

《jQuery 技术内幕》

图书基本信息

书名：《jQuery 技术内幕》

13位ISBN编号：9787111440826

10位ISBN编号：711144082X

出版时间：2014-1-1

出版社：机械工业出版社

作者：高云

页数：700

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《jQuery 技术内幕》

内容概要

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。

《jQuery 技术内幕》

作者简介

高云，阿里巴巴资深前端开发工程师（花名墨智），拥有丰富的前端开发经验，擅长 HTML5、CSS、JavaScript 等 Web 开发技术和 Java 技术，对 JavaScript 相关技术的应用和实现原理有深入研究和独到见解，曾担任项目经理、前端架构师等职。个人博客 <http://nuyssoft.com>，开源作品有 <http://mockjs.com>。

书籍目录

第1章 总体架构

1

1.1 设计理念

1

1.2 总体架构

1

1.3 自调用匿名函数

4

1.4 总结

6

第2章 构造 jQuery 对象

6

2.1 构造函数 jQuery()

7

2.1.1 jQuery(selector [, context])

7

2.1.2 jQuery(html [, ownerDocument])、jQuery(html, props)

8

2.1.3 jQuery(element)、jQuery(elementArray)

9

2.1.4 jQuery(object)

9

2.1.5 jQuery(callback)

10

2.1.6 jQuery(jQuery object)

10

2.1.7 jQuery()

10

2.2 总体结构

10

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

24

2.4 jQuery.buildFragment(args, nodes, scripts)

24

2.4.1 实现原理

24

2.4.2 源码分析

25

2.4.3 小结

30

2.5 jQuery.clean(elems, context, fragment, scripts)

31	
2.5.1 实现原理	
31	
2.5.2 源码分析	
32	
2.5.3 小结	
48	
2.6 jQuery.extend()、jQuery.fn.extend()	
48	
2.6.1 如何使用	
48	
2.6.2 源码分析	
49	
2.7 原型属性和方法	
53	
2.7.1 .selector、.jquery、.length、.size()	
54	
2.7.2 .toArray()、.get([index])	
55	
2.7.3 .each(function(index, Element))、jQuery.each(collection, callback(indexInArray, valueOfElement))	
56	
2.7.4 .map(callback(index, domElement))、jQuery.map(arrayOrObject, callback(value, indexOrKey))	
59	
2.7.5 .pushStack(elements, name, arguments)	
62	
2.7.6 .end()	
64	
2.7.7 .eq(index)、.first()、.last()、.slice(start [, end])	
64	
2.7.8 .push(value, ...)、.sort([orderfunc])、.splice(start,deleteCount, value, ...)	
65	
2.7.9 小结	
66	
2.8 静态属性和方法	
67	
2.8.1 jQuery.noConflict([removeAll])	
69	
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)	
70	
2.8.3 解析 JSON 和 XML : jQuery.parseJSON(data)、jQuery.parseXML(data)	
75	
2.8.4 jQuery.globalEval(code)	
81	
2.8.5 jQuery.camelCase (string)	
82	
2.8.6 jQuery.nodeName (elem, name)	
83	

2.8.7 jQuery.trim(str)

84

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])

85

2.8.9 jQuery.guid、 jQuery.proxy(function, context)

91

2.8.10 jQuery.access(elems, key, value, exec, fn(elem, key, value), pass)

94

2.8.11 jQuery.error(message)、 jQuery.noop()、 jQuery.now()

96

2.8.12 浏览器嗅探 : jQuery.uaMatch(ua)、 jQuery.browser

96

2.8.13 小结

98

2.9 总结

100

第3章 选择器 Sizzle

100

3.1 总体结构

101

3.2 选择器表达式

104

3.3 设计思路

105

3.4 Sizzle(selector, context, results, seed)

107

3.5 正则 chunker

119

3.6 Sizzle.find(expr, context, isXML)

121

3.7 Sizzle.filter(expr, set, inplace, not)

124

3.8 Sizzle.selectors.relative

129

3.8.1 "+"

130

3.8.2 ">"

132

3.8.3 ""

133

3.8.4 "~"

134

3.8.5 dirCheck(dir, cur, doneName, checkSet, nodeCheck, isXML)

135

3.8.6 dirNodeCheck(dir, cur, doneName, checkSet, nodeCheck, isXML)

136

3.9 Sizzle.selectors

138

3.9.1 Sizzle.selectors.order	138
3.9.2 Sizzle.selectors.match/leftMatch	139
3.9.3 Sizzle.selectors.find	151
3.9.4 Sizzle.selectors.preFilter	154
3.9.5 Sizzle.selectors.filters	162
3.9.6 Sizzle.selectors.setFilters	165
3.9.7 Sizzle.selectors.filter	166
3.10 工具方法	176
3.10.1 Sizzle.uniqueSort(results)	176
3.10.2 sortOrder(a, b)	177
3.10.3 Sizzle.contains(a, b)	182
3.10.4 Sizzle.error(msg)	182
3.10.5 Sizzle.getText(elem)	183
3.11 便捷方法	184
3.11.1 Sizzle.matches(expr, set)	184
3.11.2 Sizzle.matchesSelector(node, expr)	184
3.12 jQuery 扩展	186
3.12.1 暴露 Sizzle 给 jQuery	186
3.12.2 .find(selector)	186
3.12.3 .has(target)	188
3.12.4 .not(selector)、.filter(selector)	189
3.12.5 .is(selector)	191
3.12.6 .closest(selectors, context)	193
3.12.7 .index(elem)	195
3.12.8 .add(selector, context)	

196
3.12.9 jQuery.filter(expr, elems, not)
197
3.12.10 :animated
198
3.12.11 :hidden、 :visible
199
3.13 总结
200
第4章 异步队列 Deferred Object
201
4.1 jQuery.Callbacks(flags)
202
4.1.1 实现原理和总体结构
204
4.1.2 源码分析
205
4.1.3 小结
219
4.2 jQuery.Deferred(func)
220
4.2.1 实现原理和总体结构
223
4.2.2 源码分析
224
4.2.3 小结
232
4.3 jQuery.when(deferreds)
232
4.3.1 实现原理
234
4.3.2 源码分析
234
4.4 异步队列在 jQuery 中的应用
237
4.5 总结
238
第5章 数据缓存 Data
238
5.1 实现原理
239
5.1.1 为 DOM 元素附加数据
239
5.1.2 为 JavaScript 对象附加数据
241
5.2 总体结构
242
5.3 jQuery.acceptData(elem)
244

5.4 jQuery.data(elem, name, data, pvt)、 jQuery._data(elem, name, data, pvt)	245
5.4.1 如何使用	245
5.4.2 源码分析	246
5.4.3 jQuery._data(elem, name, data)	253
5.4.4 小结	254
5.5 .data(key, value)	256
5.5.1 如何使用	256
5.5.2 源码分析	256
5.5.3 小结	261
5.6 jQuery.removeData(elem, name, pvt)、 .removeData(key)	262
5.6.1 如何使用	262
5.6.2 源码分析	263
5.6.4 小结	269
5.7 .removeData(key)	270
5.8 jQuery.cleanData(elems)	271
5.8.1 应用场景	271
5.8.2 源码分析	273
5.8.3 小结	276
5.9 jQuery.hasData(elem)	277
5.10 总结	278
第6章 队列 Queue	279
6.1 如何使用	280
AJAX 队列	280
动画队列 + AJAX 队列	281
基于 JavaScript 对象	

281	
6.2 实现原理	
282	
6.3 总体结构	
282	
6.4 jQuery.queue(elem, type, data)	
284	
6.5 jQuery.dequeue(elem, type)	
286	
6.6 .queue(type, data)	
288	
6.7 .dequeue(type)	
290	
6.8 .delay(time, type)	
291	
6.9 .clearQueue(type)	
292	
6.10 jQuery._mark(elem, type)、jQuery._unmark(force, elem, type)	
293	
6.11 .promise(type, object)	
295	
6.11.1 如何使用	
295	
6.11.2 实现原理	
296	
6.11.3 源码分析	
297	
6.11.4 handleQueueMarkDefer(elem, type, src)	
299	
6.12 总结	
301	
第7章 浏览器功能测试 Support	
302	
7.1 总体结构	
302	
7.2 DOM 测试 (15 项)	
305	
7.2.1 leadingWhitespace	
305	
7.2.2 tbody	
307	
7.2.3 htmlSerialize	
309	
7.2.4 hrefNormalized	
311	
7.2.5 checkOn	
313	
7.2.6 noCloneChecked	
315	

7.2.7 optSelected	318
7.2.8 optDisabled	320
7.2.9 getSetAttribute	322
7.2.10 deleteExpando	326
7.2.11 enctype	328
7.2.12 html5Clone	330
7.2.13 radioValue	332
7.2.14 checkClone	334
7.2.15 appendChecked	337
7.3 样式测试 (3 项)	339
7.3.1 style	339
7.3.2 opacity	342
7.3.3 cssFloat	346
7.4 盒模型测试 (10 项)	349
7.4.1 reliableMarginRight	349
7.4.2 reliableHiddenOffsets	352
7.4.3 boxModel	355
7.4.4 inlineBlockNeedsLayout	358
7.4.5 shrinkWrapBlocks	361
7.4.6 doesNotAddBorder、doesAddBorderForTableAndCells	365
7.4.7 fixedPosition	369
7.4.8 subtractsBorderForOverflowNotVisible	372
7.4.9 doesNotIncludeMarginInBodyOffset	375
7.5 事件测试 (4 项)	378
7.5.1 noCloneEvent	

378
7.5.2 submitBubbles、changeBubbles、focusinBubbles
381
7.6 Ajax 测试 (2 项)
383
7.6.1 ajax
383
7.6.2 cors
386
7.7 总结
387
第8章 属性操作 Attributes
392
8.1 总体结构
394
8.2 jQuery.attr(elem, name, value, pass)
395
8.2.1 源码分析
396
8.2.2 boolHook
400
8.2.3 nodeHook
403
8.2.4 jQuery.attrHooks
404
8.2.5 小结
411
8.3 .attr(name, value)
412
8.4 jQuery.removeAttr(elem, value)
413
8.4.1 源码分析
414
8.4.2 小结
416
8.5 .removeAttr(name)
416
8.6 jQuery.prop(elem, name, value)
417
8.6.1 源码分析
417
8.6.2 jQuery.propHooks
420
8.6.3 小结
421
8.7 .prop(name, value)
422
8.8 .removeProp(name)
423

8.9 .addClass(className)	424
8.9.1 源码分析	425
8.9.2 小结	427
8.10 .removeClass([className])	428
8.10.1 源码分析	429
8.10.2 小结	431
8.11 .toggleClass([className] [, switch])	432
8.11.1 源码分析	432
8.11.2 小结	435
8.12 .hasClass(selector)	436
8.12.1 源码分析	436
8.12.2 小结	437
8.13 .val([value])	437
8.13.1 源码分析	438
8.13.2 jQuery.valHooks	441
8.13.3 小结	445
8.14 总结	446
第9章 事件系统 Events	448
9.1 总体结构	450
9.2 实现原理	455
9.3 jQuery 事件对象	457
9.3.1 构造函数 jQuery.Event(src, props)	459
9.3.2 原型对象 jQuery.Event.prototype	462
9.3.3 事件属性修正方法 jQuery.event.fix(event)	466
9.4 绑定事件	

475	
9.4.1	.on(events [, selector] [, data] , handler(eventObject))
475	
9.4.2	jQuery.event.add(elem, types, handler, data, selector)
479	
9.5	移除事件
491	
9.5.1	.off(events [, selector] [, handler(eventObject)])
491	
9.5.2	jQuery.event.remove(elem, types, handler, selector, mappedTypes)
495	
9.6	事件响应
504	
9.6.1	主监听函数
504	
9.6.2	jQuery.event.dispatch(event)
505	
9.7	手动触发事件
514	
9.7.1	.trigger(eventType [, extraParameters])、.triggerHandler(eventType [, extraParameters])
514	
9.7.2	jQuery.event.trigger(event, data, elem, onlyHandlers)
515	
9.8	事件修正和模拟 jQuery.event.special
528	
9.8.1	ready
529	
9.8.2	load
530	
9.8.3	focus、blur
530	
9.8.4	beforeunload
531	
9.8.5	mouseenter、mouseleave
532	
9.8.6	submit
534	
9.8.7	change
537	
9.8.8	focusin、focusout
540	
9.8.9	jQuery.event.simulate(type, elem, event, bubble)
541	
9.9	事件便捷方法
543	
9.10	组合方法
545	
9.10.1	.toggle(handler(eventObject), handler(eventObject) [, handler(eventObject)])
545	

9.10.2 .hover(handlerIn(eventObject) [, handlerOut(eventObject)])

546

9.11 ready 事件

547

9.11.1 总体结构

547

9.11.2 .ready(handler)

551

9.11.3 jQuery.bindReady()

551

9.11.4 jQuery.holdReady(hold)

556

9.11.5 jQuery.ready(wait)

557

9.12 总结

559

第10章 DOM 遍历 Traversing

562

10.1 总体结构

564

10.2 遍历函数

566

10.3 工具函数

569

10.3.1 jQuery.dir(elem, dir, until)

569

10.3.2 jQuery.nth(cur, result, dir, elem)

571

10.3.3 jQuery.sibling(n, elem)

572

10.4 模板函数

574

10.5 总结

576

第11章 DOM 操作 Manipulation

577

11.1 总体结构

577

11.2 插入元素

579

11.2.1 核心方法 .domManip(args, table, callback)

579

11.2.2 .append(content [, content])

587

11.2.3 .prepend(content [, content])

587

11.2.4 .before(content [, content])

588

11.2.5 .after(content [, content])

588	
11.2.6	.appendTo(target)、.prependTo(target)、.insertBefore(target)、.insertAfter(target)
589	
11.2.7	.html([value])
590	
11.2.8	.text([text])
595	
11.3	删除元素
598	
11.3.1	.remove(selector, keepData)
598	
11.3.2	.empty()
599	
11.3.3	.detach(selector)
599	
11.4	复制元素
600	
11.4.1	.clone(dataAndEvents, deepDataAndEvents)
600	
11.4.2	jQuery.clone(elem, dataAndEvents, deepDataAndEvents)
601	
11.4.3	cloneFixAttributes(src, dest)
606	
11.5	替换元素
609	
11.5.1	.replaceWith(value)
609	
11.5.2	.replaceAll(target)
612	
11.6	包裹元素
612	
11.6.1	.wrapAll(html)
612	
11.6.2	.wrapInner(html)
613	
11.6.3	.wrap(html)
614	
11.6.4	.unwrap()
615	
11.7	总结
615	
第12章	样式操作 CSS
618	
12.1	内联样式、计算样式
619	
12.1.1	总体结构
619	
12.1.2	.css(name, value)
621	

12.1.3 jQuery.style(elem, name, value, extra)

622

12.1.4 jQuery.css(elem, name, extra)

628

12.1.5 curCSS(elem, name)、getComputedStyle(elem, name)、currentStyle(elem, name)

631

12.1.6 jQuery.cssHooks

634

12.2 坐标 Offset

643

12.2.1 总体结构

643

12.2.2 .offset(options)

644

12.2.3 jQuery.offset.setOffset(elem, options, i)

650

12.2.4 jQuery.offset.bodyOffset(body)

654

12.2.5 .position()

655

12.2.6 .offsetParent()

656

12.2.7 .scrollLeft(val)、.scrollTop(val)

657

12.3 尺寸 Dimensions

659

12.3.1 总体结构

659

12.3.2 getWH(elem, name, extra)

660

12.3.3 .innerHeight()、.innerWidth()

664

12.3.4 .outerHeight(margin)、.outerWidth(margin)

665

12.3.5 .height(size)、.width(size)

666

12.3.6 小结

670

12.4 总结

671

第13章 异步请求 Ajax

673

13.1 总体结构

674

13.2 jQuery.ajax(url, options)

677

13.3 前置过滤器、请求发送器的初始化和执行

705

13.3.1 初始化

706	
13.3.2 执行	
709	
13.4 前置过滤器	
713	
13.4.1 json、jsonp	
713	
13.4.2 script	
716	
13.4.3 小结	
717	
13.5 请求发送器	
718	
13.5.1 script	
718	
13.5.2 XMLHttpRequest	
722	
13.5.3 小结	
733	
13.6 数据转换器	
733	
13.6.1 初始化	
734	
13.6.2 执行	
736	
13.6.3 小结	
741	
13.7 Ajax 事件	
742	
13.8 便捷方法	
744	
13.8.1 jQuery.get(url, data, callback, type)、jQuery.post(url, data, callback, type)	
744	
13.8.2 jQuerygetJSON(url, data, callback)、jQuery.getScript(url, callback)	
745	
13.8.3 .load(url, params, callback)	
745	
13.9 工具方法	
750	
13.9.1 .serialize()	
750	
13.9.2 jQuery.param(a, traditional)	
751	
13.9.3 .serializeArray()	
756	
13.10 总结	
757	
第14章 动画 Effects	
761	

14.1 总体结构	762
14.2 动画入口	766
14.2.1 .animate(prop, speed, easing, callback)	766
14.2.2 jQuery.speed(speed, easing, fn)	769
14.2.3 doAnimation()	772
14.2.4 jQuery.fx(elem, options, prop)	778
14.2.5 jQuery.fx.prototype.show()	779
14.2.6 jQuery.fx.prototype.hide()	780
14.2.7 小结	780
14.3 动画执行	781
14.3.1 jQuery.fx.prototype.custom(from, to, unit)	781
14.3.2 jQuery.fx.tick()	784
14.3.3 jQuery.fx.prototype.step(gotoEnd)	785
14.3.4 jQuery.easing	790
14.3.5 jQuery.fx.prototype.update()	791
14.3.6 jQuery.fx.step	791
14.4 停止动画 .stop(type, clearQueue, gotoEnd)	793
14.5 便捷方法	797
14.5.1 生成动画样式集 genFx(type, num)	797
14.5.2 显示隐藏 .show/hide/toggle()	798
14.5.3 渐显渐隐 .fadeIn/fadeOut/fadeTo/fadeToggle()	802
14.5.4 滑入滑出 .slideDown/slidUp/slideToggle()	803
14.6 总结	804

精彩短评

- 1、非常棒的一本书，大大降低了阅读jquery源码的难度（虽然还是非常难）。
- 2、年终最后一天读完，赞赞！
- 3、讲的挺详细的
- 4、没有完全读完此书，前端技术日益迭代更新的今天，jQuery 的时代也许已经过去了，不过这不妨碍我们吸收其中的思想，毕竟这些才是智慧。

能看得出作者在很用心地在写这本书，了解一下 jQuery 的设计思想可以看下本书。

- 5、讲解jQuery的原理，很详细。看完后觉得jQuery真是复杂，我还是就用用好了。
- 6、jQuery对象
- 7、好书
- 8、吸收不高，还需要把callback，deferred，queue，data等再看一遍
- 9、这种书，与其说吃透，不如当做工具书，但是太多的浏览器兼容性问题，现在看来反而没有必要了，还有高度的封装性，让人看起来非常吃力。读源码的意义在我看来，一是让自己更加顺畅的使用类库，做修改调整。二是让自己去深入理解作者的思想，加速自己对以后相同类型类库的学习过程，有能力的话自己再去根据这套思想再定向造轮子。
- 10、好多hack，好复杂。可以对照着网易云上的jquery源码讲解看。
- 11、挑选了几个自己感兴趣的模块阅读，对jquery有了一个新的认识

《jQuery 技术内幕》

精彩书评

- 1、对源码的解读顺序安排的很合理，而且对每个模块还配置了流程图，很棒的一本书，看的出来作者想让小白看懂，花了不少心思~，哎，豆瓣太不合理了，写的少了还不让发，太让人蛋疼了哎，豆瓣太不合理了，写的少了还不让发，太让人蛋疼了哎，豆瓣太不合理了，写的少了还不让发，太让人蛋疼了
- 2、刚好最近在研究jQuery源码，看完JavaScript程序设计之后，用原生js写了几个小库，然后感觉差不多了，就来研究一下jQuery的源码，看了的大概，并未完全理解透彻，然后看见这本书了，先去新华书店随便翻翻的，仔细一看，讲的挺细致的，果断买下。回去系统的研究一下jQuery。总的来说：不错。

章节试读

1、《jQuery 技术内幕》的笔记-第48页

看了clean里面的几段代码，发现jquery中大量用到了正则表达式，而且每一个都十分的精妙。再看作者非常详尽的解释，有一种看破天书的快感。别的不说了，就说rxhtmlTag = /<(?!area|br|col|embed|hr|img|input|link|meta|param)(([w:]+)[^>]*)\>/ig你知道是啥意思吗？

2、《jQuery 技术内幕》的笔记-第4页

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com