



# Ajax at HBS

David Lieberman

Application Software Architect

Educational Technologies Group

Harvard Business School

*[dliberman@hbs.edu](mailto:dliberman@hbs.edu)*

The slide features a light blue background with a decorative border. On the left side, there is a vertical strip showing a close-up of classical architectural columns with ornate capitals. The main content area is a white rectangle with a thin blue border.

## Why bother?

- Performance gain?
- Simpler code?
- Better, more intuitive UI
- Rising user expectations!

The background of the slide features a light blue gradient with a faint, semi-transparent image of classical architectural columns on the left side. The columns are white with detailed capitals and are set against a darker blue background. The entire slide is framed by a thin brown border.

What does AJAX mean, anyway?

- **A**ynchronous **J**avaScript **A**nd **X**ML

The background of the slide features a light blue gradient with a faint, semi-transparent image of classical architectural columns on the left side. The columns are white with detailed capitals and are set against a darker blue background.

What does AJAX mean, anyway?

- ~~A~~ynchronous ~~J~~avaScript ~~A~~nd ~~X~~ML

## What does AJAX mean, anyway?

- ~~A~~ynchronous ~~J~~avaScript ~~A~~nd ~~X~~ML
- Just plain “Ajax,” thank you.



# My Goals for Ajax

- Take the protocol out of the equation



# My Goals for Ajax

- Take the protocol out of the equation
  - Put data exchange in the foreground

# My Goals for Ajax

- Take the protocol out of the equation
  - Put data exchange in the foreground
  - If I have to write  
`request.getParameter ("...")`  
I'm not interested ...



# My Goals for Ajax

- Take the protocol out of the equation
  - Put data exchange in the foreground
  - If I have to write  
`request.getParameter ("...")`  
I'm not interested ...
- So: XML in, XML out

# My Goals for Ajax

- Take the protocol out of the equation
  - Put data exchange in the foreground
  - If I have to write  
`request.getParameter ("...")`  
I'm not interested ...
- So: XML in, XML out
- Show me the framework!
  - Struts?

## XMLHttpRequest properties

- **responseText** [String]  
Raw response text
- **responseXML** [XML Object]  
*If* response text is well-formed XML
- **readyState** [Integer]
- **status** [Integer = the HttpResponse code]

## XMLHttpRequest methods

- `open([String] method,  
[String] url,  
[boolean] asynchronous)`
- `send([String] arg)`

The background of the slide features a light blue gradient with a faint, semi-transparent image of classical architectural columns on the left side. The columns are white with detailed capitals and fluted shafts. The entire slide is framed by a thin brown border.

## XMLHttpRequest event

- `onReadyStateChange`

# XMLHttpRequest

## Set up method variables

```
function get(url, callback) {  
    var ajaxrequest;  
    var ajaxresponse;  
    . . .  
}
```

# XMLHttpRequest

## Bind the request callback method

```
function get(url, callback) {  
    . . .  
    function bindRequestChange() {  
        if (ajaxrequest.readyState == 4) {  
            if (ajaxrequest.status == 200) {  
                ajaxresult = ajaxrequest.responseXML;  
                eval(callback + "(ajaxresult)");  
            } else {  
                alert("There was a problem retrieving the XML  
                    data: \n" + ajaxresult.statusText);  
            }  
        }  
    }  
}
```

# XMLHttpRequest

## Instantiate the XMLHttpRequest object

```
function get {  
    . . .  
  
    if (window.XMLHttpRequest) {  
        ajaxrequest = new XMLHttpRequest();  
    } else if (window.ActiveXObject) {  
        ajaxrequest = new ActiveXObject("Microsoft.XMLHTTP");  
    }  
  
    if (ajaxrequest) {  
        ajaxrequest.onreadystatechange = bindRequestChange;  
        ajaxrequest.open("GET", url, true);  
        ajaxrequest.send(null);  
    }  
}
```



The background of the slide features a light blue gradient with a faint, semi-transparent image of classical architectural columns on the left side. The columns are white with detailed capitals and are set against a darker blue background. The entire slide is framed by a thin brown border.

## Three Implementations at HBS

- Application Access Admin
- Tutorial Platform Authoring
- Videotools Portal Admin
- Controlled environments with small user communities

The background of the slide features a light blue gradient with a faint, semi-transparent image of classical architectural columns on the left side. The columns are white with detailed capitals and are set against a darker blue background. The entire slide is framed by a thin brown border.

## Application Access Admin

- Utility for controlling access to applications or application modules based on group membership
- Ajax feature:
  - Auto-populating roles selector

The background of the slide features a light blue gradient with a faint, semi-transparent image of classical architectural columns on the left side. The columns are white with detailed capitals and are set against a darker blue background. The entire slide is framed by a thin brown border.

# Tutorials Platform Authoring

- Tutorials Platform: exposes Flash content to end-users delivered via XML (Ajax competitor alert!)
- Authoring environment: assembling these tutorials element-by-element
- Ajax features:
  - Metadata
  - Tree sorting
  - Rich text editor

The slide features a light blue background with a decorative border. On the left side, there is a vertical strip showing a close-up of classical architectural columns with ornate capitals. The main content area is a white rectangle with a thin brown border.

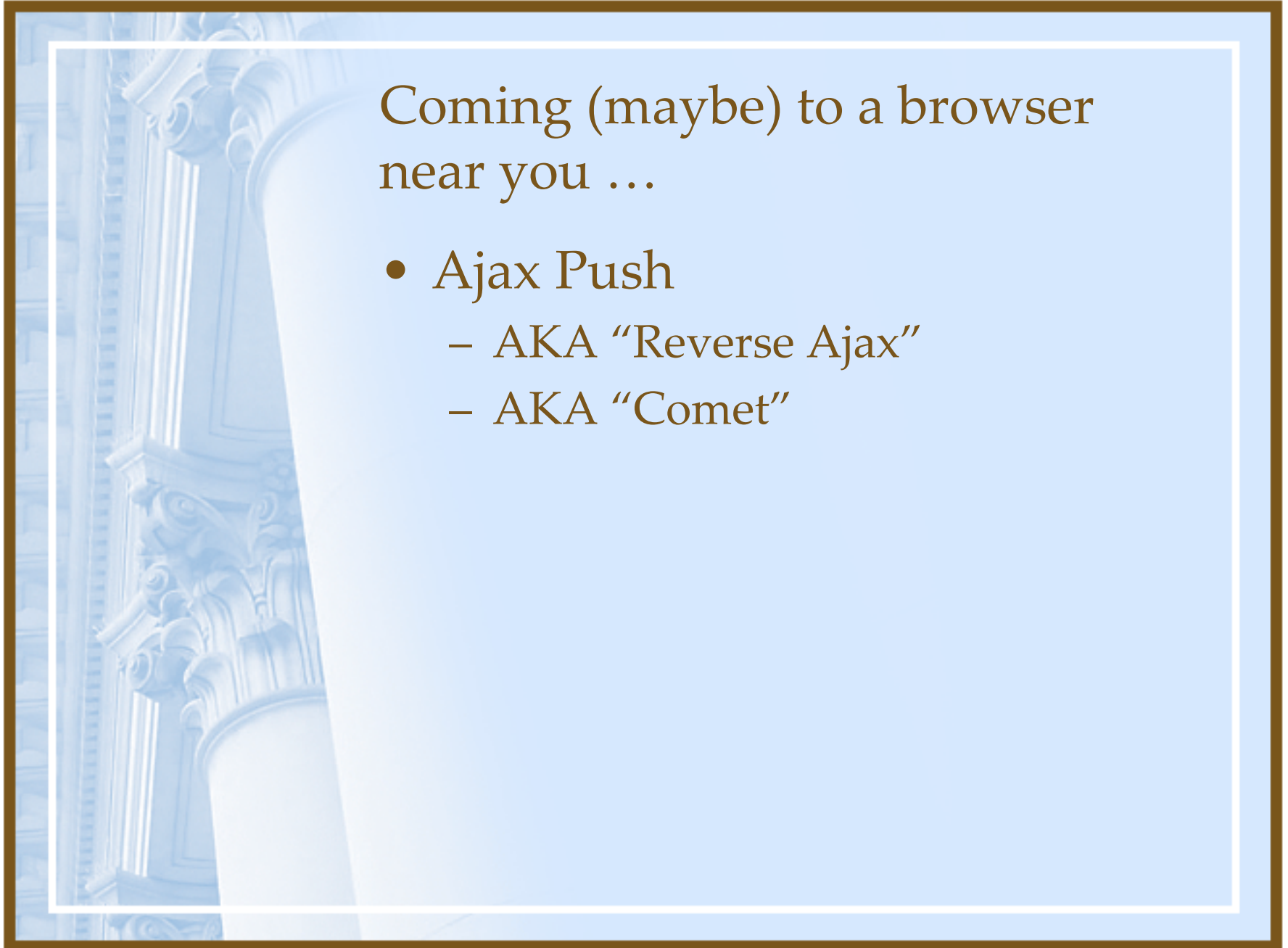
## Gotchas

- Support!
- Google we are not.
- The Javascript DOM API: more like JAXP than JDOM (cf JDOM + XPath)
- Safari!

The background of the slide features a light blue gradient with a faint, semi-transparent image of classical architectural columns on the left side. The columns are white with detailed capitals and are set against a darker blue background. The entire slide is framed by a thin brown border.

## Videotools Portal Admin

- Portal: a stylized, sorted collection of video assets targeted to a specific user community
- Admin requires: sorting, labeling, refreshing, creating RSS feeds.
- Out with XML, in with JSON!

The slide features a light blue background with a decorative border. On the left side, there is a vertical strip showing a close-up of classical architectural columns with ornate capitals. The main text is positioned on the right side of the slide.

Coming (maybe) to a browser  
near you ...

- Ajax Push
  - AKA “Reverse Ajax”
  - AKA “Comet”

# JavaScript Libraries

- Scriptaculous  
<http://script.aculo.us/>
- Prototype  
<http://prototype.conio.net/>
- Yahoo! UI Library  
<http://developer.yahoo.com/yui/>
- Zimbra Toolkit  
[http://www.zimbra.com/community/ajaxtk\\_download.html](http://www.zimbra.com/community/ajaxtk_download.html)
- Dojo <http://dojotoolkit.org/>
- DWR <http://getahead.org/dwr/getstarted>

# Other Resources

- <http://www.jdom.org>
- <http://jakarta.apache.org/commons/jxpath/>
- <http://www.json.org>
- <http://www.ajaxian.com>