



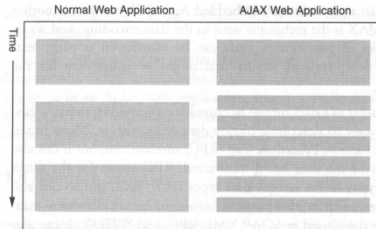
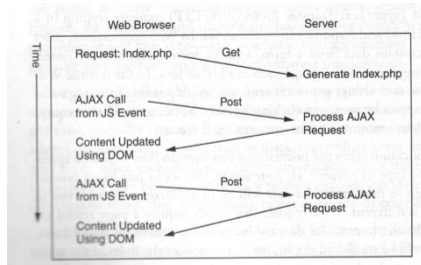
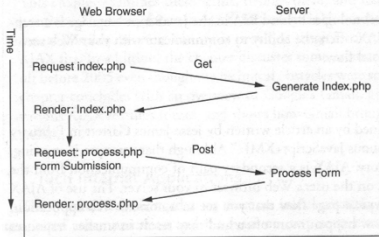
AJAX

- Asynchronous JavaScript+XML
- 2nd communication path between browser and web server
 - Remove communication bottleneck bw user and web application
 - Talk to server from JavaScript
 - Skip page reload
- Changes the typical page flow
 - More frequent requests
 - Smaller responses of non-HTML data
- Defined by Jesse James Garret in Feb. of 2005
 - Underlying technology in place since 2001
 - Google releases Gmail in March of 2004
 - One of first mainstream apps to use AJAX
 - Examples: read/tag/spell check messages without a page reload, auto save drafts

Normal Web vs. AJAX

Web Application

AJAX Application





XMLHttpRequest

- JavaScript HTTP client
 - Can load content other than XML
- Issue GET and POST requests
- Security limitation apply
 - Same point of origin; Can only connect to same domain as currently loaded page
 - Only load HTTP requests
 - File URLs don't work

XMLHttpRequest Methods

Method	Description
<code>abort()</code>	Cancels the current request.
<code>getAllResponseHeaders()</code>	Returns the complete set of HTTP headers as a string.
<code>getResponseHeader(headerName)</code>	Returns the value of the specified HTTP header.
<code>open(method, URL)</code> <code>open(method, URL, async)</code> <code>open(method, URL, async, userName)</code> <code>open(method, URL, async, userName, password)</code>	<p>Specifies the method, URL, and other optional attributes of a request.</p> <p>The method parameter can have a value of "GET", "POST", "HEAD", "PUT", "DELETE", or a variety of other HTTP methods listed in the W3C specification.^[1]</p> <p>The URL parameter may be either a relative or complete URL.</p> <p>The "async" parameter specifies whether the request should be handled asynchronously or not – "true" means that script processing carries on after the <code>send()</code> method, without waiting for a response, and "false" means that the script waits for a response before continuing script processing.</p>
<code>send(content)</code>	Sends the request.
<code>setRequestHeader(label, value)</code>	Adds a label/value pair to the HTTP header to be sent.

XMLHttpRequest Properties

Property	Description
<code>onreadystatechange</code>	An event handler for an event that fires at every state change.
<code>readyState</code>	Returns the state of the object as follows: <ul style="list-style-type: none">■ 0 = uninitialized■ 1 = open■ 2 = sent■ 3 = receiving■ 4 = loaded.
<code>responseText</code>	Returns the response as a string.
<code>responseXML</code>	Returns the response as XML. This property returns an XML document object, which can be examined and parsed using W3C DOM node tree methods and properties.
<code>status</code>	Returns the HTTP status code as a number (e.g. 404 for "Not Found" and 200 for "OK").
<code>statusText</code>	Returns the status as a string (e.g. "Not Found" or "OK").

XMLHttpRequest Events

Firefox
Specific

Attribute	Type	Description
onload	<u>nsIDOMEventListener</u>	Called when the HTTP document has been completely loaded. The event sent to the receiver has its target attribute set to the XMLHttpRequest that finished loading.
onerror	<u>nsIDOMEventListener</u>	Called when an error occurs while attempting to retrieve the HTTP document. Check the status attribute of the XMLHttpRequest specified by the target attribute in order to determine which HTTP error occurred.
onprogress	<u>nsIDOMEventListener</u>	Called periodically while the document is being loaded. The event sent to the receiver has a position attribute indicating how far through the document the process is, as well as a totalSize attribute indicating the total size of the document. In addition, it has its target attribute set to the XMLHttpRequest corresponding to the progress notification.
onreadystatechange	<u>nsIONReadyStateChangeHandler</u>	Called when the readyState changes. <i>What does this mean?</i>

- Firefox resets handlers after request completes

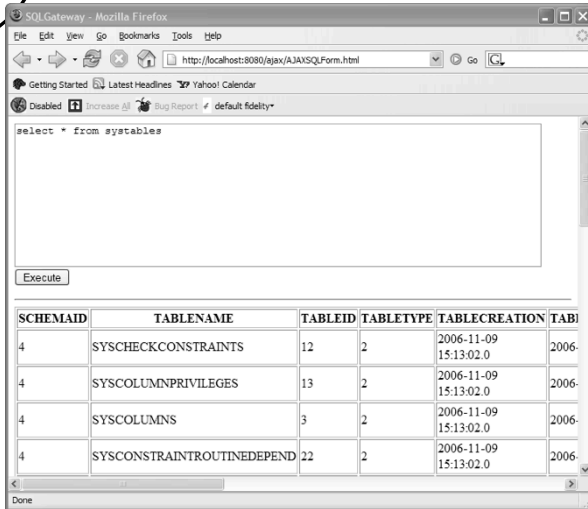
Synchronous Request

```
var req = new XMLHttpRequest();  
req.open('POST','index.php', false);  
req.setRequestHeader('Content-type','application/x-www-form-urlencoded')  
req.send('one=firstArg&two=secondArg');  
if (req.status==200)  
    alert(req.responseText);
```

Asynchronous Request

```
var req=XMLHttpRequest();
req.onreadystatechange = function () {
    if (req.readyState == 4) {
        if(req.status == 200) {
            alert(req.responseText);
        }
        else {
            alert('Loading Error: [' + req.status + ']' + req.statusText);
        }
    }
}
req.open('GET','csc309/servlet/ajaxserver',true);
req.send(null);
```


Adding Dynamically Generated HTML to a Page



SQLGateway - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://localhost:8080/ajax/AJAXSQLForm.html

Getting Started Latest Headlines Yahoo! Calendar

Disabled Increase All Bug Report default fidelity

select * from systables

Execute

SCHEMAID	TABLENAME	TABLEID	TABLETYPE	TABLECREATION	TAB
4	SYSCHECKCONSTRAINTS	12	2	2006-11-09 15:13:02.0	2006-
4	SYSCOLUMNPRIVILEGES	13	2	2006-11-09 15:13:02.0	2006-
4	SYSCOLUMNS	3	2	2006-11-09 15:13:02.0	2006-
4	SYSCONSTRAINTROUTINEDEPEND	22	2	2006-11-09 15:13:02.0	2006-

Done

IE AJAX

- XMLHttpRequest is an ActiveX object

```
var xmlhttp = new ActiveXObject('MSXML2.XMLHTTP.5.0');
```

- Multiple versions
 - Microsoft.XMLHTTP
 - MSXML2.XMLHTTP
 - MSXML2.XMLHTTP.3.0
 - MSXML2.XMLHTTP.4.0
 - MSXML2.XMLHTTP.5.0

Cross-Browser XMLHttpRequest

```
function XMLHttpRequest () {  
    try {  
        // Mozilla / Safari  
        xmlhttp = new XMLHttpRequest();  
    } catch (e) {  
        // IE  
        var XMLHTTP_IDS = new Array('MSXML2.XMLHTTP7.0'  
  
        'MSXML2.XMLHTTP6.0'  
  
        'MSXML2.XMLHTTP5.0',  
        'MSXML2.XMLHTTP4.0',  
        'MSXML2.XMLHTTP3.0',  
        'MSXML2.XMLHTTP',  
        'Microsoft.XMLHTTP' );  
  
        var success = false;  
        for (var i=0; i < XMLHTTP_IDS.length && !success; i++) {  
            try {  
                this.xmlhttp = new ActiveXObject(XMLHTTP_IDS[i]);  
                success = true;  
            } catch (e) {}  
        }  
        if (!success) {  
            throw new Error("Unable to create XMLHttpRequest.");  
        }  
    }  
    return xmlhttp;  
}
```

JQuery AJAX

- Simplifies handling of requests
- Takes care of cross-browser issues
- `$.ajax(url [, settings])`

Property	Description
url	URL of the resource to be requested
type	HTTP method to use: GET, POST, DELETE, PUT.
data	Data to send to server
dataType	Type of data expected back from server: XML, HTML, script, JSON
success(data, textStatus, jqXHR)	Function called if request succeeds
error(jqXHR, textStatus, errorThrown)	Function called if request fails
complete(jqXHR, textStatus)	Function called when request completes
timeout	Timeout (in milliseconds) for request



JQuery AJAX utility functions

- `$.get(url [, data] [, success(data, textStatus, jqXHR)] [, dataType])`
- `$.post(url [, data] [, success(data, textStatus, jqXHR)] [, dataType])`
- `$(selector).load(url [, data] [, complete(responseText, textStatus, XMLHttpRequest)]`
 - Load data from the server and place the returned HTML into the matched element.
- `$(selector).serialize()`
 - Creates a text string in standard URL-encoded notation
- `$(selector).serializeArray()`
 - Creates a JavaScript array of objects

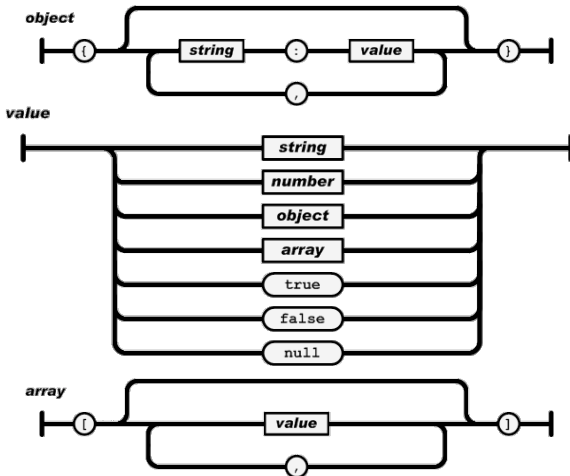
JSON



- JavaScript Object Notation
- Lightweight data-interchange format.
 - Text format
 - Language independent
- Easy for humans to read and write.
- Easy for machines to parse and generate.
- Based on JavaScript

JSON Structures

- A collection of name/value pairs.



JSON Example

PHP

```
class Stock {  
    $sym;  
    $val;  
}
```

```
$stocks[0] = new Stock("MSFT",10);  
$stocks[1] = new Stock("MOT",15);
```

JSON

```
{"stocks":[{"sym":"MSFT","val":10.0}, {"sym":"MOT","val":15.0}]}
```


JSON in PHP

- Packing

string **json_encode** (mixed \$value [, int \$options = 0])

- Unpacking

mixed **json_decode** (string \$json [, bool \$assoc = false [, int \$depth = 512 [, int \$options = 0]]])

JSON in JavaScript

- Unpacking

```
var myObject = JSON.parse(myJSONtext);
```

```
var jsonString = '{"name":"Ryan", "address":"Mountain View, CA"}';
```

```
var person = JSON.parse(jsonString);
```

```
// 'person' is now a JavaScript object with 2 properties; name and address
```

- Packing

```
var myJSONtext= JSON.stringify(myObject );
```

```
var personString = JSON.stringify(person);
```

```
// 'personString' now holds the string '{"name":"Ryan", "address":' ...
```



JSON in JQuery

- Fetching JSON

- `$.getJSON(url [, data] [, success(data, textStatus, jqXHR)])`

- Creating JSON

- `$(selector).serializeArray()`
- Creates a JavaScript array of objects, ready to be encoded as a JSON string



jQuery Timers

- jQuery plugin
- <http://jquery.offput.ca/timers/>
- `$(selector).everyTime(interval : Integer | String, [label = interval : String], fn : Function, [times = 0 : Integer])`
- `$(selector).oneTime(interval : Integer | String, [label = interval : String], fn : Function)`
- `$(selector).stopTime([label : Integer | String], [fn : Function])`

```

1 <?php
2
3 class Main extends CI_Controller {
4
5
6     function __construct() {
7         // Call the Controller constructor
8         parent::__construct();
9     }
10
11     function index() {
12         $data['title']='csc309 AJAX';
13         $data['main']='home.php';
14         $this->load->view('template',$data);
15     }
16
17     function commander() {
18         $data['title']='SQL Commander';
19         $data['main']='commanderForm.php';
20         $this->load->view('template',$data);
21     }
22
23
24     function commander1() {
25         $data['title']='SQL Commander';
26         $data['main']='commanderForm1.php';
27         $this->load->view('template',$data);
28     }
29
30     function commander2() {
31         $data['title']='SQL Commander';
32         $data['main']='commanderForm2.php';
33         $this->load->view('template',$data);
34     }
35
36     function commander3() {
37         $data['title']='SQL Commander';
38         $data['main']='commanderForm3.php';
39         $this->load->view('template',$data);
40     }
41
42     function commanderRunQuery() {
43         $this->load->library('form_validation');
44         $this->form_validation->set_rules('query','Query','required');
45
46         if ($this->form_validation->run() == true) {
47             $query = $this->db->query($this->input->get_post('query'));
48             if ($query == false) { // there was a problem with the query or the db
49                 $data['errmsg'] = $this->db->_error_message();
50                 $data['errno'] = $this->db->_error_number();
51             }
52             else {
53                 $data['query'] = $query;
54             }
55         }
56
57         if (isset($data))
58             $this->load->view('queryResult',$data);
59         else
60             $this->load->view('queryResult');
61     }
62
63
64     function stockTicker() {
65         $data['title']='Stock Ticker';
66         $data['main']='stockTicker.php';
67         $this->load->view('template',$data);
68     }
69
70     function getStocks() {
71
72

```

```
73
74 }
75
76 function stockManager() {
77     $this->load->model('stock_model');
78     $stocks = $this->stock_model->getAll();
79
80     $data['title']='Stock Manager';
81     $data['main']='stockManager.php';
82     $data['stocks']=$stocks;
83
84     $this->load->view('template',$data);
85 }
86
87 function updateStock() {
88     $stock = new Stock();
89     $stock->symbol = $this->input->post('symbol');
90     $stock->value = $this->input->post('value');
91     $this->load->model('stock_model');
92     $this->stock_model->update($stock);
93 }
94
95 function stockManager2() {
96     $this->load->model('stock_model');
97     $stocks = $this->stock_model->getAll();
98
99     $data['title']='Stock Manager';
100    $data['main']='stockManager2.php';
101    $data['stocks']=$stocks;
102
103    $this->load->view('template',$data);
104 }
105
106 function updateAllStock() {
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122 }
123
124 }
125
126
```

```
1 <?php
2 class Stock_model extends CI_Model {
3
4     function getAll()
5     {
6         $query = $this->db->get('stocks');
7         if ($query && $query->num_rows() > 0)
8             return $query->result('Stock');
9         else
10            return null;
11     }
12
13     function update($stock) {
14         $this->db->where('symbol',$stock->symbol);
15         return $this->db->update('stocks',array('value'=>$stock->value));
16     }
17
18
19 }
20 ?>
21
22
23 <?php
24 class Stock {
25     public $symbol;
26     public $value;
27 }
28 ?>
```

```
1
2<script src="<?= base_url() ?>js/jquery-latest.js"></script>
3<script>
4    $(function(){
5        $('form').submit(function(){
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22            return false;
23        });
24    });
25
26</script>
27
28<h2>SQL Commander</h2>
29
30<?php
31    echo form_open();
32    echo form_label('Query');
33    echo form_textarea('query',set_value('query'));
34    echo form_submit('submit', 'Run');
35    echo form_close();
36?>
37
38<hr />
39
40<h2>Result</h2>
41<div id="result">
42</div>
43
44
45
```



```
1
2<script src="<?= base_url() ?>js/jquery-latest.js"></script>
3<script>
4    $(function(){
5        $('form').submit(function(){
6
7
8
9
10
11
12
13
14
15
16            return false;
17        });
18    });
19
20</script>
21
22<h2>SQL Commander</h2>
23
24<?php
25    echo form_open();
26    echo form_label('Query');
27    echo form_textarea('query',set_value('query'));
28    echo form_submit('submit', 'Run');
29    echo form_close();
30?>
31
32<hr />
33
34<h2>Result</h2>
35<div id="result">
36</div>
37
38
39
```

```
1
2<script src="<?= base_url() ?>js/jquery-latest.js"></script>
3<script>
4    $(function(){
5        $('form').submit(function(){
6
7
8
9
10
11
12            return false;
13        });
14    });
15
16</script>
17
18<h2>SQL Commander</h2>
19
20<?php
21    echo form_open();
22    echo form_label('Query');
23    echo form_textarea('query',set_value('query'));
24    echo form_submit('submit', 'Run');
25    echo form_close();
26?>
27
28<hr />
29
30<h2>Result</h2>
31<div id="result">
32</div>
33
34
35
```

```
1
2<script src="<?= base_url() ?>js/jquery-latest.js"></script>
3<script>
4    $(function(){
5        $('form').submit(function(){
6
7
8
9
10           return false;
11        });
12    });
13
14</script>
15
16<h2>SQL Commander</h2>
17
18<?php
19    echo form_open();
20    echo form_label('Query');
21    echo form_textarea('query',set_value('query'));
22    echo form_submit('submit', 'Run');
23    echo form_close();
24?>
25
26<hr />
27
28<h2>Result</h2>
29<div id="result">
30</div>
31
32
33
```

```

1 <?php
2     echo validation_errors();
3
4     if (isset($errno))
5         echo "<p>DB Error: ($errno) $errmsg</p>";
6
7     if (isset($query) && $query->num_rows() > 0) {
8         echo "<table>";
9         echo "<tr>";
10
11         $fields = $query->list_fields();
12
13         foreach ($fields as $field)
14         {
15             echo "<th>" . $field . "</th>";
16         }
17         echo "</tr>";
18
19         $result = $query->result_array();
20         foreach ($result as $row) {
21             echo "<tr>";
22
23             foreach ($fields as $field)
24             {
25                 echo "<td>" . $row[$field] . "</td>";
26             }
27
28             echo "</tr>";
29         }
30         echo "<table>";
31     }
32 }
33
34
35 ?>
36
37

```

```
1
2<script src="<?= base_url() ?>js/jquery-latest.js"></script>
3
4<script>
5    $(function(){
6
7
8
9
10
11
12    });
13
14
15</script>
16
17<h2>Stock Manager</h2>
18
19<table>
20<?php
21    foreach($stocks as $stock) {
22        echo "<tr>";
23        echo "<td>" . $stock->symbol . "</td>";
24        echo "<td>";
25        echo form_input('value',$stock->value,"id='" . $stock->symbol . "'");
26        echo "</td>";
27        echo "</tr>";
28    }
29?>
30</table>
31
32
33
```

```

1
2<script src="<?= base_url() ?>js/jquery-latest.js"></script>
3
4<script>
5    $(function(){
6        $('form').submit(function(){
7
8
9
10
11
12
13
14
15
16
17
18
19            return false;
20        });
21    });
22
23
24
25</script>
26
27<h2>Stock Manager</h2>
28
29
30<?php
31    echo form_open();
32    echo "<table>";
33    foreach($stocks as $stock) {
34        echo "<tr>";
35        echo "<td>" . $stock->symbol . "</td>";
36        echo "<td>";
37        echo form_input($stock->symbol,$stock->value);
38        echo "</td>";
39        echo "</tr>";
40    }
41    echo "<tr><td>" . form_submit('submit', 'Update') . "</td></tr>";
42    echo "</table>";
43
44    echo form_close();
45?>
46
47
48

```

stockTicker.php

```
1
2<script src="<?= base_url() ?>js/jquery-latest.js"></script>
3<script src="<?= base_url() ?>js/jquery.timers.js"></script>
4
5<script>
6    var stockInfo = "No stock info";
7    var position = 0;
8
9    $(function(){
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26    });
27
28</script>
29
30<h2>Stock Ticker</h2>
31
32<p>
33<span id="ticker"> </span>
34</p>
35
36<button id="b1" >Update Ticker</button>
37
```