

AJAX

Asynchronous JavaScript and XML

Speed and Dexterity for futuristic web enabled applications...



A conventional web application essentially submits forms, filled by users to a web server, where processing is done and the processor responds by sending a new web page back. As the server sends an entirely new page every time, the application runs more slowly.

Ajax on the other hand enables web applications to respond more quickly to numerous types of user interface and to avoid continually transferring unchanged information across the network. Ajax technologies are open and therefore are compatible with all JavaScript-enabled browsers regardless of the operating systems.

What is Ajax?

Ajax is a blend of several technologies. Ajax stands for-Asynchronous JavaScript and XML. It is a technique used to develop web based applications. It incorporates:

- XHTML and CSS for marking up and styling information
- The Document Object Model, which is manipulated through JavaScript to dynamically demonstrate and interrelate with the information presented
- Retrieval of asynchronous data using XMLHttpRequest

XML is generally used, though any format works; counting preformatted HTML, plain text, JSON and even EBML.

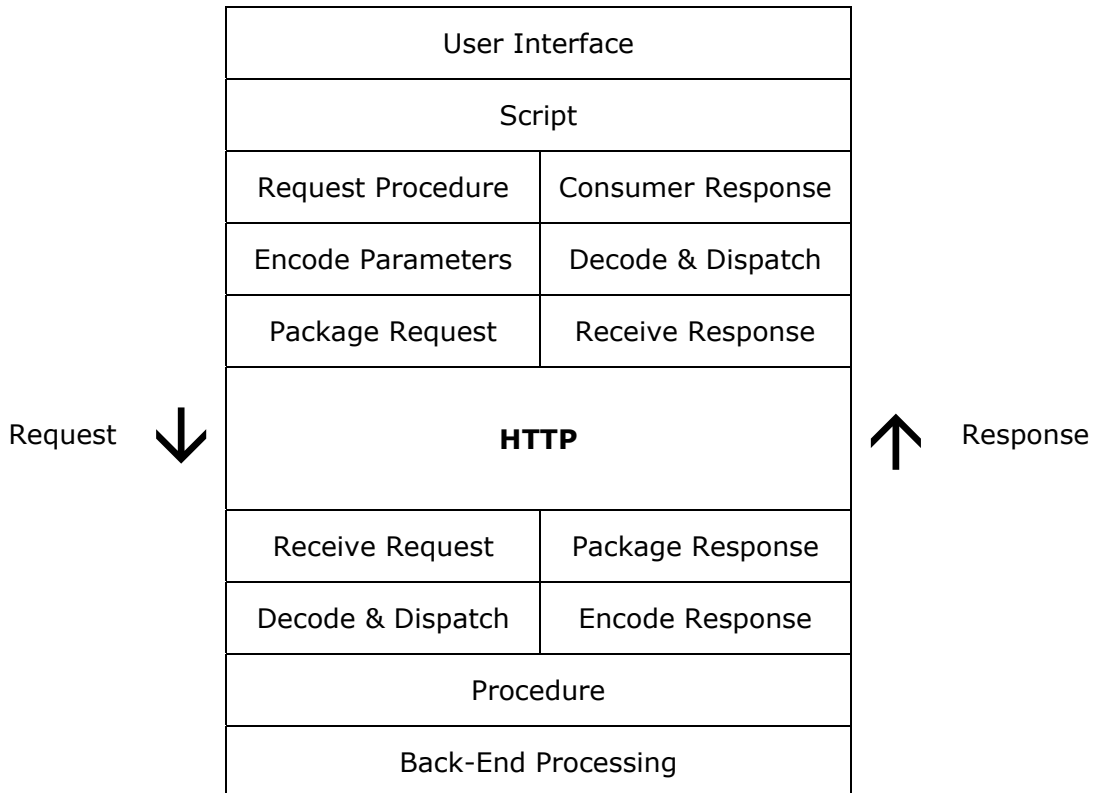
Web applications developed on Ajax technology can perform most of the functions that the native application of the user's system can. This is because while interacting with the server on which they reside, a very small amount of data is exchanged back and forth at a time. This leads to pages being updated in the user's browser only, rather than being refreshed completely. It's a model modified from the web's unique use as a hypertext medium.

Functional aspects:

In applications developed through this technology every user's action that would generally generate an HTTP request gets transformed into JavaScript and summons the Ajax engine.

To enhance the functional timing, any reaction to a user's action that does not need a trip back to the server — such as simple data confirmation, data edition in memory, and a few navigations — are handled by the engine on its own.

If the engine requires some information from the server so as to present a response, such as: submission of data for processing, loading added interface code, or recovering new data — the engine places these requests synchronously, generally using XML, without delaying a user's interaction with the application.



AJAX PROCESSING MODEL

Accessibility:

Using Ajax technology in web applications proves to be a major challenge for developers involved in adhering to WAI convenience guidelines. Developers need to offer fallback options for users on other platforms or browsers, as most methods of Ajax execution rely on characteristics, present only on desktop graphical browsers.

Web developers employ Ajax in some cases to offer content only to particular portions of a web page, allowing data management without incurring the cost of re-rendering the complete page in the web browser.

Numerous famous applications have been developed by deploying the Ajax technology such as: Google Maps and GMail, Yahoo's Flickr, America Online's AIM Mail, 24SevenOffice and Microsoft's Virtual Earth. These prestigious examples of Ajax usage demonstrate the flexibility and effectiveness of the web programming model.

Browsers Supporting Ajax:

- Apple Safari 1.2 and above
- Microsoft Internet Explorer 5.0 and above
- Mozilla/Mozilla Firefox 1.0 and above
- Netscape 7.1 and above
- Konqueror
- Opera 7.6 and above
- Opera Mobile Browser 8.0 and above

How Ajax is Different?

An Ajax application reduces the interaction time on the web by introducing a mediator — an Ajax engine — connecting the user and the server. Rather than loading a webpage, at the start of the session, the browser loads an Ajax engine which is written in JavaScript and generally stored in a hidden frame. This engine is responsible for both depicting the interface the user sees as well as communicating with the server on the user's behalf.

The Ajax engine allows the user's communication with the application to happen asynchronously without affecting communication with the server. This leads to users never staring at a blank browser window.

All user actions that would generally create an HTTP request take the form of a JavaScript call to the Ajax engine. Any reply to a user action that doesn't necessitate a trip back to the server is handled by the engine on its own. Such actions include: simple data validation, editing data in memory, and even some navigation. If the engine needs something from the server in order to respond — if it's submitting data for processing, loading additional interface code, or retrieving new data — the engine makes those requests asynchronously, usually using XML, without stalling a user's interaction with the application.