

# AJAX - Tương lai của ứng dụng Web

## Bài 7: Các công nghệ trong AJAX - XML và việc truyền dữ liệu bất đồng bộ - XML.

### XML là gì?

Năm 1969, IBM đã cho ra đời ngôn ngữ đánh dấu đầu tiên của mình với tên gọi Generalized Markup Language (GML). GML là một ngôn ngữ tự đặc tả sử dụng cho việc đánh dấu cấu trúc của một tập dữ liệu nhất định đồng thời GML được định hướng để trở thành một super language (siêu ngôn ngữ) – nghĩa là ngôn ngữ dùng để đặc tả cho ngôn ngữ khác.

Về sau GML phát triển thành SGML (Standard Generalized Markup Language). Vào năm 1986, SGML được tổ chức ISO công nhận là chuẩn lưu trữ và chuyển đổi dữ liệu. SGML được sử dụng để xây dựng những tài liệu như sách vở, báo cáo, hay cẩm nang tham khảo....Những tài liệu này sẽ được chuyển thành dạng thức biểu diễn được và sau đó gửi chúng ra thiết bị xuất như máy in, màn hình...

Tuy nhiên, trở ngại lớn nhất đối với người dùng SGML là chúng quá phức tạp và tiêu tốn nhiều công sức trong việc hiện thực. Vì vậy hầu hết người dùng cá nhân và doanh nghiệp đều không thể đáp ứng những yêu cầu để sử dụng công nghệ hữu dụng này.

Năm 1996, World Wide Web Consortium (W3C) khởi động kế hoạch xây dựng một ngôn ngữ được gọi là XML (eXtensible Markup Language – ngôn ngữ đánh dấu mở rộng) kết hợp được sức mạnh của SGML và tính phổ dụng của HTML. Sự xuất hiện của XML mang lại cho người dùng sức mạnh của SGML với tốn kém ít hơn và không phải đối mặt với sự phức tạp của SGML. Hơn nữa việc viết parser (bộ phân tích từ vựng và cú pháp) cho tài liệu XML cũng đơn giản hơn. Ngoài ra, XML tương thích với các giao thức Internet và phần mềm xử lý, chuyển đổi dữ liệu. XML được xem là một tập con của SGML, vì vậy XML có được khả năng tương thích với những hệ thống dựa trên SGML, giúp nhà phát triển vẫn có thể duy trì được những hệ thống đã được xây dựng trên nền tảng SGML mà không phải tốn kém trong việc chuyển đổi.  
(đoạn trên là copy n paste)

### Các đặc điểm của XML

1. XML tương thích với SGML
2. Dễ dàng viết được những chương trình xử lý tài liệu XML
3. Tài liệu XML dễ đọc và có tính hợp lý cao
4. XML được xây dựng với sự giảm thiểu những thuộc tính tùy chọn
5. XML dễ dàng được sử dụng trên Internet
6. XML hỗ trợ nhiều ứng dụng
7. Không đặt nặng tính hình thức trong nội dung thẻ đánh dấu XML

## Cấu trúc một tài liệu XML

Tài liệu XML chỉ chứa đựng dữ liệu và cách lưu trữ dữ liệu mà không hề đề cập tới cách thức trình bày dữ liệu. Một tài liệu XML sẽ chứa những đặc tả về cấu trúc dữ liệu. Mỗi cấu trúc gồm nhiều phần tử (element), mỗi thành phần được bắt đầu với một thẻ bắt đầu (Start-tag) và kết thúc với một thẻ kết thúc (End-tag). Giữa Start-tag và End-tag là nội dung của phần tử này. Nội dung có thể bao gồm dữ liệu văn bản hay có thể là một phần tử khác. Dưới đây là một file XML:

Trích:

```
<?xml version="1.0"?>
<Catalog>
  <Product>
    <ProductID>F10</ProductID>
    <ProductName>Shimano Calcutta </ProductName>
    <ListPrice>47.76</ListPrice>
  </Product>
  <Product>
    <ProductID>F20</ProductID>
    <ProductName>Bantam Lexica</ProductName>
    <ListPrice>49.99</ListPrice>
  </Product>
</Catalog>
```

Một tài liệu HTML có thể tồn tại một số thẻ không đúng quy định (trình biên dịch sẽ bỏ qua những thẻ này). Tuy nhiên với một tài liệu XML thì điều này không thể xảy ra. Khi xây dựng một tài liệu XML, nó phải tuân thủ theo một số quy luật nào đó. Những tài liệu XML tuân thủ đúng những quy luật này được gọi là well-formed (tạm dịch là định dạng đúng). Với một tài liệu không phải là well-formed, Internet Explorer sẽ thông báo lỗi khi nạp tài liệu này.

Một tài liệu XML well-formed chưa chắc là một tài liệu hợp lệ. Một tài liệu XML được xem là hợp lệ nếu nó đảm bảo những quy tắc đặc tả trong tài liệu Document Type Definition (DTD) hay giản đồ (schema). Một DTD hay schema sẽ định nghĩa mọi thứ từ cấu trúc dữ liệu tới kiểu dữ liệu, những thuộc tính được yêu cầu, và những ràng buộc về thành phần và thuộc tính được kết hợp trong tài liệu. Phương thức kiểm tra tài liệu này thường được sử dụng trong giao tiếp giữa ứng dụng - ứng dụng, đảm bảo dữ liệu trao đổi hợp lệ tránh dẫn tới những ảnh hưởng của dữ liệu không hợp lệ trên toàn hệ thống.

## XSLT - (eXtensible Style Language Transforming)

XSLT là một phần của XSL (eXtensible Style Language). XSL là một ngôn ngữ nền tảng XML và ra đời với mục đích chuyển đổi một tài liệu XML thành một tài liệu XML khác hay chuyển

một tài liệu XML thành những đối tượng có thể thể hiện được. Internet Explorer được tích hợp XSL transformer giúp tự động chuyển đổi tài liệu XML thành tài liệu HTML. Để sử dụng XSLT, chúng ta phải xây dựng tài liệu XSL chứa những template. Trong những template chúng ta sẽ kết hợp những phần tử HTML sẽ được xuất ra.

## Bài 8: Công nghệ trong AJAX - XML và việc truyền dữ liệu bất đồng bộ - XMLHttpRequest

*Các ứng dụng web truyền thống đều phải tải nạp toàn bộ trang web khi gửi yêu cầu đến server, điều này thường dẫn đến một sự ngắt quãng cho công việc của người dùng. Vì thế yêu cầu đặt ra là làm cho các request của server không đồng bộ và người dùng vẫn có thể làm việc trong khi đợi tín hiệu trả lời từ server. Giải pháp đầu tiên để giải quyết vấn đề này là dùng IFrame. Cho đến gần đây, XMLHttpRequest đã được đưa ra như là một giải pháp mạnh mẽ và hiệu quả.*

### XmlDocument và XMLHttpRequest

Các đối tượng XmlDocument và XMLHttpRequest không phải là các chuẩn DOM mở rộng của trình duyệt web nhưng vẫn được hỗ trợ bởi rất nhiều trình duyệt. Các đối tượng đều dựa trên ActiveX hay các đối tượng JavaScript. Các trình duyệt khác đều có thực thi các đối tượng trên có các chức năng tương đương và các lời gọi hàm API.

Một ví dụ sử dụng XmlDocument:

Trích:

```
function getXMLDocument(){
var xDoc=null;
if(document.implementation&&document.implementation.
createDocument){
xDoc=document.implementation.createDocument("", "", null);
}else if (typeof ActiveXObject != "undefined"){
var msXmlAx==null;
try{
msXmlAx=new ActiveXObject("Msxml2.DOMDocument");
}catch (e){
msXmlAx=new ActiveXObject("Msxml.DOMDocument");
}
xDoc=msXmlAx;
}
if (xDoc==null || typeof xDoc.load=="undefined"){
xDoc=null;
```

```
}  
return xDoc;  
}
```

Hàm vừa thực hiện trên sẽ trả về một đối tượng XmlDocument với các hàm API giống nhau trong các trình duyệt hiện nay.

Ví dụ sau có chức năng tương tự nhưng dùng cho đối tượng XMLHttpRequest.

Trích:

```
function getXMLHttpRequest() {  
    var xRequest=null;  
    if (window.XMLHttpRequest) {  
        xRequest=new XMLHttpRequest();  
    }else if (typeof ActiveXObject != "undefined"){  
        xRequest=new ActiveXObject("Microsoft.XMLHTTP");  
    }  
    return xRequest;  
}
```

## Gửi một yêu cầu đến server

Gửi một yêu cầu đến server từ một đối tượng XMLHttpRequest rất dễ dàng. Tất cả những việc ta cần làm là truyền cho nó địa chỉ URL của server. Ta sẽ làm như sau:

Trích:

```
function sendRequest(url,params,HttpMethod){  
    if (!HttpMethod){  
        HttpMethod="POST";  
    }  
    var req=getXMLHttpRequest();  
    if (req){  
        req.open(HttpMethod,url,true);  
        req.setRequestHeader("Content-Type","application/  
x-www-form-urlencoded");  
        req.send(params);  
    }  
}
```

XMLHttpRequest hỗ trợ một miền rất rộng các lời gọi HTTP, gồm các tham số truy vấn cho các trang web động.

### Dùng các hàm callback để giám sát các yêu cầu

Các hàm callback đón bắt các sự kiện trong cách tiếp cận lập trình hướng sự kiện trong hầu hết các bộ công cụ UI hiện nay – như nhấn bàn phím, click chuột, vì không thể biết được các sự kiện nào sẽ xảy ra, nên các lập trình viên phải đón bắt các sự kiện có thể xảy ra. Khi tạo các giao diện người dùng UI bằng JavaScript, ta gán các hàm onkeypress, onmouseover, và đặt tên tương tự cho các thuộc tính tương đương cho các đối tượng. Khi lập trình cho các yêu cầu cho server, chúng ta gặp các thuộc tính tương tự onload và onreadystatechange.

## Bài 9: Công nghệ trong AJAX - Javascript

JavaScript là một ngôn ngữ lập trình đa năng, nó tương đối giống C. JavaScript được biết dưới dạng một ngôn ngữ có kiểu tự do, thông dịch, ngôn ngữ kịch bản đa năng.

Kiểu tự do nghĩa là các biến không được khai báo cụ thể như string, integer, hay object, và các biến giống nhau có thể được gán bởi các kiểu khác nhau. Ví dụ, đoạn mã sau là hợp lệ:

Trích:

```
var x=3.1415926;  
x='pi';
```

Biến x lúc đầu được định nghĩa là giá trị số và sau đó được gán lại bởi giá trị xâu kí tự.

Thông dịch nghĩa là các mã nguồn không được biên dịch thành các đoạn mã nhị phân có thể thực thi được, và nó được thực thi trực tiếp, cụ thể là qua các trình duyệt. Khi triển khai một ứng dụng JavaScript, ta đặt mã nguồn trên web server, và mã nguồn này được truyền trực tiếp qua Internet tới web browser.

Đa năng nghĩa là ngôn ngữ này thích hợp để lập trình theo hầu hết các thuật toán và các tác vụ. JavaScript cơ bản hỗ trợ các kiểu số - number, string, date và time, array, các biểu thức toán học được xử lý trong văn bản, và các hàm toán học như các hàm lượng giác và bộ tạo số ngẫu nhiên. Hoàn toàn có thể định nghĩa cấu trúc một đối tượng bằng JavaScript, điều này mang đến nguyên lý cơ bản cho lập trình và để viết lên nhưng đoạn mã phức tạp.

Trong môi trường trình duyệt web, các chức năng cơ bản của trình duyệt, gồm CSS, DOM, và các đối tượng XMLHttpRequest, được coi là các phương tiện của JavaScript, cho phép các nhà phát triển điều khiển các trang ở các mức độ khác nhau.

Trong chõng công nghệ (tức là các công nghệ sử dụng liệt kê từ trên xuống dưới:D) của Ajax, JavaScript là chất kết dính các thành phần lại với nhau.

Học Javascript tiếng việt:

Trích:

[http://crazivn.com/uds\\_JS.rar](http://crazivn.com/uds_JS.rar)

Học bằng tiếng Anh:

Trích:

<http://crazivn.com/Oreilly.Java.Script.And.DHTML.Cookbook.eBook-LiB.rar>

Như vậy là chúng ta đã học xong tất cả các công nghệ trong AJAX, từ bài sau chúng ta bắt đầu nghiên cứu về **Đối tượng XMLHttpRequest**. và sau đó có lẽ sẽ là: **Gửi các request và xử lý response**.