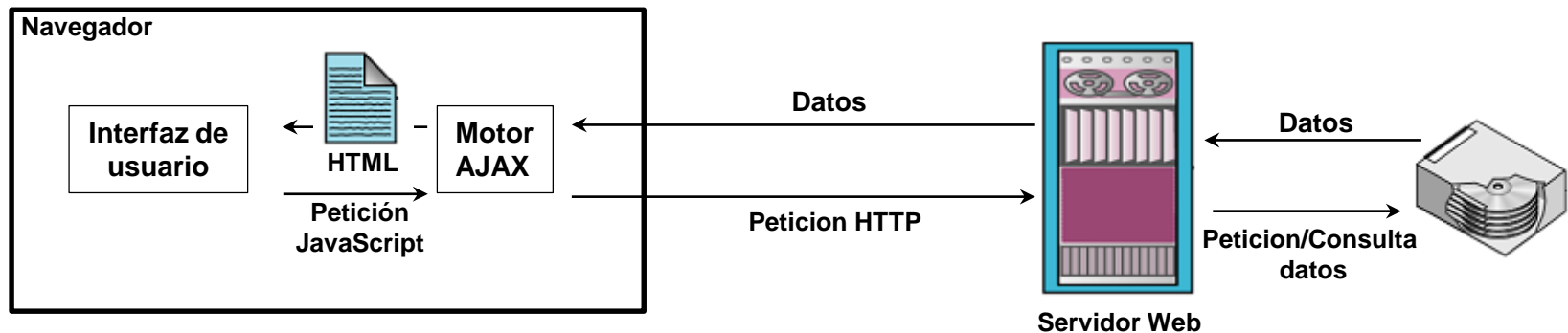
JavaScript vs AJAX

AJAX.- *Asynchronous JavaScript + XML*

Nace en febrero de 2005 de la mano de *Jesse James Garret*, como consecuencia de un artículo titulado: “[*Ajax a new approach to web applications*](#)”

Ajax permite:

- Actualizar una página web, sin recargarla.
- Hacer peticiones de datos a un servidor, después de que la página esté cargada.
- Recibir datos de un servidor, después de que la página esté cargada.
- Enviar datos a un servidor, en *background*.



Motor AJAX.- objeto o función *JavaScript* que es llamada siempre que se realiza una petición de información al servidor.

JavaScript vs AJAX

AJAX.- *Asynchronous JavaScript + XML*

AJAX utiliza una combinación de:

- Un objeto XMLHttpRequest construido dentro de un navegador.
- JavaScript y HTML DOM para mostrar los datos en el navegador.

XMLHttpRequest → objeto utilizado para intercambio de datos con el servidor.

Creación:

```
var= new XMLHttpRequest();
```

JavaScript vs AJAX

AJAX.- *Asynchronous JavaScript + XML*

El objeto XMLHttpRequest.

Métodos:

- abort() → cancela solicitud.
- getAllResponseHeaders() → devuelve información de cabecera
- getResponseHeader() → devuelve información de cabecera específica
- open(method, url, async, user, psw()) → especifica el tipo de solicitud
- send() → envía solicitud al servidor, en modo GET
- send(string) → envía solicitud al servidor en modo POST
- setRequestHeader() → añade un par etiqueta/valor a la cabecera para envío

Atributos:

- onreadystatechange → define una función para ser llamada cuando el readyState cambie
- readyState → indica el estado de XMLHttpRequest
- .responseText → retorna respuesta en formato texto
- responseXML → retorna respuesta en formato XML
- status → retorna el número de estado de la solicitud
- .statusText → retorna el estado de la solicitud

JavaScript vs AJAX

AJAX.- *Asynchronous JavaScript + XML*

El objeto XMLHttpRequest.

```
<body>
  <h1>ETSE</h1>
  <h2 id="demo" onclick="cargaTexto()">Cambia contenido</h2>
</body>
```

```
<h1>AJAX USC</h1>
```

fileTXT.txt

```
<script>
  function cargaTexto() {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
      if (this.readyState == 4 && this.status == 200) {
        var txt= this.responseText;
        document.getElementById("demo").innerHTML= txt;
      }
    };
    xhttp.open("GET", "fileTXT.txt", true);
    xhttp.send();
  }
</script>
```

ETSE

AJAX USC

JavaScript vs AJAX

AJAX.- Asynchronous JavaScript + XML

El objeto XMLHttpRequest.

```
<body>
  <h1>ETSE</h1>
  <h2 id="demo" onclick="cargaTexto()">Cambia contenido</h2>
</body>
```

```
<script>
function cargaTexto(){
  var xhttp= new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
      var xmlDoc= this.responseXML;
      var txt= "<ul><li>";
      var x= xmlDoc.getElementsByTagName("TITULO");
      for (var i = 0; i < x.length; i++)
        txt= txt + x[i].childNodes[0].nodeValue + "</li><li>";
      document.getElementById("demo").innerHTML= txt;
    }
  };
  xhttp.open("GET", "fileXML.xml", true);
  xhttp.send();
}
</script>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<UNIVERSIDAD>
- <CORUÑA>
  <AÑO>1985</AÑO>
  <RECTOR>Enrique Pérez</RECTOR>
  - <TITULACIONES>
    <TITULO>Derecho</TITULO>
    <TITULO>Fisioterapia</TITULO>
    <TITULO>Ingeniería Informática</TITULO>
  </TITULACIONES>
</CORUÑA>
- <SANTIAGO>
  <AÑO>1500</AÑO>
  <RECTOR>Enrique Pérez</RECTOR>
  - <TITULACIONES>
    <TITULO>Farmacia</TITULO>
    <TITULO>Historia</TITULO>
    <TITULO>Ingeniería Informática</TITULO>
    <TITULO>Medicina</TITULO>
  </TITULACIONES>
</SANTIAGO>
- <VIGO>
  <AÑO>1985</AÑO>
  <RECTOR>Enrique Pérez</RECTOR>
  - <TITULACIONES>
    <TITULO>Ingeniería Informática</TITULO>
    <TITULO>Ingeniería de Telecomunicaciones</TITULO>
    <TITULO>Traducción e Interpretación</TITULO>
  </TITULACIONES>
</VIGO>
</UNIVERSIDAD>
```

fileXML.xml

ETSE

- Derecho
- Fisioterapia
- Ingeniería Informática
- Farmacia
- Historia
- Ingeniería Informática
- Medicina
- Ingeniería Informática
- Ingeniería de Telecomunicaciones
- Traducción e Interpretación

JavaScript vs AJAX

AJAX.- *Asynchronous JavaScript + XML*

El objeto XMLHttpRequest.

```
<body>
  <h1>ETSE</h1>
  <h2 id="demo" onclick="cargaTexto()">Cambia contenido</h2>
</body>
```

```
<script>
  function cargaTexto() {
    var xhttp= new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
      if (this.readyState == 4 && this.status == 200) {
        var jsonObj= JSON.parse(this.responseText);
        var txt= "<ul>";
        for (x in jsonObj.Registros)
          txt= txt + "<li>" + jsonObj.Registros[x].Nombre
            + " -> " + jsonObj.Registros[x].Ciudad
            + "</li>";
        txt= txt + "</ul>";
        document.getElementById("demo").innerHTML= txt;
      }
    };
    xhttp.open("GET", "fileJSON.json", true);
    xhttp.send();
  }
</script>
```

```
{
  "Registros":[
    {"Nombre":"Pedro",
      "Ciudad":"Santiago",
      "Pais":"España"},
    {"Nombre":"Pablo",
      "Ciudad":"León",
      "Pais":"España"}
  ]
}
```

fileJSON.json

ETSE

- **Pedro -> Santiago**
- **Pablo -> León**