

Escuela Politécnica de Ingeniería  
Grado de Ingeniería Informática en Tecnologías de la  
Información

# Tecnologías Web

Tema 2

Tecnologías web de cliente: DOM y JQuery



# Índice

- DOM
- JQuery

# Document Object Model (DOM)

# Introducción (1)

- DOM
  - API orientado a objetos que permite que los programas y *scripts* interactúen dinámicamente con el documento HTML
    - Acceso o modificación de la estructura o del contenido
    - Acceso o modificación a estilos CSS
    - Gestión de eventos
  - Niveles (versiones) del DOM (<http://www.w3.org/DOM/DOMTR>)
    - 0: impuesto por los navegadores (finales de los 90)
    - 1, 2, 3: estándar del W3C (actualmente está en desarrollo el nivel 4).

# Introducción (2)

- Módulos del DOM
  - A partir del nivel 1 la API se subdivide en módulos, por ejemplo:
    - **Núcleo** (core): común a documentos HTML y XML.
    - **HTML**: objetos, propiedades y métodos para documentos HTML.
    - **Traversal and Range**: permite recorrer y manejar fragmentos de documentos.
    - **Events**: objetos, propiedades y métodos de generación de eventos.
    - **Style**: objetos, propiedades y métodos para interactuar con las hojas de estilo (CSS).

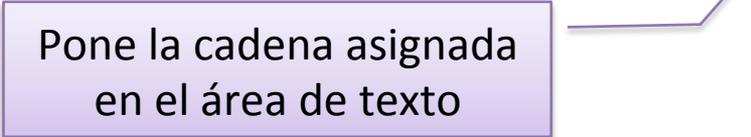
# DOM HTML (1)

- DOM HTML
  - Facilita el trabajo con documentos HTML
    - A mayor nivel pierde importancia en favor del **núcleo**, que ofrece una API común a los documentos HTML y XML.
  - Hay varios ejemplos disponibles en el tutorial
    - [http://www-aii.dyndns.org/aii/ejemplos.php?id\\_codigo=8&codigo=DOM+HTML](http://www-aii.dyndns.org/aii/ejemplos.php?id_codigo=8&codigo=DOM+HTML)
    - De éstos sólo tienen interés los que hacen referencia al navegador: objeto **Window** y objeto **Navigator** (para el resto se utilizará el API común con los documentos XML). En ocasiones la API correspondientes a éstos dos objetos se diferencia del DOM HTML (objetos del documento HTML), la API denominada *Browser Object Model* (BOM).

# DOM HTML (2)

- Propiedades de objetos del DOM HTML
  - Entre las propiedades de estos objetos están disponibles los atributos del elemento HTML correspondiente.
    - Esto nos permite especificar, modificar u obtener valores de los atributos de los elementos HTML.
    - Ejemplo: `<input type="text" name="campo">`. Si `obj` es el objeto correspondiente al elemento `input`:
      - `obj.type;` → retorna la cadena "text"
      - `obj.value = "Valor del campo";` → pone el atributo value

Pone la cadena asignada en el área de texto



# Núcleo del DOM (1)

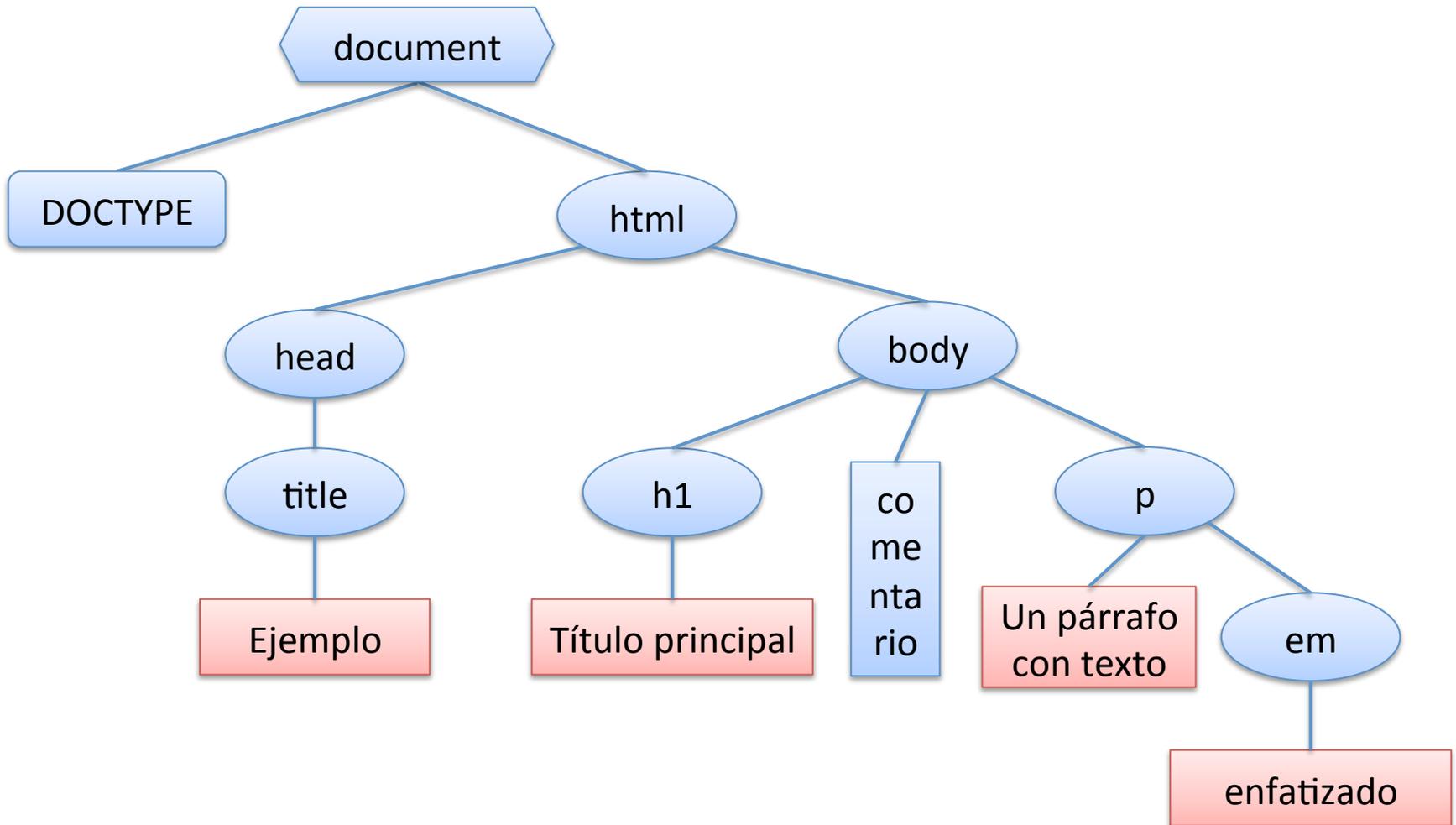
- En el DOM un documento (HTML o XML) se representa como el árbol correspondiente a su estructura. Los nodos del árbol son las distintas partes que componen su estructura (DOCTYPE, elementos, comentarios, etc.) y su contenido (texto).
  - El estándar exige nodos de texto adicionales (*whitespace nodes*) para representar los espacios en blanco (espacios, tabuladores, retornos de carro, etc.) entre etiquetas del documento.

# Núcleo del DOM (2)

## Ejemplo

```
<!DOCTYPE html>
<html>
<head>
  <title>Ejemplo</title>
</head>
<body>
  <h1>Título principal</h1>
  <!-- ejemplo de comentario -->
  <p>Un párrafo con texto
  <em>enfaticado</em></p>
</body>
</html>
```

# Núcleo del DOM (3)



# Núcleo del DOM (4)

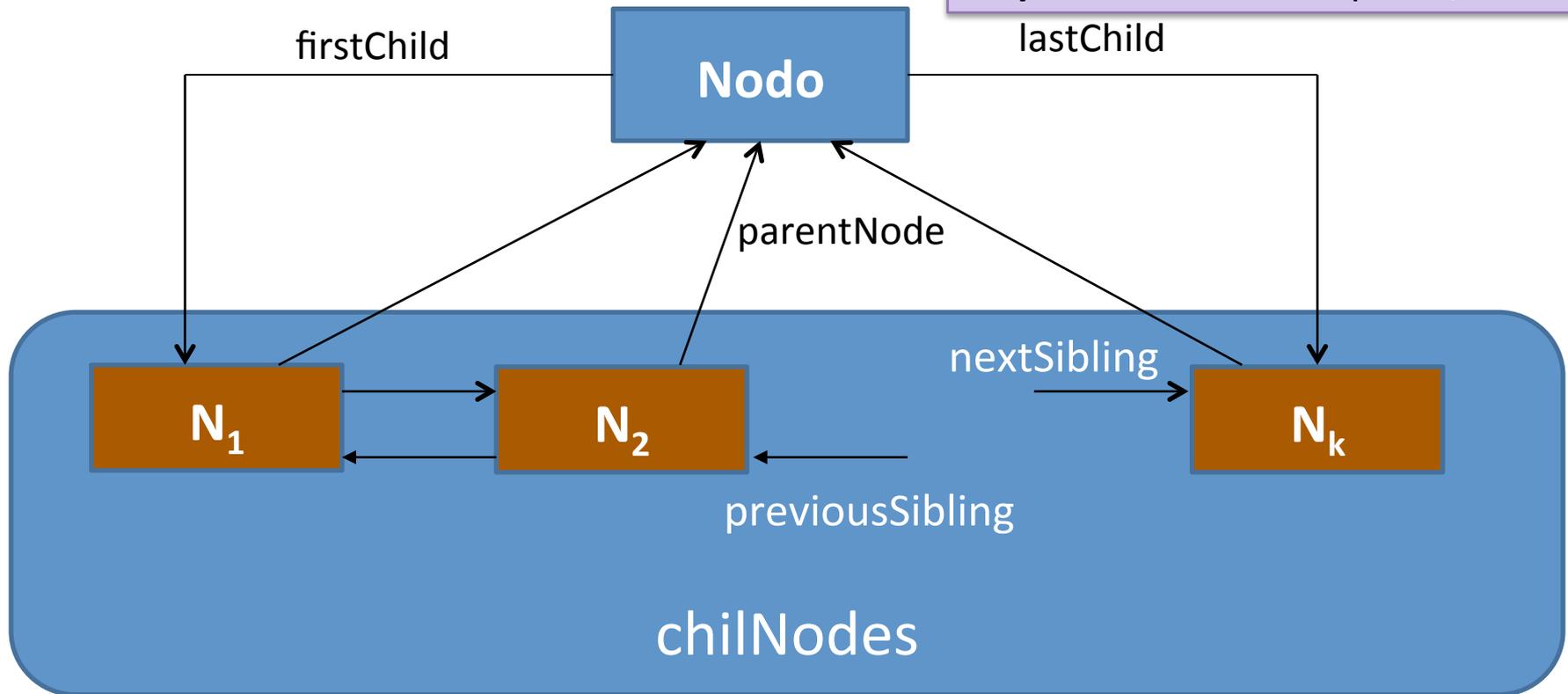
- La parte del API correspondiente al núcleo permite acceder y actualizar la estructura y contenido de los documentos (HTML y XML).
  - Para manipular los documentos es necesario manipular los nodos del árbol, siendo éste (**Node**) el objeto primario del DOM.
    - Los nodos pueden ser de varios tipos: *Document*, *DocumentType*, *Element*, *Comment*, *Attr*, *Text*, etc.
      - Los atributos de las etiquetas son nodos de tipo *Attr* pero no están en el árbol. Para acceder a estos hay que utilizar los métodos de los nodos que los contienen.
  - Ejemplos, en el tutorial:  
[http://www-aii.dyndns.org/aii/ejemplos.php?id\\_codigo=4&codigo=DOM#DOM59](http://www-aii.dyndns.org/aii/ejemplos.php?id_codigo=4&codigo=DOM#DOM59)

# Núcleo del DOM (5)

- Estructura de los nodos

La estructura se forma mediante propiedades del objeto **Node**:

- **firstChild**: primer hijo
- **childNodes**: lista de nodos hijos
- **parentNode**: nodo padre, etc.



# Núcleo del DOM (6)

- Algunas propiedades de los nodos
  - Además de las propiedades que confieren al conjunto de éstos de una estructura jerárquica (árbol) y que se indicaron en la transparencia previa, tienen entre otras las propiedades:
    - **nodeType**: entero que representa el tipo de nodo.
    - **nodeName**: nombre del nodo.
    - **nodeValue**: valor del nodo.
  - Los valores de estas tres propiedades dependen del tipo de nodo.

# Núcleo del DOM (7)

Tipo de nodo	nodeType	nodeName	nodeValue
Etiqueta	1 (Node: ELEMENT_NODE)	Nombre de la etiqueta en mayúsculas	null
Texto	3 (Node: TEXT_NODE)	#Text	Texto del nodo
Comentario	8 (Node: COMMENT_NODE)	#Comment	Texto del comentario
DOCTYPE	10 (Node: DOCUMENT_TYPE_NODE)	Nombre de la etiqueta raíz del DOCTYPE	null
Documento	9 (Node: DOCUMENT_NODE)	#Document	null

# Núcleo del DOM (8)

- Acceso a los nodos del documento
  - Para acceder a los nodos de un documento habitualmente se utilizan los métodos del objeto **document**:
    - **document.getElementById(*identificador\_str*)**: en documentos HTML el nodo elemento cuyo valor del atributo **id** es el especificado. En documentos XML el nodo elemento que contiene un atributo de tipo ID con el valor especificado.
    - **document.getElementsByTagName(*etiqueta\_str*)**: lista de nodos (objeto **NodeList**) de tipo elemento del documento cuya etiqueta es la especificada. Para acceder a cada nodo se pueden utilizar las propiedades y métodos de **NodeList** o, en JavaScript, tratar éste como un *array* JavaScript.

# Núcleo del DOM (9)

- Una vez que se accede a la referencia de un nodo o nodos específicos es posible acceder a otros mediante las relaciones de parentesco en la jerarquía o mediante métodos del DOM.
- El método **getElementsByTagName** también se puede utilizar con nodos de tipo elemento, quedando la lista de nodos obtenida a la búsqueda en el subárbol de raíz el nodo elemento (elementos que contiene con la etiqueta especificada).
- Ejemplos en el tutorial:

[http://www-aii.dyndns.org/aii/ejemplos.php?id\\_codigo=4&codigo=DOM](http://www-aii.dyndns.org/aii/ejemplos.php?id_codigo=4&codigo=DOM)

# Núcleo del DOM (10)

- Modificación de la estructura
  - Métodos de **Node** para insertar nodos
    - **appendChild(node)**: añade un nodo hijo al final de childNodes.
    - **insertBefore(node, current)**: inserta un nodo hijo en childNodes antes del nodo current.
    - **setAttribute(attr, value)**: crea o modifica el atributo attr del nodo con el valor dado.
  - Métodos de **Node** para eliminar nodos
    - **removeChild(node)**
    - **replaceChild(node, current)**: reemplaza el hijo current por el nodo especificado.

# Núcleo del DOM (11)

## – Métodos de **Node** para crear nodos

- **document.createElement(tag\_str)**: crea un nodo de tipo elemento de etiqueta especificada.
- **document.createTextNode(texto)**: crea un nodo de texto con el contenido especificado.
  - En ambos casos, sólo se producen cambios en el documento una vez que los nodos se insertan de forma adecuada en el árbol.



# jQuery (1)

- jQuery
  - Librería JavaScript que simplifica tareas comunes y proporciona una capa de compatibilidad entre navegadores.
  - Facilidades que incorpora
    - Manejo del DOM
    - Eventos
    - AJAX
    - *Widgets* y efectos visuales
  - Es extensible mediante *plugins*.

# jQuery (2)

- Incluir jQuery

- La librería consta de un único archivo que debe referenciarse en los documentos HTML.

- `<script type="text/javascript" src="js/jquery-1.8.3.min.js">`

- La función \$

- Es la función que permite el acceso a la mayor parte de las funcionalidades
  - Abreviatura de la función jQuery.

# jQuery (3)

- Estructura típica de una sentencia jQuery

- Selector + comando + parámetros

```
$('#parrafo').css('color', 'blue');
```

Selector

Comando

Parámetros

- La mayor parte de los comandos devuelven un resultado, por lo que se pueden encadenar.

```
$('#parrafo1').addClass('destacado').show('slow')
```

# jQuery (4)

- Selección de nodos
  - Se utiliza la sintaxis de CSS3. La selección está implementada en la librería, por lo que no depende del soporte del navegador.
  - Los elementos seleccionados no son directamente nodos del DOM (utiliza una clase envoltorio propia).
  - Se puede acceder a los nodos del DOM por índice (como en un *array*) o con el método `get`.

# jQuery (5)

```
$('#a')[0].href = 'http://www.uniovi.es/';  
$('#a').get(0).href = 'http://www.uniovi.es/';  
// Podemos saber el tamaño con size()  
alert('Hay ' + $('#a').size() + ' enlaces');
```

- Bucles

- Implícito

- Cuando se seleccionan varios nodos, el comando jQuery realiza un bucle para todos ellos.

```
// Ocultar todos los enlaces  
$('#a').hide();
```

# jQuery (6)

## - Explícito

- Pasando una función mediante `each()`. La función recibirá automáticamente como parámetro el índice del elemento actual.

```
$('.a').each(function (n) {  
    alert('a.href[' + n + '] = ' + this.href);  
})
```

## • Navegación por el árbol

```
//devuelve los hijos de los nodos seleccionados  
$('.ul').children()  
//devuelve todos los <div> que contienen párrafos  
$('.p').parent('div')  
//devuelve el antecesor más inmediato  
$('#imagen').closest('div')
```

# jQuery (7)

- Añadir o eliminar clases

```
// Añade la clase o clases indicadas a todos los párrafos  
$('p').addClass('destacado');  
// Quita la clase o clases indicadas en todos los párrafos  
$('p').removeClass('destacado')
```

- Cambiar estilos CSS

- css(propiedad, valor)

```
// todos los párrafos pasan a tener color azul  
$('p').css('color', 'blue');  
// se le puede pasar una función para que cambie el  
// valor a partir del actual  
$('a').css('width', function(n, actual) {  
    return parseInt(actual) + 1 + "px"; })
```

# jQuery (8)

- Acceso y modificación del contenido

```
// Equivalente al innerHTML  
$("#cabecera").html("<h1>hola</h1>");  
// Valor actual  
var cab = $("#cabecera").html();  
// Contenido de texto  
alert($("#cabecera").text());
```

- Edición y eliminación de nodos

```
// Añadir  
$("#cabecera").append("<h1>hola</h1>");  
// Eliminar todos los párrafos  
$('p').remove();  
// Eliminar los párrafos de clase 'borra'  
$('p').remove('.borra');
```

# jQuery (9)

## AJAX

```
$(document).ready(function(){
    $.ajax({
        type: "GET",
        url: "links.xml",
        dataType: "xml",
        success: function (xml) {
            $(xml).find('link').each(function() {
                var titulo = $(this).find('titulo').text();
                var href = $(this).find('href').text();
                $('ul#links').append(
                    '<li><a href="' + href + '">' +
                    titulo + '</li>'
                );
            });
        }
    });
});
```