

CS520 Web Programming

Introduction to AJAX

Chengyu Sun
California State University, Los Angeles

Browser As The New OS

- ◆ Easy application distribution and deployment
- ◆ Greatly simplifies system administration
 - No software to download, install, and update
 - Centralized data management
- ◆ Application can be used from anywhere
- ◆ So why it didn't happen??

Interactivity Issue

- ◆ Conventional GUI application
 - Rich event model
 - Responsive
 - No network delay
 - Partial redraw
- ◆ Web application
 - Simple request-response model
 - Not so responsive
 - Send request, wait for response
 - Full page refresh

HTML Event Models

- ◆ HTML 4 Event Model
 - HTML 4.01 Specification - <http://www.w3.org/TR/REC-html40/>
 - Limited features but portable
- ◆ Standard Event Model
 - DOM Level 2 HTML Specification - <http://www.w3.org/TR/2003/REC-DOM-Level-2-HTML-20030109/>
 - Fully featured but less portable
- ◆ Vendor specific event models

Events and Event Handler

- ◆ Events
 - onfocus, onblur, onkeypress, onkeydown, onkeyup, onclick, ondblclick, onmousedown, onmouseup, onmousemove, onmouseover ...
- ◆ Specify event handler
 - `<element event="code">`
 - For example:

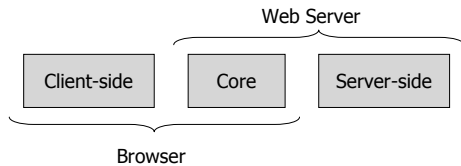
```
<button onclick="clickHandler();" >click</button>
```

Example: Event Handling with JavaScript

- ◆ j1.html in jex
- ◆ Disclaimer: all my JavaScript code is only tested under Firefox 1.07

JavaScript

- ◆ Interpreted language
- ◆ Originally developed by Netscape
- ◆ Syntax is similar to Java



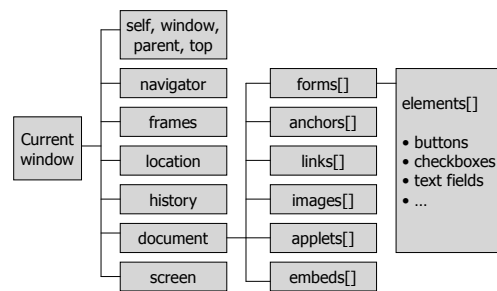
Core JavaScript

- ◆ Mainly covers language syntax, which is kind of similar to Java
- ◆ Global Object
 - Created by a JavaScript interpreter
 - *Global variables* and *global methods* are simply variables and methods of this object

Client-Side JavaScript

- ◆ Embed JavaScript in HTML
 - `<script>`
 - `wtype="text/javascript"`
 - `wlanague="JavaScript"`
 - `wsrc="path_to_script_file"`
- ◆ Run inside a browser
- ◆ Window is the global object

Window



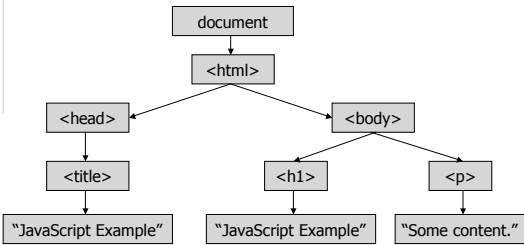
Document Object Model (DOM)

- ◆ Representing documents as objects so they can be manipulated in a programming language.

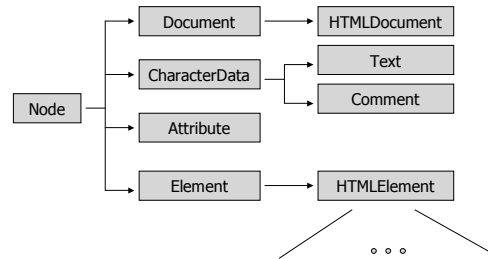
DOM HTML Representation - HTML

```
<html>
<head><title>JavaScript Example</title></head>
<body>
  <h1>JavaScript Example</h1>
  <p>Some content.</p>
</body>
</html>
```

DOM HTML Representation - DOM



Nodes



Find Element

- ◆ `document.getElementsByTagName()`
- ◆ `document.getElementById()`

Modify Element

- ◆ **HTMLElement properties and methods**
 - n IE
 - w `innerHTML`
 - w `innerText`
 - w `insertAdjacentHTML()`
 - w `insertAdjacentText()`
 - n Netscape/Mozilla
 - w `innerHTML`
 - n Element-specific

Communicate with Server

- ◆ The request-response model is still a limiting factor in user interactivity
- ◆ **Solution: XMLHttpRequest**
 - n A JavaScript object
 - w Send HTTP request
 - w Parse XML response
 - n *Response can be handled asynchronously*

XMLHttpRequest Example

- ◆ A1 in jex
 - n A client that sends XMLHttpRequest
 - n A servlet that replies with XML response
- ◆ Non-blocking - the server response is handled asynchronously with a *callback* function
- ◆ Partial page update

About the Example

- ◆ `getReadyStateHandler()`
 - Function parameter
 - Function return type

XMLHttpRequest - Methods

- ◆ `abort()`
- ◆ `getAllResponseHeaders()`
- ◆ `getResponseHeader(header)`
- ◆ `open(method, url, asyncFlag, username, password)`
 - `asyncFlag`, `username`, `password` are optional
- ◆ `send(responseBody)`
- ◆ `setRequestHeader(name, value)`

XMLHttpRequest - Properties

- ◆ `onreadystatechange`
- ◆ `readyState`
 - 0 – uninitialized
 - 1 – loading
 - 2 – loaded
 - 3 – interactive
 - 4 – complete
- ◆ `status`
- ◆ `statusText`
- ◆ `responseBody`
- ◆ `responseStream`
- ◆ `responseText`
- ◆ `responseXML`

AJAX

- ◆ AJAX = JavaScript + XMLHttpRequest
- ◆ **A**synchronous **J**avaScript and **X**ML
- ◆ The technologies have been around for several years
- ◆ The recent buzz seems to be started by Google Maps
 - Vs. Yahoo Maps
- ◆ Now it's "Web 2.0"!

AJAX Frameworks and Libraries

- ◆ http://ajaxpatterns.org/Ajax_Frameworks
- ◆ Some interesting ones (*)
 - Ajax JSP Tag Library - <http://ajaxtags.sourceforge.net/>
 - Ajax Faces - <http://myfaces.apache.org/>
 - OpenRico - <http://openrico.org>
 - Pragmatic Objects - <http://pragmaticobjects.com>
 - Prototype - <http://prototype.conio.net>
 - Scriptaculous - <http://script.aculo.us>
 - Taconite - <http://taconite.sourceforge.net>

(*) I bias toward Java related ones and the ones with online demos.

Using Google Maps API

- ◆ <http://www.google.com/apis/maps>
- ◆ A JavaScript API
- ◆ Host restriction
- ◆ Some missing functions
 - Geocoding
 - Routing

Google Maps in Evelyn

- ◆ Longitude-latitude lookup using <http://geocoder.us>
- ◆ Distance calculation and other spatial operations
 - PostGIS - <http://postgis.refrations.net>
- ◆ Mixing JSP and JavaScript