

# AJAX

## 定义

Ajax即Asynchronous Javascript And XML（异步JavaScript和XML）在2005年被Jesse James Garrett提出的新术语，用来描述一种使用现有技术集合的新方法，包括: [HTML](#) 或 [XHTML](#), [CSS](#), [JavaScript](#), [DOM](#), XML, [XSLT](#), 以及最重要的[XMLHttpRequest](#)。使用Ajax技术网页应用能够快速地将增量更新呈现在[用户界面](#)上，而不需要重载（刷新）整个页面，这使得程序能够更快地回应用户的操作。

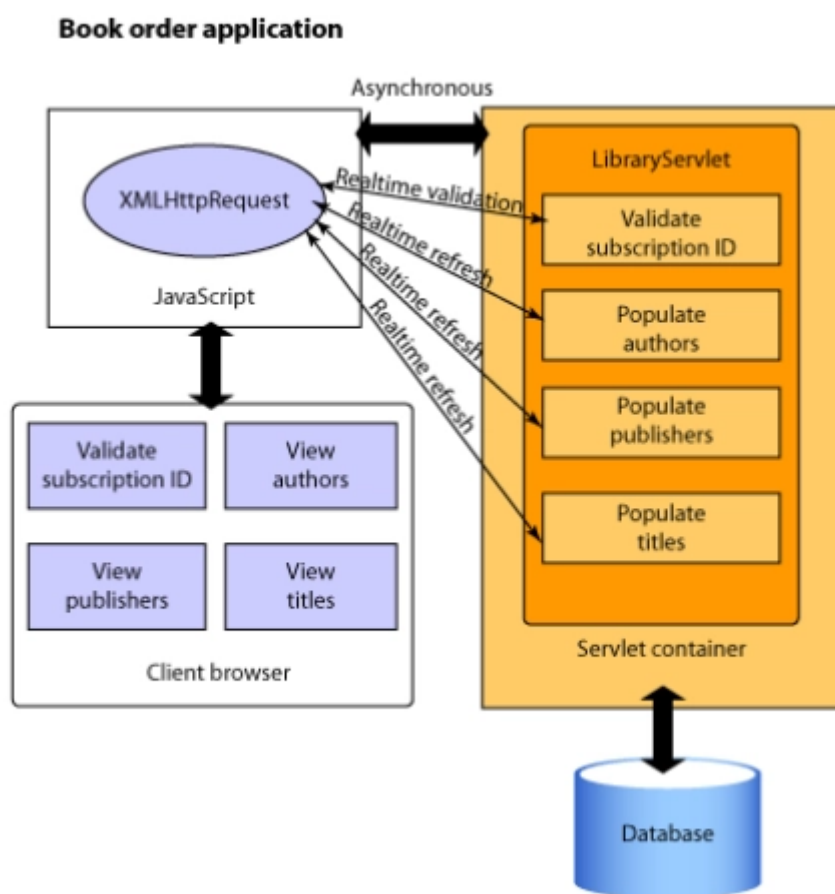
## 来源

Ajax这个术语源自描述从基于 Web 的应用到基于数据的应用。

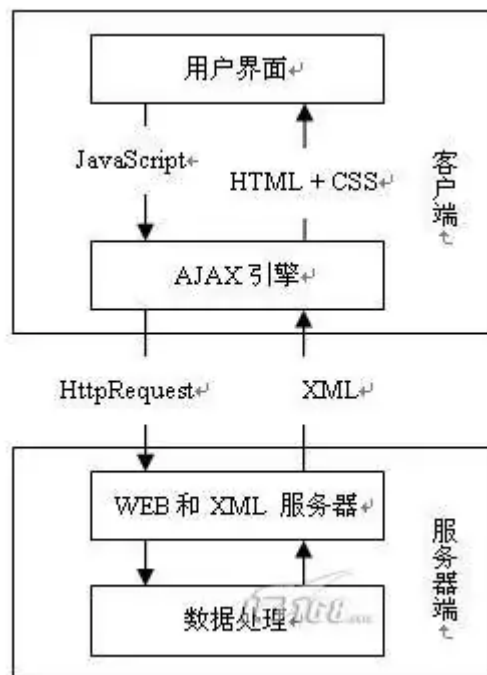
Ajax 不是一种新的编程语言，而是一种用于创建更好更快以及交互性更强的Web应用程序的技术。

使用 JavaScript 向服务器提出请求并处理响应而不阻塞用户核心对象XMLHttpRequest（xhr）。通过这个对象，您的JavaScript可在不重载页面的情况与Web服务器交换数据，即在不需要刷新页面的情况下，就可以产生局部刷新的效果。

Ajax 在浏览器与Web服务器之间使用异步数据传输（HTTP 请求），这样就可使网页从服务器请求少量的信息，而不是整个页面。



使用ajax 构建应用程序



AJAX 工作原理

Ajax可使因特网应用程序更小、更快，更友好。

Ajax 是一种独立于 Web 服务器软件的浏览器技术。 Ajax 基于下列 Web 标准：

JavaScript、XML、[HTML](#)与 [CSS](#) 在 Ajax 中使用的 Web 标准已被良好定义，并被所有的主流浏览器支持。Ajax 应用程序独立于浏览器和平台。

Web 应用程序较[桌面应用程序](#)有诸多优势；它们能够涉及广大的用户，它们更易安装及维护，也更易开发。

不过，因特网应用程序并不像传统的桌面应用程序那样完善且友好。通过 Ajax，因特网应用程序可以变得更完善，更友好。

## 模拟Ajax

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>伪造Ajax</title>
  <script src="../lib/jquery-3.6.0.js"></script>
</head>
<body>
<div>
  <p>输入要加载的地址: <span id="currentTime"></span></p>
  <p>
    <input type="text" id="url" value="">
    <input type="button" value="提交" onclick="load()">
  </p>
</div>
<div>
  <h3>加载页面的位置</h3>
  <iframe style="width: 100%;height: 600px" id="iframePosition"></iframe>
</div>
<script>
  $(function () {
    var myDate = new Date();

```

```

        document.getElementById('currentTime').innerText = new
Date(myDate.getTime());
    });
    function load() {
        var targetURL = document.getElementById('url').value;
        document.getElementById('iframePosition').src = targetURL;
    }
</script>
</body>
</html>

```

url data success

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <script src="../static/js/jquery-3.6.0.js"></script>
</head>
<body>
    用户名: <input type="text" onblur="a1()" id="txtName">

    <script>
        //所有参数:
        // url:待载入页面的URL地址
        // data:待发送key /value参数
        // success:载入成功时回调函数
        function a1() {
            $.ajax({
                url:"/ajax/a1",
                data:{"name":$("#txtName").val()},
                success:function (data, status) {
                    alert(data);
                    alert(status);
                }
            });
        }
    </script>
</body>
</html>

```