AJAX

Bank of America

What is AJAX?

- AJAX , is a web development technique used for creating interactive web applications .
- The intent is to make web pages feel more responsive by exchanging small amounts of data with the server behind the scenes, so that the entire web page does not have to be reloaded each time the user requests a change.
- This is intended to increase the web page interactivity, speed, functionality, and usability

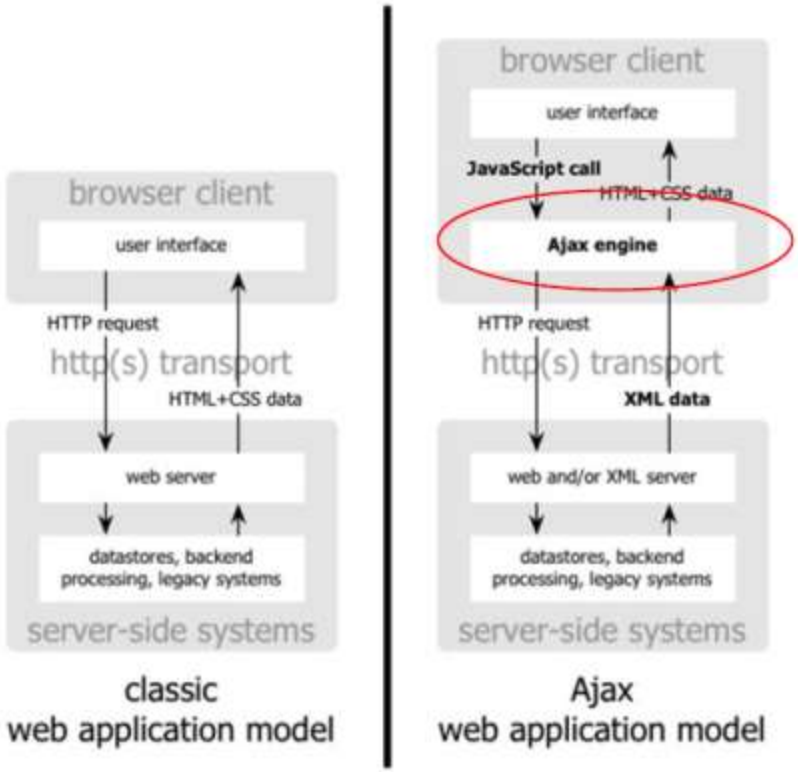
Components

- **XMLHttpRequest.**
  - Retrieving data asynchronously from the web server.
- **Cascading Style Sheets (CSS).**
  - Presenting information.
- **Document Object Model (DOM).**
  - Dynamic display and interaction with information.
- **JavaScript.**
  - Binding everything together.
- **XML**
  - Often used as the format for transferring data

How does AJAX work?

- **Classic Web Model**
  - Browser interacts directly with Web Server.
  - Each Request Causes Page Refresh.
  - Synchronous.
  - Refresh Looses Place on Page.

- **Ajax Application Model:**
  - Browser interacts with Ajax Engine.
  - Ajax Engine communicates with Web Server
  - Asynchronous
  - Page manipulated by Ajax Engine (no refresh)



XMLHttpRequest Object

- The XMLHttpRequest object is the key to AJAX

- XMLHttpRequest (XHR) is an API that can be used by JavaScript, JScript, VBScript and other web browser scripting languages to transfer and manipulate XML data to and from a web server using HTTP, establishing an independent connection channel between a web page's Client-Side and Server-Side.

- The XMLHttpRequest object is used to exchange data with a server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.

XMLHttpRequest Object (cont...)

Creating an XMLHttpRequest Object

var xmlhttp=new XMLHttpRequest();

Old versions of Internet Explorer(IE5 and IE6) uses an ActiveXObject

var xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");

Send a Request To a Server

Method	Description
open(*method,url,async*)	Specifies the type of request, the URL, and if the request should be handled asynchronously or not. *method*: the type of request: GET or POST *url*: the location of the file on the server *async*: true (asynchronous) or false (synchronous)
send(*string*)	Sends the request off to the server. *string*: Only used for POST requests
setRequestHeader(*header,value*)	Adds HTTP headers to the request. *header*: specifies the header name *value*: specifies the header value

```
xmlhttp.open("GET","ajax_info.aspx",true);
xmlhttp.send();
```

Server Response

- **Onreadystatechange**

  An event handler for an event that fires at every state change.
- **readyState**

  The readyState property defines the current state of the XMLHttpRequest object.
  Here are the possible values for the readyState property:

State	Description
0	The request is not initialized
1	The request has been set up
2	The request has been sent
3	The request is in process
4	The request is completed

- **responseText**

  Returns the response as a string.
- **responseXML**

  Returns the response as XML.
- **status**

  Returns the status as a number (e.g. 404 for "Not Found" and 200 for "OK").
- **statusText**

  Returns the status as a string (e.g. "Not Found" or "OK").

Example

```
function ajax_request(url, data, callback)
    {
     var xhrobj;
      try {
        xhrobj = new XMLHttpRequest();
      } catch (e) {
        try {
          xhrobj=new ActiveXObject("Msxml2.XMLHTTP");
        } catch (e) {
          xhrobj=new ActiveXObject("Microsoft.XMLHTTP");
        }
      }  xhrobj.open("GET", url);
      xhrobj.setRequestHeader("X-Requested-With", "XMLHttpRequest");

      xhrobj.onreadystatechange = function()
      {
        if (xhrobj.readyState == 4 && xhrobj.status == 200)
        {
          if (xhrobj.responseText)
          {
              callback(xhrobj.responseText);
          }
        }
      };
      xhrobj.send(data);
    }
```

Demo on ASP.NET AJAX

Bank of America

AJAX ASP.Net Control Toolkit

- The ASP.NET AJAX Control Toolkit is a shared-source community project consisting of samples and components that make it easier to work with AJAX-enabled controls and extenders.
- The Control Toolkit provides both ready-to-run samples and a powerful SDK to simplify creating custom ASP.NET AJAX controls and extenders.
- Demo

Page Methods

- Server code can be bound to ASP.NET pages
- Easy way to communicate asynchronously with the server using ASP.NET AJAX technologies
- Alternate to calling the page which is reliable and carries a low risk.
- Expose the web methods to the client script
- **Key Points to Remember**
  - The Page Method should be static using the attribute [WebMethod].
  - The Page Method can be in the code behind file of the page and in aspx file as well.
  - The EnablePageMethods property of the script manger should be marked as “True”.
  - The call back methods are mandatory to receive the processed result on the client side.
  - Page Methods don’t support all the datatypes to return to client.
  - We can access the Session. Application and Cache variables using Page Methods.

Demo on Page Methods

Thank You