## Ajax的发展历史

2005年Jesse James Garrett 发表了一篇文章，标题为: Ajax: A new Approach to Web Application。在他发表这篇文章里介绍了一种技术，用他的话说，就叫：AJAX，是Asynchronous

Javascript+xml的简写。然后就席卷全球了。

## 什么是Ajax

AJAX即“**A***synchronous***J***avascript****A****nd***X***ML*”（异步JavaScript和XML），是指一种创建交互式[网页](http://baike.baidu.com/view/828.htm" \t "_blank)应用的网页开发技术。

AJAX = 异步 [JavaScript](http://baike.baidu.com/view/16168.htm)和[XML](http://baike.baidu.com/view/63.htm" \t "_blank)（[标准通用标记语言](http://baike.baidu.com/view/5286041.htm)的子集）。

AJAX 是一种用于创建快速动态网页的技术。

通过在后台与服务器进行少量数据交换，AJAX 可以使网页实现异步更新。这意味着可以在不重新加载整个网页的情况下，对网页的某部分进行更新。

## XMLHttpRequest对象

Ajax技术核心就是XMLHttpRequest对象（简称XHR），这是有微软首先引入的一个特性，其他浏览器提供商后来也都提供了相同的实现。在XHR出现之前，ajax式的通信必须借助一些hack手段来实现，大多数是使用隐藏的框架或内嵌的框架实现的。

XHR的出现，提供了向服务器发送请求和解析服务器响应提供了流畅的接口，能够以异步方式从服务器获取更多的信息，这样就意味着，用户只要出发某事件，在不刷新王爷的情况下，更行服务器最新的数据。

但是，虽然AJAX中的X代表的是xml，但ajax通信和数据格式无关。也就是这种技术不一定适用xml。

## HTTP协议

[超文本传输协议](http://baike.baidu.com/view/468465.htm)

是[互联网](http://baike.baidu.com/view/6825.htm" \t "_blank)上应用最为广泛的一种[网络协议](http://baike.baidu.com/view/16603.htm" \t "_blank)。所有的[WWW](http://baike.baidu.com/view/1453.htm" \t "_blank)文件都必须遵守这个标准。设计HTTP最初的目的是为了提供一种发布和接收[HTML](http://baike.baidu.com/view/692.htm" \t "_blank)页面的方法。1960年美国人[Ted Nelson](http://baike.baidu.com/view/6412907.htm" \t "_blank)构思了一种通过[计算机](http://baike.baidu.com/view/3314.htm" \t "_blank)处理文本信息的方法，并称之为超文本（hypertext）,这成为了HTTP超文本传输协议标准架构的发展根基。Ted Nelson组织协调万维网协会（World Wide Web Consortium）和[互联网工程工作小组](http://baike.baidu.com/view/1941215.htm" \t "_blank)（Internet Engineering Task Force ）共同合作研究，最终发布了一系列的[RFC](http://baike.baidu.com/view/6108.htm" \t "_blank)，其中著名的RFC 2616定义了HTTP 1.1。

## 什么是异步

异步就是不需要在相同时间内统一完成某件事。可以各自分工，相继的各自完成独立的工作；互相不影响。此概念是相对于同步而讲的。

一般来说异步操作都是启动一个新的线程去处理你的请求，完成之后调用你指定的回调函数来完成整个异步操作。

## 什么是同步

"同步"。 的意思是说，你发起一个操作，程序需要一直等待，直到操作完成，程序才能继续执行进行下一步。

## 什么是进程和线程

进程和线程都是由操作系统所体现的程序运行的基本单元，系统利用该基本单元实现系统对应用的并发性。   
进程和线程的区别在于：   
简而言之,一个程序至少有一个进程,一个进程至少有一个线程.   
线程的划分尺度小于进程，使得多线程程序的并发性高。   
另外，进程在执行过程中拥有独立的内存单元，而多个线程共享内存，从而极大地提高了程序的运行效率。   
线程在执行过程中与进程还是有区别的。每个独立的线程有一个程序运行的入口、顺序执行序列和程序的出 口。但是线程不能够独立执行，必须依存在应用程序中，由应用程序提供多个线程执行控制。   
从逻辑角度来看，多线程的意义在于一个应用程序中，有多个执行部分可以同时执行。但操作系统并没有将多个线程看做多个独立的应用，来实现进程的调度和管理以及资源分配。这就是进程和线程的重要区别。   
  
进程是具有一定独立功能的程序关于某个数据集合上的一次运行活动,进程是系统进行资源分配和调度的一个独立单位. 线程是进程的一个实体,是CPU调度和分派的基本单位,它是比进程更小的能独立运行的基本单位.线程自己基本上不拥有系统资源,只拥有一点在运行中必不可少的资源(如程序计数器,一组寄存器和栈),但是它可与同属一个进程的其他的线程共享进程所拥有的全部资源.   
一个线程可以创建和撤销另一个线程;同一个进程中的多个线程之间可以并发执行.

* 进程的概念
  + 进程**≠**程序
    - 程序从开始到结束的一次执行过程叫做进程
    - 一个进程当中，程序同时运行的多个分支，叫做线程
  + 多线程异步执行，可以提高程序的效率

## 什么是同源策略

同源策略是客户端脚本（尤其是Javascript）的重要的安全度量标准。它最早出自Netscape Navigator2.0，其目的是防止某个文档或脚本从多个不同源装载。这里的同源指的是：同协议，同域名和同端口。[www.baidu.com/index.html](http://www.baidu.com/index.html) 和 [www.baidu.com/default.html](http://www.baidu.com/default.html)