

# TỔNG QUAN VỀ AJAX\*

Do Ngọc Minh

This work is produced by OpenStax-CNIX and licensed under the Creative Commons Attribution License 3.0<sup>†</sup>

## Tóm tắt nội dung

Vào những năm 90, trên thế giới, khi mà công nghệ “Web động” cho phép các lập trình viên nhanh chóng phát triển các ứng dụng trên nền Web với khả năng tương tác 2 chiều với người sử dụng, nhiều người đã có thể hy vọng rằng đến một lúc nào đó tất cả các ứng dụng mà chúng ta sử dụng sẽ là các ứng dụng Web thay vì các phần mềm chạy độc lập trên các máy tính đơn lẻ (ứng dụng desktop). Quả thật, với sự phát triển chóng mặt của mạng Internet cùng với những ưu điểm của các ứng dụng Web (truy cập tại mọi nơi, không cần nâng cấp nhiều về phía máy sử dụng,...), tương lai của các phần mềm chắc chắn sẽ gắn chặt với các ứng dụng Web, nếu không muốn nói là có thể sẽ bị thay thế. Tuy nhiên, cho đến giờ, hy vọng đó vẫn chưa được đáp ứng một cách triệt để bởi một số điểm đặc trưng của ứng dụng Web lại chính là những giới hạn tương chừng như không thể vượt qua nổi. Đó chính là cách thức mà người dùng và ứng dụng Web tương tác với nhau.

## 1 Tổng quan về AJAX

Khác với các phần mềm chạy độc lập ở máy khách (có khả năng tương tác gần như tức thời với người dùng), các ứng dụng Web bị giới hạn bởi chính nguyên lý hoạt động của nó: tất cả các giao dịch phải thực hiện thông qua phương thức giao dịch HTTP (HyperText Transport Protocol - Giao thức truyền tải qua các siêu liên kết) trong một mô hình có tên Client/Server. Bất kỳ một tác động nào của người dùng lên ứng dụng Web thông qua trình duyệt đều cần thời gian gửi về Server và sau khi xử lý, Server sẽ trả về những thông tin người dùng mong đợi. Như vậy, độ trễ trong trường hợp này chính là điều mà các ứng dụng Web khó có thể sánh với như các phần mềm chạy trên máy tính đơn lẻ (đặc biệt là trong các ứng dụng như Bản đồ trực tuyến, soạn thảo văn bản trực tuyến, sát hạch trực tuyến có tính thời gian làm bài...).

Thuật ngữ AJAX được xuất hiện vào ngày 18/2/2005 trong một bài báo có tên AJAX : A New Approach to Web Applications [1] của tác giả Jesse James Garrett, công ty AdapativePath. Ông định nghĩa và tóm gọn lại từ cụm từ “Asynchronous JavaScript+CSS+DOM+XMLHttpRequest”. Ngay sau đó thuật ngữ AJAX được phổ biến cực kỳ nhanh chóng trong cộng đồng phát triển Web và cho đến nay nó là một trong những từ khóa được tìm kiếm nhiều nhất trên Internet.

Nội dung định nghĩa của Garrett về AJAX như sau:

AJAX không phải là một công nghệ. Nó là tập hợp của nhiều công nghệ với thế mạnh của riêng mình để tạo thành một sức mạnh mới. AJAX bao gồm:

- Thể hiện Web theo tiêu chuẩn XHTML và CSS;
- Nâng cao tính năng động và tương tác bằng DOM (Document Object Model);
- Trao đổi và xử lý dữ liệu bằng XML và XSLT;
- Truy cập dữ liệu theo kiểu bất đồng bộ (asynchronous) bằng XMLHttpRequest;

---

\*Version 1.1: Jul 12, 2009 9:59 am -0500

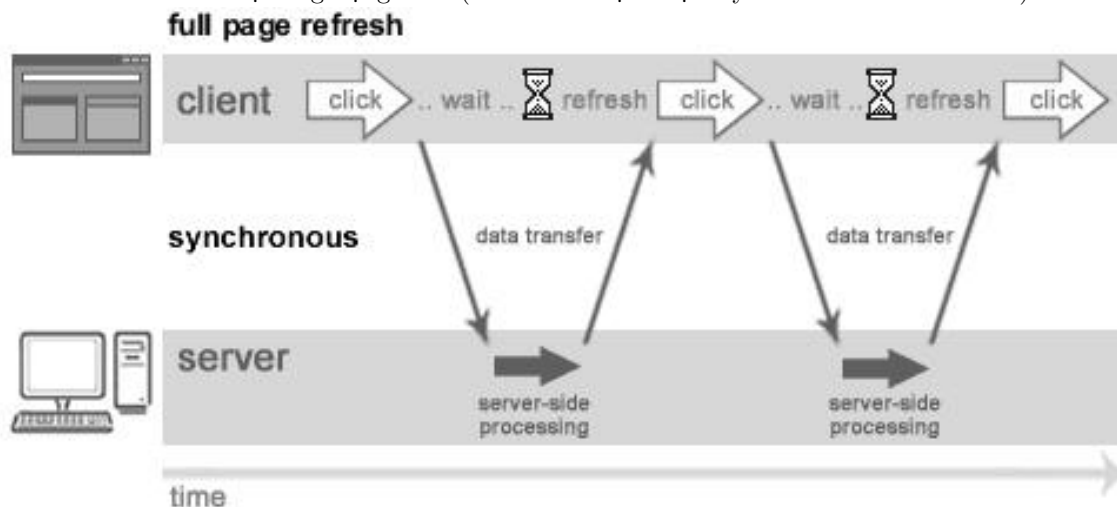
<sup>†</sup><http://creativecommons.org/licenses/by/3.0/>

- Và tất cả các kỹ thuật trên được liên kết lại với nhau bằng JavaScript.[1]

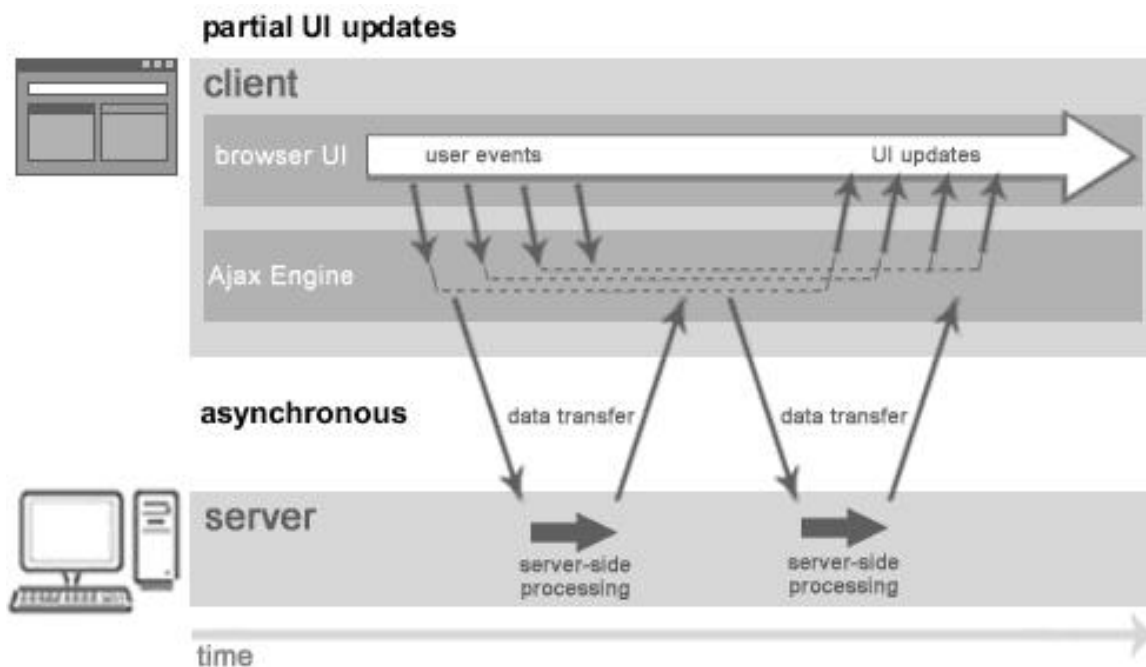
Trong các thành phần cấu thành trên, điểm mấu chốt của AJAX nằm ở XMLHttpRequest. Đây là một kỹ thuật do Microsoft khởi xướng và tích hợp lần đầu tiên vào IE5 dưới dạng một ActiveX. Mozilla tích hợp công nghệ này vào Mozilla 1.0/Netscape 6 sau đó (đương nhiên toàn bộ các version sau này của Firefox đều có XMLHttpRequest) và hiện nay đã có trong trình duyệt Safari 1.2 (Apple) và Opera 7 trở lên. Các vấn đề về XMLHttpRequest và cách sử dụng nó trên các trình duyệt khác nhau sẽ được chúng ta quay trở lại ở các phần tiếp theo. Sau đây là một số nét khác biệt cơ bản giữa các ứng dụng Web truyền thống và ứng dụng Web sử dụng AJAX.

Trong các ứng dụng Web truyền thống, khi người dùng có một cần thay đổi dữ liệu trên trang Web, yêu cầu thay đổi được gửi về server dưới dạng HTTP request (hay còn gọi postback), server sẽ xử lý yêu cầu này và gửi trả lại trang HTML khác thay thế trang cũ. Quy trình này được mô tả là nhấp-chờ và tải lại (click-wait-and-refresh): ví dụ người dùng sau khi nhấn một nút “Submit” trên trang Web phải chờ cho đến khi server xử lý xong mới có thể tiếp tục công việc. Ngược lại, trong các ứng dụng AJAX, người dùng có thể nhấp chuột, gõ phím liên tục mà không cần chờ đợi. Nội dung tương ứng với từng hành động của người dùng sẽ gần như ngay lập tức được hiển thị vào vị trí cần thiết (đáp ứng gần như tức thời) trong khi trang Web không cần phải refresh lại toàn bộ nội dung. Để tìm hiểu kỹ hơn điều này, ta sẽ xem xét 2 mô hình ứng dụng như [1] đã đề cập, Mô hình cổ điển và Mô hình AJAX-based:

Mô hình cổ điển của một ứng dụng Web (hình minh họa được lấy từ bài báo của Garrett)



Mô hình ứng dụng Web sử dụng AJAX (theo bài báo của Garrett)



Rõ ràng điểm khác biệt là thay vì phải tải cả trang Web thì với AJAX trình duyệt phía người dùng chỉ cần tải về phần của trang Web mà người dùng muốn thay đổi. Điều này giúp cho ứng dụng Web phản hồi nhanh hơn, thông minh hơn. Ngoài ra, điểm đặc biệt quan trọng trong công nghệ AJAX nằm ở chữ A (Asynchronous) – không đồng bộ – tức là người dùng cứ gửi yêu cầu của mình tới server và quay lại với công việc của mình mà không cần chờ trả lời. Khi nào server xử lý xong yêu cầu của phía người dùng, nó sẽ báo hiệu và người dùng có thể “thu nhận lấy” để thể hiện những thay đổi cần thiết. Vậy tất cả cơ chế này hoạt động thực sự thế nào? AJAX cho phép tạo ra một AJAX Engine nằm giữa giao tiếp này. Khi đó, các yêu cầu gửi (request) và nhận (response) do AJAX Engine thực hiện. Thay vì trả dữ liệu dưới dạng HTML và CSS trực tiếp cho trình duyệt, Web server có thể gửi trả dữ liệu dạng XML và AJAX Engine sẽ tiếp nhận, phân tách và chuyển hóa thành XHTML + CSS cho trình duyệt hiển thị. Việc này được thực hiện trên client nên giảm tải rất nhiều cho server, đồng thời người sử dụng cảm thấy kết quả xử lý được hiển thị tức thì mà không cần nạp lại trang. Mặt khác, sự kết hợp của các công nghệ Web như CSS và XHTML làm cho việc trình bày giao diện trang Web tốt hơn nhiều và giảm đáng kể dung lượng trang phải nạp. Đây là những lợi ích hết sức thiết thực mà AJAX đem lại. Chúng ta sẽ xem xét cụ thể các thành phần cấu thành AJAX, nguyên lý hoạt động và việc sử dụng Javascript kết nối chúng trong phần tiếp theo.