# 5

# AJAX

In this chapter we will look at:

- AJAX basics
- Limitations of traditional web applications
- Items for implementing AJAX

## 5.1 AJAX BASICS

Ajax stands for Asynchronous JavaScript and XML. Usually in a traditional web application, when we want to access certain data from the database on the server, an HTTP request from the client is made to the server either by GET or POST method. After receiving the data from the server, the web page needs to be *reloaded* to show the fetched data. In AJAX technology we can request and receive the data from the server in the background and can display it on the page without a reload.

With AJAX, JavaScript communicates directly with the server, through the JavaScript XMLHttpRequest object (XML over HTTP) and it is with the help of this object that a web page can make a request to, and get a response from a web server without reloading the page.

A traditional web page takes a longer time to get the desired results because of the *round trip*: all the information entered by the user on the form is sent from the client to the web server. The web server processes the data and the desired information is sent back to the client. Even if small changes are made in the form, all the data on the form is sent to the web server and the entire page is refreshed. In a traditional web application, we don't have a facility to refresh only a small portion of the web page; instead the complete page is refreshed, which is very time consuming

In Figure 5.1, we can clearly see the three steps of communication performed in the traditional web application model:

The client makes an HTTP request to the web server.

The web server searches for the desired data from the database and

The fetched data (from the database) is sent back (postback) to the client (the whole page is reloaded with the new information).

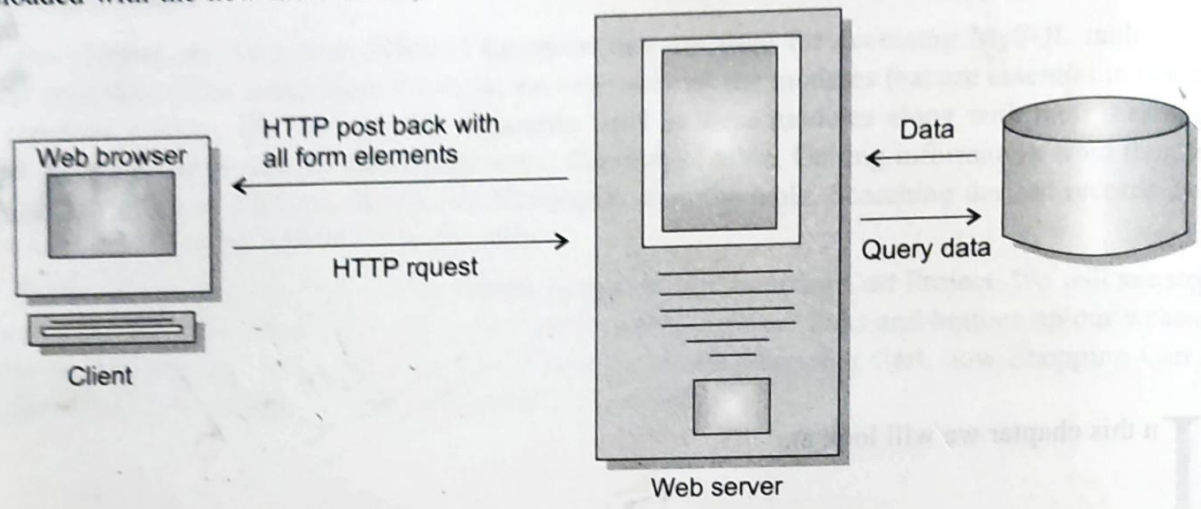


**Figure 5.1** *Traditional Web Application Model*

We face the following limitations in a conventional web application.

## 5.2 LIMITATIONS OF TRADITIONAL WEB APPLICATIONS

- All the data of the form is sent to the web server—even if small changes are made
- Results in network congestion because of large amount of data transferred during a postback
- Until and unless the user clicks a button or another control that posts data back to the server, no result will be displayed

In AJAX, A stands for *Asynchronous* and it means getting server response without refreshing the whole page. We can even update a portion of a web page with this technology. Let's try for a deeper understanding of what AJAX consists of.

With AJAX all the above limitations of traditional web applications are removed as it consists of following items:

- XmlHttpRequest
- JavaScript
- DOM (Document Object Model)
- CSS (Cascading Style Sheets)

## 5.3 ITEMS FOR IMPLEMENTING AJAX

### 5.3.1 XmlHttpRequest

It is the object that is used to talk to the server asynchronously, meaning it allows the browser to talk to the server without requiring a postback of the entire web page. The way this capability is provided varies from browser to browser. For example, in Internet Explorer, this capability is provided by the MSXML ActiveX component. With Mozilla Firefox and other web browsers, this capability is provided by the object called XmlHttpRequest.

### 5.3.2 JavaScript

Ajax applications use JavaScript code for the following reasons:

- It is a scripting language which is interpreted
- The syntax of commands is easier to learn
- It can automate several tasks like validation of userid, email id etc.

**Note**: JavaScript is the client side scripting language which is supported by all major browsers.

### 5.3.3 DOM (Document Object Model)

It provides a tree-like structure to a web page as a set of programmable objects which can be manipulated using JavaScript code. It allows dynamic updating of a part of a web page.

### 5.3.4 CSS (Cascading Style Sheets)

It is a centralised way of defining all the styles to be applied to different elements of a web page at one place. It makes web applications appear more consistent and attractive. CSS styles are implemented by defining style rules in a separate document which is then referred to by the web page where styles have to be applied.

In Figure 5.2, we can clearly see that XMLHttpRequest plays a major role in performing asynchronous request in the Ajax web application model:

- The client makes an XMLHTTP Request object to communicate with the web server (fetching data from server without postback). The client may use:
- JavaScript to automate validation or navigation tasks
- CSS for applying styles uniformly
- DOM to update a part of the web page
- The Web Server searches for the desired data from the database and

- The fetched data (it may constitute a part of the whole page) is sent back to the client so as to update a portion of the web page without delay.

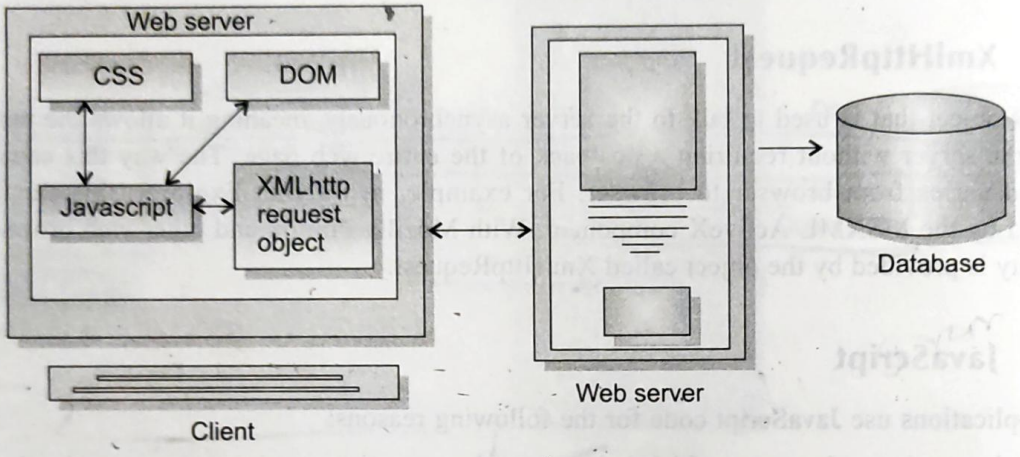


**Figure 5.2** ***Ajax Web Application Model***

So, it is clear that Ajax applications make use of XmlHttpRequest object for talking to the server asynchronously and to retrieve only the data that is needed (while the user is working). And on the client side, JavaScript plays a major role as it processes the server's response and modifies the document contents by manipulating its DOM to specify that action is completed. The job of CSS is to provide a consistent look and feel to the web application.

**Note**: The data are mostly communicated to the web server in XML format but it may also be in JSON (JavaScript Object Notation) format (we will see it in Chapter 8).

## SUMMARY

In this chapter we learnt that the main components that make up AJAX are Asynchronous JavaScript and XML and it is a technique which makes a web page highly responsive. It makes use of a XMLHttpRequest object with the help of which a web page can make a request to, and get a response from a web server without reloading the page. Also in this technique, even a small portion of a web page can be refreshed instead of the entire page. We also saw the limitation of the traditional web page in this chapter. Finally we had an introduction to all the four items used in implementing AJAX: XmlHttpRequest, JavaScript, DOM and CSS.

In the next chapter, we will see the practical demonstration of AJAX with the help of running examples. We will learn the basics of JavaScript, role of DOM in accessing form elements, the way in which an asynchronous request is placed to the server, handling of Key Events, using JavaScript in accessing form elements, performing XMLHttp Post Requests, separating JavaScript code from the PHP program, specifying our own function in JavaScript file, using CSS (cascading style sheets), sending data selected from the combo box, items selected from a list box, items selected from radio and checkboxes to the server asynchronously. Finally, we will also see how AJAX, PHP and MySQL all combine for accessing a database.