AJAX – Asynchronous JavaScript And XML

Overview

- What is AJAX ?
- Why AJAX ?
- AJAX Vs Classical Web Approach.
- How AJAX Works ?
- XMLHttpRequest ?
- How to make an AJAX application ?
- Live AJAX Examples.

What is AJAX ?

- Ajax isn’t a new technology or programming language.
- It is a technique used to develop interactive web applications that are able to process a user request immediately.
- Ajax can selectively modify a part of a page displayed by the browser, and update it without the need to reload the whole document with all images, menus, etc.

What is AJAX ?(cont)

- The word "**Asynchronous**" in AJAX means that the request to the server will be made. The response will be made available by the server after it has finished processing the request, without having to wait for it explicitly, to come back. i.e. you *don't have to wait for an answer.*

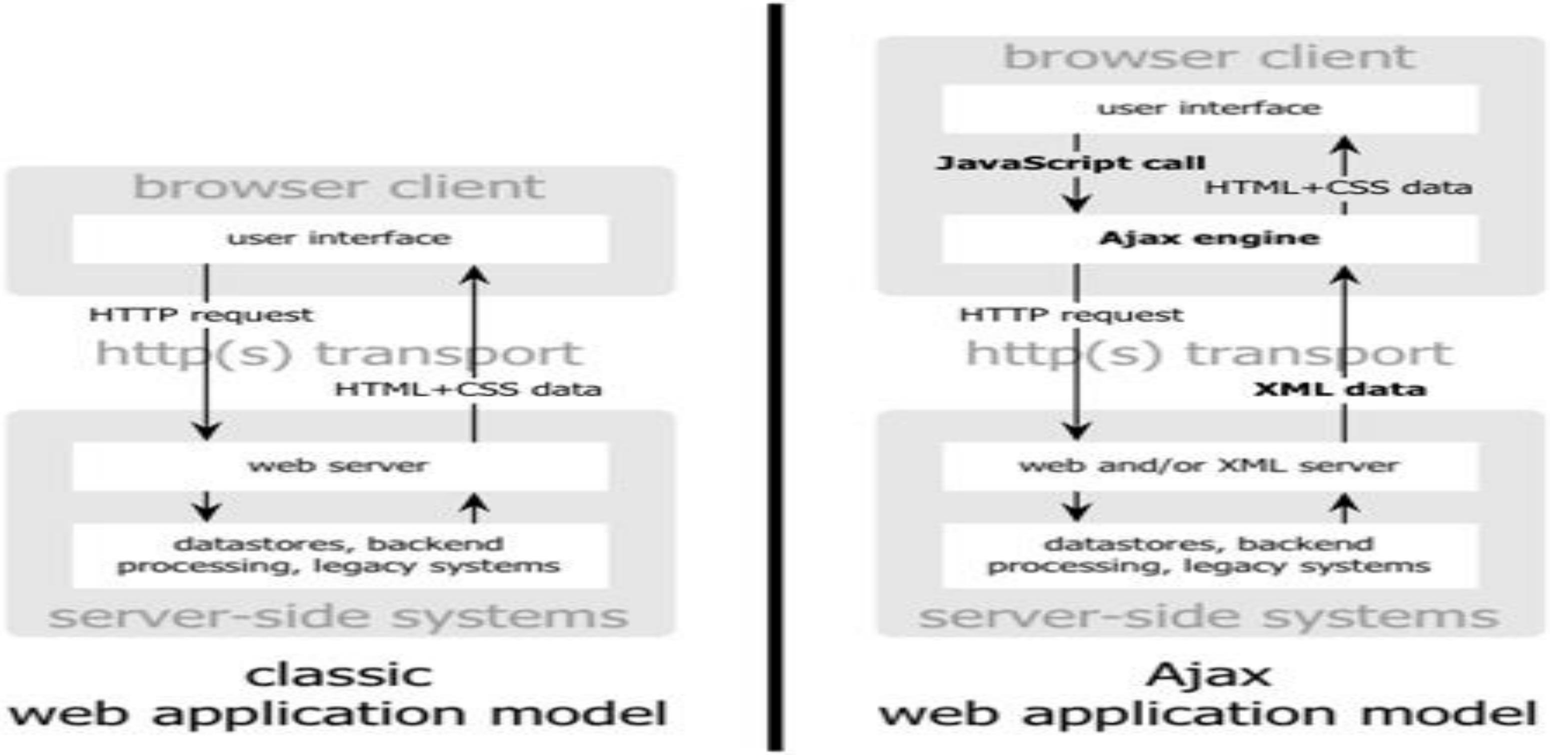
Why AJAX?

- The common problem is WebPages always "RELOAD" and never get "UPDATED", creating a break in user interaction.
- Users wait for the entire page to load even if a single piece of data is needed
- Restriction on single Request/Response.
- Need for complicated UI components.

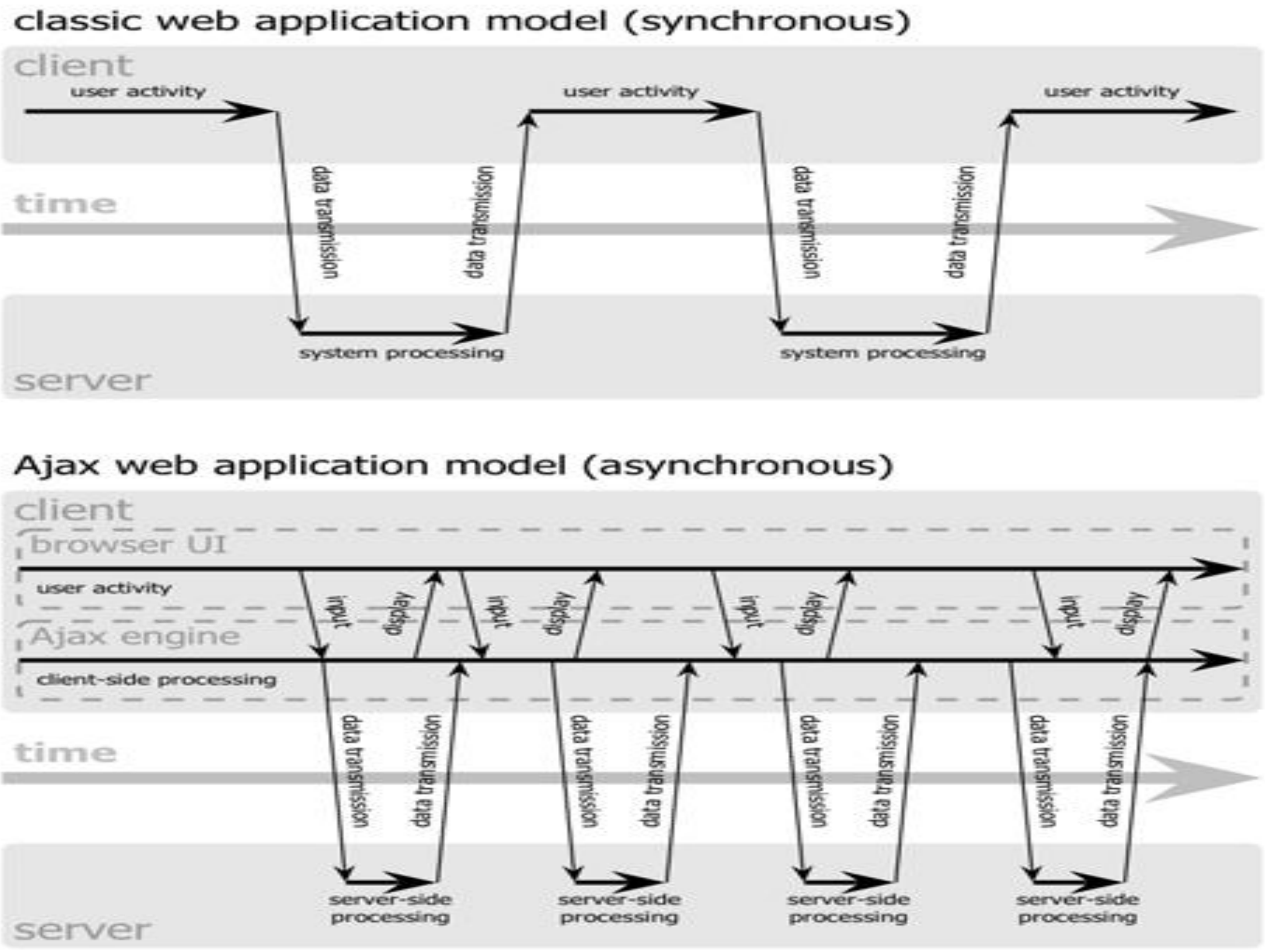
Technologies Involved

- **JavaScript** – for binding everything together.
- **HTML & Cascading Style Sheets** (CSS), for presentation and to provide style classes for the component styling.
- **XMLHttpRequest** – provides asynchronous data retrieval.
- **Document Object Model** (DOM), for dynamic display and interaction and provides a tree structure for the components placed in the web page.
- **Extensible Markup Language** (XML), Format to send data from client to server, though other formats like JSON may also be used.

AJAX approach Vs Classical approach



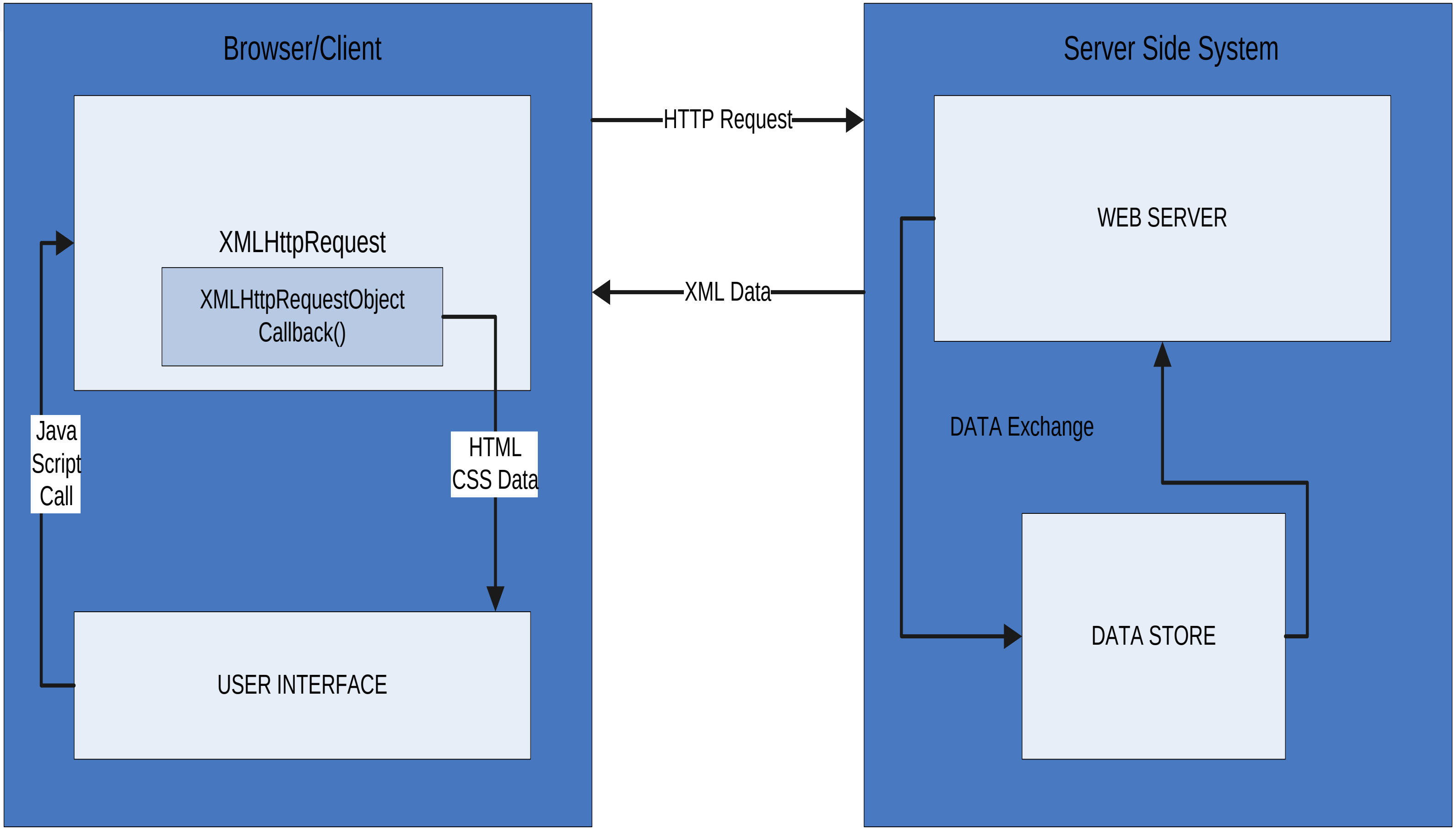
AJAX approach Vs Classical approach (Cont)



How AJAX Works ?

- A JavaScript function creates and configures an XMLHttpRequest object on the client, and specifies a JavaScript callback function.
- The XMLHttpRequest object makes an asynchronous call to the web server.
- The web server processes the request and returns an XML document that contains the result.
- The XMLHttpRequest object calls the callback function and exposes the response from the web server so that the request can be processed.
- The client updates the HTML DOM representing the page with the new data.

How AJAX Works ? (cont)



XMLHttpRequest ?

- XMLHttpRequest object is the key to Ajax programming.
- It’s main purpose is to put an asynchronous http request to the web server.
- Because of this asynchronous call to the web server, you are allowed to continue using the page without the interruption of a browser refresh and the loading of a new or revised page.
- This object has few properties.

Properties of XMLHttpRequest

- Property 1: objXML*Http.onreadystatechange*

  This property holds the reference of function which is going to process the response from the server*.*

objXMLHttp.onreadystatechange = procRequest;

** "procRequest " is the function which will process the response*

Properties of XMLHttpRequest

- Property 2 : objXML*Http. readyState*

  This property holds the status of server response.

  objXMLHttp.readyState = [state];

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 complete

Properties of XMLHttpRequest

- Property 3: objXML*Http.responseText*

  This property retrieves the data sent back from server.

  var objVal = objXMLHttp. responseText;

  While the responseText is used to return text ,

  responseXML can be used to return an XML document object.

  var xmlDoc;

  xmlDoc = objXMLHttp.responseXML.documentElement

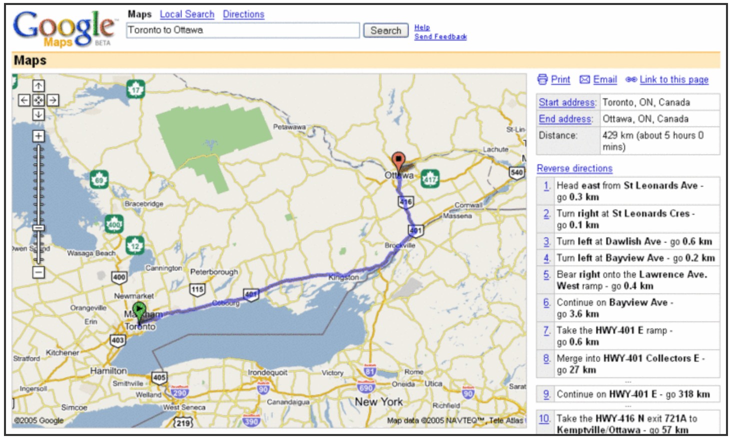
How to make an Ajax application ?

```
<html>
 <head>
  <title>Ajax at work</title>
  <script language = "javascript">
   var XMLHttpRequestObject = false;
   if (window.XMLHttpRequest) {
    XMLHttpRequestObject = new XMLHttpRequest(); //for mozilla
   } else if (window.ActiveXObject) {
 XMLHttpRequestObject = new ActiveXObject("Microsoft.XMLHTTP"); // IE
   }
   function getData(dataSource, divID)
   {
    if(XMLHttpRequestObject) {
     var obj = document.getElementById(divID);
     XMLHttpRequestObject.open("GET", dataSource);
     XMLHttpRequestObject.onreadystatechange = function()
     {
      if (XMLHttpRequestObject.readyState == 4 &&
       XMLHttpRequestObject.status == 200) {
        obj.innerHTML = XMLHttpRequestObject.responseText
```

Sample Code (cont)

```
}      } XMLHttpRequestObject.send(null);
    }
   }

</script>
 </head>

 <body>

  <H1>Fetching data with Ajax</H1>

  <form>
   <input type = "button" value = "Display Message"
    on click = "getData('data.txt', 'targetDiv')“ >
  </form>

  <div id="targetDiv">
   <p>The fetched data will go here.</p>
  </div>

 </body>
</html>
```

Live Examples of AJAX



Maps [Local Search](#) [Directions](#)

Toronto to Ottawa

[Help](#)
[Send Feedback](#)

Maps

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

[Start address](#):	Toronto, ON, Canada
[End address](#):	Ottawa, ON, Canada
Distance:	429 km (about 5 hours 0 mins)

[Reverse directions](#)

1.	Head **east** from **St Leonards Ave** - go **0.3 km**
2.	Turn **right** at **St Leonards Cres** - go **0.1 km**
3.	Turn **left** at **Dawlish Ave** - go **0.6 km**
4.	Turn **left** at **Bayview Ave** - go **0.2 km**
5.	Bear **right** onto the **Lawrence Ave. West** ramp - go **0.4 km**
6.	Continue on **Bayview Ave** - go **3.6 km**
7.	Take the **HWY-401 E** ramp - go **0.6 km**
8.	Merge into **HWY-401 Collectors E** - go **27 km**

...

9.	Continue on **HWY-401 E** - go **318 km**

...

10.	Take the **HWY-416 N** exit **721A** to **Kemptville/Ottawa** - go **57 km**

- Google suggest is a simple google application which uses Ajax.

References

- http://en.wikipedia.org/wiki/AJAX
- http://www.w3schools.com/ajax/default.asp
- http://www.adaptivepath.com/ideas/essays/archives/000385.php
- http://www.w3.org/TR/XMLHttpRequest/
- http://developer.mozilla.org/en/docs/AJAX

End