用 visibilitychange 事件判断页面可见性 – 使用 PageVisibility API

发表于 [2016年05月24日](http://www.css88.com/archives/6103) by [愚人码头](http://www.css88.com/archives/author/admin) 被浏览 31,059 次 分享到：

**小编推荐：**[掘金是一个面向程序员的高质量技术社区，从 一线大厂经验分享到前端开发最佳实践，无论是入门还是进阶，来掘金你不会错过前端开发的任何一个技术干货。](https://juejin.im/welcome/frontend?utm_source=css88&utm_medium=word&utm_content=word&utm_campaign=q2_website)

这里了将介绍一下 页面可见性（PageVisibility）API的简单应用。

visibilitychange事件介绍

简单的说，浏览器标签页被隐藏或显示的时候会触发visibilitychange事件。

这个事件可以满足一些用户需求，比如标签页隐藏的时候停止播放音乐视频、停止一些不必要的轮询，还有停止一些诸如轮播等循环动画效果等等。这些可以节省服务器和本地的开销。

这个事件已经得到现代浏览器广泛的支持，不过一些老版本的浏览器需要加相应的前缀。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Chrome (Webkit)** | **Firefox (Gecko)** | **Internet Explorer** | **Opera** | **Safari (WebKit)** |
| 13前缀webkit 33无需前缀 | 10前缀moz 18无需前缀 | 10前缀ms | 12.10 | 最新的测试已经支持 |

注：Opera 12.10浏览器在最小化时不触发visibilitychange事件,也不将hidden属性设置为true.

document的可见性属性

[Page Visibility (Second Edition)](https://www.w3.org/TR/page-visibility/)中定义了2个只读的document属性：hidden和visibilityState。

其中document.hidden是一个布尔值，简单的表示标签页显示或者隐藏。而document.visibilityState属性更为详细，目前有四个可能的值：

* visible : 页面内容至少部分可见。这意味着在实际情况中，该网页是一个非最小化窗口的可见标签页。
* hidden : 页面内容是对用户不可见。实际上，这意味着该文档是后台标签页或最小化窗口的一部分，或者系统锁屏是时的状态。
* prerender : 网页内容被预渲染并且用户不可见。
* unloaded : 如果文档被卸载，那么这个值将被返回。

一般情况下我们使用document.hidden就能满足通常的需求。

为了支持老版本的浏览器，我们需要对document.hidden在做一些前缀处理：

js 代码:

1. function getHiddenProp(){
2. var prefixes = ['webkit','moz','ms','o'];
4. // if 'hidden' is natively supported just return it
5. if ('hidden' in document) return 'hidden';
7. // otherwise loop over all the known prefixes until we find one
8. for (var i = 0; i < prefixes.length; i++){
9. if ((prefixes[i] + 'Hidden') in document)
10. return prefixes[i] + 'Hidden';
11. }
13. // otherwise it's not supported
14. return null;
15. }

同样的，我们可以获取document.visibilityState属性：

js 代码:

1. function getVisibilityState() {
2. var prefixes = ['webkit', 'moz', 'ms', 'o'];
3. if ('visibilityState' in document) return 'visibilityState';
4. for (var i = 0; i < prefixes.length; i++) {
5. if ((prefixes[i] + 'VisibilityState') in document)
6. return prefixes[i] + 'VisibilityState';
7. }
8. // otherwise it's not supported
9. return null;
10. }

这样我们可以写一个跨浏览器的函数，检查文档是否可见。

js 代码:

1. function isHidden() {
2. var prop = getHiddenProp();
3. if (!prop) return false;
5. return document[prop];
6. }

visibilitychange监听事件

你可以在document对象上注册一个监听visibilitychange事件，根据document.hidden或者document.visibilityState属性做一些业务逻辑：

js 代码:

1. // use the property name to generate the prefixed event name
2. var visProp = getHiddenProp();
3. if (visProp) {
4. var evtname = visProp.replace(/[H|h]idden/, '') + 'visibilitychange';
5. document.addEventListener(evtname, function () {
6. document.title = document[getVisibilityState()]+"状态";
7. },false);
8. }

上面的代码会在页面可见性发生变化时修改document.title的值。