AJAX – A new pattern to build dynamic web applications

AJAX, or Asynchronous JavaScript and XML, is emerging as a popular pattern to create highly interactive and responsive web interfaces. It is already more than just a buzz. AJAX is one piece of so called Web 2.0. Impressive web applications using AJAX are already in place and available. Just take a look at:

- Google Suggest: http://www.google.com/webhp?complete=1&hl=en
- Google Maps: http://maps.google.com/
- Flickr: http://www.flickr.com/

The basic idea of AJAX is to establish a communication channel between the web client and the web server, where data can be transferred asynchronously, usually as XML. Because of the fact that the communication is asynchronous, the web page that represents the user interface does not have to be transferred completely. This technology avoids flickering effects because of complete page reloads and allows the user to continue to use the page as normal, while data (e.g.: cell values) are being transferred in the background. AJAX is a means to enhance and streamline web applications by eliminating page loads. The following two figures depict the building blocks of web applications / web user interfaces built on AJAX.

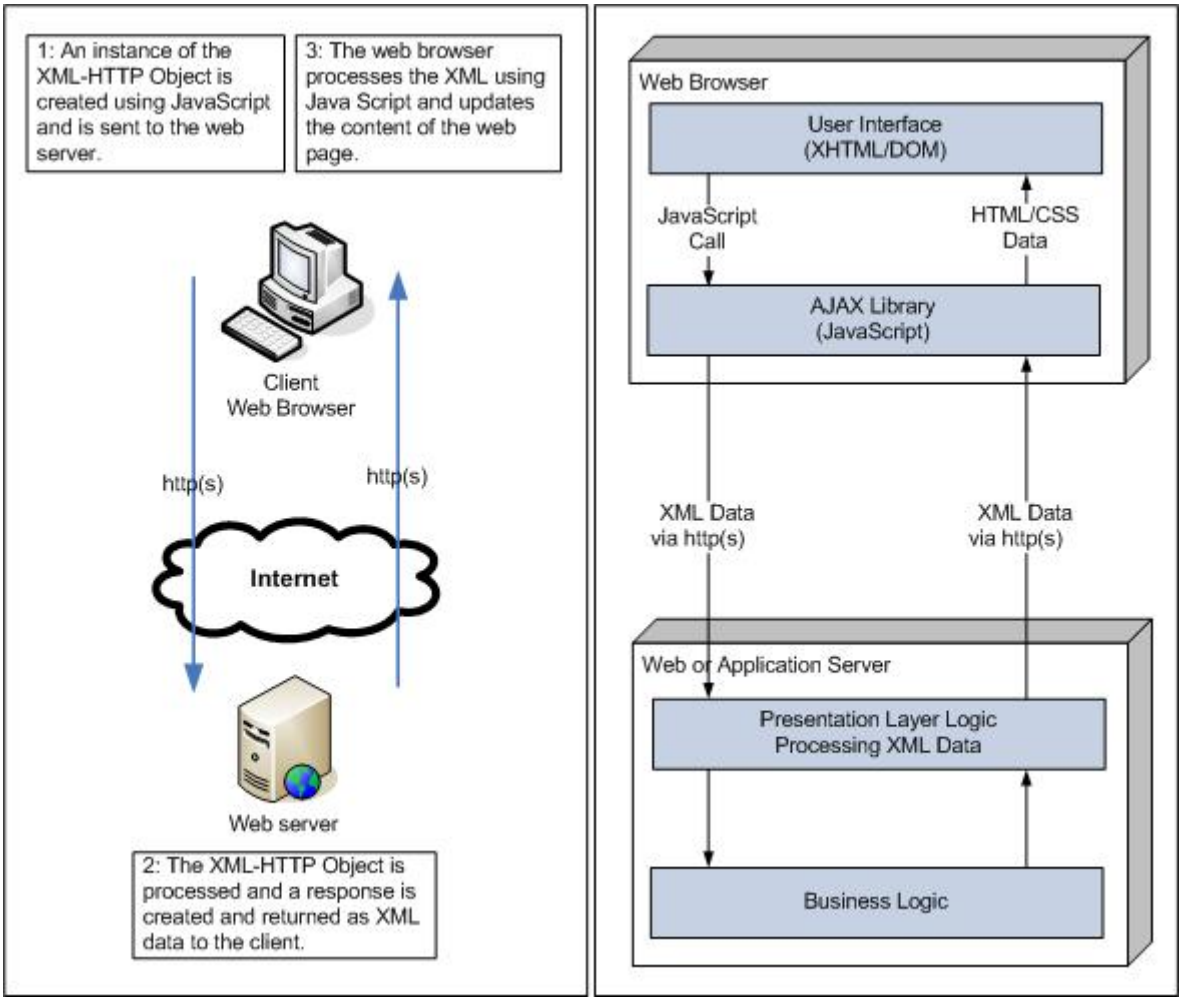

Figure 1: Basic AJAX Functionality

A typical use case is the update of table cell values or other page components without a complete page reload or frame content reload. Basically, AJAX as a pattern is a composition of existing technologies which are well known in the domain of web application development. AJAX encompasses:

- Web browser presentation based on XHTML and CSS
- Interaction, navigation and manipulation using the Document Object Model (DOM)
- Asynchronous data fetching using the XML-HTTP Object (Microsoft.XMLHTTP, XMLHttpRequest)
- Data handling and transformation based on XML and XSLT
- Interchange and manipulation logic based on JavaScript

The JavaScript libraries are the core components of an AJAX based solution at the client side (web browser). They encapsulate the asynchronous data interchange logic. And, it needs the corresponding presentation logic on the web server side.

Availability and Frameworks

Both components (client and server side libraries / controls) are evolving as frameworks in both worlds of web application development (ASP.NET and J2EE). Microsoft has already announced to release an add-on called “Atlas” encompassing the AJAX functionality. It is smart business to check on these implementations in order to base development efforts on a mature and supported framework.

AJAX & ATLAS @ASP.NET:

http://beta.asp.net/default.aspx?tabindex=0&tabid=1

AJAX Framework proposal @Apache:

http://www.mail-archive.com/general@incubator.apache.org/msg05985.html

AJAX @STRUTS:

http://struts.sourceforge.net/

Challenges and constraints

AJAX can improve performance and usability of a web application substantially. But a couple of constraints have to be considered to avoid pitfalls. Some of these problems will disappear as AJAX matures.

1. XML-HTTP Object availability and implementation

The web browser must support the XML-HTTP object. The Internet Explorer as well as the Firefox/Mozilla does support it but with different implementations. This has to be taken into account regarding the provisioning of the libraries. In addition, security setting of the web browser can affect the functionality of the client side scripting. This is especially true in a scenario where the data are intended to fetch from a server different than the server where the scripts come from. A signed script can ease this problem.

2. Server Load

An AJAX based web application may increase the number of requests made to the web server. The resulting server load has to be considered. Intelligent logic distributions (e.g.: validation on client side) and smart caching and buffering can mitigate the server load.

3. Programming asynchronous web applications

Asynchronous data transfer using HTTP has to be understood and considered in the implementation. It is not ensured that the XML-HTTP requests will be completed in the order they were sent to the web server.

4. Usability

This is just a minor issue but also worth to mention. Web application (web page) behavior may change for the user. Typical indications for requests and page updates do not apply by using AJAX. One means to inform the user on processing and updates could be a message saying “update in progress”.