

Day Two. Ajax 101

A Look Under the Hood



Bill W. Scott, Y! Ajax Evangelist

bscott@yahoo-inc.com



Operation. Using XHR

Simple Ajax 'Hello World'

Ajax Hello World

Clicking the link below will use XHR to fetch the data and then show the result in the box below.

[Make an ajax request for data](#)

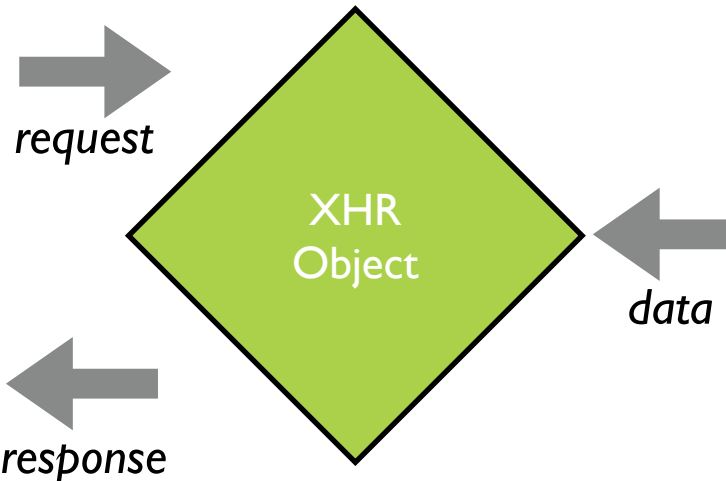
This data was brought to you by Ajax!

```
<h2>Ajax Hello World</h2>
<p>Clicking the link below will use XHR to fetch the data and then show the result in the box below.</p>

<span class="ajaxlink" onclick="makeRequest('test.xml')">
Make an ajax request for data
</span>

<div id="helloArea"></div>
```

/index.html



```
<?xml version="1.0" ?>
<root>
This data was brought
to you by Ajax!
</root>
```

/response.xml

- Clicking the link makes an XHR request
- Response is inserted into the area outlined in blue

Ajax How To

1. Create a request object
2. Write a callback
3. Make the request
4. Parse the response

if browser is mozilla or safari or opera then
create a `new XMLHttpRequest`

otherwise it is IE then
create a `new ActiveXObject`

otherwise

error - browser does not support XMLHttpRequest

- IE5, 5.5, 6 implements XHR as an ActiveX object (Msxm12.XMLHTTP/Microsoft.XMLHTTP)
- Mozilla 1.0+, Safari 1.2+, Opera 8+, IE7 provide XMLHttpRequest object natively
- All XHR objects have same methods & properties

Method	Description
<code>open("method", "url", [, asyncFlag [, "username" [, "password"]]])</code>	Sets up the request object for sending a request
<code>send(content)</code>	Sends the request to the server. Can be null.
<code>abort()</code>	Stops the request.
<code>getAllResponseHeaders()</code>	Returns all response headers for the HTTP request as key/value pairs.
<code>getResponseHeader("header")</code>	Returns the string value of the specified header.
<code>setRequestHeader("header", "value")</code>	Sets the specified header to the supplied value.

Source: *Foundations of Ajax* - APress



Property	Description
onreadystatechange	The event handler that fires at every state change.
readyState	The state of the request: 0=uninitialized, 1=loading, 2=loaded, 3=interactive, 4=complete
responseText	The response from the server as a text string
responseXML	The response from the server as an XML document
status	The HTTP status code from the server for the request object: 200: Ok; 201: Created; 400: bad request, 403: Forbidden; 500: internal sever error
statusText	The text version of the HTTP status code

```
function handleAjaxResponse  
begin  
    do something with the data that is returned from XHR  
end
```

- JavaScript callback function is invoked when the readystate changes on the XMLHttpRequest object

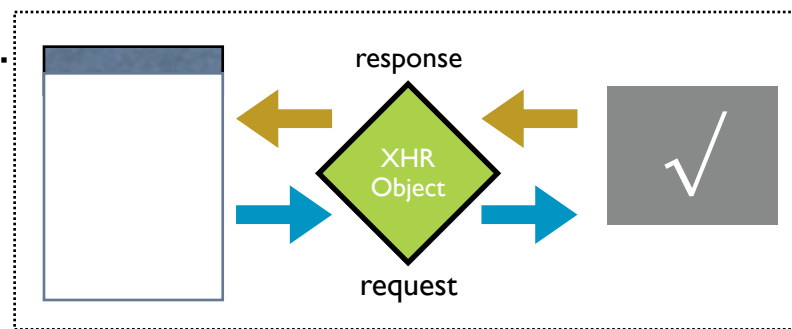
set `onreadystatechange` to callback function - `handleAjaxResponse`

open a request on the `xhr` object

send the request through the `xhr` object

- The JavaScript function *handleAjaxResponse* will be invoked when the `readystatechange` property changes on the XHR object
- Same site rule
- 'GET' or 'POST'
- Asynchronous flag

server must be accessible via relative url



```
function handleAjaxResponse
begin
    if response is valid then
        get responseXML
        get rootNode
        get helloArea on the page
        stuff the rootNode value into the helloArea DIV
    endif
end
```

- **readystatechange** values

- 0 - Uninitialized
- 1 - Loading
- 2 - Loaded
- 3 - Interactive
- 4 - Completed

- XML version must be first line

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

- Set up response header's Content-type
“Content-Type”, “text/xml”
- Use well-formed XML

- Use XHR property **responseXML** to get the response as an XML DOM (XmlDocument)
- Use standard JavaScript DOM methods
 - Mozilla, Safari, Opera & IE support a common set of methods and properties
 - Watch out for IE only stuff (e.g., **children** property)
 - Watch out for whitespace issues
- Other options
 - Use XML library for JS (e.g., XML for <script>)

Property/Method	Description
documentElement	Returns the root element of the document
firstChild	Is the first element within another element (the first child of the current node)
lastChild	Is the last element within another element (the last child of the current node)
nextSibling	Is the next element in the same nested level as the current one
previousSibling	Is the previous element in the same nested level as the current one
nodeValue	The value of a document element
getElementsByTagName	Used to place all elements into an object ????????

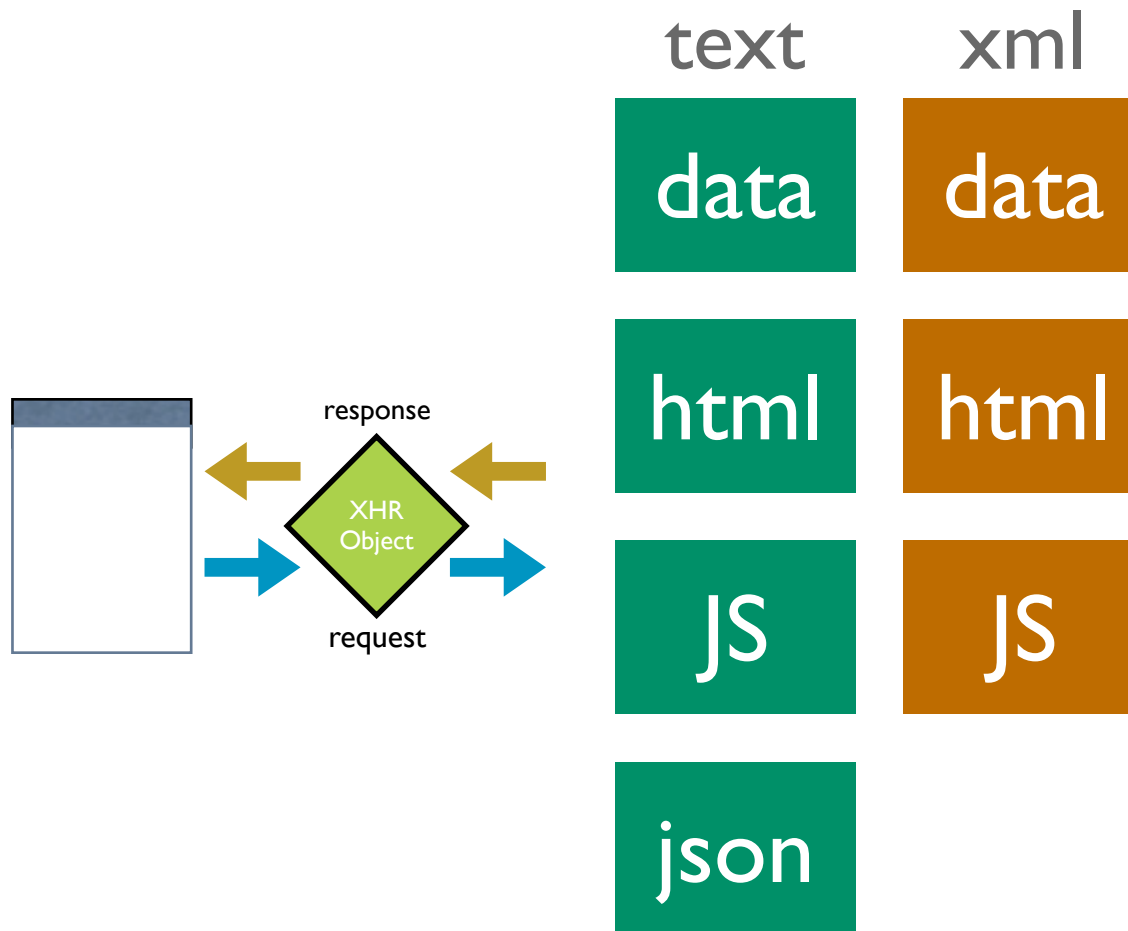
4. Parse the Response: A little more parsing.

```
<?xml version="1.0" ?>
<root>
  <contents>
    This data was brought to you by Ajax!
  </contents>
  <style>
    height:40px;width:250px;margin-top:20px;padding:20px;border:8px solid red;
  </style>
</root>
```

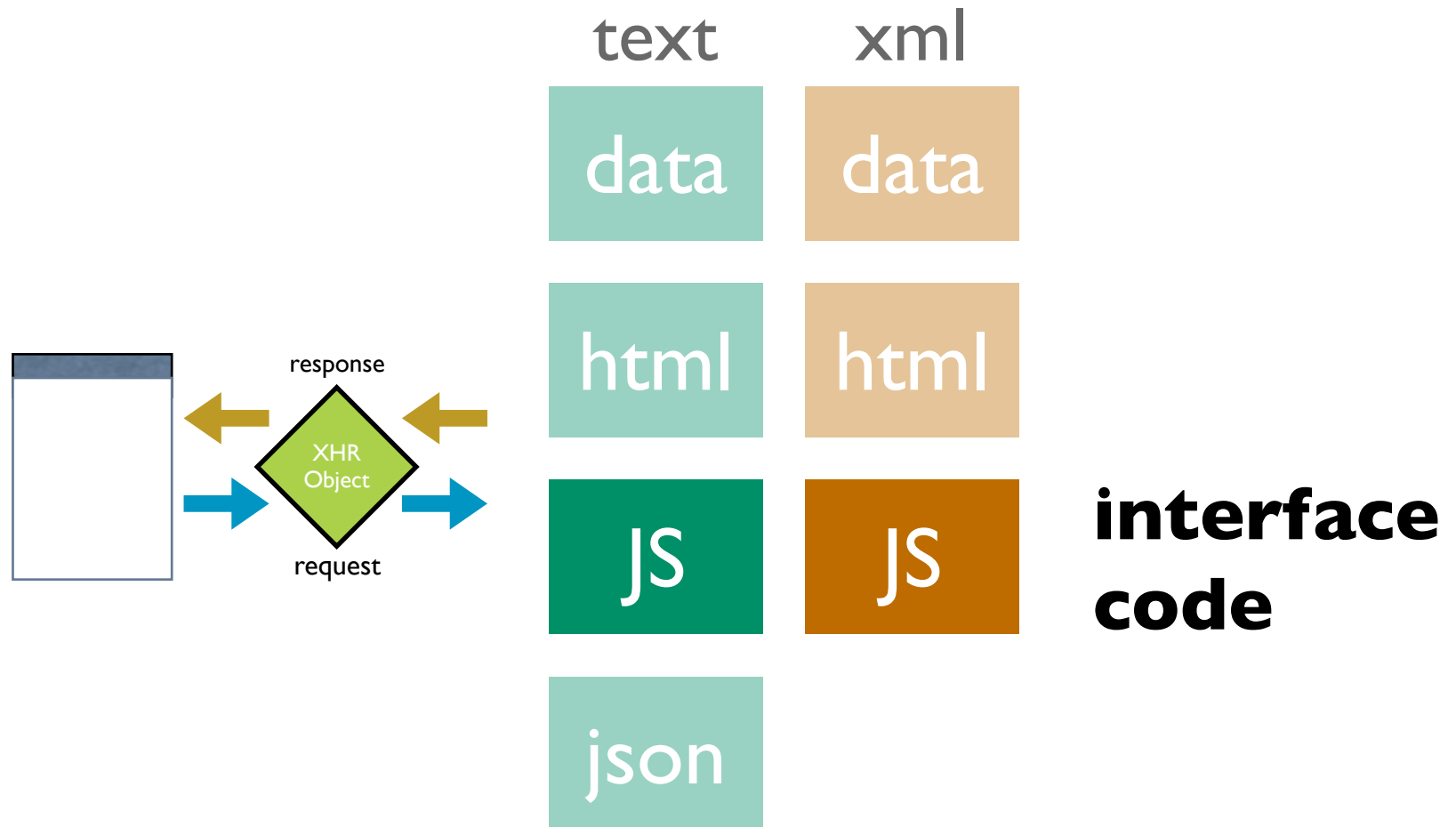
```
function alertContents() {
    if (xhr.readyState == 4) {
        if (xhr.status == 200) {
            var response = xhr.responseXML.documentElement;
            var content =
                response.getElementsByTagName('contents')[0].firstChild.data;
            var style =
                response.getElementsByTagName('style')[0].firstChild.data;
            var helloArea = document.getElementById("helloArea");
            helloArea.innerHTML=content;
            helloArea.style.cssText=style;
        } else {
            alert('There was a problem with the request.');
```

This data was brought to you by Ajax!

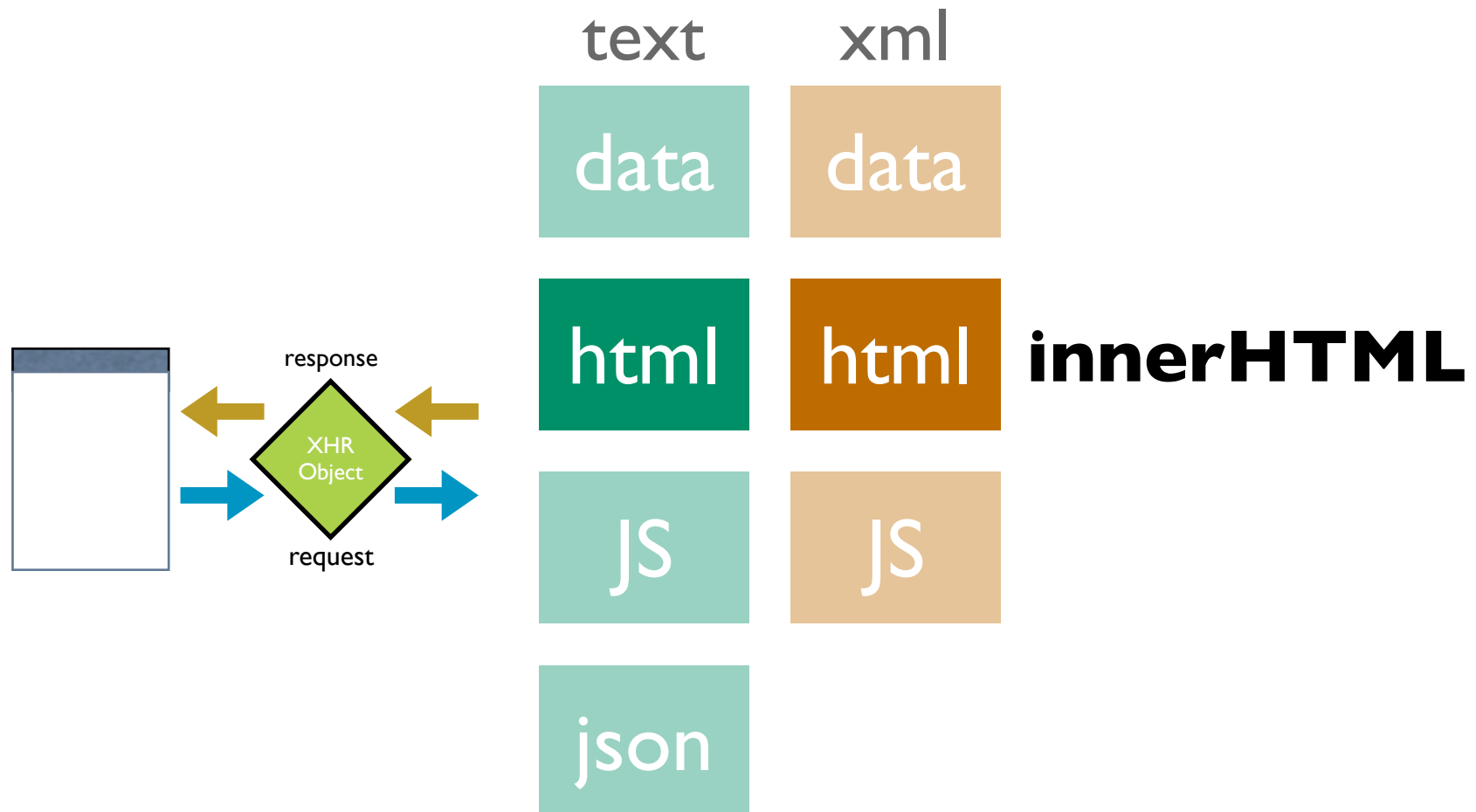
- Two types: text, xml
- Many formats



- JavaScript generated on server-side
- JavaScript is eval'ed on client side to generate GUI
- Dangerous!



- HTML generated on server side
- Stuffed into innerHTML of DOM element



Handling the Response: Direct-HTML Method

```
// var helloArea = YAHOO.util.Dom.get("hello-area");  
var helloArea = document.getElementById("hello-area");  
helloArea.innerHTML="<p>Hello <em>World</em></p>";
```

```
<p>A message will be inserted below:</p>  
<div id="hello-area" style="height:20px;width:220px;  
margin-top:8px;padding:4px;border:1px solid blue"></div>
```

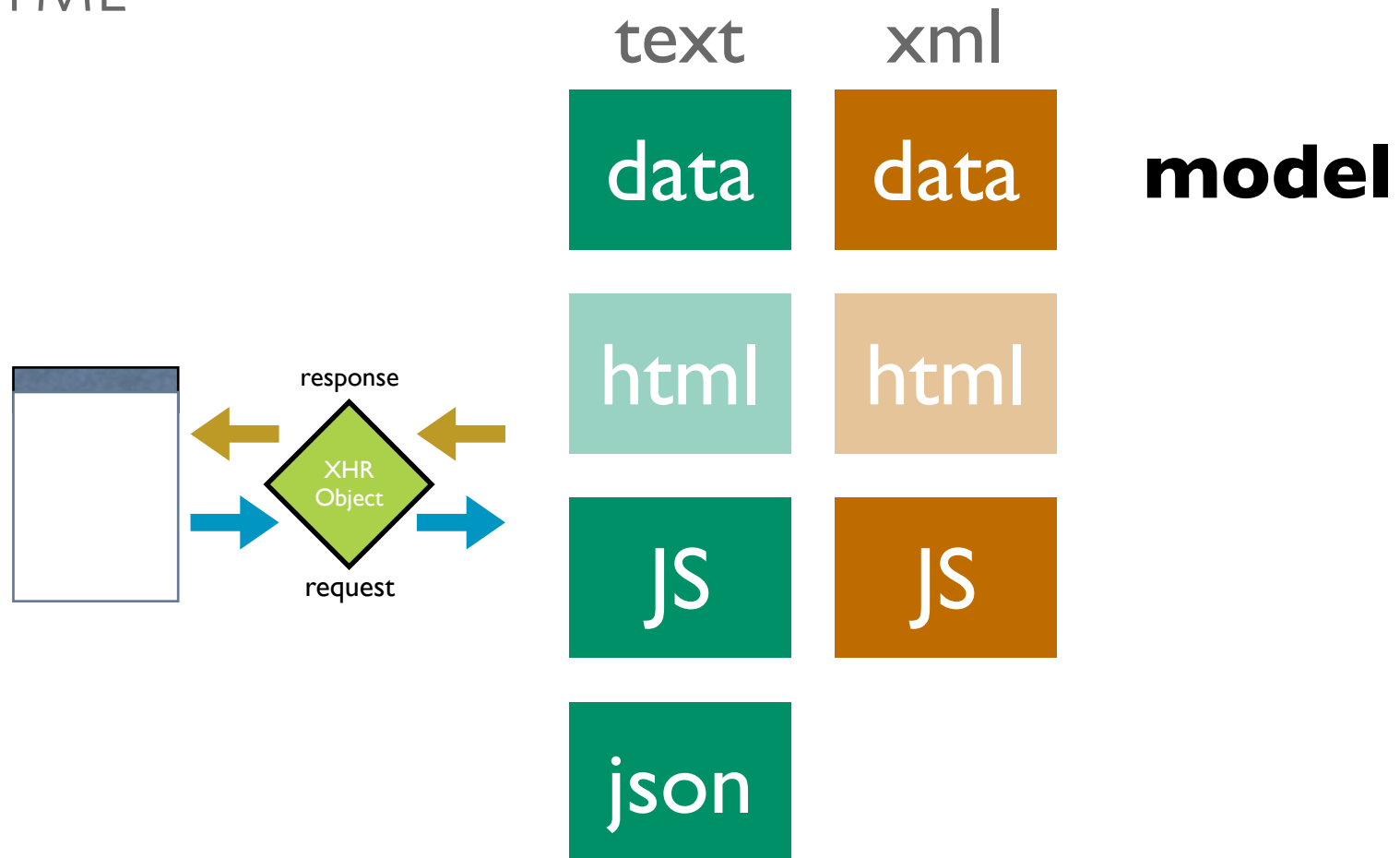
A message will be inserted below:



A message will be inserted below:



- Model data generated on server side
 - w3c DOM elements created from model data
 - Substitute model data into HTML strings and insert via innerHTML



```
// var helloArea = YAHOO.util.Dom.get("hello-area");  
var helloArea = document.getElementById("hello-area");  
var para = document.createElement("p");  
var paraText = document.createTextNode("Hello ");  
var em = document.createElement("em");  
var emText = document.createTextNode("World");  
helloArea.appendChild(para);  
para.appendChild(paraText);  
para.appendChild(em);  
em.appendChild(emText);
```

```
<p>A message will be inserted below:</p>  
<div id="hello-area" style="height:20px;width:220px;  
margin-top:8px;padding:4px;border:1px solid blue"></div>
```

A message will be inserted below:



A message will be inserted below:



JSON

- JavaScript supports several string based notations
 - Allows the basic types to be represented as string literals
 - Strings are easy to pass over the wire
- JSON (JavaScript Object Notation - json.org)

```
{"name": "Jack B. Nimble", "at large": true, "grade": "A", "level": 3}
```

name	Jack B. Nimble
at large	true
grade	A
level	3

```
["Sunday", "Monday", "Tuesday", "Wednesday",  
"Thursday", "Friday", "Saturday"]
```

array of 7 named days

```
[  
[0, -1, 0],  
[1, 0, 0],  
[0, 0, 1]  
]
```

3x3 array



- JSON's simple values are the same as used in JavaScript
- No restructuring is requested: JSON's structures are JavaScript!
- JSON's object is the JavaScript object
- JSON's array is the JavaScript array

- Parsing is simple, native

- Obtain responseText
- Parse the responseText

```
responseData = eval('(' + responseText + ')');
```

OR

```
responseData = JSON.parseJSON(responseText);
```

- <http://api.search.yahoo.com/WebSearchService/V1/webSearch?appid=YahooDemo&query=finances&start=1&results=1&output=json>

```
{
  "ResultSet":
  {
    "totalResultsAvailable": "69200000",
    "totalResultsReturned": "1",
    "firstResultPosition": "1",
    "Result":
    {
      "Title": "Yahoo! Finance",
      "Summary": "manage the market and your money with Yahoo! Finance. Includes stock market quotes, business news, mutual funds, online bill pay, banking tools, loans, insurance, retirement planning, and tax tips and advice.",
      "Url": "http://finance.yahoo.com/",
      "ClickUrl": "http://finance.yahoo.com/",
      "ModificationDate": "1137225600",
      "MimeType": "text/html"
    }
  }
}
```



Getting info from a JSON response

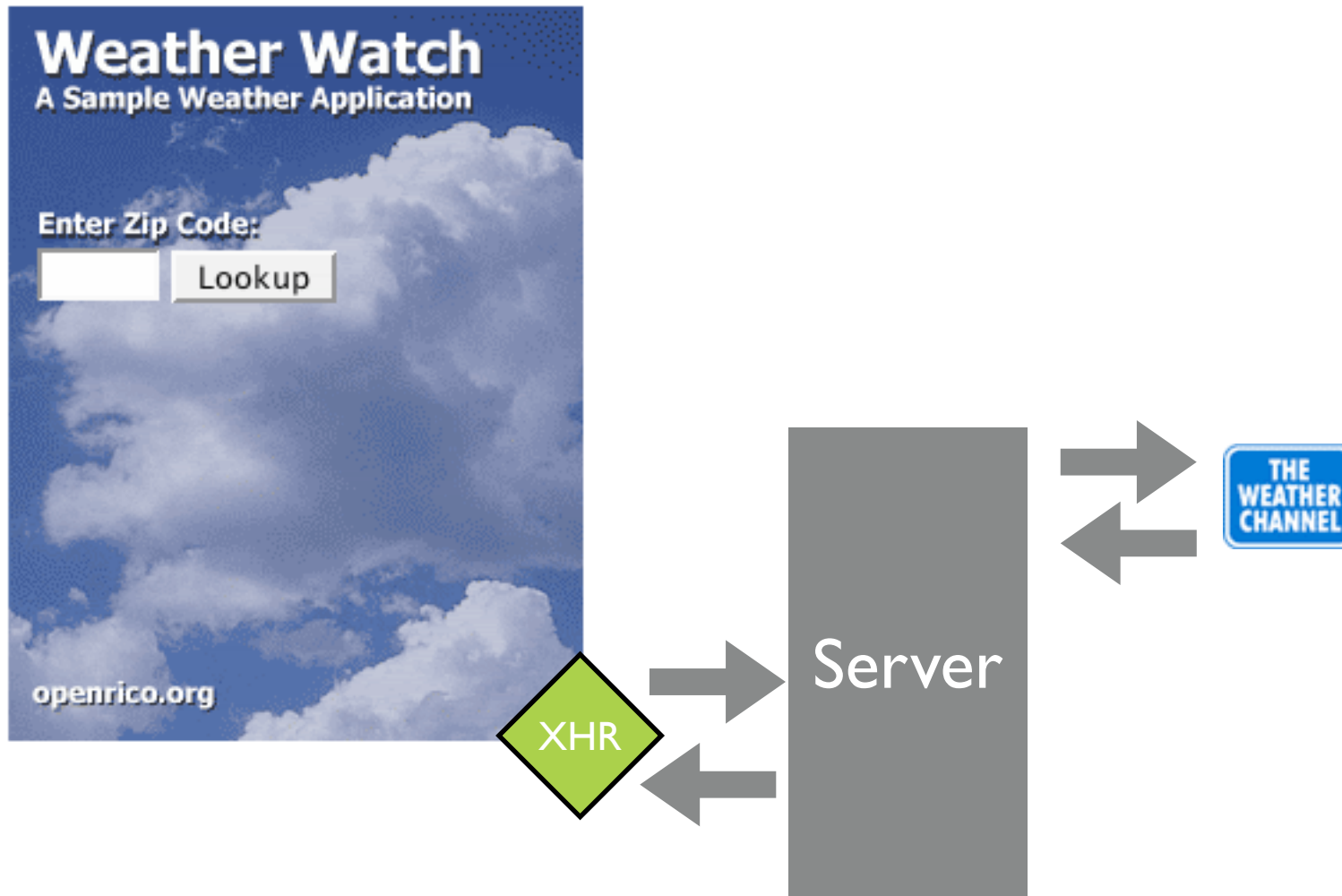
```
{
  "ResultSet":
  {
    "totalResultsAvailable": "69200000",
    "totalResultsReturned": "1",
    "firstResultPosition": "1",
    "Result":
    {
      "Title": "Yahoo! Finance",
      "Summary": "manage the market and your money with Yahoo! Finance. Includes stock
market quotes, business news, mutual funds, online bill pay, banking tools, loans, insurance,
retirement planning, and tax tips and advice.",
      "Url": "http://finance.yahoo.com/",
      "ClickUrl": "http://finance.yahoo.com/",
      "ModificationDate": "1137225600",
      "MimeType": "text/html"
    }
  }
}
```

```
responseData = JSON.parseJSON(responseText);
var resultsAvail = responseData.ResultSet.totalResultsAvailable;
var resultsReturned = responseData.ResultSet.totalResultsReturned;
var firstResultTitle = responseData.ResultSet.Result.Title;
var firstResultSummary = responseData.ResultSet.Result.Summary;
```



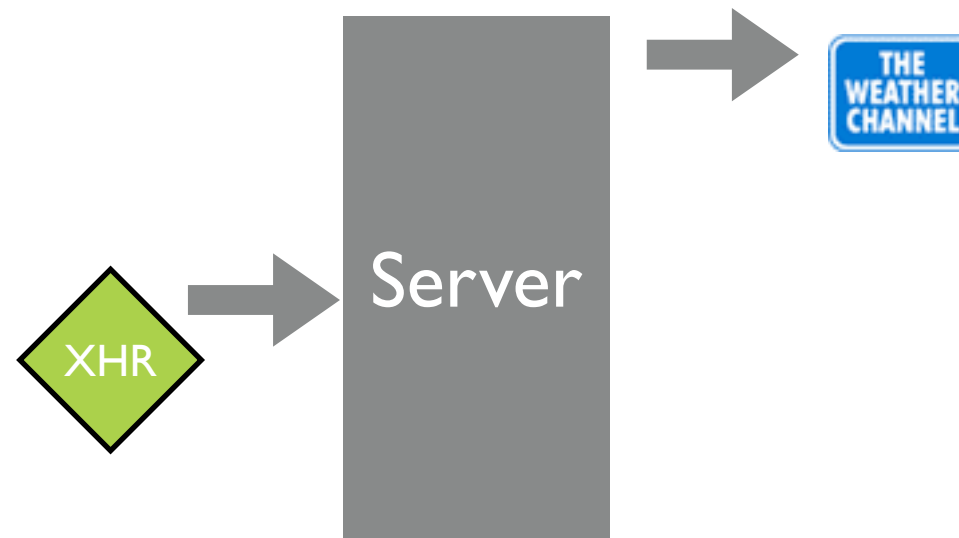
Building an Ajax Weather Widget

Using an XML Service



weather.com exposes its weather service via an URL that includes a partner key and license key

`http://xoap.weather.com/weather/local/95132?cc=*&link=xoap&prod=xoap&par=PARTNER_KEY&key=LICENSE_KEY`



```

<?xml version="1.0" encoding="ISO
-8859-1"?>
<weather ver="2.0">
  <head>
    <locale>en_US</locale>
    <form>MEDIUM</form>
    <ut>F</ut>
    <ud>mi</ud>
    <us>mph</us>
    <up>in</up>
    <ur>in</ur>
  </head>
  <loc id="95132">
    <dnam>San Jose, CA (95132)</
dnam>
    <tm>8:37 PM</tm>
    <lat>37.4</lat>
    <lon>-121.86</lon>
    <sunr>7:18 AM</sunr>
    <suns>5:20 PM</suns>
    <zone>-8</zone>
  </loc>

```

```

<cc>
  <lsup>1/21/06 7:53 PM PST</lsup>
  <obst>San Jose, CA</obst>
  <tmp>49</tmp>
  <flik>46</flik>
  <t>Mostly Cloudy</t>
  <icon>27</icon>
  <bar>
    <r>30.27</r>
    <d>rising</d>
  </bar>
  <wind>
    <s>7</s>
    <gust>N/A</gust>
    <d>350</d>
    <t>N</t>
  </wind>
  <hmid>80</hmid>
  <vis>10.0</vis>
  <uv>
    <i>0</i>
    <t>Low</t>
  </uv>
  <dewp>43</dewp>
  <moon>
    <icon>21</icon>
    <t>Waning Gibbous</t>
  </moon>
</cc>
</weather>

```

weather.com responds with an XML response that describes the weather's current conditions

Server



```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ajax-response>
  <response type="element" id="ccInfo">
    <div class="weatherTitle">San Jose, CA (95132)</div>
    <div>
      <span></img></span>
      <span class="weatherTemp">49&#176;F</span>
    </div>
    <div class="weatherDescr">Mostly Cloudy</div>
    <div>
      <table cellpadding="0" cellspacing="0" border="0">
        <tr class="weatherDetail">
          <td class="weatherDetailTitle">Humidity:</td>
          <td>80&#37;</td></tr>
        <tr class="weatherDetail">
          <td class="weatherDetailTitle">Barometer:</td>
          <td>30.27&quot;</td></tr>
        <tr class="weatherDetail">
          <td class="weatherDetailTitle">Wind:</td>
          <td>From N at 7 mph</td></tr>
        <tr class="weatherDetail">
          <td class="weatherDetailTitle"></td>
          <td>gusting to N/A mph</td></tr>
        <tr class="weatherDetail">
          <td class="weatherDetailTitle">Dewpoint:</td>
          <td>43&#176;F</td></tr>
        <tr class="weatherDetail">
          <td class="weatherDetailTitle">Heat Index:</td>
          <td>46&#176;F</td></tr>
      </table>
    </div>
  </response>
</ajax-response>
```

My server code translates the weather.com XML into a Rico/Prototype ajax-response. Notice the response is an **HTML code snippet**. The response is of type element, and mapped to **id="ccInfo"**



```
<div id="weather">
  <div id="current">
    <div id="currentHeader" class="accordionTabTitleBar">
      Current Conditions
    </div>
    <div class="weatherTabContentBox">
      <div class="weatherPanel" id="ccInfo">
      </div>
    </div>
  </div>
</div>
```

```
<div id="moon">
  <div id="moonHeader" class="accordionTabTitleBar">
    Moon
  </div>
  <div class="weatherTabContentBox">
    <div class="weatherPanel" id="moonInfo">
    </div>
  </div>
</div>
```

```
<div id="sun">
  <div id="sunHeader" class="accordionTabTitleBar">
    Sunlight Summary
  </div>
  <div class="weatherTabContentBox">
    <div class="weatherPanel" id="sunInfo">
    </div>
  </div>
</div>
</div>
```



```
<script>
onloads.push( accord );
function accord() {
  new Rico.Accordion( 'weather',
    {panelHeight:220, expandedBg:'#39497b'} );
}
</script>
```



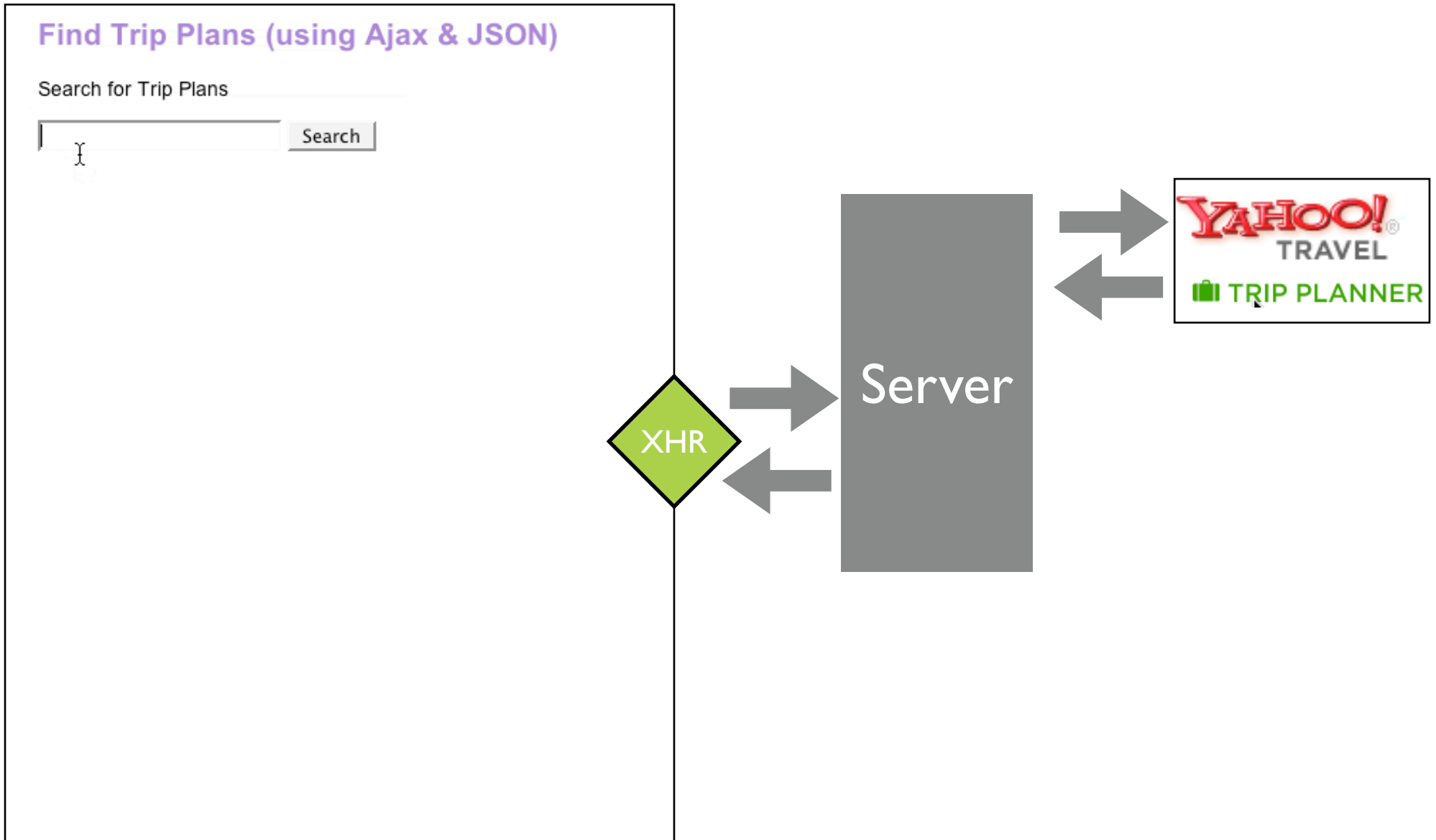
```
function registerAjaxStuff() {
    ajaxEngine.registerRequest( 'getWeatherInfo', 'ajax_weather_info' );
    ajaxEngine.registerAjaxElement( 'ccInfo' );
    ajaxEngine.registerAjaxElement( 'moonInfo' );
    ajaxEngine.registerAjaxElement( 'sunInfo' );
    ajaxEngine.registerAjaxElement( 'sponsoredLinks' );
    $('zip').onkeydown = handleEnterKey.bindAsEventListener($('zip'));
}
function getWeatherInfo() {
    $('checkanother').style.visibility='visible';
    new Rico.Effect.Position( $('zipinput'), 200, null, 100, 10);

    new Rico.Effect.FadeTo( 'frontdoor', 0, 100, 10,
        {complete:function() {$('frontdoor').style.display = 'none';}}
    );

    ajaxEngine.sendRequest( 'getWeatherInfo', "zip=" + $('zip').value);
}
function resetWeather() {
    $('zipinput').style.left = '12px';
    $('checkanother').style.visibility='hidden';

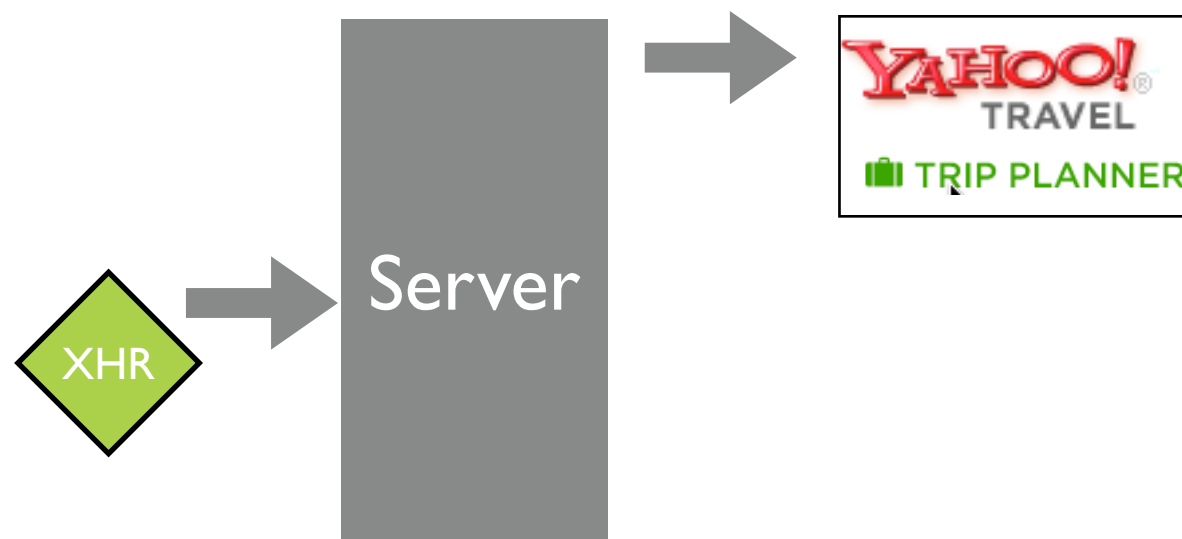
    $('frontdoor').style.display = ''
    $('zip').focus();
    new Rico.Effect.FadeTo( 'frontdoor', .99999, 100, 10, {complete:emptyContents});
}
```

JSON Trip Finder



Yahoo! Trip Planner exposes its service via an URL that requires a free partner key. Notice output=json.

`http://api.travel.yahoo.com/TripService/V1/tripSearch?appid=PARTNER_KEY&query=alaska&output=json`



```

{"ResultSet":{"firstResultPosition":
1,"totalResultsAvailable":"16","totalResultsReturned":
10,"Result":[

{"id":"351102","YahooID":"jk_95661","Title":"Cruise to Alaska
from Vancouver","Summary":"Things to do: Whistler Blackcomb -
Hiking,... Hotel: Hyatt Regency Vancouver, Howard
Jho...","Destinations":"Vancouver, Anchorage,
Vancouver, Ketchikan,
Se...", "CreateDate":"1130437928", "Duration": "10", "Image": {
{"Url":"http://us.i1.yimg.com/us.i1.yimg.com/
lp/cd/
100x100_cde24409e413d4d6da27953f6a
"Width":"66"}, "Geocode":{"Latitude":
-123.120499", "precision":"not available"}, "Url": "http://
travel.yahoo.com/trip/?pid=351102"}

{"id":"400333","YahooID":"brdway_grl","Title":"Alaska","Summa
ry":"Things to do: Moose's Tooth Pub and Pizzer...
Restaurant: Orso, Simon's & Seafort's
Salo...", "Destinations":"Anchorage", "CreateDate":"1134676973"
, "Duration":"10", "Image": {"Url":"http://
us.yimg.com/i/travel/tg/poi/ca/
100x100_cde605e44b47e29e731d3093b94afa37e.jpg", "Height":"75", "
Width":"149", "Geocode":{"Latitude":"61.190361", "Longitude":
:"not available"}, "Url":"http://
travel.yahoo.com/trip/?pid=400333&action=view"},
...
]}}
    
```

Y! service responds with a JSON text string, valid JavaScript object notation. This is passed directly back as the XHR result



```
<script>
function showResponse() {
    if (xhr.readyState == 4) {
        if (xhr.status == 200) {
            var jsonText = xhr.responseText;
            var helloArea = document.getElementById("helloArea");
            var theTrip = eval( '(' + jsonText + ')' );
            var tripPlanHTML = "";
            for(var i=0; i<theTrip.ResultSet.totalResultsReturned; i++) {
                var result = theTrip.ResultSet.Result[i];
                tripPlanHTML = tripPlanHTML + '<div style="padding:4px;
border:1px solid gray;width: '+result.Image.Width+';"></img></div>'+
'<div ><a href="' +result.Url+' "><span style="font-weight:bold;font-size:
18px;">' +result.Title+' </span></a></div><div>' +result.Summary+' </div><br/>';
            }
            helloArea.innerHTML=tripPlanHTML;
        } else {
            alert('There was a problem with the request.');
```



Trigger.

JavaScript Events

Trigger. JavaScript Events

- Ajax interactions are kicked off by event & timer triggers
- There are issues with event management within the browsers
- You **do** need to understand these implications to write good Ajax applications

First, the Events

onAbort	onBlur
onChange	onClick
onDbClick	onDragDrop
onError	onFocus
onKeyDown	onKeyPress
onKeyUp	onLoad
onMouseDown	onMouseMove
onMouseOut	onMouseOver
onMouseUp	onMove
onReset	onResize
onSelect	onSubmit
onUnload	

Problem in a Nutshell

- Different event models
- Timing Issues
- Confusing madness around the **this** pointer
- Browser incompatibilities
- Memory leak issues

- Element Attributes
 - `<element onclick="func()" >`
- Element Properties (function reference)
 - `element.onclick=func;`
- Event Binding
 - w3c: `addEventListener`, `removeEventListener`
 - IE: `attachEvent`, `detachEvent`

- Element Attributes

```
<a href="#" onclick="clickHandler(this)">
function clickHandler(anchorDOMElem) {
    // this == window --> window owns the function
    // anchorDOMElem was set to anchor DOM element
    this = anchorDOMElem; // fix this pointer
}
```

- Element Properties

```
myAnchor.onclick=clickHandler;
function clickHandler() {
    //this == anchorDOMElem --> anchorDOMElem owns the function
}
```

- Event Binding

```
function AnchorLink(anchorDOMElem) {
    this.anchorDOMElem = anchorDOMElem;
    this.anchorDOMElem.onclick = this.clickHandler;
    this.anchorDOMElem.anchorLinkObj = this;
}
AnchorLink.prototype.clickHandler = function() {
    // this == anchorDOMElem, not AnchorLink object
    // confusing since this normally refers to AnchorLink
    // grab our normal this
    anchorLinkObj = this.anchorLinkObj;
}
```

- Supports arbitrary number of event handlers
- Provides a way to remove events
- The model defines capture & bubble flow
- But, no way to get list of handlers
- And remember the browsers play differently:

Internet Explorer	Mozilla (FF), Safari, Opera [W3C]
<code>attachEvent()</code> <code>detachEvent()</code>	<code>addEventListener()</code> <code>removeEventListener()</code>
<code>this == window object</code>	<code>this == DOM event object</code>

- Attempting to attach events before the page is loaded will cause problems
- Do it on the onload

- IE's garbage collection does simple reference counting
- Attaching events and not removing them when finished will cause memory leaks
- Can use an Observer style pattern to cache event handlers and at the end clean them up

- Standardizes event binding

- Attach an event

```
var oElement = document.getElementById("elementid");  
function fnCallback(e) { alert("click"); }  
YAHOO.util.Event.addListener(oElement, "click", fnCallback);
```

- Attach multiple events

```
var ids = ["el1", "el2", "el3"]; // array can contain object references, element ids, or both  
function fnCallback(e) { alert(this.id); }  
YAHOO.util.Event.addListener(ids, "click", fnCallback);
```

- Handles automatic scope correction

- By default "this" refers to the DOM element that the event was attached to

- Can be overridden

- First parameter is the event object

- Pass arbitrary objects to event handler

- Automatic Listener Cleanup
- Custom Events
 - For custom objects or overriding events on widgets
- onAvailable event
 - Define a function that will execute as soon as an element is detected in the DOM (during load)

```
<script type="text/javascript">  
  
function TestObj(id) { YAHOO.util.Event.onAvailable(id, this.handleOnAvailable, this); }  
  
TestObj.prototype.handleOnAvailable = function(me) { alert(this.id + " is available"); }  
  
var obj = new TestObj("myelementid"); </script>  
  
<div id="myelementid">my element</div>
```




Update.

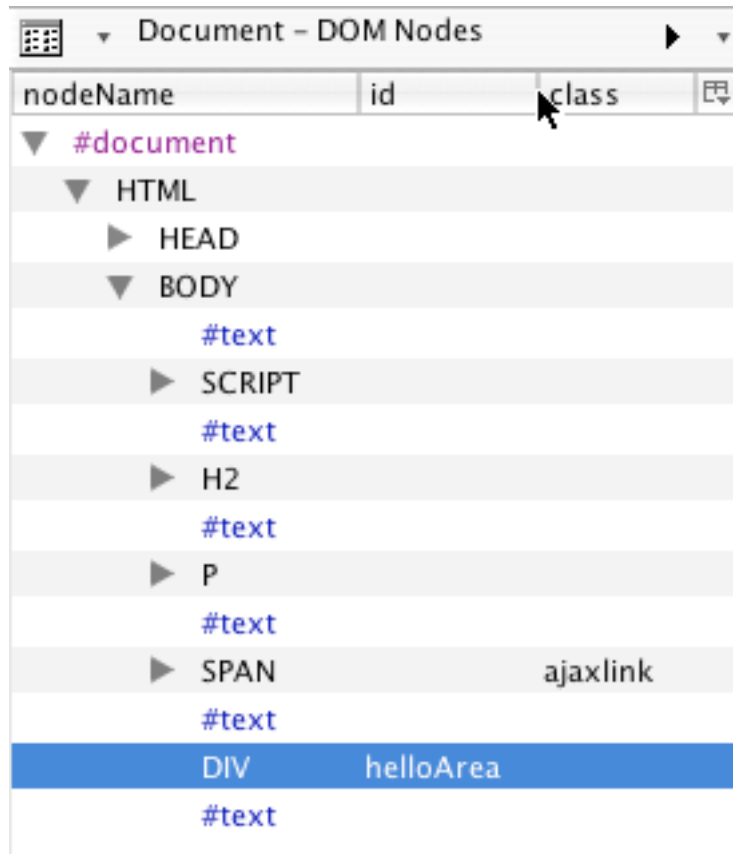
The DOM

Update. The DOM

- Browsers represent the user interface as a set of objects (or elements)
- Since a page has a hierarchy of containment, each of these elements are related as parent-child or siblings
- This takes the shape of a tree
- The tree of elements is called the *document object model* or DOM. Specifically the Browser DOM.
- Any change to the DOM structure or a DOM element is reflected immediately on the web page

DOM Example

- Represented as a tree of nodes



nodeName	id	class
#document		
HTML		
HEAD		
BODY		
#text		
SCRIPT		
#text		
H2		
#text		
P		
#text		
SPAN		ajaxlink
DIV	helloArea	
#text		

Ajax Hello World

Clicking the link below will use XHR to fetch the data and then show the result in the box below.

[Make an ajax request for data](#)

Using the DOM

- JavaScript is the DOM manipulator
- Use it to
 - Find DOM elements
 - Add new elements
 - Remove elements
 - Modify element attributes
 - Position elements
 - Style elements

Finding DOM Elements

- `document.getElementById`
 - Prototype library shortcut: `$("#idName")` or `$(id)`
 - Yahoo! library shortcuts:
 - `YAHOO.util.Dom.get("idName")`
 - `getElementByClassName`
- `parentNode`
- `childNodes`
 - Yahoo! library has ways to find ancestors

DOM Manipulation

- Creating new interface elements
 - `innerHTML`, `createElement()`, `createTextNode()`, `appendChild()`
- Changing element styles
 - Visual attributes
 - Geometry
- Y!UI positioning, region detection
- Y!UI classname & style normalization

```
<h2>Ajax Hello World</h2>
```

```
<p>Clicking the link below will use XHR to fetch the data and  
then show the result in the box below.</p>
```

```
<span class="ajaxlink" onclick="makeRequest('response.jsp')">  
Make an ajax request for data  
</span>
```

```
<div id="helloArea"></div>
```

Ajax Hello World

Clicking the link below will use XHR to
fetch the data and then show the
result in the box below.

[Make an ajax request for data](#)

This data was brought to you by
Ajax!

```
var helloArea = document.getElementById("helloArea");  
helloArea.innerHTML=rootNode.firstChild.data;
```

Keeping it Clean

- Separate presentation style from content with CSS
 - Supports degradability
 - Supports accessibility
 - Simplifies maintenance
- <http://www.mezzoblue.com/css/cribsheet/>

- Avoid these elements:
 - `b`, `big`, `hr`, `i`, `small`, `sub`, `sup`, `tt`
 - `basefont`, `center`, `dir`, `font`, `isindex`, `menu`, `s`, `strike`, `u`, `tfoot`
 - `br` is ok for content breaks; not layout
- Make your HTML meaningful
 - List structures as `ul`, `li`
 - Tree structures as nested `ul`, `li`
 - Containers as `div`
- Use css selectors (contextual)
 - Reduces the number of classes
 - Avoid inline styles

```
<div id="weather">
  <div id="current">
    <div id="currentHeader" class="accordionTabTitleBar">
      Current Conditions
    </div>
    <div class="weatherTabContentBox">
      <div class="weatherPanel" id="ccInfo">
      </div>
    </div>
  </div>

  <div id="moon">
    <div id="moonHeader" class="accordionTabTitleBar">
      Moon
    </div>
    <div class="weatherTabContentBox">
      <div class="weatherPanel" id="moonInfo">
      </div>
    </div>
  </div>

  <div id="sun">
    <div id="sunHeader" class="accordionTabTitleBar">
      Sunlight Summary
    </div>
    <div class="weatherTabContentBox">
      <div class="weatherPanel" id="sunInfo">
      </div>
    </div>
  </div>
</div>
```



```
new Rico.Accordion( 'weather', {panelHeight:220, expandedBg:'#39497b'}
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head><title>Make this page more semantic</title></head>
<body>
<div class="page-title">Make this page more semantic</div>
<div>
<div class="module-title">Module Title</div>

<span class="time"><i>12:40pm Wed Mar 29</i></span></i>
<div class="para">Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod
tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci </
div>
</div>
<p><div><i>Web users ultimately want to get at data quickly and easily. They don't care as much about
attractive sites and pretty design.</i></div> says <b>Tim Berner's Lee</b>, inventor of the World Wide Web,
although I'm sure many visual designers would beg to differ. <u>"The web isn't print!"</u> - Chanel.</p>

<table>
  <tr>
    <td class="heading">column 1</td>
    <td class="heading">column 2</td>
    <td class="heading">column 3</td>
  </tr>
  <tr>
    <td class="data">data cell 1</td>
    <td class="data">data cell 2</td>
    <td class="data">data cell 3</td>
  </tr>
</table>
</body>
</html>
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head><title>Make this page more semantic</title></head>
<body>
<h1>Make this page more semantic</h1>

<h2>Module Title</h2>

<em class="time" title="time stamp">12:40pm Wed Mar 29</em>
<blockquote>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt
ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci </blockquote>
<blockquote cite="Tim Berner's Lee">

<p>&#39;Web users ultimately want to get at data quickly and easily. They don't care as much about attractive
sites and pretty design.&#39;</p></blockquote> says <cite>Tim Berners Lee</cite>, inventor of the World Wide
Web, although I'm sure many visual designers would beg to differ. <q>The web isn't print!</q> -
<cite>Chanel</cite>.

<table summary="this month's numbers">
  <caption>Q1 Financials</caption>
  <thead> <tr>
    <th scope="col">column 1</th>
    <th scope="col">column 2</th>
    <th scope="col">column 3</th>
  </tr> </thead>
  <tbody>
    <td class="data">data cell 1</td>
    <td class="data">data cell 2</td>
    <td class="data">data cell 3</td>
  </tr>
</tbody>
</table>
</body>
</html>
```

```
...
<script type="text/javascript">
YAHOO.example.DDApp = function() {
    var dd;
    return {
        init: function() {
            var dd = new YAHOO.util.DD("dragDiv");
        }
    }
} ();
YAHOO.util.Event.addListener(window, "load", YAHOO.example.DDApp.init);
</script>

<style type="text/css">
#dragDiv {
    background:url(images/macosexlogo.gif) 0 0 no-repeat;
    height:100px;
    width:70px;
}
</style>
</head>

<body>
<h2>Drag and Drop</h2>
<p>The Mac logo is draggable</p>

<div id="dragDiv"></div>

</body>
...
```

Drag and Drop

The Mac logo is draggable



```

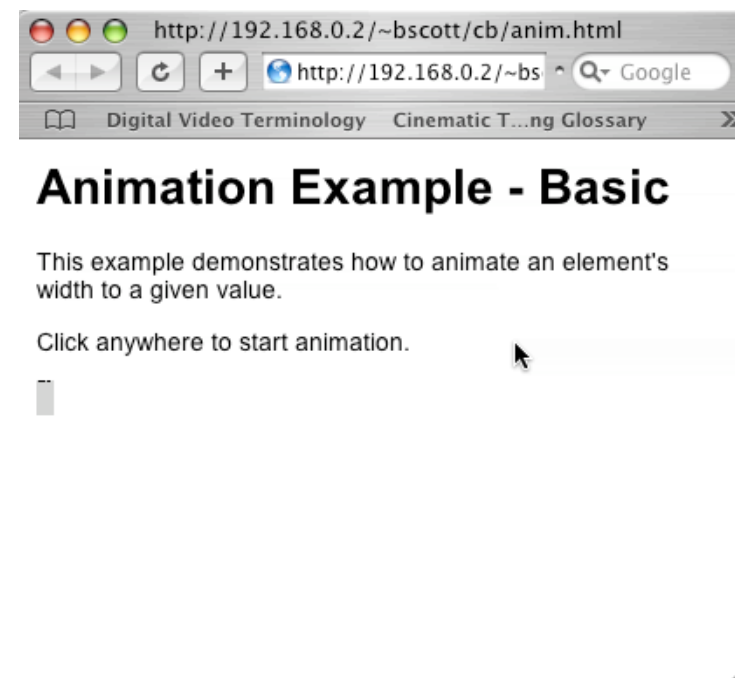
<html>
<head>
<style>
#anim { background:#ccc;width:10px;height:20px;font-size:10%;}
</style>
<script type="text/javascript">
YAHOO.namespace('example.anim');
YAHOO.example.anim.init = function() {
    var myAnim = new YAHOO.util.Anim('anim', {
        width: {to: 200},
        height: {to: 150},
        fontSize: {from: 20, to: 120, unit: '%'},
        opacity: {from: 0.25, to: 1.0 }},
        1,
        YAHOO.util.Easing.easeOut);

    var animate = function(e) {
        myAnim.animate();
        return false;
    }
    YAHOO.util.Event.addListener(document, 'click', animate);
}
YAHOO.util.Event.addListener(window, 'load', YAHOO.example.anim.init);
</script>
</head>
<body>
<h1>Animation Example - Basic</h1>
<p>This example demonstrates how to animate an element's width to a given value.</p>
<p>Click anywhere to start animation.</p>

<div id="anim">Lorem ipsum dolor</div>

</body>
</html>

```



Learning Ajax

- Learning Ajax, means understanding
 - **Triggers** (event or timer)
 - **Operations** (ajax xhr)
 - **Updating** (dom)

*Triggering
Events &
Timers*

*Using XHR
for Ajax
Operations*

*Updating
the DOM*

Yahoo! UI Library

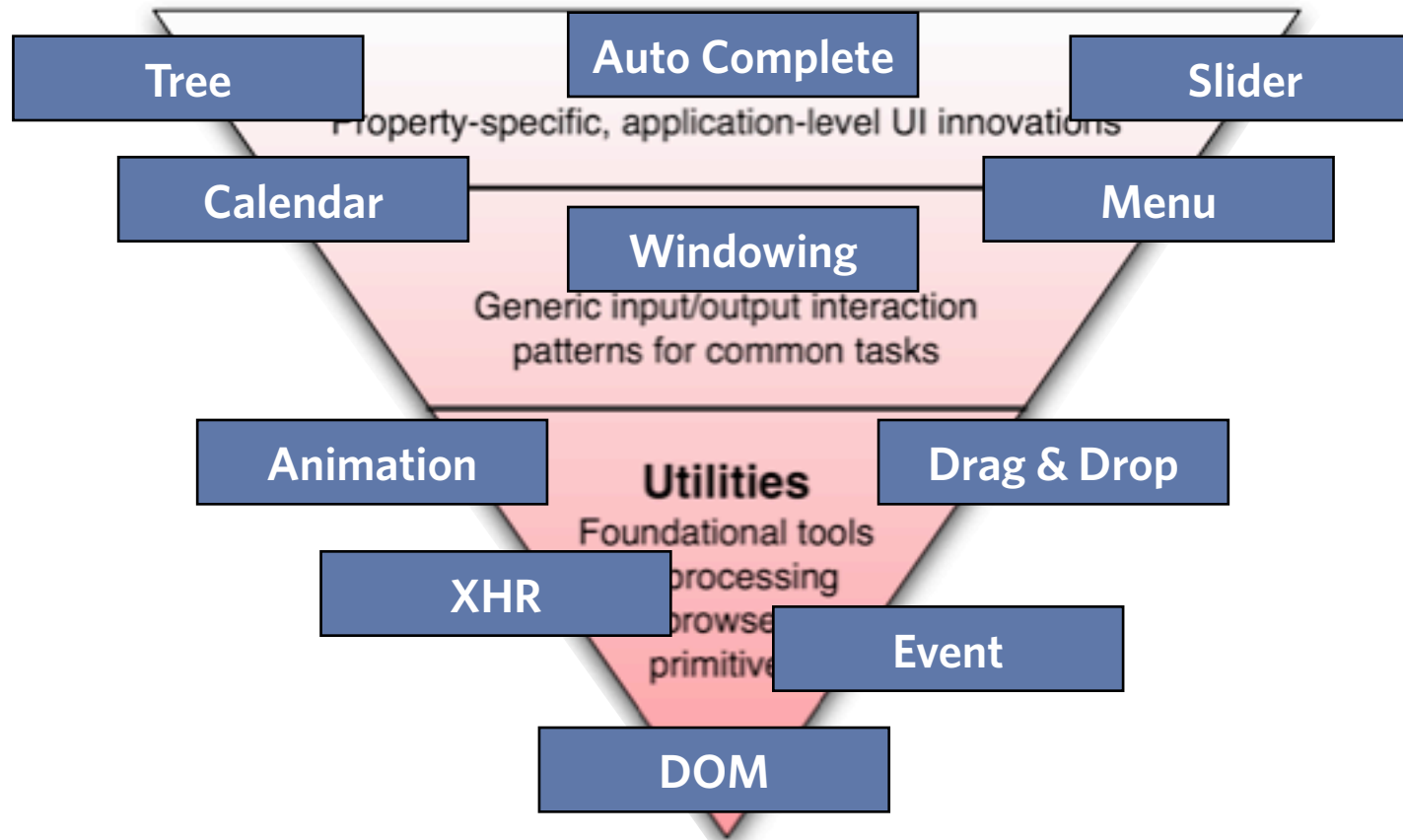
Designing and Building Ajax Applications with Y!UI Library



Bill W. Scott, Y! Ajax Evangelist

bscott@yahoo-inc.com

Yahoo! UI Library



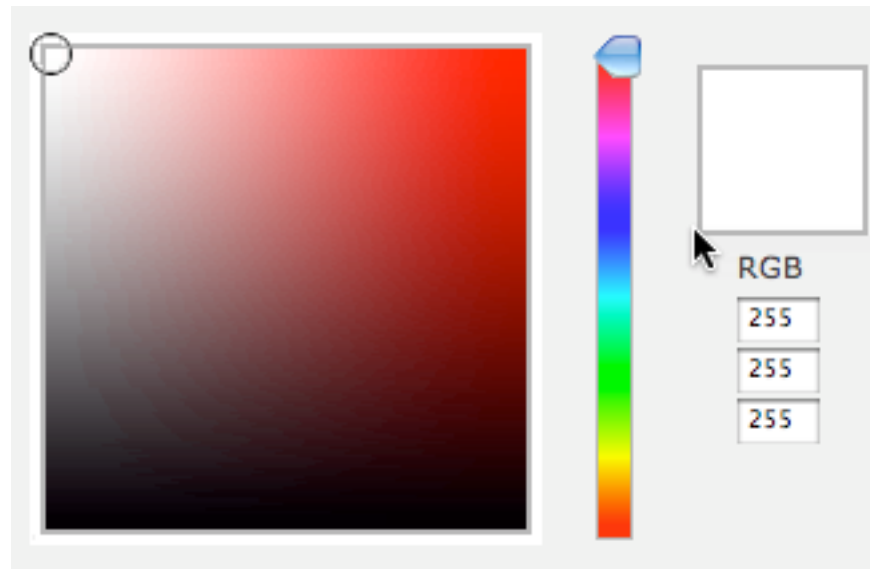
Yahoo! UI Library

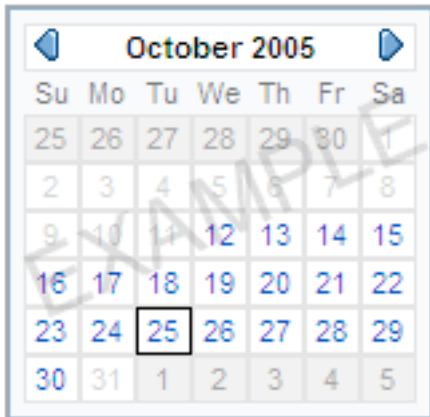
- Normalization (Event Utility)
- Messaging (Connection Manager for AJAX)
- Direct Representation (Animation and Position)
- Direct Manipulation (Drag and Drop)
- Rich Interactive Widgets (Slider, TreeView, Calendar, AutoComplete)



```
<div>  
  <div id="horizHandleDiv">  
  </div>  
</div>
```

```
var mySlider = ygSlider.getHorizSlider("horizBGDiv",  
    "horizHandleDiv", 100, 100);  
mySlider.onChange = function(offsetFromStart) {  
  document.getElementById("horizVal").value =  
    offsetFromStart;  
}
```





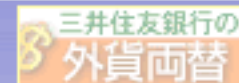
```
<script>
  var myCal;
  function init() {
    myCal = new ygCalendar("myCal", "myCalContainer");
    myCal.render();
  }
</script>
<div id="myCalContainer"></div>
```



こんにちは、ゲストさん

[ログイン](#)

Yahoo!トラベル 海外航空券予約



[トップ](#) > [海外航空券予約](#) > [アメリカ、カナダ](#) > [アメリカ合衆国](#) > [ヒューストン](#)

[PR] [癒しのニュージーランド旅行で豪華賞品&ボーナスマイルをゲット!](#)

出発日別最安料金検索 出発日（当月を含む6か月以内）を選択すると、選択された出発日の前後3日間の最安料金を検索できます。

2005年 10月 26日

このリストから条件を指定して絞り込む

・ 出発地:

・ 航空会社:

・ 料金: 円以上 円未満

・ 有効期間: 日以上

・ 座席クラス: エコノミー ビジネス ファースト

・ 旅行形態: 往復（帰国便変更不可） 往復（帰国便を現地で変更可能） 片道

・ その他の条件: 直行便 カード決済可能 子ども料金あり オンライン空席照会、予約可能



flickr Calendar Widget

Photos taken in May 2005

Su	Mo	Tu	We	Th	Fr	Sa
 1	 2	 3	 4	 5	 6	 7
 8	 9	 10	 11	 12	 13	 14
 15	 16	 17	 18	 19	 20	 21
 22	 23	 24	 25	 26	 27	 28
 29	 30	 31	1	2	3	4
5	6	7	8	9	10	11











Yahoo! UI Auto Complete

To: [Show BCC](#)

Cc:

Subject: [Plain Text](#)

Times New Roman 12 **B** *I* U         .

APPLICATION NEEDS TO

▼ CALL ATTENTION

- ▶ Help
- ▶ Invitation
- ▶ Jump to Details
- ▶ Progress Indication
- ▶ Transitions
- ▶ Zoom

▶ IMPROVE READABILITY

▶ GROUP RELATED ITEMS

▶ ORGANIZE SCREEN/PAGE

[Click here for more in EXAMPLE](#)

Follow this link to read more about the Tooltip control

Sprockets



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis in risus. Vestibulum a nibh lincidunt elit lacus laoreet. In pede metus, tristique non, consectetur a, vestibulum vel, enim. Curabitur neque magna, pulvinar sit amet, lobortis viverra, porta non, dui. Donec tempor, arcu quis interdum pellentesque, dui dolor gravida pede, in laculis justo tortor venenatis ligula. Cras non pede vel risus egestas faucibus. Nam pulvinar lobortis risi. In vehicula venenatis sapien.

Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Etiam et risus ut est adipiscing congue. Praesent ut justo. Proin hendrerit urna eget eros. Curabitur condimentum tempor diam. Pellentesque tortor odio, sollicitudin eget, pellentesque ut, ultramcoper a, justo. Nam tellus pede, fermentum ac, sollicitudin sed, mollis vel, turpis. Fusce pulvinar nibh et erat. Praesent purus nibh, fermentum ac, imperdiet vel, aliquet eu, mi. Praesent non nisi ac mauris tempor uterque. Nulla mollis tellus porta leo. Nunc accumsan dapibus tellus. Nunc nec nunc.

Some footer text

EXAMPLE

Please enter your information

Please enter your personal contact information:

First Name:

Last Name:

E-mail:

State:

Radio buttons: 1 2

Single checkbox: 1

Text area:

Multi checkbox: 1 2

EXAMPLE

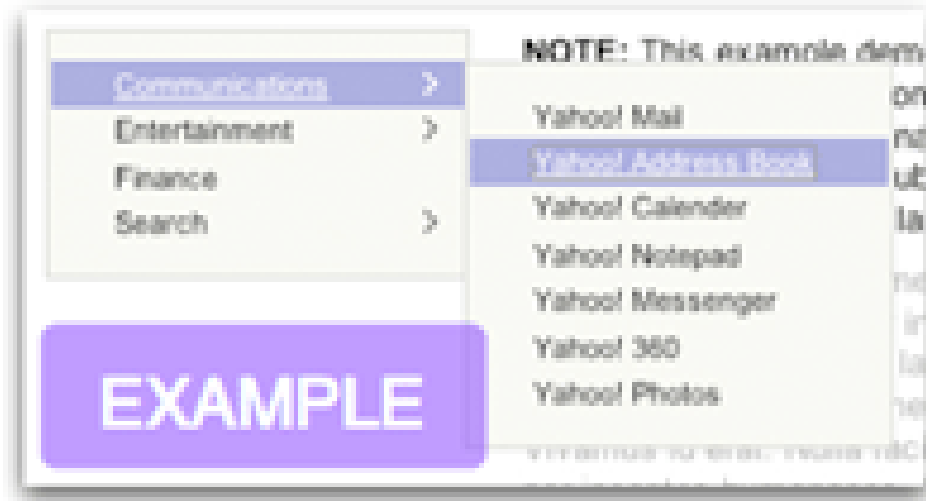
```
to instantiate this SimpleDialog example looks like this: dlg =  
.widget.SimpleDialog("dlg", { width: "20em",  
titleEffect: YAHOO.widget.ContainerEffect.PADDED,  
center: true, modal: true, close: true, closeText: "Close" });  
.exampleSimpleDialog.dlg.cfg.setProperty("title", "Warning!");  
.exampleSimpleDialog.dlg.cfg.setProperty("message", "Are you sure you want to do this?");  
.exampleSimpleDialog.dlg.cfg.setProperty("buttons", [{"label": "OK", "type": "button"}, {"label": "Cancel", "type": "button"}]);  
.exampleSimpleDialog.dlg.show();
```

Warning!

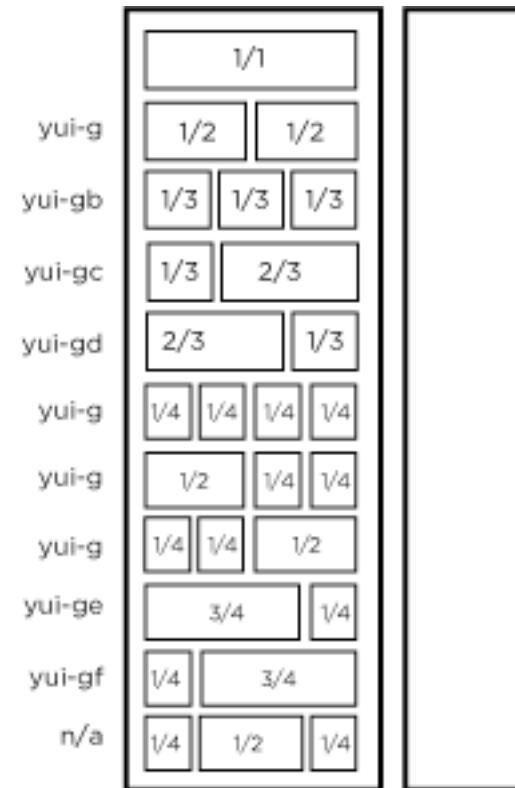
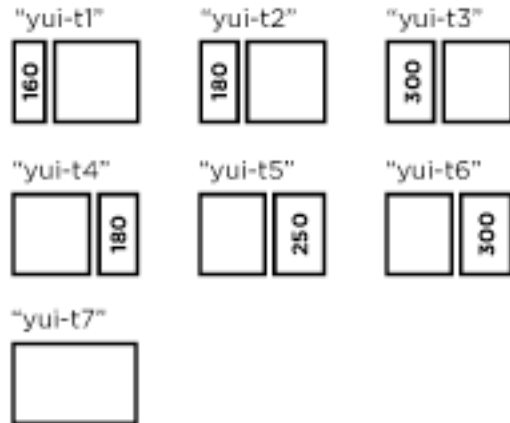
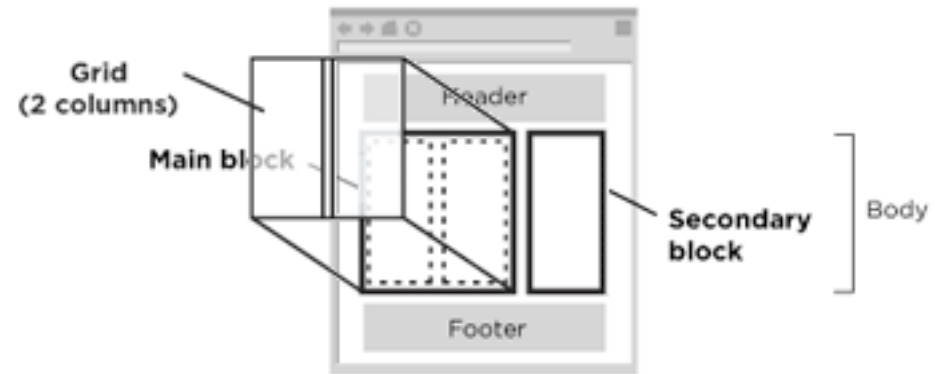
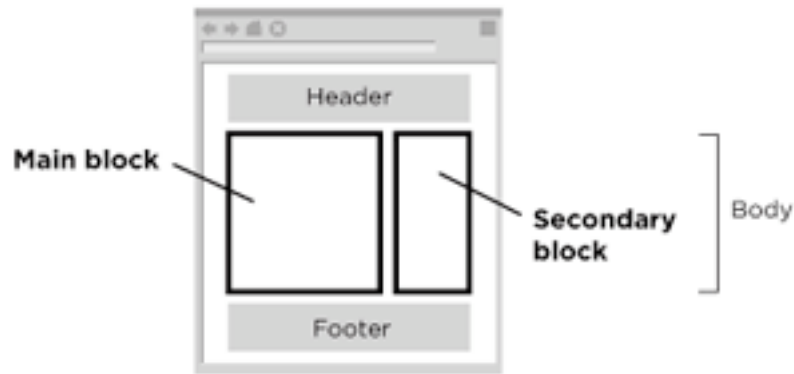
Are you sure you want to do this?

OK Cancel

EXAMPLE



Y!UI Page Grids



- <http://billwscott.com/proto/>
- <http://billwscott.com/carousel/>

Advanced Topics

Problems and Challenges with Building Ajax Applications



Bill W. Scott, Y! Ajax Evangelist

bscott@yahoo-inc.com

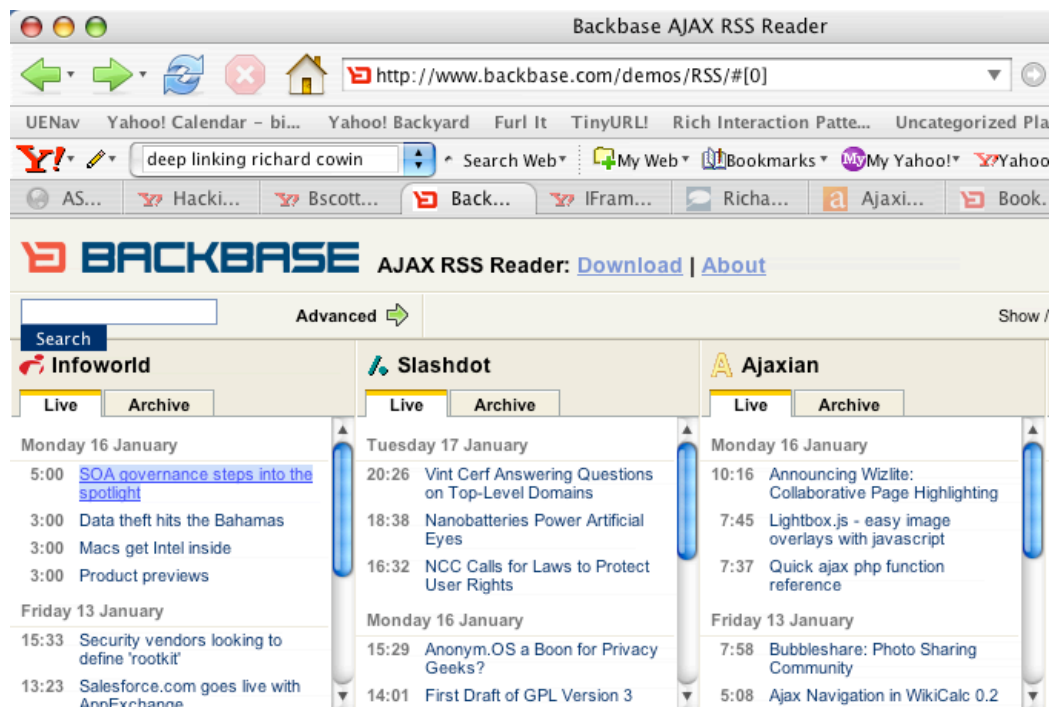
How Ajax Changes Things

Classic Web	Ajax/Web 2.0	Problems/Challenges
Page to page navigation	Micro content	Back button, SEO, bookmarking, accessibility, security
URL/Link = User's location	Application state	Back button, SEO, bookmarking, accessibility
Browser history	Application history	Back button, bookmarking, accessibility
Back button = Undo	Is unpredictable	Back button, bookmarking, accessibility
Little JavaScript	More JavaScript	Accessibility, degradability, security, memory, performance, debug, obfuscation, error handling
Document model	Application model	Back button, SEO, bookmarking, accessibility
Static content	Dynamic content	SEO, bookmarking, accessibility
Course-grained events	Micro states	Back button
Synchronous	Asynchronous	Error handling
Browser chrome	Application controls	Back button, bookmarking, accessibility
Page Services	Web Services	Security, XML vs. JSON

Back Button

- Problem: Ajax application state changes, URL does not. No browser history of state changes/navigation
- What does the user expect?
 - Often confused with Undo
 - True context view changes should be part of history
 - Navigation tabs; but not content tabs?
 - Steps in a process
 - Tree selection when views change
 - Impact of tabbed browsers?

- URL hash, fragment identifier (**http://a.com#loc** does not trigger a page reload
- Fragment identifier (string after '#') can be used to record state changes in the URL



- Yahoo! Maps Beta also uses this technique
- Bottom line: tricky to implement
 - Dojo, Backbase provide direct support
 - One approach:
<http://www.contentwithstyle.co.uk/Articles/38/fixing-the-back-button-and-enabling-bookmarking-for-ajax-apps>
 - All links have fragment
 - Clicking link changes URL, not page
 - Timer monitors window.location.href & updates
 - RSH - Really Simple History.
<http://www.onjava.com/pub/a/onjava/2005/10/26/ajax-handling-bookmarks-and-back-button.html>

- Technique: use iframes to control browser history for recording state changes
- Tricky issues abound
 - Fragile across browsers
 - onload issues
 - audible transition (on IE if sound enabled)
- Bottom line: problematic

Search Engine Optimization (Deep Linking)

- All the content may not be statically available for search engine crawlers
 - Won't find content to index your pages correctly
- Possible solutions
 - **Lightweight Indexing:** leverage existing tags such as meta, title and h1
 - **Extra Links:** extra links are placed on the site.
 - **Secondary Site:** a secondary site is fully accessible to the search engine. (See degraded experience)

source: backbase.com



Bookmarking

- Since we have broken the history and URL paradigm, bookmarking become problematic
- What does the user expect?
 - Do they expect to bookmark application state? content viewed?
 - Desktop apps often support bookmarking. It is always content based.

- Allow the user to save a bookmark at an interesting moment in an Ajax application
- Perhaps dynamically generate a link for bookmarking
- The URL generated for the bookmark is sufficient to restore the state

- Google Maps
 - Link is generated on each new map address
 - Link contains URL parameters to return to the page

 [Print](#)  [Email](#)  [Link to this page](#)

- OpenRico LiveGrid

http://richardcowin.typepad.com/blog/2005/07/there_has_been_.html

Listing movies

#	Title	Year
1	Mr and Mrs Smith	0
2	Shichinin no samurai	1954
3	The Lord of the Rings: T	2003
4	Buono, y il brutto, il cattivo, Il	1966
5	The Lord of the Rings: The Fellowship of the Ring	2001
6	Star Wars	1977
7	The Lord of the Rings: The Two Towers	2002
8	Star Wars: Episode V - The Empire Strikes Back	1980
9	Raiders of the Lost Ark	1981
10	Apocalypse Now	1979

Name: Listing movies 61 - 70 of 894

Create in: Bookmarks

Cancel Add

<http://openrico.org/rico/livegrid.page?>

[data_grid_index=60&data_grid_sort_col=rating&data_grid_sort_dir=ASC](http://openrico.org/rico/livegrid.page?data_grid_index=60&data_grid_sort_col=rating&data_grid_sort_dir=ASC)

Accessibility

DHTML Provides	Accessibility Expects	Problem
JavaScript enabled markup, new user interface controls	Simple markup	Markup has more meaning than expected. How does the assistive technology understand this is a tree control? How does it understand the state of a control?
Dynamic pages & content	Fairly static pages	How do I announce dynamic content?
Weak support for keyboard navigation	Keyboard navigation	Sight-impaired users will not be using rich interactions; they will use the keyboard (or another device that simulates the keyboard) But how to tab key, arrow key, select items with the keyboard?

- IBM/Mozilla Accessible DHTML API/Spec
 - Direct support for keyboard traversal
 - Supports tab key to a container, arrow keys to navigate inside the container and enter key to select item
 - Setting tabindex=-1 allows focus to be set on a object without it being in the tab order
 - A way to add metadata to HTML markup for assistive technologies to understand:
 - Widget roles, properties, values and events
 - Working with assistive technology vendors to make them aware of this API (funded development)
- Microsoft's plans

tabindex value	Purpose	Tab key navigable?
Not set	Accept default behavior	Default behavior Form elements
tabindex = “-1”	For child elements (nodes in a tree)	No. You must set the focus with JavaScript
tabindex = “0”	To tab in HTML code order	Tabbing moves you field to field in the order they are added to the page
tabindex > “0”	To specify an exact order for each field	Value describes the tab order

- Roles

```
<span tabindex="0" xhtml2:role="wairole:checkbox"
  property:checked="true"
  onkeydown="return checkBoxEvent(event);"
  onclick="return checkBoxEvent(event);">
```

Any checkbox label

```
</span>
```

- Provides clues to assistive technologies

Progressive Enhancement

- Degradability - Managing the user experience as you move down in device/browser capability
- At Yahoo! we grade the browser by experience we will provide (A, B, X grade)
 - A-grade experience (majority of our users; greatest visual fidelity and richest interactions)
 - B-grade experience (almost A-grade, but either they are on bleeding edge, low adoption or low economic value). Not considered in development or testing
 - X-grade experience (baseline of support; content but little style or interaction). Crawlers and old browsers get this experience.



Graded Browser Support @ Yahoo!

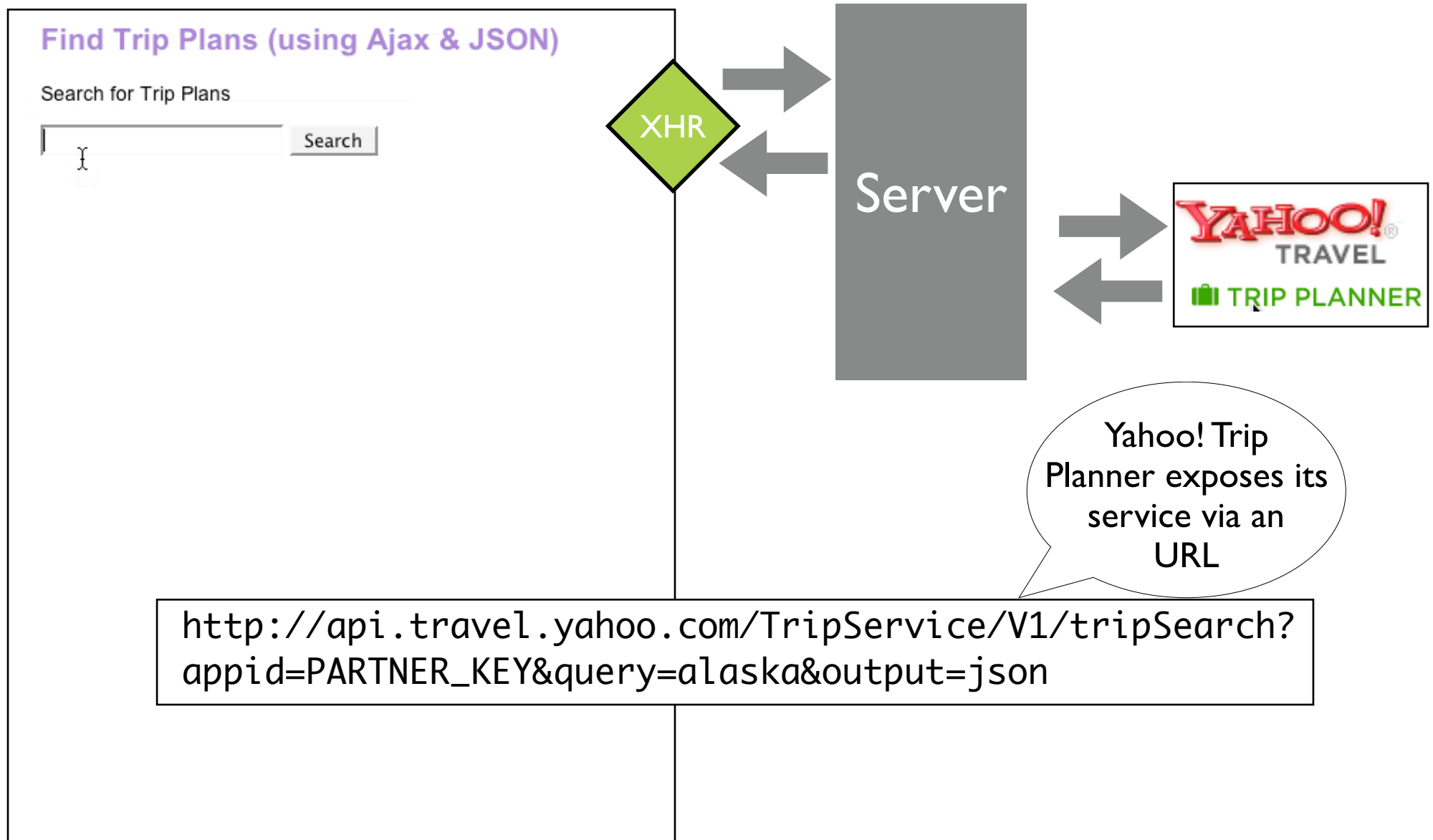
	<u>Win 98</u>	<u>Win 2000</u>	<u>Win XP</u>	<u>Mac 10.0</u>	<u>Mac 10.2</u>	<u>Mac 10.3</u>	<u>Mac 10.3.x</u>	<u>Mac 10.4</u>
<u>IE 7.0</u>	n/a	n/a	A-grade	n/a	n/a	n/a	n/a	n/a
<u>IE 6.0</u>	A-grade	A-grade	A-grade	n/a	n/a	n/a	n/a	n/a
<u>IE 5.5</u>	A-grade	A-grade	n/a	n/a	n/a	n/a	n/a	n/a
<u>IE 5.0</u>	C-grade	C-grade	n/a	C-grade	C-grade	C-grade	C-grade	C-grade
<u>Netscape 8.0</u>	X-grade	X-grade	A-grade	n/a	n/a	n/a	n/a	n/a
<u>Firefox 1.5</u>	A-grade	A-grade	A-grade	A-grade	A-grade	A-grade	A-grade	A-grade
<u>Firefox 1.0.7</u>	A-grade	A-grade	A-grade	A-grade	A-grade	A-grade	A-grade	A-grade
<u>Mozilla 1.7.12</u>	X-grade	X-grade	A-grade	X-grade	X-grade	X-grade	X-grade	X-grade
<u>Opera 8.5</u>	X-grade	X-grade	A-grade	C-grade	C-grade	C-grade	X-grade	X-grade
<u>Safari 1.0</u>	n/a	n/a	n/a	X-grade	n/a	n/a	n/a	n/a
<u>Safari 1.1</u>	n/a	n/a	n/a	X-grade	X-grade	n/a	n/a	n/a
<u>Safari 1.2</u>	n/a	n/a	n/a	X-grade	X-grade	X-grade	n/a	n/a
<u>Safari 1.3</u>	n/a	n/a	n/a	n/a	n/a	X-grade	A-grade	n/a
<u>Safari 2.0</u>	n/a	n/a	n/a	n/a	n/a	n/a	n/a	A-grade



- Pre-emptive nag bar
- Semantic Markup
 - What's good for accessibility is good for degradability
- Design Issues
 - What is the experience for C-grade browsers?
 - What is the experience for non-browsers?

Web Services

- Web 2.0 - Web as a Platform. Lots of web services!
-  **DEVELOPER NETWORK**
 - del.icio.us, Finance, flickr, HotJobs, Maps, Merchant Solutions, Music, Design Patterns, RSS Feeds, Search, Search Marketing, Shopping, Travel, Traffic, upcoming.org, UI library, weather, webjay, widgets, JS Developer Center, PHP Developer Center
-  **Google Web APIs (beta)**
 - maps, search, desktop, sitemaps, adwords, finance, calendar



- JSON means
 - Trivial parsing
 - Faster than XML
 - Friendlier for developer
 - Runs everywhere
 - It's JavaScript, simpler programming model
 - Very stable... never will change!
 - Strong community, wide acceptance
- Yahoo! behind it

Security: Same Site Rule

- The domain of the URL request destination must be same as one that serves up page containing script
- Why? XHR requests to a different site would carry cookies. Cookies would spoof authentication.
- Solutions
 - Proxies
 - `<script>` hack
 - Other remote scripting tricks
 - XMLHttpRequest (coming)

- I want to access multiple services (from different domains) without setting up a separate server (scalable, simpler to implement)
- Solution: Protocol that POSTs JSON text, gets response, parses the response into JavaScript value
 - No cookies sent
 - No other file formats accepted. Must be valid JS
 - JSON is safe JavaScript (data not functions)
 - Little or no error information provided to a miscreant
 - Accumulates random delays before trying again to frustrate denial of service attacks
 - Can support duplex!

Memory Management

- Hey, Internet Explorer is leaky!
 - Its memory garbage collector is the culprit
 - Uses reference counting; JS uses Mark & Sweep
 - Often closures are blamed
- Common problem that causes memory leaks
 - DOM <--> Application object reference each other
 - Event handlers left hanging around
- Tool for IE: Drip

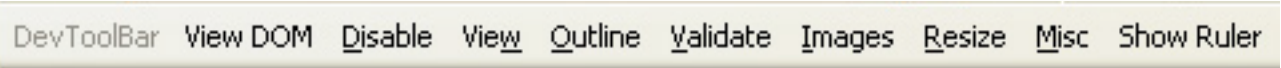
JavaScript Performance

- JavaScript requires common sense
 - Cache repeatedly accessed objects
 - Always use var for local variables
 - Use numbers for number and strings for strings
 - Avoid using eval() for accessing properties
eval("obj."+propName) --> obj[propName]
 - Look to your loops
- And a few surprises
 - Build DOM trees downward
 - Use array.join for lengthy string concatenations

source: Adam Platti



Debugging Tools

- Microsoft Script Editor
- Instant Source (IE Plugin)
- IE Dev Toolbar The image shows a screenshot of the Internet Explorer Developer Toolbar menu. The menu items are: DevToolBar, View DOM, Disable, View, Outline, Validate, Images, Resize, Misc, and Show Ruler. Each item has a small icon to its left.
- Venkman (Mozilla)
- Firebug (Mozilla)
- DOM Inspector (Mozilla)
- Web Developer Tools (Mozilla)
- Safari JavaScript Console
- JSLint

DHTML Load - Carousel

An array of images are randomly chosen on load and 'next' events, simulating loading data from within the client space.



Also see the [documentation](#)

Done



- Lint are syntax checkers and validators
- JavaScript needs lint
- <http://www.crockford.com/jslint/lint.html>
- Scans source file & looks for common syntax problems
 - Nice way to catch silly mistakes
 - Try it at: <http://jslint.com>

Obsfucation or Minification

- JavaScript is easy to see with a Save or a <ctrl>-U
- JavaScript can increase page size & page load
 - Obsfucators mangle the code into unreadable form
 - Minifiers strip white space & comments
- Obsfucators go too far
 - Makes development way too hard
- Minification & compression do just enough
 - JSMin (<http://www.crockford.com/javascript/jsmin.html>)

Error Handling

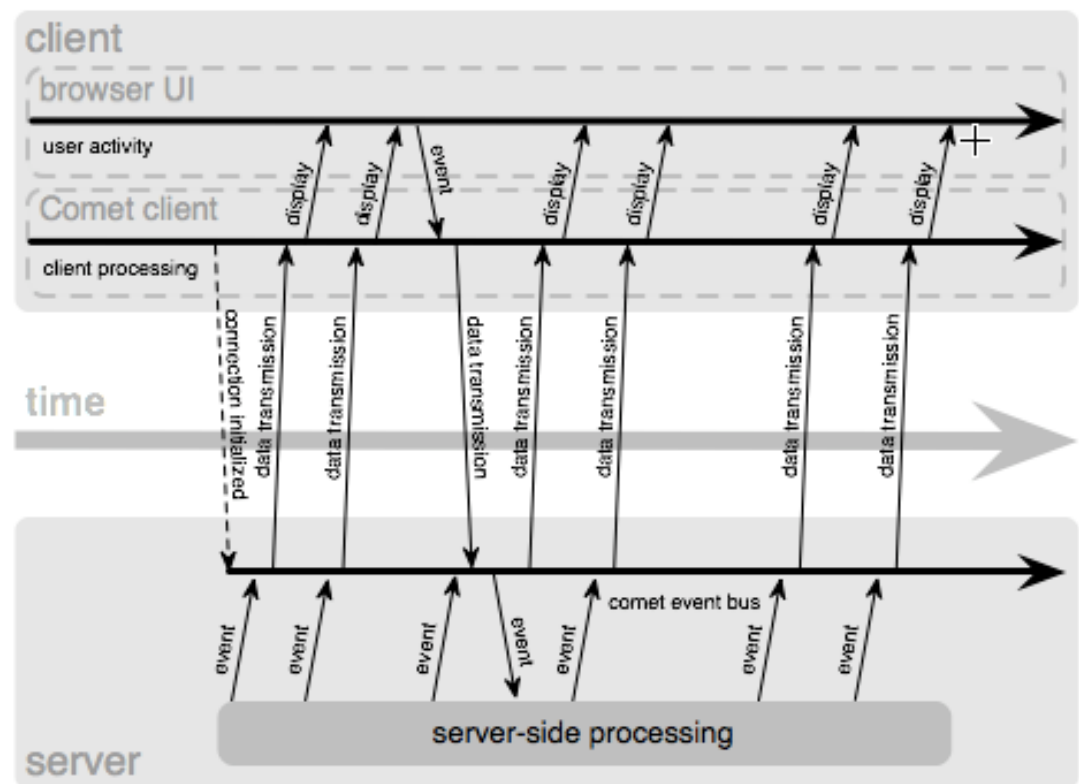
- Asynchronous error handling
 - Keep design simple (do you need multiple requests going at once?)
 - Will increase implementation issues
- Normal error handling
 - Check for error codes ($\neq 200$)
 - Roll your own HTTP status error codes
- Minimize server-side errors with intentional validation (error prevention)

- Difficult to correct XML, JSON or even HTML over the wire. XML & JSON are slightly harder. Server issue, but how to handle
- Use error handlers provided by Ajax frameworks
 - Prototype try.these
 - JavaScript try/catch
 - DWR has global error handler
 - Dojo error handlers

Comet

- Can deliver data to client any time
 - Delivered over single, previously opened connection
- Example applications
 - gmail's GTalk integration
 - Jot Live
 - Renkoo
 - Meebo

Comet web application model



What about Flash?

- Adobe Flash/JS Bridge (finance.google.com)
- Adobe Flex/Ajax Bridge (8.5)
- Flash sites (maps.yahoo.com/beta)
- Flex: 2.0 is serious environment (Flash 8.5; free)
- Laszlo: Generates either Flash or DHTML/Ajax
- Communication Bridge
- Storage Mechanism
- Event Extender
- Rich Media Extender (sound, video, streaming)



Questions?