

jQuery Internals

Architecture and Implementation in Depth

jQuery 技术内幕

深入解析 jQuery 架构设计与实现原理



高云 著



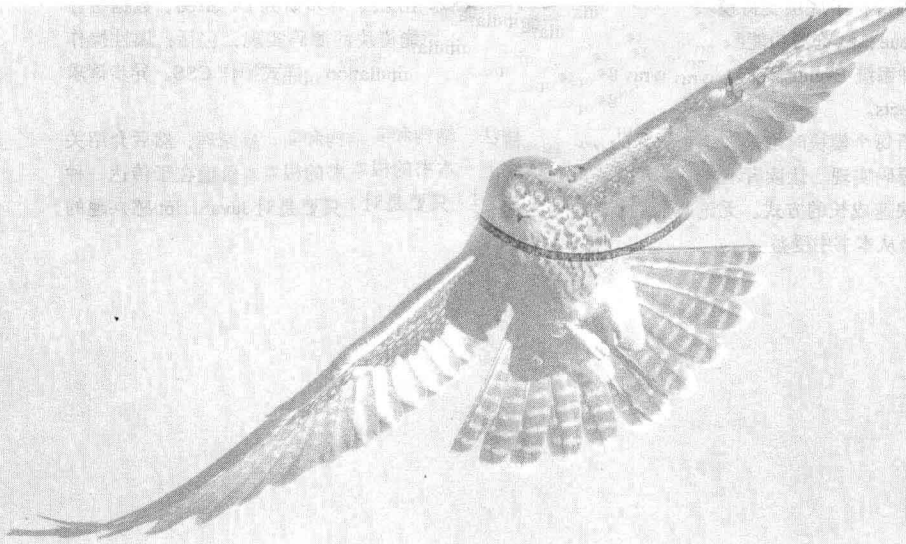
机械工业出版社
China Machine Press

jQuery Internals

Architecture and Implementation in Depth

jQuery技术内幕

深入解析jQuery架构设计与实现原理



高云 著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

jQuery 技术内幕：深入解析 jQuery 架构设计与实现原理 / 高云著. —北京：机械工业出版社，2013.11

ISBN 978-7-111-44082-6

I. j… II. 高… III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2013) 第 221662 号

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问 北京市展达律师事务所

本书由阿里巴巴资深前端开发工程师撰写，从源代码角度全面而系统地解读了 jQuery 的 17 个模块的架构设计理念和内部实现原理，旨在帮助读者参透 jQuery 中的实现技巧和技术精髓，同时本书也对广大开发者如何通过阅读源代码来提升编码能力和软件架构能力提供了指导。

本书首先通过“总体架构”梳理了各个模块的分类、功能和依赖关系，让大家对 jQuery 的工作原理有大致的印象；进而通过“构造 jQuery 对象”章节分析了构造函数 jQuery() 的各种用法和内部构造过程；接着详细分析了底层支持模块的源码实现，包括：选择器 Sizzle、异步队列 Deferred、数据缓存 Data、队列 Queue、浏览器功能测试 Support；最后详细分析了功能模块的源码实现，包括：属性操作 Attributes、事件系统 Events、DOM 遍历 Traversing、DOM 操作 Manipulation、样式操作 CSS、异步请求 Ajax、动画 Effects。

本书在分析每个模块时均采用由浅入深的方式，先概述功能、用法、结构和实现原理，然后介绍关键步骤和分析源码实现。让读者不仅知其然，而且知其所以然。事实上，本书的根本价值在于传达一种通过阅读源码快速成长的方式。无论是前端新人，还是经验丰富的老手，只要是对 JavaScript 感兴趣的开发人员，都会从本书中受益。

机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码 100037）

责任编辑：朱秀英

藁城市京瑞印刷有限公司印刷

2014 年 1 月第 1 版第 1 次印刷

186mm × 240mm · 39.5 印张

标准书号：ISBN 978-7-111-44082-6

定 价：99.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：(010) 88378991 88361066

投稿热线：(010) 88379604

购书热线：(010) 68326294 88379649 68995259

读者信箱：hzjsj@hzbook.com

jQuery 是业界最流行的 JavaScript 库，其 API 非常精致和优雅，但是 jQuery 的源码却庞大且晦涩难懂，在本书开始写作时发布的 1.7.1 版本有 9266 行代码，涉及 17 个模块，读起来常常是一头雾水、有心无力。本书尝试对 jQuery 的源码进行系统、完整的介绍和分析，阐述 jQuery 的设计理念、实现原理和源码实现。

为什么要写这本书

笔者在 2010 年参与了一款卫星机顶盒用户界面的设计和开发，程序运行在机顶盒中间件供应商提供的一款定制浏览器上，在开发过程中，发现这款浏览器的行为类似于古老的 IE 5，各种缺陷和 bug 折磨得笔者苦不堪言，所以希望引入 jQuery 作为基础库，并开发一些通用组件和接口来简化开发过程，可是很快又发现这款浏览器对正则表达式的支持非常粗糙，导致选择器引擎 Sizzle 根本无法运行。此时，对 jQuery 进行简单改造已经满足不了需求。

然而令人惊艳的是，这款浏览器提供了与操作系统、文件系统、中间件、播放器、智能卡和卫星接收器等交互的 JavaScript API，例如，待机 & 关机、文件读写、计费、卫星锁频、数据接收等。鉴于这种复杂的体系架构，以及对浏览器缺陷的完善也非短期可以完成，笔者开始为这款机顶盒浏览器移植 jQuery，从而开始了对 jQuery 源码的学习和分析。

从 2011 年 6 月开始，笔者开始把心得和记录整理成《jQuery 1.6.1 源码分析系列》，陆续发表在程序员社区 ITEye[Ⓔ]和博客园[Ⓕ]上，本书最初的内容也是基于这个系列而来的。《jQuery 1.6.1 源码分析系列》成体系但尚粗糙不堪，因此本书基于 jQuery 1.7.1 几乎全部重写，在内容上更加完善和严谨。

希望本书对读者能有所帮助。

读者对象

本书适合初级、中级、高级前端开发工程师，以及对前端开发感兴趣的读者。

在阅读本书之前，读者应该初步掌握 JavaScript、HTML、CSS 的基础知识，初步掌握 jQuery 的使用，或者有其他语言基础。

Ⓔ <http://iteye.com>, <http://nuyssoft.iteye.com/>。

Ⓕ <http://cnblog.com>, <http://cnblogs.com/nuyssoft/>。

如何阅读本书

本书共分为四大部分，首先介绍了 jQuery 的总体架构，然后分别分析了构造 jQuery 对象模块、底层支持模块和功能模块的源码实现。在阅读本书时，首先建议读者建立一个源码阅读和调试环境，在阅读过程中进行各种尝试和验证，加深对源码的理解；在阅读本书的每个章节前，建议读者先仔细阅读相应的官方文档，并验证官方示例，掌握 API 的功能和用法。

第一部分（第 1 章）对 jQuery 的设计理念、总体架构和源码结构进行了介绍和分析，让读者对 jQuery 有整体的认识。

第二部分（第 2 章）详细介绍和分析了构造函数 jQuery() 的用法、构造过程、原型属性和方法、静态属性和方法。

第三部分（第 3 ~ 7 章）详细分析了底层支持模块的源码实现，包括选择器 Sizzle、异步队列 Deferred Object、数据缓存 Data、队列 Queue、浏览器功能测试 Support。

第四部分（第 8 ~ 14 章）详细分析了功能模块的源码实现，包括属性操作 Attributes、事件系统 Events、DOM 遍历 Traversing、DOM 操作 Manipulation、样式操作 CSS、异步请求 Ajax、动画 Effects。

勘误和支持

由于笔者水平有限，再加上写作时的疏漏，书中难免存在许多需要改进之处。在此，欢迎读者朋友们指出书中存在的问题，并提出指导性意见，不胜感谢。提交地址为 <https://github.com/nuysoft/jquery-errata-support/issues>，勘误内容将在 <http://nuysoft.com/jquery.html> 上发布。

致谢

首先要向 jQuery 作者 John Resig、jQuery 团队和 jQuery 社区致敬，是你们繁荣了 JavaScript。

感谢机械工业出版社的杨绣国（Lisa）编辑，她对本书进行了大量润色，修订和批注的内容比我写的内容要多得多。本书的进度和内容在写作期间不断变更，因为她的不断激励和反复修订，才使本书得以呈现在广大读者面前。感谢朱秀英编辑，她修正了书稿中诸多不完善之处。感谢为本书拾遗补缺的诸多幕后编辑。

感谢为本书初稿给出反馈意见的所有人：古西风（叶克良）、左莫（徐波）、逸才（陈养剑）、崇志（李德强）、李志博、王阳光、符宝（徐丽丽）、余鹏、许杰。你们的宝贵意见使本书内容更加完善、更加准确。

本书最初的内容以及得到出版的机遇，源自发表在 ITEye 和博客园的一系列博客，感谢这两个社区，以及因此结识的朋友们。

高云（nuysoft）

目 录

前言

第一部分 总体架构

第 1 章 总体架构	2
1.1 设计理念	2
1.2 总体架构	2
1.3 自调用匿名函数	4
1.4 总结	6

第二部分 构造 jQuery 对象

第 2 章 构造 jQuery 对象	8
2.1 构造函数 jQuery()	8
2.1.1 jQuery(selector [, context])	9
2.1.2 jQuery(html [, ownerDocument]), jQuery(html, props)	9
2.1.3 jQuery(element), jQuery(elementArray)	10
2.1.4 jQuery(object)	10
2.1.5 jQuery(callback)	11
2.1.6 jQuery(jQuery object)	11
2.1.7 jQuery()	11
2.2 总体结构	11
2.3 jQuery.fn.init(selector, context, rootjQuery)	13
2.3.1 12 个分支	13
2.3.2 源码分析	14
2.3.3 小结	21
2.4 jQuery.buildFragment(args, nodes, scripts)	22
2.4.1 实现原理	22

2.4.2	源码分析	22
2.4.3	小结	26
2.5	jQuery.clean(elems, context, fragment, scripts)	27
2.5.1	实现原理	27
2.5.2	源码分析	27
2.5.3	小结	39
2.6	jQuery.extend()、jQuery.fn.extend()	40
2.6.1	如何使用	40
2.6.2	源码分析	40
2.7	原型属性和方法	43
2.7.1	.selector、.jquery、.length、.size()	44
2.7.2	.toArray()、.get([index])	45
2.7.3	.each(function(index, Element))、jQuery.each(collection, callback(indexInArray, valueOfElement))	46
2.7.4	.map(callback(index, domElement))、jQuery.map(arrayOrObject, callback(value, indexOrKey))	47
2.7.5	.pushStack(elements, name, arguments)	49
2.7.6	.end()	51
2.7.7	.eq(index)、.first()、.last()、.slice(start [, end])	51
2.7.8	.push(value, ...)、.sort([orderfunc])、.splice(start,deleteCount, value, ...)	52
2.7.9	小结	53
2.8	静态属性和方法	54
2.8.1	jQuery.noConflict([removeAll])	55
2.8.2	类型检测: jQuery.isFunction(obj)、jQuery.isArray(obj)、jQuery.isWindow(obj)、jQuery.isNumeric(value)、jQuery.type(obj)、jQuery.isPlainObject(object)、jQuery.isEmptyObject(object)	56
2.8.3	解析 JSON 和 XML: jQuery.parseJSON(data)、jQuery.parseXML(data)	60
2.8.4	jQuery.globalEval(code)	65
2.8.5	jQuery.camelCase(string)	65
2.8.6	jQuery.nodeName(elem, name)	66
2.8.7	jQuery.trim(str)	67
2.8.8	数组操作方法: jQuery.makeArray(obj)、jQuery.inArray(value, array [, fromIndex])、jQuery.merge(first, second)、jQuery.grep(array, function(elementOfArray, indexInArray) [, invert])	68
2.8.9	jQuery.guid、jQuery.proxy(function, context)	72
2.8.10	jQuery.access(elems, key, value, exec, fn(elem, key, value), pass)	74
2.8.11	jQuery.error(message)、jQuery.noop()、jQuery.now()	75

2.8.12	浏览器嗅探: jQuery.uaMatch(ua)、jQuery.browser	76
2.8.13	小结	77
2.9	总结	77

第三部分 底层支持模块

第 3 章	选择器 Sizzle	80
3.1	总体结构	81
3.2	选择器表达式	83
3.3	设计思路	84
3.4	Sizzle(selector, context, results, seed)	86
3.5	正则 chunker	94
3.6	Sizzle.find(expr, context, isXML)	94
3.7	Sizzle.filter(expr, set, inplace, not)	99
3.8	Sizzle.selectors.relative	103
3.8.1	"+"	105
3.8.2	<td>106</td>	106
3.8.3	" "	108
3.8.4	"~"	108
3.8.5	dirCheck(dir, cur, doneName, checkSet, nodeCheck, isXML)	109
3.8.6	dirNodeCheck(dir, cur, doneName, checkSet, nodeCheck, isXML)	111
3.9	Sizzle.selectors	112
3.9.1	Sizzle.selectors.order	112
3.9.2	Sizzle.selectors.match/leftMatch	113
3.9.3	Sizzle.selectors.find	122
3.9.4	Sizzle.selectors.preFilter	123
3.9.5	Sizzle.selectors.filters	129
3.9.6	Sizzle.selectors.setFilters	132
3.9.7	Sizzle.selectors.filter	133
3.10	工具方法	140
3.10.1	Sizzle.uniqueSort(results)	140
3.10.2	sortOrder(a, b)	141
3.10.3	Sizzle.contains(a, b)	144
3.10.4	Sizzle.error(msg)	145
3.10.5	Sizzle.getText(elem)	145
3.11	便捷方法	146
3.11.1	Sizzle.matches(expr, set)	146

3.11.2	Sizzle.matchesSelector(node, expr)	146
3.12	jQuery 扩展	147
3.12.1	暴露 Sizzle 给 jQuery	147
3.12.2	.find(selector)	148
3.12.3	.has(target)	149
3.12.4	.not(selector)、.filter(selector)	150
3.12.5	.is(selector)	152
3.12.6	.closest(selectors, context)	153
3.12.7	.index(elem)	154
3.12.8	.add(selector, context)	155
3.12.9	jQuery.filter(expr, elems, not)	156
3.12.10	:animated	157
3.12.11	hidden、:visible	157
3.13	总结	158
第 4 章	异步队列 Deferred Object	160
4.1	jQuery.Callbacks(flags)	161
4.1.1	实现原理和总体结构	162
4.1.2	源码分析	163
4.1.3	小结	174
4.2	jQuery.Deferred(func)	174
4.2.1	实现原理和总体结构	176
4.2.2	源码分析	177
4.2.3	小结	183
4.3	jQuery.when(deferreds)	184
4.3.1	实现原理	185
4.3.2	源码分析	185
4.4	异步队列在 jQuery 中的应用	187
4.5	总结	188
第 5 章	数据缓存 Data	189
5.1	实现原理	189
5.1.1	为 DOM 元素附加数据	189
5.1.2	为 JavaScript 对象附加数据	191
5.2	总体结构	192
5.3	jQuery.acceptData(elem)	193
5.4	jQuery.data(elem, name, data, pvt)、jQuery._data(elem, name, data, pvt)	194

5.4.1	如何使用	194
5.4.2	源码分析	194
5.4.3	jQuery._data(elem, name, data)	199
5.4.4	小结	201
5.5	.data(key,value)	201
5.5.1	如何使用	201
5.5.2	源码分析	202
5.5.3	小结	206
5.6	jQuery.removeData(elem,name,pvt)、.removeData(key)	207
5.6.1	如何使用	207
5.6.2	源码分析	207
5.6.3	小结	212
5.7	.removeData(key)	213
5.8	jQuery.cleanData(elems)	213
5.8.1	应用场景	213
5.8.2	源码分析	214
5.8.3	小结	217
5.9	jQuery.hasData(elem)	217
5.10	总结	218
第 6 章	队列 Queue	219
6.1	如何使用	219
6.1.1	Ajax 队列	220
6.1.2	动画队列 + Ajax 队列	220
6.1.3	基于 JavaScript 对象	221
6.2	实现原理	221
6.3	总体结构	222
6.4	jQuery.queue(elem,type,data)	223
6.5	jQuery.dequeue(elem,type)	224
6.6	.queue(type,data)	227
6.7	.dequeue(type)	228
6.8	.delay(time,type)	229
6.9	.clearQueue(type)	230
6.10	jQuery._mark(elem,type)、jQuery._unmark(force,elem,type)	230
6.11	.promise(type,object)	232
6.11.1	如何使用	232
6.11.2	实现原理	233

6.11.3	源码分析	233
6.11.4	handleQueueMarkDefer(elem,type,src)	235
6.12	总结	237
第 7 章	浏览器功能测试 Support	238
7.1	总体结构	238
7.2	DOM 测试 (15 项)	241
7.2.1	leadingWhitespace	241
7.2.2	tbody	242
7.2.3	htmlSerialize	243
7.2.4	hrefNormalized	245
7.2.5	checkOn	246
7.2.6	noCloneChecked	248
7.2.7	optSelected	250
7.2.8	optDisabled	251
7.2.9	getSetAttribute	253
7.2.10	deleteExpando	256
7.2.11	enctype	258
7.2.12	html5Clone	259
7.2.13	radioValue	261
7.2.14	checkClone	263
7.2.15	appendChecked	264
7.3	样式测试 (3 项)	266
7.3.1	style	266
7.3.2	opacity	268
7.3.3	cssFloat	272
7.4	盒模型测试 (10 项)	273
7.4.1	reliableMarginRight	273
7.4.2	reliableHiddenOffsets	276
7.4.3	boxModel	278
7.4.4	inlineBlockNeedsLayout	280
7.4.5	shrinkWrapBlocks	282
7.4.6	doesNotAddBorder、doesAddBorderForTableAndCells	285
7.4.7	fixedPosition	287
7.4.8	subtractsBorderForOverflowNotVisible	290
7.4.9	doesNotIncludeMarginInBodyOffset	292
7.5	事件测试 (4 项)	294

7.5.1	noCloneEvent	294
7.5.2	submitBubbles、changeBubbles、focusinBubbles	296
7.6	Ajax 测试 (2 项)	298
7.6.1	ajax	298
7.6.2	cors	300
7.7	总结	301

第四部分 功能模块

第 8 章	属性操作 Attributes	306
8.1	总体结构	307
8.2	jQuery.attr(elem, name, value, pass)	308
8.2.1	源码分析	308
8.2.2	boolHook	311
8.2.3	nodeHook	313
8.2.4	jQuery.attrHooks	314
8.2.5	小结	319
8.3	.attr(name, value)	319
8.4	jQuery.removeAttr(elem, value)	321
8.4.1	源码分析	321
8.4.2	小结	322
8.5	.removeAttr(name)	323
8.6	jQuery.prop(elem, name, value)	323
8.6.1	源码分析	323
8.6.2	jQuery.propHooks	325
8.6.3	小结	326
8.7	.prop(name, value)	327
8.8	.removeProp(name)	327
8.9	.addClass(className)	328
8.9.1	源码分析	328
8.9.2	小结	330
8.10	.removeClass([className])	330
8.10.1	源码分析	331
8.10.2	小结	333
8.11	.toggleClass([className][, switch])	333
8.11.1	源码分析	334
8.11.2	小结	336

8.12	.hasClass(selector)	336
8.12.1	源码分析	336
8.12.2	小结	337
8.13	.val([value])	338
8.13.1	源码分析	338
8.13.2	jQuery.valHooks	340
8.13.3	小结	343
8.14	总结	344
第 9 章	事件系统 Events	346
9.1	总体结构	346
9.2	实现原理	350
9.3	jQuery 事件对象	353
9.3.1	构造函数 jQuery.Event(src, props)	355
9.3.2	原型对象 jQuery.Event.prototype	357
9.3.3	事件属性修正方法 jQuery.event.fix(event)	360
9.4	绑定事件	367
9.4.1	.on(events [, selector] [, data] , handler(eventObject))	367
9.4.2	jQuery.event.add(elem, types, handler, data, selector)	370
9.5	移除事件	379
9.5.1	.off(events [, selector] [, handler(eventObject)])	379
9.5.2	jQuery.event.remove(elem, types, handler, selector, mappedTypes)	382
9.6	事件响应	388
9.6.1	主监听函数	388
9.6.2	jQuery.event.dispatch(event)	390
9.7	手动触发事件	396
9.7.1	.trigger(eventType [, extraParameters])、.triggerHandler(eventType [, extraParameters])	396
9.7.2	jQuery.event.trigger(event, data, elem, onlyHandlers)	397
9.8	事件修正和模拟 jQuery.event.special	406
9.8.1	ready	408
9.8.2	load	408
9.8.3	focus、blur	409
9.8.4	beforeunload	409
9.8.5	mouseenter、mouseleave	410
9.8.6	submit	412
9.8.7	change	413

9.8.8	focusin、focusout	416
9.8.9	jQuery.event.simulate(type, elem, event, bubble)	417
9.9	事件便捷方法	418
9.10	组合方法	419
9.10.1	.toggle(handler(eventObject), handler(eventObject) [, handler(eventObject)])	419
9.10.2	.hover(handlerIn(eventObject) [, handlerOut(eventObject)])	421
9.11	ready 事件	421
9.11.1	总体结构	421
9.11.2	.ready(handler)	424
9.11.3	jQuery.bindReady()	424
9.11.4	jQuery.holdReady(hold)	427
9.11.5	jQuery.ready(wait)	428
9.12	总结	430
第 10 章	DOM 遍历 Traversing	433
10.1	总体结构	434
10.2	遍历函数	435
10.3	工具函数	437
10.3.1	jQuery.dir(elem, dir, until)	437
10.3.2	jQuery.nth(cur, result, dir, elem)	439
10.3.3	jQuery.sibling(n, elem)	440
10.4	模板函数	441
10.5	总结	443
第 11 章	DOM 操作 Manipulation	444
11.1	总体结构	444
11.2	插入元素	445
11.2.1	核心方法 .domManip(args, table, callback)	445
11.2.2	.append(content [, content])	451
11.2.3	.prepend(content [, content])	452
11.2.4	.before(content [, content])	452
11.2.5	.after(content [, content])	452
11.2.6	.appendTo(target)、.prependTo(target)、.insertBefore(target)、 .insertAfter(target)	453
11.2.7	.html([value])	454
11.2.8	.text([text])	458
11.3	删除元素	459

11.3.1	.remove(selector, keepData)	459
11.3.2	.empty()	460
11.3.3	.detach(selector)	460
11.4	复制元素	461
11.4.1	.clone(dataAndEvents, deepDataAndEvents)	461
11.4.2	jQuery.clone(elem, dataAndEvents, deepDataAndEvents)	461
11.4.3	cloneFixAttributes(src, dest)	465
11.5	替换元素	467
11.5.1	.replaceWith(value)	467
11.5.2	.replaceAll(target)	469
11.6	包裹元素	469
11.6.1	.wrapAll(html)	469
11.6.2	.wrapInner(html)	470
11.6.3	.wrap(html)	471
11.6.4	.unwrap()	471
11.7	总结	472
第 12 章 样式操作 CSS		474
12.1	内联样式、计算样式	475
12.1.1	总体结构	475
12.1.2	.css(name, value)	476
12.1.3	jQuery.style(elem, name, value, extra)	477
12.1.4	jQuery.css(elem, name, extra)	481
12.1.5	curCSS(elem, name)、getComputedStyle(elem, name)、 currentStyle(elem, name)	483
12.1.6	jQuery.cssHooks	486
12.2	坐标 Offset	492
12.2.1	总体结构	492
12.2.2	.offset(options)	493
12.2.3	jQuery.offset.setOffset(elem, options, i)	498
12.2.4	jQuery.offset.bodyOffset(body)	500
12.2.5	.position()	501
12.2.6	.offsetParent()	502
12.2.7	.scrollLeft(val)、.scrollTop(val)	503
12.3	尺寸 Dimensions	504
12.3.1	总体结构	504
12.3.2	getWH(elem, name, extra)	505

12.3.3	.innerHeight()、.innerWidth()	508
12.3.4	.outerHeight(margin)、.outerWidth(margin)	509
12.3.5	.height(size)、.width(size)	509
12.3.6	小结	513
12.4	总结	513
第 13 章	异步请求 Ajax	516
13.1	总体结构	517
13.2	jQuery.ajax(url, options)	519
13.3	前置过滤器、请求发送器的初始化和执行	540
13.3.1	初始化	540
13.3.2	执行	543
13.4	前置过滤器	545
13.4.1	json、jsonp	545
13.4.2	script	548
13.4.3	小结	549
13.5	请求发送器	549
13.5.1	script	549
13.5.2	XMLHttpRequest	552
13.5.3	小结	560
13.6	数据转换器	561
13.6.1	初始化	561
13.6.2	执行	562
13.6.3	小结	566
13.7	Ajax 事件	567
13.8	便捷方法	568
13.8.1	jQuery.get(url, data, callback, type)、jQuery.post(url, data, callback, type)	569
13.8.2	jQuerygetJSON(url, data, callback)、jQuery.getScript(url, callback)	569
13.8.3	.load(url, params, callback)	570
13.9	工具方法	573
13.9.1	.serialize()	573
13.9.2	jQuery.param(a, traditional)	574
13.9.3	.serializeArray()	577
13.10	总结	579
第 14 章	动画 Effects	582
14.1	总体结构	583

14.2	动画入口	586
14.2.1	.animate(prop, speed, easing, callback)	586
14.2.2	jQuery.speed(speed, easing, fn)	588
14.2.3	doAnimation()	590
14.2.4	jQuery.fx(elem, options, prop)	595
14.2.5	jQuery.fx.prototype.show()	595
14.2.6	jQuery.fx.prototype.hide()	596
14.2.7	小结	596
14.3	动画执行	597
14.3.1	jQuery.fx.prototype.custom(from, to, unit)	598
14.3.2	jQuery.fx.tick()	599
14.3.3	jQuery.fx.prototype.step(gotoEnd)	600
14.3.4	jQuery.easing	604
14.3.5	jQuery.fx.prototype.update()	604
14.3.6	jQuery.fx.step	605
14.4	停止动画 .stop(type, clearQueue, gotoEnd)	606
14.5	便捷方法	609
14.5.1	生成动画样式集 genFx(type, num)	609
14.5.2	显示隐藏 .show/hide/toggle()	610
14.5.3	渐显渐隐 .fadeIn/fadeOut/fadeTo/fadeToggle()	613
14.5.4	滑入滑出 .slideDown/slidUp/slideToggle()	614
14.6	总结	615