



ΔΙΚΤΥΑΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ & ΥΠΗΡΕΣΙΕΣ ΔΙΑΔΙΚΤΥΟΥ

Ajax Framework

Κωνσταντίνος Τσερπές

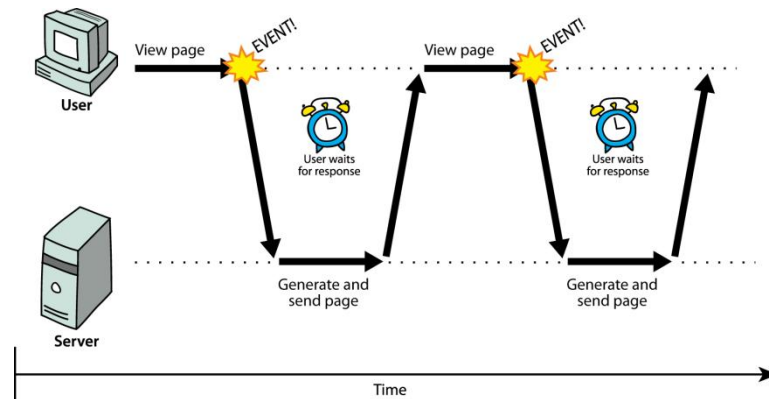
Τμήμα Πληροφορικής και Τηλεματικής

Χαροκόπειο Πανεπιστήμιο



ΣΥΓΧΡΟΝΗ ΔΙΑΔΙΚΤΥΑΚΗ ΕΠΙΚΟΙΝΩΝΙΑ

- synchronous: ο χρήστης περιμένει όσο φορτώνουν οι νέες σελίδες
 - Τυπικό πρότυπο επικοινωνίας στις ιστοσελίδες (click, wait, refresh)





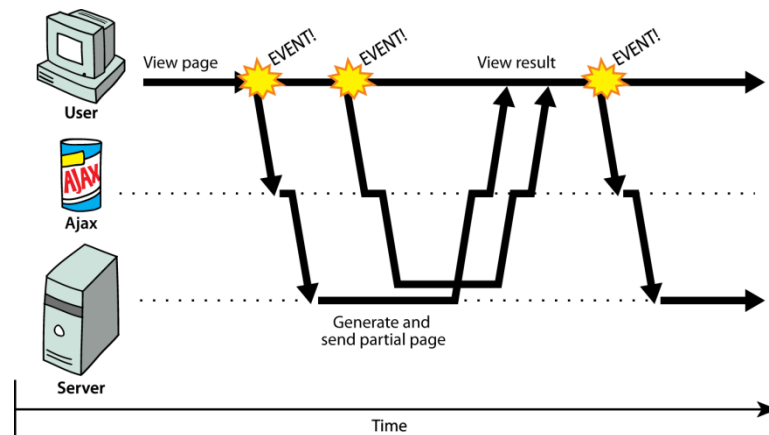
ΔΙΑΔΙΚΤΥΑΚΕΣ ΕΦΑΡΜΟΓΕΣ ΚΑΙ AJAX

- web application: (πολλά πράγματα, μεταξύ άλλων...) μία δυναμική ιστοσελίδα η οποία αναπαριστά μία κοινή desktop εφαρμογή
 - Παρουσιάζει μία συνεχή εμπειρία προς το χρήστη παρά ασύνδετες σελίδες
 - Παραδείγματα: Gmail, Google Maps, Google Docs and Spreadsheets, Flickr, Facebook
- Ajax: Asynchronous JavaScript and XML
 - Δεν είναι γλώσσα προγραμματισμού, είναι τρόπος χρήσης της JavaScript
 - “κατεβάζει” δεδομένα από ένα εξυπηρετητή χωρίς να το αντιλαμβάνεται ο χρήστης
 - Επιτρέπει τη δυναμική ανανέωση του περιεχομένου μίας σελίδας χωρίς να περιμένει ο χρήστης
 - Δεν ακολουθεί (απαραίτητα) το πρότυπο “click-wait-refresh”
 - Παραδείγματα: Facebook, Google Suggest (Google apps γενικά)



ΑΣΥΓΧΡΟΝΗ ΔΙΑΔΙΚΤΥΑΚΗ ΕΠΙΚΟΙΝΩΝΙΑ

- asynchronous: ο χρήστης μπορεί να αλληλεπιδρά με τη σελίδα ανεξάρτητα αν φορτώνουν τα δεδομένα
 - Εφικτό μέσω του Ajax





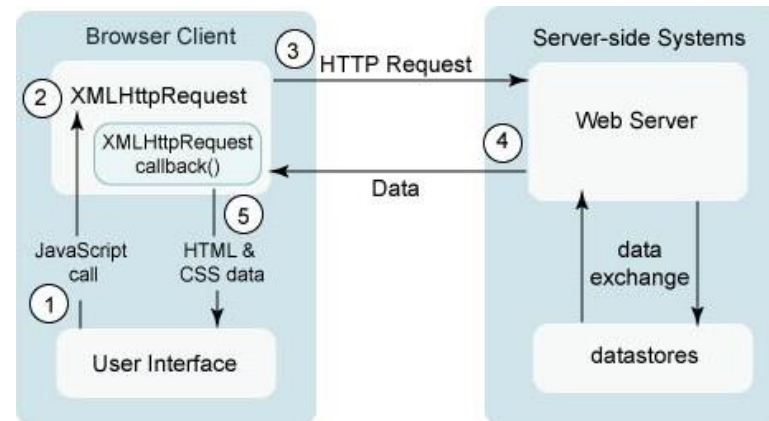
ΚΕΝΤΡΙΚΕΣ ΕΝΝΟΙΕΣ ΤΟΥ AJAX

- Το XMLHttpRequest object της JavaScript μπορεί να φέρει αρχεία από ένα server
 - Υποστηρίζεται από όλους τους γνωστούς browsers
- Μπορεί να γίνει ασύγχρονα χωρίς ο χρήστης να το αντιλαμβάνεται
- Τα περιεχόμενα του αρχείου μπορούν να εμφανιστούν στην τρέχουσα ιστοσελίδα χρησιμοποιώντας το DOM
- Οι σελίδες του χρήστη ανανεώνονται δυναμικά χωρίς να ξαναφορτώνεται η σελίδα



ΕΝΑ ΤΥΠΙΚΟ AJAX REQUEST

- Ο χρήστης ενεργοποιεί ένα event handler
- 1. Ο κώδικας χειρισμού του handler δημιουργεί ένα XMLHttpRequest object
- 2. Το XMLHttpRequest object ζητά ένα κείμενο από τον εξυπηρετητή
- 3. Ο εξυπηρετητής συγκεντρώνει τα απαραίτητα δεδομένα και τα στέλνει πίσω
- 4. Το XMLHttpRequest ενεργοποιεί ένα event για να ενημερώσει για την άφιξη των δεδομένων
 - Αυτό αναλαμβάνει η callback function
 - Μπορούμε να αναθέσουμε σε ένα handler να ενημερώνει για το πότε τα δεδομένα έφτασαν
- 5. Η callback function επεξεργάζεται τα δεδομένα (και τα απεικονίζει)





TO XMLHttpRequest OBJECT*

Method	Description
abort()	Cancels the current request
getAllResponseHeaders()	Returns header information
getResponseHeader()	Returns specific header information
open(method,url,async,uname,pswd)	<p>Specifies the type of request, the URL, if the request should be handled asynchronously or not, and other optional attributes of a request</p> <p>method: the type of request: GET or POST url: the location of the file on the server async: true (asynchronous) or false (synchronous)</p>
send(string)	<p>send(string) Sends the request off to the server.</p> <p>string: Only used for POST requests</p>
setRequestHeader()	Adds a label/value pair to the header to be sent

Property	Description
onreadystatechange	Stores a function (or the name of a function) to be called automatically each time the readyState property changes
readyState	<p>Holds the status of the XMLHttpRequest. Changes from 0 to 4:</p> <p>0: request not initialized 1: server connection established 2: request received 3: processing request 4: request finished and response is ready</p>
responseText	Returns the response data as a string
responseXML	Returns the response data as XML data
status	Returns the status-number (e.g. "404" for "Not Found" or "200" for "OK")
statusText	Returns the status-text (e.g. "Not Found" or "OK")

* More in <https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest>



ΠΙΘΑΝΟΙ ΣΥΝΔΥΑΣΜΟΙ XMLHTTPREQUEST OBJECT

- synchronous, text-only (SJAT?)
- asynchronous, text-only (AJAT?)
- asynchronous w/ XML data (AJAX)



SYNCHRONOUS REQUESTS (ΒΑΣΙΚΗ ΙΔΕΑ)

```
var xHRObj = new XMLHttpRequest();  
xHRObj.open("GET", url, false);  
xHRObj.send(null);  
// at this point, the request will have  
//    returned with its data  
do something with xHRObj.responseText;
```



SYNCHRONOUS REQUEST

○ Quiz

- Φτιάξτε μία σελίδα με 2 textboxes και ένα κουμπί το οποίο μόλις πατιέται θα καλεί μία php σελίδα (<http://galaxy.hua.gr/~tserpes/PhpProject1/sum.php>) η οποία θα υπολογίζει το άθροισμα των 2 αριθμών που θα βρίσκονται στα textboxes. Το αποτέλεσμα θα εκτυπώνεται στην αρχική σελίδα (κάτω από τα textboxes και το κουμπί) χωρίς να ανανεώνεται η σελίδα
- Δώστε ένα παράδειγμα που να αποδεικνύει ότι τα synchronous requests μπορεί να είναι προβληματικά



ASYNCHRONOUS REQUESTS (ΒΑΣΙΚΗ ΙΔΕΑ)

```
var ajax = new XMLHttpRequest();  
ajax.onreadystatechange = functionName;  
ajax.open("get", url, true);  
ajax.send(null);  
// don't process ajax.responseText here, but in  
    your function  
...
```



Η ΙΔΙΟΤΗΤΑ READYSTATE

- Περιέχει το status του XMLHttpRequest
- Όταν το readyState changes καλείται ο handler του onreadystatechange
- Το επιτυχές είναι το readyState == 4

State	Description
0	not initialized
1	set up
2	sent
3	in progress
4	complete



ASYNCHRONOUS XMLHttpRequest TEMPLATE

```
var ajax = new XMLHttpRequest();  
ajax.onreadystatechange = function() {  
  if (ajax.readyState == 4) {    // 4 means  
    request is finished  
    do something with ajax.responseText;  
  }  
};  
ajax.open("get", url, true);  
ajax.send(null);
```

○ Quiz:

- Γιατί βολεύει να ορίζουμε τον event handler ως εσωτερική, ανώνυμη συνάρτηση;



ΕΛΕΓΧΟΣ ΣΦΑΛΜΑΤΩΝ ΣΤΟ REQUEST

```
var ajax = new XMLHttpRequest();  
ajax.onreadystatechange = function() {  
    if (ajax.readyState == 4) {  
        if (ajax.status == 200) {    // 200 means  
            request succeeded  
            do something with ajax.responseText;  
        } else {  
            code to handle the error;  
        }  
    }  
};
```

○ Quiz

- Ποιο θα είναι το status του request αν η σελίδα δεν υπάρχει;



XMLHttpRequest Περιορισμοί Ασφαλείας

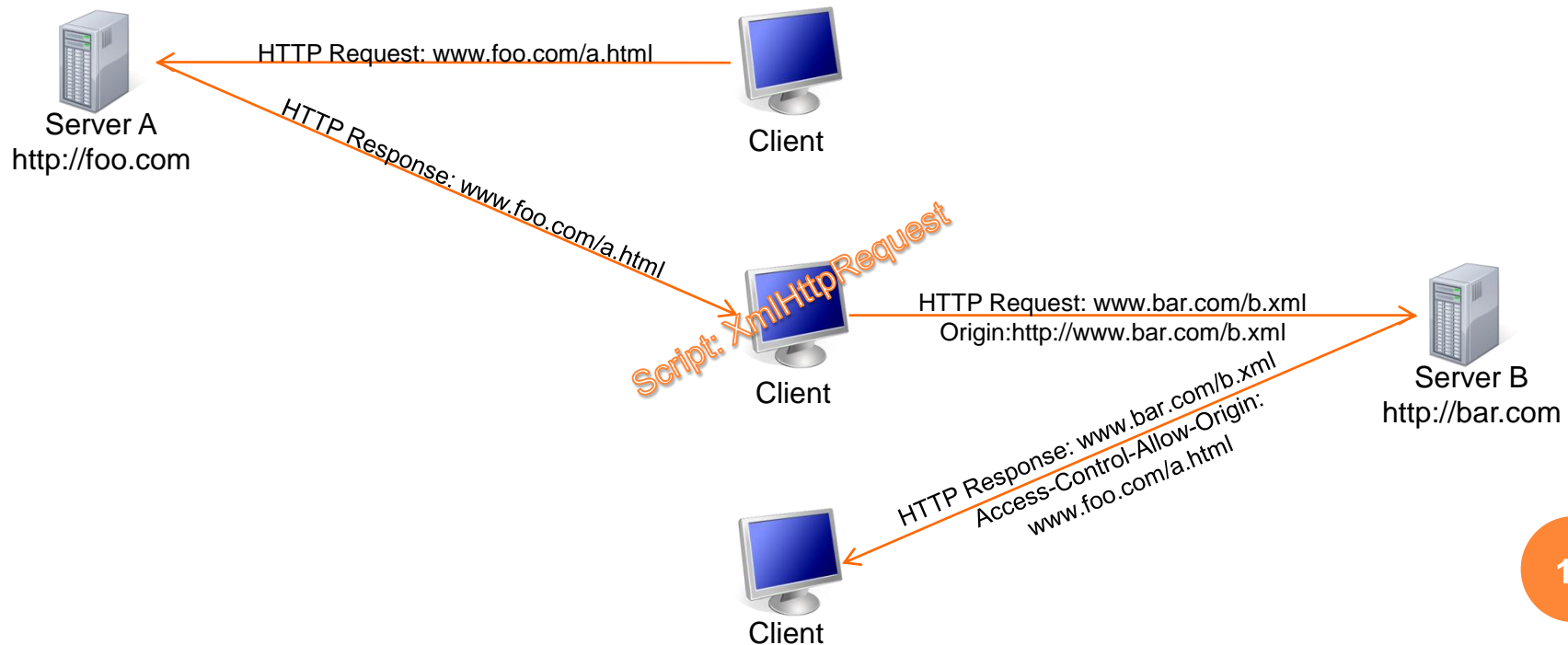
○ Same Origin Policy

- Η ιστοσελίδα που κάνει το request και το αρχείο το οποίο καλεί, πρέπει να προέρχονται από το ίδιο domain name, application layer protocol και port number
- Π.χ. ένα script στο `www.foo.com/a/b/c.html` μπορεί να δεχτεί HTTP responses μόνο από το: `www.foo.com`



CROSS ORIGIN RESOURCE SHARING (CORS)

- Cross-Site HTTP Requests
- Origin HTTP header





TO XMLHttpRequest LEVEL2 OBJECT*

- Latest version:
<http://www.w3.org/TR/XMLHttpRequest/>
- Important points (for starters):
 - withCredentials
 - More event handlers than onreadystatechange
- In brief: [XMLHttpRequestLvl2Reference.txt](#)



`XMLHttpRequest2` EXAMPLE (FROM `HTML5ROCKS.COM`)

```
// Create the XHR object.
function createCORSRequest(method, url) {
    var xhr = new XMLHttpRequest();
    if ("withCredentials" in xhr) {
        // XHR for Chrome/Firefox/Opera/Safari.
        xhr.open(method, url, true);
    } else if (typeof XDomainRequest != "undefined") {
        // XDomainRequest for IE.
        xhr = new XDomainRequest();
        xhr.open(method, url);
    } else {
        // CORS not supported.
        xhr = null;
    }
    return xhr;
}
// Helper method to parse the title tag from the response.
function getTitle(text) {
    return text.match('<title>(.*?)</title>')[1];
}
```

```
// Make the actual CORS request.
function makeCorsRequest() {
    // All HTML5 Rocks properties support CORS.
    var url = 'http://updates.html5rocks.com';
    var xhr = createCORSRequest('GET', url);
    if (!xhr) {
        alert('CORS not supported');
        return;
    }
    // Response handlers.
    xhr.onload = function() {
        var text = xhr.responseText;
        var title = getTitle(text);
        alert('Response from CORS request to ' + url + ': ' + title);
    };
    xhr.onerror = function() {
        alert('Woops, there was an error making the request.');
```



ASYNCHRONOUS REQUESTS

○ Quiz

- Φτιάξτε το προηγούμενο παράδειγμα με χρήση asynchronous request



ΠΗΓΕΣ

- http://www.webstepbook.com/supplements/slides/ch10-ajax_xml.shtml
- Chris Ullman, Lucinda Dykes, Beginning Ajax, Wiley Publishing Inc, 2007