

Web前端开发 最佳实践

党建 著

Web Front-End Development
Best Practices

Web前端开发领域经典著作，深入分析Web前端开发中与HTML、CSS、JavaScript以及Web移动端开发相关的技术方案、技巧和最佳实践，包含大量范例，极具指导意义



机械工业出版社
China Machine Press

Web前端开发 最佳实践

Web Front-End Development
Best Practices

党建 著

图书在版编目 (CIP) 数据

Web 前端开发最佳实践 / 党建著. —北京: 机械工业出版社, 2015.1
(Web 开发技术丛书)

ISBN 978-7-111-48843-9

I. W… II. 党… III. ①超文本标记语言 - 程序设计 ②网页制作工具 ③ JAVA 语言 - 程序设计 IV. ① TP312 ② TP393.092

中国版本图书馆 CIP 数据核字 (2014) 第 293448 号

Web 前端开发最佳实践

出版发行: 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: 谢晓芳

责任校对: 董纪丽

印 刷: 三河市宏图印务有限公司

版 次: 2015 年 1 月第 1 版第 1 次印刷

开 本: 170mm × 242mm 1/16

印 张: 13.75

书 号: ISBN 978-7-111-48843-9

定 价: 59.00 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: hzsj@hzbook.com

版权所有 · 侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

为什么要写这本书

Web 前端开发入门难度并不高，但是初学者如果没有一个很好的学习和编码习惯，则开发水平的提高速度会变得很慢。下面几点是影响 Web 前端开发者技术提高的主要因素。

其一是开发者缺乏良好的实践指导。Web 前端兴起的时间不长，很多大学都还没有来得及开一门专门讲解 Web 前端的课程，因此，大部分的 Web 前端开发者都是通过自学的方式来了解 Web 前端相关的技术。开发者学习前端技术的渠道很多，其中很大一部分是通过查找网络资源的方式，而网络上充斥着大量的错误或者过时的实践方法，这些实践方法很容易误导初学者，使得初学者养成了一些不良的开发习惯。因此，Web 前端开发的初学者非常需要一些能帮助他们快速、正确地进行 Web 前端开发的最佳实践方法。

其二是开发者容易产生浮躁的情绪。Web 前端虽然兴起时间不长，但是发展很迅速，技术的更新换代也非常快，这本来是一件好事情，但也使得很多 Web 前端开发者产生了浮躁的情绪。开发者疲于追逐各种最新的技术，而忽略了最基础的技术概念。这些开发者对 Node.js、AngularJS、Polymer 等相关概念相当熟悉，但对 HTML 标准规范、AJAX 的原理、CSS 选择器的优先级等基本的概念却理解得不够深入。实质上，虽然 Web 前端技术更新换代速度很快，但也离不开 HTML、CSS、JavaScript 这 3 个基本的技术。Web 前端的开发者应该明白：只有掌握了扎实的基础知识，才能在学习新技术的时候更容易上手，并且更快地理解新技术的原理，否则只是接触新技术的皮毛而已。

其三是开发者不善于利用各种开发工具。人类文明之所以快速发展，和人类善于使用工具是分不开的。人类是感性动物，并不能保证每次都能正确地做同样的事情，但工具不一样，只要环境相同，工具每次处理的结果应该是一样的。类比到 Web 前端开发中，聪明的程序员应该学会利用各种辅助开发工具来提高开发效率，减少不必要的错误。Web 前端开发中的工具覆盖了开发过程的方方面面，包括设计、开发、测试、部署等阶段的辅助工具。善于使用工具也是 Web 前端开发中必备的一项技能。

笔者从 2002 年开始接触软件开发，先后接触过各种主流的开发技术，包括各种 Web 开发技术。从 2008 年开始，主要从事 Web 前端开发的工作。在这么多年的 Web 网站开发过程中，笔者深知技术基础、编码规范、辅助开发工具对软件开发的重要性，因此，在平时的开发中会特别注重这方面的积累，也非常乐意分享个人的开发经验以及使用过的效果较好的开发工具。

本书综合了笔者这些年来积累的各种 Web 前端开发经验以及各种高效的 Web 前端开发实践。如果把这些实践方法称为最佳实践，则有可能会引来争议，但有些实践方法的确是认可度很高的方法。从内容上说，本书是一本介绍如何高效地从事 Web 前端开发的书籍，而不是讲解 Web 前端技术的书籍。虽然技术更新非常快，但高效的开发实践方法则相对会更新得慢一些。开发者掌握这些高效的实践方法，会在开发过程中节省大量的成本，另外，在学习新技术的过程中，也会应对自如。

读者对象

本书主要适合以下读者：

- 对 Web 前端有浓厚兴趣的读者
- Web 前端开发工程师
- Web 项目的管理人员

如何阅读本书

本书将分为五大部分，第一部分为概述，第二、三、四部分分别介绍 HTML、CSS、JavaScript 相关的实践方法，最后一部分简单介绍移动 Web 开发相关的实践方法。每个部分相对独立，读者可以从头至尾通读，也可以选择感兴趣的部分

阅读。

第一部分（第 1、2 章）为高效 Web 前端开发综述。这部分将介绍前端开发中的基本规范、代码组织、重构及测试相关话题，同时也介绍一些好用的开发辅助工具。

第二部分（第 3 ~ 5 章）为 HTML 相关开发最佳实践。这部分贴合 W3C 标准，介绍如何编写符合标准的、语义化的、可维护性高的 HTML 代码，并简单介绍 HTML5 规范中新添加的标签和属性的使用。

第三部分（第 6 ~ 8 章）为 CSS 相关开发最佳实践。这部分介绍如何编写高维护性和高性能的 CSS 代码。并简单介绍如何高效地使用 CSS3 中新添加的特性。

第四部分（第 9 ~ 11 章）为 JavaScript 相关开发最佳实践。该部分同样包含了如何编写高维护性和高性能的 JavaScript 代码。同时，编写 JavaScript 代码也要考虑代码的安全性，因此，该部分也会介绍 Web 前端中的各种攻击及防范方法。

第五部分（第 12 章）为移动 Web 开发相关最佳实践。移动 Web 开发兴起的时间不长，很多开发实践方法也在总结和摸索阶段。前面四部分介绍的大部分实践方法同样适用于移动 Web 开发。另外，这个部分会简单介绍在移动 Web 开发时特别需要关注的实践方法。

勘误和支持

由于作者的水平有限，书中难免会出现一些错误或者不准确的地方，恳请读者批评指正。本书引用了大量的网络资源，但随着时间的推移，很难保证所有的资源都可以访问，因此，我把部分网络引用资源备份到我的个人网站：<http://www.dangjian.com>，供读者查阅。同时，在该网站上也提供了本书中的源代码的下载。另外，在该网站上开辟了一个专门的栏目，用于勘误和问题咨询，欢迎读者提交在本书中发现的错误。如果对本书中的观点有不同的见解，也欢迎来该网站与笔者交流讨论。为方便读者，也欢迎发送邮件至我的邮箱：hunterdang@gmail.com。

致谢

首先，感谢西安葡萄城信息技术有限公司，正是借助该公司在博客园上的官方技术博客，我才能够将很多的技术思想表达出来。另外，通过在这个博客上写作，促成了本书的出版。

感谢经常和我在一起讨论技术的小伙伴们：张凌、许佳瑟、黄海涛、杨波、梁真。他们在本书的写作过程中提供了很多宝贵的意见，为本书增色不少。

感谢机械工业出版社的编辑杨绣国，感谢你容忍我多次推迟交稿时间，并且非常耐心地纠正各种表达上的错误。从你身上，我感受到了一个编辑的敬业精神。

谨以此书，献给我的妻子李莹鸽。她为了让我有更多的写作时间，毫无怨言地承担起了更多的家庭责任。如果没有她的鼓励和督促，我都不知道什么时候才能完成本书的写作。我儿子的出生也给了我很大的动力。每当我苦恼于如何把一些开发理念表达清楚时，儿子调皮的笑声缓解了我不小的压力。

党 建

马来西亚吉隆坡

前言

第一部分 高效 Web 前端开发综述

第1章 Web前端开发概述	2
1.1 Web 前端开发的范畴	2
1.1.1 什么是 Web 前端开发	2
1.1.2 Web 前端开发需要具备的技能	3
1.2 Web 前端开发现状	6
1.3 规范的 Web 前端代码：更易维护、更高性能和更安全	9
第2章 高效Web前端开发	11
2.1 前端代码的结构组织和文件的命名	11
2.2 前端代码重构	13
2.3 合理使用前端框架	15
2.4 多浏览器测试：多测试，早测试	16
2.5 Web 前端代码开发和调试	19
2.5.1 Web 前端集成开发环境	19
2.5.2 Web 前端代码调试	20
2.5.3 Web 前端性能分析	21

2.5.4	代码和资源的压缩与合并	22
2.6	前端代码基本命名规范和格式规范	25
2.6.1	HTML 命名规范及格式规范	25
2.6.2	CSS 命名规范及格式规范	27
2.6.3	JavaScript 命名规范及格式规范	30

第二部分 HTML 最佳实践

第3章	标准的HTML代码	36
3.1	验证代码是否符合标准	36
3.2	标准的 HTML 页面结构	39
3.3	正确闭合 HTML 标签	40
3.4	停止使用不标准的标签和属性, 简化 HTML 代码	41
3.5	样式与结构分离	43
3.6	添加 JavaScript 禁用的提示信息	45
3.7	添加必要的 <meta> 标签	49
第4章	高可读性的HTML	53
4.1	HTML 语义化	53
4.2	如何设置网页标题层级	58
4.3	如何正确设计表单	60
4.4	精简 HTML 代码	65
4.5	过时的块状元素和行内元素	68
第5章	积极拥抱HTML5	71
5.1	HTML5 新特性的使用	71
5.1.1	使用 HTML5 中简化的定义方式	72
5.1.2	使用 HTML5 中的新标签和新属性	73
5.2	不要使用 HTML5 中已废弃的标签和属性	78
5.3	如何处理浏览器的兼容问题	79

第三部分 CSS 最佳实践

第6章 高维护性的CSS	84
6.1 如何高效地组织 CSS 代码	84
6.2 使用 CSS Reset: 统一浏览器显示效果	89
6.3 给 CSS 样式定义排序	94
6.4 合理利用 CSS 的权重: 提高代码的重用性	97
6.5 CSS 代码的兼容性: 如何兼容 IE 浏览器	101
6.6 em、px 还是 %	104
第7章 高性能的CSS	109
7.1 使用高效的 CSS 选择器	109
7.2 CSS 相关的图片处理	112
7.3 减少 CSS 的代码量	116
7.4 其他 CSS 高性能实践	120
第8章 CSS3相关最佳实践	122
8.1 查看浏览器的支持情况	122
8.2 添加必要的浏览器前缀	123
8.3 做好 CSS3 中新特性的兼容处理	127

第四部分 JavaScript 最佳实践

第9章 高维护性的JavaScript	132
9.1 养成良好的编码习惯, 提高代码的可维护性	132
9.2 使用更严格的编码格式	138
9.3 事件处理和业务逻辑分离	141
9.4 配置数据和代码逻辑分离	142
9.5 逻辑与结构样式分离	145
9.5.1 从 JavaScript 代码逻辑中分离 CSS 样式	145

9.5.2 从 JavaScript 代码逻辑中分离 HTML 结构	146
9.6 JavaScript 模板的使用	149
9.7 在 JavaScript 开发中应用 MVC 模式	152
9.8 JavaScript 模块化开发	154
9.9 合理使用 AJAX 技术	156
第10章 高性能的JavaScript	159
10.1 加快 JavaScript 文件的加载速度	159
10.2 养成良好的编码习惯，提高代码运行速度	162
10.3 使用高性能的变量或属性值读取方式	165
10.4 高效的 DOM 操作	168
10.5 使用辅助工具优化 JavaScript 代码性能	173
第11章 高安全性的JavaScript	178
11.1 常见的 Web 前端攻击方式	179
11.2 不要轻易信任任何外部传入的数据	180
11.3 其他前端安全防范实践	183

第五部分 移动 Web 前端开发

第12章 移动Web前端开发最佳实践	186
12.1 移动 Web 前端开发概述	187
12.1.1 主流的移动终端和移动浏览器	187
12.1.2 移动端和桌面端 Web 前端开发的差异	188
12.1.3 如何让桌面 Web 页面兼容移动设备	190
12.1.4 开发移动 Web 站点的准备工作	196
12.2 移动 Web 前端开发相关技术最佳实践	199
12.2.1 HTML 相关最佳实践	200
12.2.2 CSS 相关最佳实践	202
12.2.3 JavaScript 相关最佳实践	205



第一部分 *Part 1*

高效 Web 前端 开发综述

- 第1章 Web前端开发概述
- 第2章 高效Web前端开发

Web 前端开发概述

本章首先会简单介绍 Web 前端开发的历史由来，以及前端开发的概念及其范畴，让读者对前端开发有个整体的认识，同时还会对前端开发的一些错误认识进行更正。其次会介绍一下 Web 前端的现状，包括互联网整体环境、浏览器的发展、网站的设计和开发现状等，让读者认识到前端开发的现状和趋势，以便更好地提高自身的前端开发技能。

1.1 Web 前端开发的范畴

1.1.1 什么是 Web 前端开发

什么是 Web 前端开发？也许有人会说 Web 前端开发就是网页设计，就是美工，也有人会说 Web 前端的工作就是使用工具拖曳生成各种界面，然后导出为网页。其实这些都是对 Web 前端的误解，或者说还是停留在过去对前端开发的认识上。记得某个求职节目中，某互联网公司负责人对前来求职的前端工程师说他们理解的 Web 前端是需要用 Java 的，是需要用这种语言编写的。一个互联网公司的负责人竟然对自己公司使用的核心技术概念知之甚少，这实在是让人匪夷所思，更是闹了不小的笑话。此事件也激起了业内针对 Web 前端概念的大讨论，不过也从侧面说明 Web 前端这一概念还没有深入人心。的确，Web 前端兴起的时间还很短，这也就导致了很大一部分人不太了解这个新型的职业，所以我在这里再次解释一下 Web 前端这个词语的来龙去脉。

对于 Web 前端，业内公认的说法是从 2005 年开始兴起的。2005 年以前，因为 Web 网页主要以展示为主，内容基本都是静态的，所以客户端开发工作的目的就是让页面展现得更加整齐和漂亮，没有太多花哨的内容。网站的用户也只是以浏览为主，并不会会有复杂的交互。正因为如此，一般的美工仅依靠 Photoshop 和 Dreamweaver 等工具就可以制作出外观漂亮的静态网页。2005 年之后，互联网进入 Web 2.0 时代。Web 2.0 更注重用户的交互作用，用户不再只是读者，同时也是作者。用户不再是仅仅阅读静态的网页，同时也为网站贡献内容。随着这一概念的发展，人们开始重新审视网站的设计，制作的网页慢慢变得生动起来，页面也有了大量的交互，不再是简单地展示静态的文字和图片。Google Gmail 的发布，使得 AJAX 技术大红大紫，这也把对 Web 2.0 的概念的认识推上了一个新的高度。AJAX 无刷新技术极大地增强了网页的用户体验，使得用户操作页面更流畅，操作体验更接近于本地应用。此外，搜索引擎的普及使得网站的搜索引擎优化受到了重视。搜索引擎对网站的外观并不感兴趣，只识别网站的 HTML 代码，这就要求网站的设计者和开发者不仅要重视网站外在的用户体验，还要重视网站内在的代码质量。

随着网站功能的丰富、设计风格的发展以及网站代码质量的要求，网页端的开发也变得复杂起来，其代码量和逻辑复杂度都增加不少，同时还需要考虑网站的性能、浏览器兼容及网站安全性方面的问题。传统的网站开发者仅仅会使用网页制作工具已经不能够满足目前的需求了，此时的网站开发更接近于后端开发，需要有专门的软件开发工程师来做网站开发相关的工作，于是原来的网页制作这一职业就演变成了 Web 前端开发。从职责上讲，Web 前端开发要涉及网站开发的方方面面，从前端 UI 到和后端的数据交互都属于前端开发的范畴。因此，Web 前端开发是兼具艺术气息和逻辑思维的综合体，既要考虑页面的美感和操作体验，又要关注前端代码的质量。

1.1.2 Web 前端开发需要具备的技能

由于 Web 前端技术兴起的时间不长，因此它还没有明确的界限定义，不同的 Web 项目中可能要求的 Web 前端开发技术会有所不同。例如，某些项目可能需要前端开发人员了解一些后端技术，这样前端开发人员才可以更好地与后端开发人员配合，如在页面上留下一些后端需要调用的“钩子”等，而某些项目可能需要前端开发人员懂一些 UI 设计、Photoshop 工具的使用方法等，以便于和 UI 设计师沟通和配合。虽然 Web 前端开发的范畴广泛，并且界限模糊，但是以下 7 点是 Web 前端开发必备的技能。

1. 页面标记 (HTML)

由于页面 HTML 代码结构基本固定，HTML 的标签数量也不多，因此，从学习的难易程度来说，HTML 应该是前端技术中非常容易学习的技术。即使是一个新手，也能在较短的时间里学会编写一个结构良好的页面。虽然说入门容易，但是要编写语义良好、简洁整齐的 HTML 代码则需要大量的实践才能掌握。HTML 是页面的基本结构组成部分，是网站的基础，臃肿混乱的 HTML 代码不但会影响其本身的展现，而且与其对应的 CSS 与 JavaScript 代码也会变得难以编写和维护。

2. 页面样式

CSS 是 Cascading Style Sheet (层叠样式表) 的简称。在标准页面设计中，因为 CSS 负责网页内容的表现，所以 CSS 也是前端开发需要掌握的核心内容之一。丰富的 CSS 样式能让平淡的 HTML 展现出绚丽的效果，使得页面更为友好。好的样式可以让用户在页面上停留的时间更久一些，也可以帮助用户更好地阅读网站内容，同时，还可以让用户在不同浏览器上有着相同的体验。CSS 和 HTML 代码一样，没有复杂的逻辑，上手也比较容易，其主要的难点在于如何合理地利用 CSS 的组合和继承特性来编写简洁、可维护性好的 CSS 代码。以上这两项基本技能是前端 UI 开发必备的技能。

3. 前端编程

前端编程技能主要是指 JavaScript 编程。JavaScript 是一种基于对象和事件驱动的客户脚本语言，是页面实时动态交互的技术基础。相较于 HTML 和 CSS，编写 JavaScript 代码更能让前端开发人员找到后端程序员的感觉。JavaScript 是非常灵活的脚本语言，包含了高阶函数、动态类型以及灵活的对象模型等强大的语言特性，当然，JavaScript 的灵活性也可能导致代码不易维护。此外，浏览器的兼容性也增加了 JavaScript 编码的难度。同一个功能，可能在不同的浏览器中有不同的实现。例如，在 IE 浏览器中，事件绑定使用的是 `attachEvent()` 方法，但其他浏览器则使用的是 `addEventListener()` 方法。开发人员在熟悉 JavaScript 基本语法和基本的编码规范之外，还应该了解并解决在不同浏览器中的 JavaScript 的兼容性问题。另外，作为前端开发工程师，必定会涉及后端的编程，一些原因是目前流行的 Web 编程方式会有部分后端代码存在于前端页面中，和前端的 HTML、JavaScript 等混合在一起，如 PHP、JSP、ASP.NET 等，因此，前端开发工程师也有必要了解一些后端编程技术。

4. 跨平台，跨浏览器

前端代码本来不存在跨平台方面的问题，但是随着移动 Web 平台的兴起，跨

平台的问题就逐渐显现出来了。移动设备（如智能手机和平板电脑）在近几年发展迅猛，用户通过移动设备访问 Web 站点的比率也是逐年增高。如何在众多移动平台、众多屏幕尺寸上展现友好的 Web 站点成为一项前端技能。不过，目前跨浏览器没有像几年前表现得那么突出了，这主要是因为 IE 6、IE 7 浏览器的占有率下降和众多浏览器对标准的重视，另外，目前流行的前端框架已经很好地解决了浏览器的兼容问题。尽管如此，但是还需要熟悉常见的浏览器兼容方法，主要包括：IE 7、IE 8 的兼容，HTML5 中新特性的兼容等。

5. 前端框架

各种前端框架的出现，在很大程度上降低了前端开发的难度。框架统一了编码的方式，封装了浏览器兼容问题并添加大量的扩展功能。如今的 Web 项目中前端框架应用非常广泛，在开源社区 GitHub 上排名靠前的开源框架也是以前端框架居多。优秀的前端框架可以在很大程度上缩短项目开发的周期，尤其是 jQuery，几乎成为 Web 项目默认的前端框架。但是，前端框架的接口众多，各种框架的使用方式和编码方式也不尽相同，作为前端开发工程师，需要熟悉一些常用框架的使用方法，并且要了解如何编写常用框架的扩展插件，如 jQuery、YUI、Ext JS 等。

6. 调试工具

对于前端代码，在调试过程中需要查看页面的 HTML 结构变化、CSS 渲染效果、JavaScript 代码的执行情况以及 HTTP 请求和返回的数据，并且要了解网站各个部分的性能等，甚至需要动态更改 HTML、CSS 代码来查看预期的效果，模拟发起 HTTP 请求来查看后端返回的数据。主流浏览器都会有对应的浏览器插件来辅助完成这些工作，如 IE 中的 IE Dev Toolbar、Chrome 中的 Developer Tools、Firefox 中的 Firebug 等，此外还有 HTTP 请求监控和模拟工具，如 Fiddler 等。开发工程师需要熟练使用这些工具来辅助完成前端代码的调试。

7. 沟通能力

沟通是开发人员必备的一项基本技能，尤其是对于前端开发工程师来说。Web 前端开发介于 UI 和后端逻辑开发之间，因此，Web 前端工程师在开发过程中必定会和 UI 设计师及后端工程师合作：前端工程师需要和 UI 设计师沟通，确定效果是否可以实现以及实现的代价，并对 UI 设计提出建议；还需要和后端工程师沟通，确定前后端交互的接口以及传输的数据实体的结构等，良好的沟通会让这些过程变得轻松许多。

1.2 Web 前端开发现状

前端开发虽然起步时间晚，但是发展势头迅猛，在各种新技术、新标准的推动下，各大互联网公司也开始重视 Web 产品的前端重构与开发，如淘宝、腾讯、新浪、百度、搜狐等都对自己的网站进行了重构并同时使用了 HTML5 中的新特性。现在开发的 Web 新产品的页面交互也越来越丰富，视觉效果也越来越绚丽。互联网公司除了重视前端开发之外，还在积极推动互联网技术的发展，2001 ~ 2012 年，腾讯、百度及奇虎 360 公司先后加入了万维网联盟（W3C），共同参与互联网技术标准的研究和制定，促进了国内互联网产业的发展。

在 Web 前端发展的过程中，浏览器的发展起着至关重要的作用。浏览器厂商在这场 Web 浪潮中表现突出，具体表现在浏览器的版本升级周期短、对前端标准的支持也越来越好、浏览器的性能也有大幅的提升等。IE 浏览器也开始发力，最新发布的 IE 10 浏览器无论是标准的支持还是性能都表现突出。浏览器的快速发展使得低端的浏览器如 IE 6、IE 7 等快速退出市场。

下面看一下全球范围内浏览器的占有率情况。根据 StatCounter 的统计，2012 年 10 月份，全球浏览器占有率前 5 位的分别是 Chrome、IE、Firefox、Safari 及 Opera，占有率分别为 34.77%、32.08%、22.32%、7.81% 和 1.63%。按照浏览器的版本统计，在低端浏览器中，IE 8 占有率为 12%，IE 7 的占有率仅为 0.88%，IE 6 的占有率则更低。

此外，移动设备上的浏览器在整个浏览器占有率中上升速度很快。根据 StatCounter 的统计，Safari iPad 在浏览器的占有率已经达到了 2.76%。目前移动设备的普及率极高，用户已经逐渐习惯于通过移动设备来浏览网页。主流的移动平台主要为 Android 和 iOS 系统，对应的浏览器都是以 WebKit 为核心的，性能和对标准的支持都不错。

以上介绍的是浏览器在全球范围内的占有率情况。从上面的统计可以看出，在全球范围内，低端浏览器的占有率已经很低了，但国内浏览器占有率的状况则比较复杂，低端浏览器如 IE 8、IE 7 和 IE 6 等还占有不小的份额。根据 StatCounter 的统计，2012 年 10 月份，国内浏览器占有率前 5 位分别为 IE 8、IE 9、IE 6、搜狗浏览器和 360 安全浏览器，占有率分别为 48.7%、14.09%、8.29%、5.74% 和 2.72%。其中低端浏览器 IE 8、IE 7 和 IE 6 的总占有率大约为 58%，低端浏览器依然占有“半壁江山”，而高级浏览器如 Chrome、Safari、Firefox 等占有率则很低。值得称道的是，360 安全浏览器和搜狗浏览器这两款浏览器都是基于开源的浏览器内核，对标准支持较好，性能表现突出。这两款浏览器的占有率目前虽然不高，但发展势头迅猛，是推动老旧浏览器退出市场的中坚力量。

除各大互联网公司和浏览器厂商的积极推动外，Web 前端开源社区的参与热