

图片整合（精灵图、css sprites、css精灵、雪碧图）

1. 把网站里面的小图标有规则的整合在一起，利用 background-position 改变背景图的位置，每个图标应用。
1. 优点：
 2. (1) CSS Sprites能很好地减少网页的http请求，从而大大的提高页面的性能，这是CSS Sprites最大的优点，也是其被广泛传播和应用的主要原因；
 3. (2) CSS Sprites能减少图片的字节；
 4. (3) CSS Sprites解决了网页设计师在图片命名上的困扰，只需对一张集合的图片命名，不需要对每一个小图片进行命名，从而提高了网页制作效率。
 5. (4) CSS Sprites只需要修改一张或少张图片的颜色或样式来改变整个网页的风格。
- 6.
7. 缺点：
 8. (1) 图片合并麻烦：图片合并时，需要把多张图片有序的合理的合并成一张图片，并留好足够的空间防止版块出现不必要的背景。
 9. (2) 图片适应性差：在高分辨的屏幕下自适应页面，若图片不够宽会出现背景断裂。
 10. (3) 图片定位繁琐：开发时需要通过工具测量计算每个背景单元的精确位置。
 11. (4) 可维护性差：页面背景需要少许改动，可能要修改部分或整张已合并的图片，进而要改动css。在避免改动图片的前提下，又只能（最好）往下追加图片，但这样增加了图片字节。

兼容

浏览器



百度



傲游



UC



谷歌



火狐



搜狗



QQ



360



欧朋



海豚

□

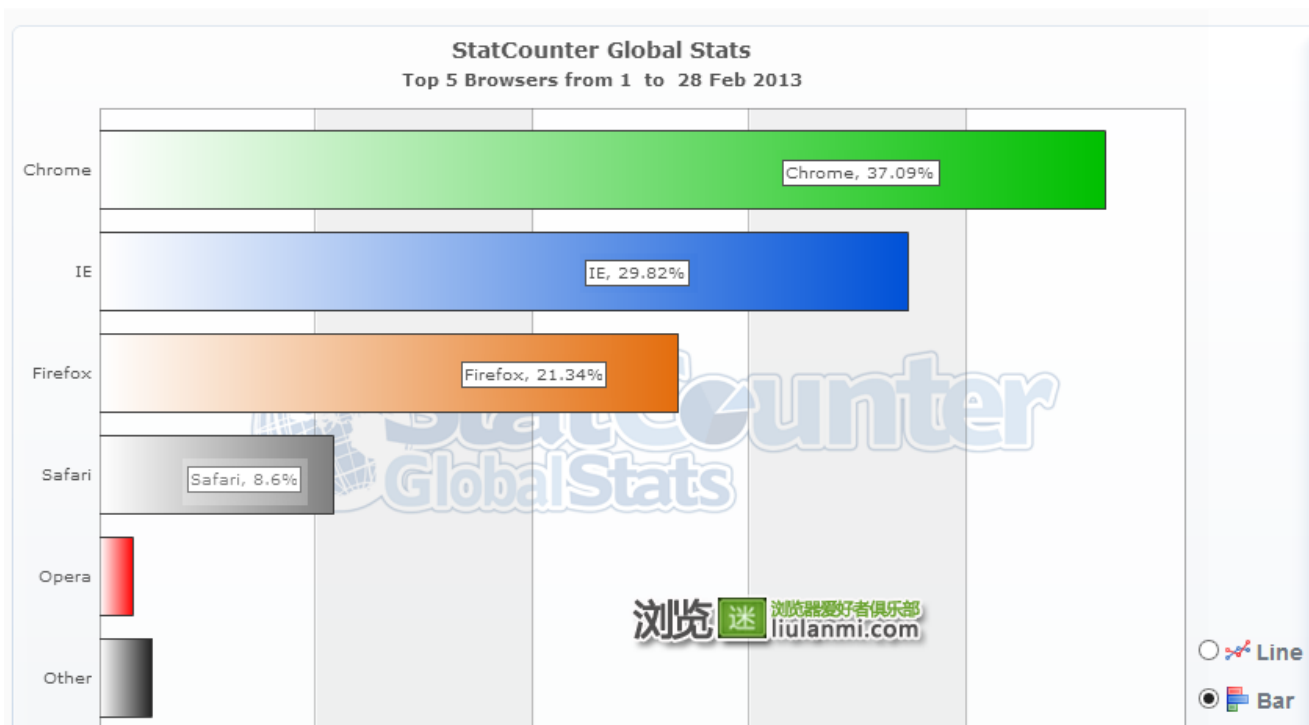
浏览器图标

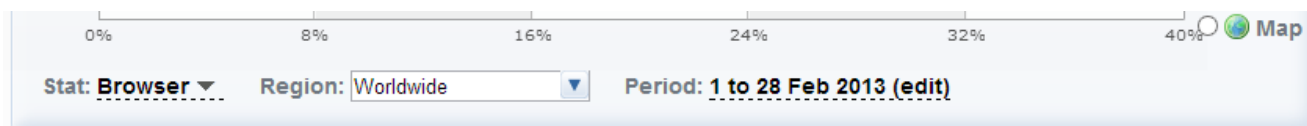
浏览器大战



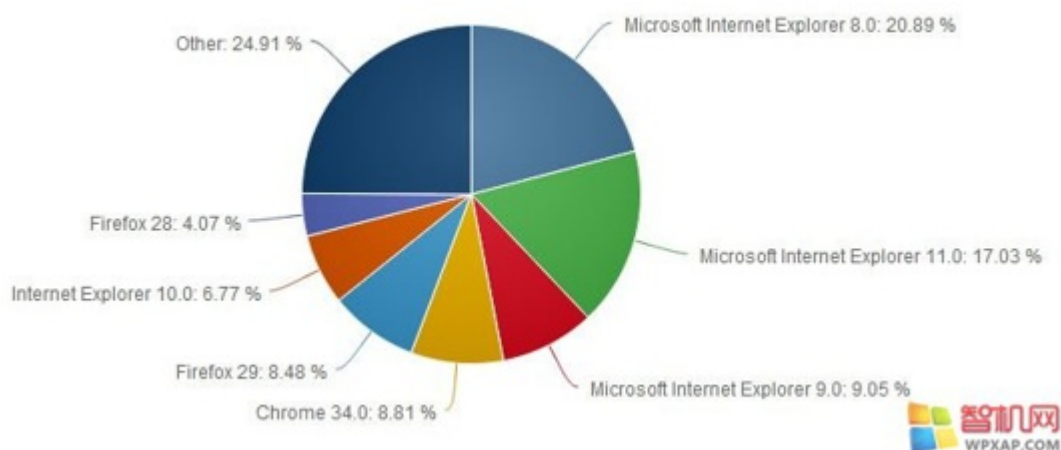
1. 第一次浏览器大战发生在上个世纪90年代，微软发布了它的IE浏览器，和网景公司的Netscape Navigator浏览器大打出手。
- 2.
3. 第二次浏览器大战发生在20世纪。
- 4.
5. 战争产物：Internet Explorer 9

13年市场比重

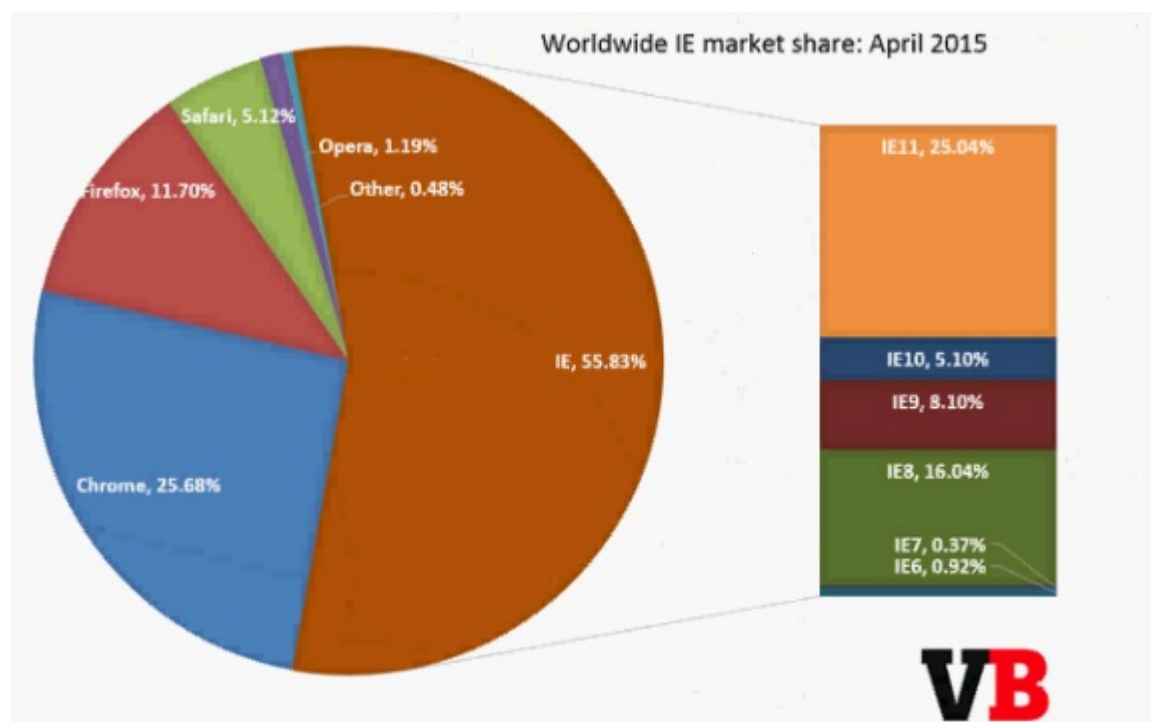




14年市场比重



15年市场比重



17年市场比重

2017年4月，全球PC浏览器市场份额排行榜▼

第1名：Google的Chrome浏览器，其全球市场份额为**63.44%**；

第2名：Mozilla Firefox，其市场份额为**14.54%**；

第3名：微软的IE浏览器，其市场份额为8.98%；

第4名：苹果的Safari浏览器，其市场份额为5.20%；

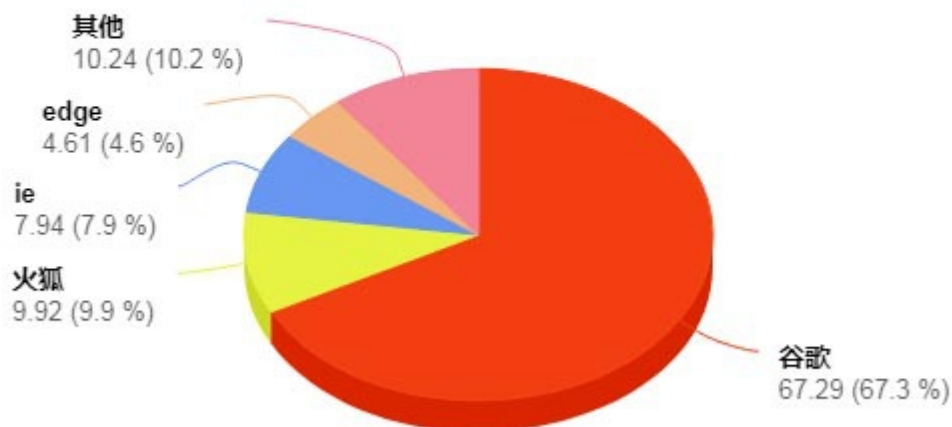
第5名：微软的Edge浏览器，其市场份额为3.72%；

第6名：Opera浏览器，其市场份额为2.09%；

其他浏览器的市场份额合计为2.04%。

19年市场比重

2019 浏览器市场份额



浏览器内核（现代4大内核）

Trident 代表作：IE

1. 元老级内核之一，由微软开发，并于1997年10月首次在ie 4.0中使用，凭借其windows垄断优势，Trident市场占有率一直很高。然而垄断并非，没有竞争就没有进步，长期以往，Trident内核一度停滞不前，更新缓慢，甚至一度与W3C标准脱节。2011年，从ie 9开始，Trident开始支持HTML5和CSS3。因此我们也经常看到有些网站在浏览器会报三思户（左图），右图。

符HTML5和CSS 3，因此我们也经常看到有些网站在浏览时会提示用户（在Internet Explorer 9.0+以上浏览效果最佳）。前端程序员做浏览器兼容一般也不再会考虑ie 8之前的浏览器了。

Gecko 代表作：Mozilla

1. 元老级内核之一，由Netscape公司Mozilla组织开发。1998年，Netscape在于IE浏览器竞争失利之后，成立了非正式组织Mozilla，由其开发新一代内核，后命名为“Gecko”。FireFox也是这班人开发出来了，因此这也就是Mozilla一直使用的内核。
2. Gecko的特点是代码完全公开，因此其开发程度很高，全世界的程序员都可以为其编写代码，增加功能。

WebKit：苹果 & 谷歌旧版本

1. 这是苹果公司开发的内核，也是其旗下产品Safari浏览器使用的内核。WebKit引擎包含了WebCode排版引擎和JavaScriptCode解析引擎，分别是KDE的KHTML和KJS衍生而来，它们都是自由软件，在GPL条约下授权，同时支持BSD系统开发。
2. Chrome、360极速浏览器以及搜狗高速浏览器也使用WebKit作为内核（在脚本理解方面，Chrome使用自己研发的V8引擎）。

Blink：代表作：谷歌 & 欧鹏

1. 这是由Google和Opera Software开发的浏览器排版引擎，Google计划将这个渲染引擎作为Chromium计划的一部分，并且在2013年4月公布了这一消息。这一渲染引擎是开源引擎WebKit中WebCore组件的一个分支，并且在Chrome（28及往后版本）、Opera（15及往后版本）和Yandex浏览器中使用

Presto（Opera前内核 已经废弃）

为什么会出现浏览器兼容问题？

1. 由于各大主流浏览器由不同的厂家开发，所用的核心架构和代码也很难重合，这就为各种莫名其妙的Bug（代码错误）提供了温床。再加上各大厂商出于自身利益考虑而设置的种种技术壁垒，都让CSS应用起来比想象得要麻烦。浏览器的兼容问题是我们必须去克服的。

CSS Bug、CSS Hack和Filter

- CSS Bug: CSS样式在各浏览器中解析不一致的情况，或者说CSS样式在浏览器中不能正确显示的问题称为CSS bug。
- CSS Hack: CSS中，Hack是指一种兼容CSS在不同浏览器中正确显示的技巧方法，因为它们都属于个人对CSS代码的非官方的修改，或非官方的补丁。有些人更喜欢使用patch（补丁）来描述这种行为。
- Filter: 表示过滤器的意思，它是一种对特定的浏览器或浏览器组显示或隐藏规则或声明的方法。本质上讲，Filter是一种用来过滤不同浏览器的Hack类型。

常见的BUG

IE低版本常见CSS解析Bug及hack

1. 1) 图片有边框BUG
2. 当图片加[“#”](#)在IE上会出现边框
3. Hack: 给图片加border:0;或者border:0 none;
- 4.
- 5.
6. 2) 图片间隙
7. div中的图片间隙BUG
8. 描述: 在div中插入图片时, 图片会将div下方撑大大约三像素。
9. hack1: 将</div>与写在一行上;
10. hack2: 将转为块状元素, 给添加声明: display:block;
- 11.
- 12.
13. 3) 双倍浮向 (双倍边距) (只有IE6出现)
14. 描述: 当IE6及更低版本浏览器在解析浮动元素时, 会错误地把浮向边边界 (margin) 加倍显示。
15. hack: 给浮动元素添加声明: display:inline;
- 16.
- 17.
18. 4) 默认高度 (IE6、IE7)
19. 描述: 在IE6及以下版本中, 部分块元素拥有默认高度 (在16px左右;)
20. hack1: 给元素添加声明: font-size:0;
21. hack2: 给元素添加声明: overflow:hidden;

非IE BUG

1. 5) 表单元素对齐不一致
2. 描述: 表单元素行高对齐方式不一致
3. hack: 给表单元素添加声明: float:left;
- 4.
- 5.
6. 6) 按钮元素默认大小不一
- 7.
8. 描述: 各浏览器中按钮元素大小不一致
9. hack1: 统一大小/ (用a标记模拟)
10. hack2: input外边套一个标签, 在这个标签里写按钮的样式, 把input的边框去掉。
11. hack3: 如果这个按钮是一个图片, 直接把图片作为按钮的背景图即可。
- 12.
- 13.

- 13.
14. 8) 鼠标指针bug
15. 描述: cursor属性的hand属性值只有IE9以下浏览器识别, 其它浏览器不识别该声明, cursor属性的pointer属性值IE6.0以上版本及其它内核浏览器都识别该声明。
16. hack: 如统一某元素鼠标指针形状为手型,
17. 应添加声明: cursor:pointer
- 18.
- 19.
20. cursor: ;
21. auto默认
22. crosshair加号
23. text文本
24. wait等待
25. help帮助
26. progress过程
27. inherit继承
28. move移动
29. ne-resize向上或向右移动
30. pointer手形
- 31.
- 32.
- 33.
34. 9) 透明属性
35. 兼容其他浏览器写法: opacity:value; (value的取值范围0-1;
36. 例: opacity:0.5;)
37. IE浏览器写法: filter:alpha(opacity=value); 取值范围 1-100(整数)

过滤器

1. 1. 下划线属性过滤器
2. 当在一个属性前面增加了一个下划线后, 由于符合标准的浏览器不能识别带有下划线的属性而忽略了
了这个声明, 但是在IE6及更低版本浏览器中会继续解析这个规则。
- 3.
4. 语法: 选择符 {_属性: 属性值; }
- 5.
- 6.
7. 2. !important关键字过滤器
- 8.
9. 它表示所附加的声明具有最高优先级的意思。但由于IE6及更低版本不能识别它,
10. 我们可以利用IE6的这个Bug作为过滤器来兼容 I E 6 和其它标准浏览器。
- 11.

12. 语法：选择符{属性：属性值!important;}

13.

14.

15. 3. *属性过滤器

16.

17. 当在一个属性前面增加了*后，该属性只能被IE7浏览器识别，其它浏览器混略该属
18. 性的作用。

19.

20. 语法：选择符{*属性：属性值；}

21.

22.

23. 4. \9 ： IE版本识别；其它浏览器都不识别

24. 语法：选择符{属性：属性值\9;}

25.

26. 5. \0 ： IE8 及以上版本识别；其它浏览器都不识别