

Google 成功背后的技术 Ajax 网页程序设计



沈时宇
李宏伟
飞思科技产品研发中心

著
改编
监制

- ➔ 教您写出与GMail、Google Maps、Yahoo字典、Flickr相簿一样方便好用的网页
- ➔ 实例演练：会员申请账号即时检查、邮政编码查询、在线英文词典、拖放式购物车、简易版WIKI系统等



网站开发专家
DEVELOPMENT EXPERTS

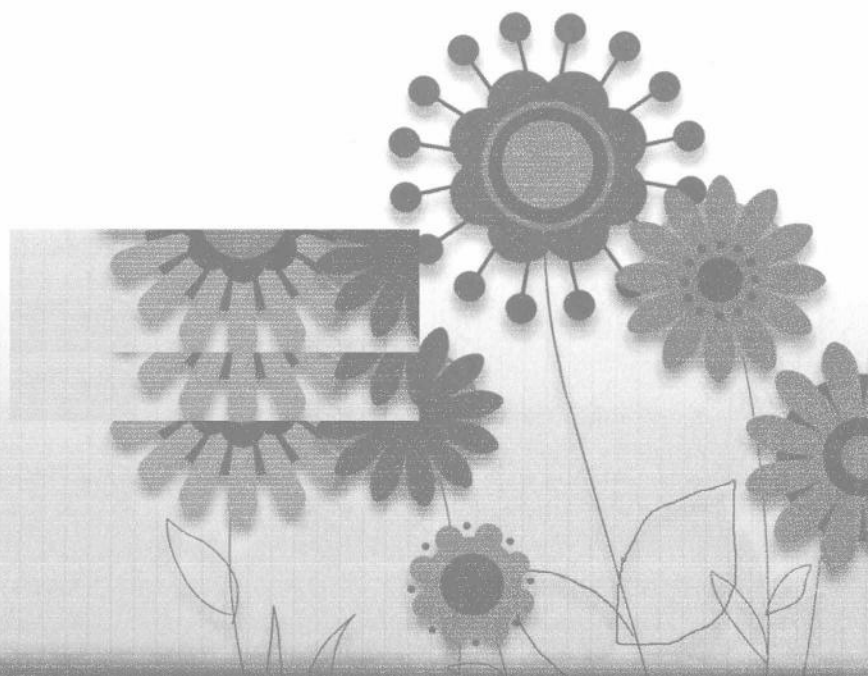
Google 成功背后的技术

Ajax

网页程序设计

沈时宇
李宏伟
飞思科技产品研发中心

著
改编
监制



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内容简介

本书详细介绍了Google背后的技术——Ajax，主要内容包括：Ajax简介、Ajax 基础、prototype.js、rico、xajax、登录系统之地址查询与用户账号验证、在线英汉词典、购物车、Wiki、Google Maps API、Google Maps 与 Ajax。本书内容丰富、实例典型、讲解详尽，为了便于读者学习，本书配套光盘中包含全部实例的源代码。

本书实例源文件请到www.fecit.com.cn的“下载专区”中下载。

本书适合于所有具有一定Web开发经验的读者阅读。

本书繁体字版名为《AJAX 網頁程式設計——Google 成功背後的技术》，由旗标出版股份有限公司出版，版权属旗标出版股份有限公司所有。本中文简体字版由旗标出版股份有限公司授权电子工业出版社独家出版发行。未经本书原版出版者和本书出版者书面许可，任何单位和个人均不得以任何方式或任何手段复制或传播本书的部分或全部。

版权贸易合同登记号 图字：01-2006 6651

图书在版编目（CIP）数据

Google 成功背后的技术：Ajax 网页程序设计 / 沈时宇著；李宏伟改编.—北京：电子工业出版社，2008.9
（网站开发专家）

ISBN 978-7-121-07115-7

I. G… II.①沈…②李… III.计算机网络—程序设计 IV.TP393.09

中国版本图书馆 CIP 数据核字（2008）第 106811 号

责任编辑：王树伟 李新承

印 刷：北京智力达印刷有限公司

装 订：北京中新伟业印刷有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：787×1092 1/16 印张：16.25 字数：416 千字

印 次：2008 年 9 月第 1 次印刷

印 数：6 000 册 定价：36.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

序

最早接触到在 2005 年时全世界有关网页开发的 Blog 都在谈论 Ajax 什么是 Ajax 呢？于是在 Google 上下，的确能找到很多与 Ajax 相关的介绍。现在用户在 Google 搜索关键字“Ajax”，会找到超过 150 000 000 条的数据。这在中的人们也是如此。在笔者编写本书能够带您步入 Ajax 殿堂。当你再看到一个类似的 Internet 服务时，也能大概猜测出其背后的实现方式。不过笔者更希望的是，通过本书中的示例引导激发读者的创造力，毕竟 Ajax 只是一种工具，如何拿它来应用才是最有价值的。

本书是笔者编写的第一本书，因为是第一次接触，有许多地方不了解，在思考如何写这本书时，不知道该怎么描述才会让读者更容易理解，幸好有赖于编辑黄大哥的帮忙润饰，也感谢我周围的其他人，在最忙时还能一直支持我。

沈时宇

说在前头

前言

万维网（WWW）自从 1990 年初期开始发展以来，历经了不少变动，但是网页的基本浏览方式是大同小异的。在早期，网页是由 HTML 格式的文件所组成，由 HTTP 传输数据的；在客户端是由浏览器（Browser）呈现这种文件格式的。这样的类型是静态的，不论是谁来浏览同一页面都会看到相同的结果。后来，随着 CGI 和 Server Side Script 的发展，以及与数据库的结合，使得 WWW 能够呈现更丰富、更有弹性的信息，而 Client Side Script 的出现（如 JavaScript 或 VBScript 等），以及与 CSS 的结合，则使得网页的呈现更加动态。

不过，就如前面所讲的，不管 WWW 如何演变，基本的浏览方式仍然是不变的。受限于 HTTP 的设计，网页的浏览方式一向都是“点击”、“页面刷新”，然后“显示结果”，这种设计可以说是一种优点，因为正是这种超链接（Hyperlink）的概念让万维网真的达到 World Wide 的发展。但以其他观点来看，也可算是一种缺点，因为它跟我们所使用软件的习惯不同。

现在，有种称为“Ajax”的技术正悄悄成形，并以铺天盖地之势席卷因特网，逐渐改变人们浏览网页的习惯。而本书正是要介绍这种看似新鲜的技术，并且通过几个网上成功的案例来带领大家了解何谓 Ajax，并让大家简简单单学会它。

谁适合看这本书

本书主要是写给对网站设计及 JavaScript 网页技术有兴趣的读者，不过所包含的层面会涵盖整体的网站设计，不只是客户端的网页编写，也包括服务器端的应用，所以阅读本书前最好先具备一些相关知识，例如 HTTP、HTML 文件格式，以及 JavaScript 等。对于这些技术的相关知识，读者如果不甚了解，建议先参考本书后面的附录。有了这些观念、基础后，相信必定能够对本书的重点 Ajax 技术有更深刻的了解。

章节安排

我自己有个习惯，在阅读一本程序语言技术书籍时最喜欢从示例开始着手，简单且清楚的示例总是让我可以快速上手，甚至常常认为一行代码可以抵过数十行的文字说明，而且经过实际演练后，从中学习到的新知识也比较能记在脑海里。所以当编写这本书时，就决定这本书应该以大量且实用的示例作为辅助，激发大家的兴趣。

不过学习还是应该以渐进的方式来进行，所以在前面的章节中笔者会介绍一些目前已经应用 Ajax 的实例，等大家从这些实例中感受到 Ajax 所带给网站的全新感受后，再开始进入 Ajax 的核心，以及如何利用目前网络上流行的 Framework 加速 Ajax 软件的开发。下面是本书的具体章节安排。

第 1 章 Ajax 简介

本章将介绍一些目前应用 Ajax 相当成功的网站,从这些应用中我们可以感受到 Ajax 带给网站的全新感觉。此外,改善网页的呈现方式并非一定得用 Ajax,你也可以采取其他不同的方案,这些方案各有优缺点,它们跟 Ajax 相比孰优孰劣在这章中都会有详细介绍。

第 2 章 Ajax 基础

本章我们将开始正式进入 Ajax 世界,如什么是“异步通信”,什么又是“Ajax 三层式架构”,这些问题都可以在这章中找到答案。此外,笔者也会带领大家了解 Ajax 的核心——XMLHttpRequest,详尽介绍 XMLHttpRequest 的使用及各种限制等。当了解了 XMLHttpRequest 后,也就已经掌握了 Ajax 的精髓。

第 3 章 prototype.js

实现 Ajax 困难吗?在本章中笔者将介绍几个网上的 Ajax Framework,利用这些 Framework,你将会了解 Ajax 的开发并不是那么困难,同时学习也变得有趣多了。

在这些 Framework 中,prototype.js 是目前在网络上最流行的,因为它具有易学易用的特性,还让 JavaScript 的面向对象观念更为完整,而且它 also 支持几个 Ajax 对象,让 Ajax 开发变得更容易,所以笔者在本章中将会详细介绍 prototype.js,其后的示例全部采用 prototype.js 以加速开发的速度。

第 4 章 rico

rico 这个 Framework 除了包含有一些视觉特效的应用外,也具备了 Ajax,不过在概念的实现上比 prototype.js 更符合三层式的架构。

第 5 章 xajax

xajax 是前端的 JavaScript 与后端的 PHP 程序整合得最完美的一套 Ajax Framework,它让开发者从另外一种角度来思考 Ajax 软件的实现方式,即使它目前尚未成为一个稳定的版本,但是笔者还是希望介绍这个有趣的 Framework 给大家。

第 6 章 登录系统地址查询与用户账号验证

从第 6 章开始,笔者设计了几个应用示例,目的就是希望读者能够从这些示例中快速学习 Ajax,进而刺激大家的创意,从而创造出更多有趣的应用来。有时,大家可能会看到一些类似的

Ajax 应用概念不断地出现在各个网站上，本书第一个完整的示例就是 Ajax 应用中常见的 Auto-Complete 功能，希望能够通过这个示例让大家了解 Auto-Complete 背后的实现原理。

第 7 章 在线英汉词典

笔者以前曾在网络中上传过一个以 Ajax 方式实现的英汉词典，得到非常热烈的响应。因此，在本书中也特别收录了这个字典的实现方式，并一步一步地指导大家。

第 8 章 购物车

在一般的电子商务网站中，购物车是常见的功能，但是随着 Ajax 的出现，我们是不是也能够整合购物车与 Ajax 呢？这个示例就是一个简单的实现，让购物车变得更具亲和力。

第 9 章 Wiki

在 Wiki 示例中，笔者将导入“就地编辑”（Edit in place）及草稿保存等概念，这些概念也都是常见的 Ajax 应用，不过传统网页上要实现这些功能并不容易，但有了 Ajax 后，一切都变得简单了。

第 10 章 Google Maps API

Google Maps 是 Google 的重要服务之一，而且它所采用的 Ajax 的方式也让全世界的网页开发者眼睛为之一亮。在 2005 年，Google 推出的这项服务备受用户欢迎，不仅仅是它炫丽的 Ajax 界面，更由于它趁势推出了 API 程序库让全世界有兴趣的人都能够一同参与，所以在本书最后两章中笔者将以实例的方式一步一步地介绍 Google Maps API 的使用。本章为 Google Maps API 的简要概述。

第 11 章 Google Maps 与 Ajax

除了一般利用 API 实现地图的介绍外，笔者在本章中也会将 Google Maps API 与 Ajax 整合利用，并借助实例进行介绍。

注：①本书所用地图仅为说明编程技术而用，因图书出版时间关系，若与现在的实际地图有差异，请以实际地图为准，本书中的地图不作为地理方位指示之用。

②本书中涉及到的地名，部分为虚构地名，如有雷同，实属巧合。

③本书中用到的地图、地名仅为说明编程技术，部分地图、地名与原书有差异，同样也不作为地理方位指示之用。

目 录

第 1 篇 Ajax 入门

第 1 章 Ajax 简介	3
1.1 传统网站设计上的优缺点	3
1.1.1 等待时间	3
1.1.2 页面刷新	3
1.2 其他替代方案	4
1.2.1 Flash	5
1.2.2 Java Applet	6
1.3 Ajax 的诞生	7
1.3.1 Google Maps	7
1.3.2 Google Suggest	10
1.3.3 GMail	11
1.3.4 Flickr	14
1.3.5 异步通信	17
1.4 Flash、Java Applet 及 Ajax 的比较	19
第 2 章 Ajax 基础	21
2.1 Ajax 基本原理	21
2.1.1 使用 Ajax 之前	21
2.1.2 使用 iframe	23
2.1.3 Ajax 三层架构	25
2.2 Ajax 的核心 XMLHttpRequest	29
2.2.1 设置异步传输完成函数	32
2.2.2 使用 open() 函数初始设置	33
2.2.3 使用 send() 传输	37
2.2.4 取得服务器的响应	38
2.3 不同浏览器上的 XMLHttpRequest	39
2.4 Ajax 的安全性问题	41

2.4.1	网络服务与安全问题	41
2.4.2	Ajax 的限制	43
2.4.3	如何突破 Ajax 的安全限制	45
2.4.4	访问需要账号和密码的网页	49

第 2 篇 AjaxFramework

第 3 章	prototype.js	53
3.1	为什么要使用 Framework	53
3.2	prototype.js 简介	53
3.2.1	prototype.js 的版权	54
3.2.2	如何应用 prototype.js	54
3.2.3	prototype.js 的面向对象概念	55
3.3	JavaScript 原有对象的扩展	57
3.3.1	Array 对象	57
3.3.2	Number 对象	61
3.3.3	String 对象	61
3.3.4	Event 对象	63
3.4	新增加的对象	66
3.4.1	哈希 (Hash) 对象	66
3.4.2	PeriodicalExecuter 对象	68
3.5	Ajax 对象	71
3.5.1	prototype.js 的 Ajax 示例	71
3.5.2	Ajax 对象	79
3.5.3	Ajax.Base 对象	80
3.5.4	Ajax.Request 对象	82
3.5.5	Ajax.Responders 对象	82
3.5.6	Ajax.Updater 对象	83
3.5.7	Ajax.PeriodicalUpdater 对象	85
3.6	prototype.js 后记	86
第 4 章	rico	87
4.1	rico 简介	87
4.2	rico 的版权	87
4.3	rico 示例: 旅游网站	87

4.4	rico 旅游网站示例改进	91
4.5	rico 后记	94
第 5 章	xajax	95
5.1	xajax 简介	95
5.2	xajax 版权	95
5.3	使用 xajax	95
5.4	xajax 的第 1 个示例	96
5.5	xajax 的第 2 个示例	100
5.6	xajax 重要对象及函数整理	103
5.6.1	xajax 对象	103
5.6.2	xajaxResponse 对象	103
5.7	xajax 后记	104

第 3 篇 Ajax 实例演练

第 6 章	登录系统之地址查询与用户账号验证	107
6.1	简介	107
6.2	数据准备	109
6.3	程序架构	111
6.4	地址查询后端处理	111
6.4.1	列举地址	111
6.4.2	深入筛选	112
6.5	前端的用户处理	115
6.5.1	版面设置	117
6.5.2	事件处理	117
6.5.3	下拉菜单	122
6.5.4	检验用户账号	126
6.6	改进空间	128
第 7 章	在线英汉词典	129
7.1	简介	129
7.2	数据准备	129
7.3	Ajax 词典后端代码	132
7.3.1	寻找单词	132

7.3.2 列举单词	135
7.4 Ajax 词典前端代码	137
7.4.1 版面设置	139
7.4.2 查询单词	140
7.5 其他改进的方法	144
第 8 章 购物车	145
8.1 简介	145
8.2 程序说明	147
8.2.1 版面设置	148
8.2.2 拖放功能	150
第 9 章 Wiki	159
9.1 什么是 Wiki	159
9.2 程序架构	160
9.3 代码说明	161
9.3.1 Wiki 引擎前端代码	161
9.3.2 后端代码	174
9.4 本章小结	176
第 10 章 Google Maps API	177
10.1 Google Maps 简介	177
10.2 开始的第一步: 申请 API 密钥	178
10.3 开始使用 Google Maps API	179
10.3.1 引用 API	180
10.3.2 定义 Container	180
10.3.3 建立 GMap 对象	181
10.3.4 加上控件	184
10.4 在地图上增加其他信息	185
10.4.1 标示地点	185
10.4.2 使用自定义图标标示地点	187
10.4.3 显示信息	189
10.5 地图彩妆——画上线条	191
10.5.1 程序说明	192
10.5.2 运行结果	193

10.6	Google Maps API 的事件处理	194
10.6.1	单击按钮触发事件	194
10.6.2	显示经纬度	196
10.7	本章小结	200
第 11 章	Google Maps 与 Ajax	201
11.1	Ajax 相关对象	201
11.1.1	GXmlHttp	201
11.1.2	GXslt	202
11.1.3	XML 文件	205
11.1.4	XSLT 文件	205
11.2	利用 Ajax 加载地点数据	207
11.3	本章小结	214

第 4 篇 附录

附录 A	HTTP	217
A.1	HTTP 简介	217
A.2	HTTPS	219
附录 B	JavaScript	221
B.1	JavaScript 简介	221
B.2	语法简介	222
B.2.1	变量	223
B.2.2	基本语法	223
B.2.3	函数	224
B.3	DOM	225
B.4	JavaScript 实例演练	227
B.5	面向对象	230
B.5.1	数组 (Array)	231
B.5.2	字符串 (String)	232
B.5.3	对象 (Object)	233
附录 C	XMLHttpRequest API	237
C.1	XMLHttpRequest	237

C.1.1	IE 上取得该对象的方法	237
C.1.2	其他浏览器上取得该对象的方法	237
C.2	open()	238
C.3	send()	238
C.4	onreadystatechange	238
C.5	responseXML	239
C.6	responseText	240
C.7	readyState	240
C.8	status	241
C.9	statusText	241
C.10	abort()	242
C.11	getAllResponseHeaders()	243
C.12	getResponseHeader()	243
C.13	setRequestHeader()	244
C.14	onload	245

第 1 篇

Ajax 入门

第 1 章 Ajax 简介

第 2 章 Ajax 基础

第 1 章 Ajax 简介

1.1 传统网站设计上的优缺点

传统网站的运作机制与架构都是建立在 HTTP 上的。HTTP 是个简单的协议，自然会有许多限制存在，不管是利用 CGI、Server Side Script (PHP 或 ASP 等) 来设计任何网站，都必须限制于 HTTP 一个请求 (Request)，然后一个响应 (Response) 的设计。在 WWW 刚起步的阶段，这样的设计并无不妥，浏览器的“上一页”、“下一页”及“历史记录” (History) 的功能也都是因此而存在的。

但是随着网站的应用越来越广，甚至已经有越来越多的产品不再使用传统 Windows 的使用界面，而是以网页来操作产品，我们就有必要来重新思考，目前的网站设计存在着什么缺陷。

1.1.1 等待时间

现在多数网站背后所要做的事情包罗万象，CGI 或 Server Side Script 处理需要时间，单击超链接后就会产生一段等待时间。虽然说网站可以利用与后端数据库整合，提高服务器的等级，或者改善整体服务的程序架构等办法缩短等待时间，但是一段不少的时间仍是不可避免的。

1.1.2 页面刷新

正如前面所提到的，HTTP 的设计是返回整个新的页面，所以在浏览器上就必须刷新整个页面来显示新的页面如图 1-1 所示，这样的设计会有几个缺点。

(1) 页面的刷新会带来一种不流畅的感觉，用户每次单击后都使页面清空再重来，这样的用户界面设计基本上来说并不合理，或许可以利用其他 HTML 的技巧进行改进，如框架 (Frame) 的使用，但是可改进的空间并不大，甚至会带来其他缺点。

(2) 如果想要设计一种会动态更新的页面，例如，动态显示所有上线用户数，传统做法都是做个框架 (Frame)，利用更新 (Refresh) 的方式来定期更新该框架中的数据，但是这种做法的最大缺点是造成页面的闪烁。正如前面所提到的，浏览器每次收到网站服务器的响应时，都是先清空页面再重新显示，所以当更新的频率快到某种程度时会给用户一种闪烁的感觉，这是不太合理的用户界面设计。

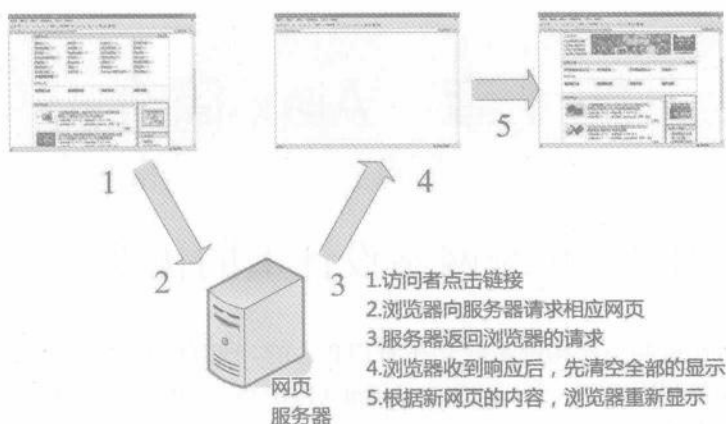


图 1-1

(3) 上一页、下一页的操作。其实严格来说，笔者并不认为网页“上一页”和“下一页”的设计不好，相反这是可以让用户随时查看过往记录的好工具。但是在复杂的网页设计中，如果对浏览器的“上一页”、“下一页”没有好的设计，就会给用户带来不便。

举例来说，可能大部分用户都有这样的经验，当使用某些有搜索功能的网站输入了搜索的关键字符串后，浏览器上出现了一堆相关的数据，因为数据众多需要进行分页，在浏览的过程中需要不断地单击“下一页”，但是当你再使用浏览器的“上一页”功能时，浏览器可能会重新弹出一个窗口告诉你再次发送信息。不过即使单击了“确定”按钮，所产生的网页还可能会出现“请求超时”等错误消息，而这些不愉快的体验，也往往让用户在使用“上一页”功能时疑虑再三。

但是利用网页作为用户界面还是有很多优点的，最主要的优点是开发时间短，容易开发（开发环境及语言繁多，开发者可以挑选自己熟悉进行开发），容易在不同平台间互相兼容；而对于一般软硬件的管理界面来说，网页式的管理也利于远程管理，同时因为 HTTP 也是广泛使用的标准通信协议，用 HTTP 不需要考虑穿透防火墙等棘手的问题，所以多数网络设备都是以 HTML 加上 HTTP 的组合来开发网页界面的，而安全性的问题也由 HTTPS 来解决，所以当用户使用网页作为界面时只需要考虑数据及页面即可，是非常方便的。

1.2 其他替代方案

如果在网页上开发用户界面，还有哪些替代方案呢？其他替代方案很多，这也是网页的另一个优点——扩展性高，目前在 WWW 上有很多各式各样的浏览器插件（Plug-In）来弥补原本 HTML 的不足，其中使用最多的应该是 Flash 和 Java Applet。