

A标签常见兼容性问题

标签：a,link,visited,hover,active,href,font,specificity,特殊性,L-V-H-A

操作系统版本：

Windows 7 Ultimate build 7600

浏览器版本：

IE6

IE7

IE8

Firefox 3.6.2

Safari 4.0.4

Chrome 5.0.356.2 dev

受影响的浏览器：

所有浏览器

本文目录

本文目录	1
一、关于A标签的说明	1
二、差异及可能产生的问题	2
1、A标签伪类书写顺序	2
2、A标签伪类选择器特殊性计算	2
3、A标签缺失href属性后，伪类是否依然有效	4
4、A标签嵌套Font标签后字体和下划线的颜色	5
三、如何避免受此问题影响	5

一、关于A标签的说明

1、A标签是web应用中重要的核心标签之一，它有两个常用的用途：

- 作为锚点：定义页面上的一个锚点，可通过其它A标签的href属性访问。
- 作为链接：导航到A标签href属性所指定的目标地址（URI地址、或同一文档的锚点）

2、A标签的渲染，有四个常用的伪类，如下：

- 'link'：适用于未被访问的链接。
- 'visited'：适用于已经访问过的链接。
- 'hover'：在可视化客服端上，适用于光标（鼠标指针）指向一个元素，但还未激活它时。
- 'active'：适用于一个元素被用户激活时。

3、A标签最常用的属性，莫过于href。绝大多数情况下，A标签都是作为链接来使用的，此时必须定义href属性，如果缺乏，将被视为普通标签。

关于A元素的更多信息，请参考 w3c html401 文档第 [12.2](#) 节

关于链接伪类的更详细信息，请参考 w3c css21 文档第 [5.11.2](#) 节

关于动态伪类的更详细信息，请参考 w3c css21 文档第 [5.11.3](#) 节

二、差异及可能产生的问题

在实际开发中，由于疏忽、使用不当、或其它原因，导致A标签的应用在不同浏览器下的表现存在差异。下面枚举一些常见的现象加以分析说明：

1、A标签伪类书写顺序





w3c css21规范中指出：A: hover必须放置在A: link和A: visited之后，否则将隐藏A: hover内定义的相同规则。同理，A: active应在A: hover之后，否则A: active中的相同规则将被隐藏。

按照规范，正确的顺序应为：L-V-H-A。下面举一个反例，例子中A标签伪类的顺序为 H-L-V-A，观察其在各浏览器下的表现，代码如下：

```
<style type="text/css">
a: hover{background-color:red;}
a: link{background-color:yellow;}
a: visited{background-color:blue;}
a: active{background-color:green;}
</style>

<a href="#">put your mouse on</a>
```

从代码中看，在鼠标未置上A链接时，A链接应用“a: link”中的规则（背景为黄色），当鼠标置上后，“a: hover”中的规则被应用。然而由于“a: hover”在“a: link”前出现，按照标准，“a: hover”中的背景规则被隐藏。下面我们来观察以上代码在各浏览器下的表现，是否遵循w3c css21规范中的L-V-H-A顺序规则：

	IE6(S)(Q)/IE7(Q)/IE8(Q)	IE7(S)/IE8(S)/Firefox(S)(Q)/Chrome(S)(Q)/Safari(S)(Q)
鼠标未置上		
鼠标置上后		

从上表，我们可以看出在IE7(S)/IE8(S)/Firefox(S)(Q)/Chrome(S)(Q)/Safari(S)(Q)下，A标签的伪类遵循w3c css21规范中的L-V-H-A顺序标准。而在IE6(S)(Q)/IE7(Q)/IE8(Q)下则不遵循，关于此问题，实为IE6(S)(Q)/IE7(Q)/IE8(Q)下的“A: hover”的bug，在下一小节中将详细讨论此bug。

关于链接伪类的更详细信息，请参考 w3c css21 文档第 5.11.2 节
关于动态伪类的更详细信息，请参考 w3c css21 文档第 5.11.3 节

2、A标签伪类选择器特殊性计算

为说明方便，排除其它因素对选择器优先级的影响，我们仅关注选择器的特殊性和出现的顺序。在此前提下，根据 w3c css21 规范，选择器的优先级由它的特殊性和出现的顺序决定，特殊性越多优先级越大，相同特殊性下，后者优先。




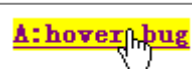
这里，我们将讨论IE6(S)(Q)/IE7(Q)/IE8(Q)下“A: hover”的bug，在上一小节中已经有所提及。

问题来了！在IE6(S)(Q)/IE7(Q)/IE8(Q)下，“A: hover”的优先级是否真的无法超越？请看如下代码：

```
<style type="text/css">
a{background-color: #d4d4d4;font-size:14px;font-weight:bold;}
a: hover{background-color: red;} /* [0,0,1,1] */
div.div1 a{background-color: yellow;} /* [0,0,1,2] */
</style>

<div class="div1">
  <a class="a1" href="#">A: hover bug</a>
</div>
```

各浏览器结果如下：



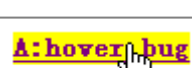
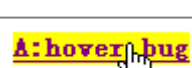
	IE6(S)(Q)/IE7(Q)/IE8(Q)	IE7(S)/IE8(S)/Firefox(S)(Q)/Chrome(S)(Q)/Safari(S)(Q)
鼠标未置上		
鼠标置上后		

观察代码，选择器“a: hover”的特殊性为“[0,0,1,1]”，选择器“div.div1 a”的特殊性为“[0,0,1,2]”，很明显选择器“div.div1 a”中的背景色（yellow）应该覆盖选择器“a: hover”中的背景色（red），然而从上表得到的测试结果来看，在IE6(S)(Q)/IE7(Q)/IE8(Q)下却无法覆盖。难道真如传说中所言IE6(S)(Q)/IE7(Q)/IE8(Q)下“A: hover”无法超越！

修改选择器“div.div1 a”为“.div1 a.a1”，如下：

```
.div1 a.a1{background-color: yellow;} /* [0,0,2,1] */
```

此时，选择器“.div1 a.a1”的特殊性为“[0,0,2,1]”。各浏览器结果如下：

	IE6(S)(Q)/IE7(Q)/IE8(Q)	IE7(S)/IE8(S)/Firefox(S)(Q)/Chrome(S)(Q)/Safari(S)(Q)
鼠标未置上		
鼠标置上后		

神奇，竟然超越了！传说被打破！经过反复测试我们得出结论：当选择器明确指向A标签，且特殊性c的值大于选择器“*: hover”特殊性c的值时（当然a，b值要相等），IE6(S)(Q)/IE7(Q)/IE8(Q)下“A: hover”优先级被超越！

那么，如果选择器没有明确的指向A标签时，IE6(S)(Q)/IE7(Q)/IE8(Q)下“A: hover”是否可超越？修改选择器“.div1 a.a1”为“div.div1 .a1”，如下：

```
div.div1 .a1{background-color: yellow;} /* [0,0,2,1] */
```

此时，选择器“div.div1 .a1”的特殊性也为“[0,0,2,1]”。观察各浏览器下的结果：

	IE6(S)(Q)/IE7(Q)/IE8(Q)	IE7(S)/IE8(S)/Firefox(S)(Q)/Chrome(S)(Q)/Safari(S)(Q)
鼠标未置上	A: hover bug	A: hover bug
鼠标置上后	A: hover bug	A: hover bug

竟然无法超越，没天理了！再次调整代码，给div设置一个ID，代码调整后如下：

```
<style type="text/css">
a{background-color: #d4d4d4;font-size:14px;font-weight:bold;}
a: hover{background-color: red;} /* [0,0,1,1] */
#divID .a1{background-color: yellow;} /* [0,1,1,0] */
</style>

<div id="divID">
  <a class="a1" href="#">A: hover bug</a>
</div>
```

此时，选择器“#divID .a1”的特殊性为“[0,1,1,0]”。观察各浏览器下的结果：

	IE6(S)(Q)/IE7(Q)/IE8(Q)	IE7(S)/IE8(S)/Firefox(S)(Q)/Chrome(S)(Q)/Safari(S)(Q)
鼠标未置上	A: hover bug	A: hover bug
鼠标置上后	A: hover bug	A: hover bug

小样，顶不住了吧！经反复测试，我们得出结论：当选择器没有明确指向A标签，且特殊性b的值大于选择器“*:hover”特殊性b的值时（当然a值要相等），IE6(S)(Q)/IE7(Q)/IE8(Q)下“A: hover”优先级被超越！

关于选择器特殊性计算的更多信息，请参考 w3c css21 文档第 [6.3.4](#) 节
关于层叠顺序的更多信息，请参考 w3c css21 文档第 [6.4.1](#) 节

3、A标签缺失href属性后，伪类是否依然有效

当A标签缺失href属性后，它相当于普通的标签，在IE6下，普通标签是不支持“:hover”伪类的。因此，对缺失href属性的A标签设置“:hover”伪类是无效的。

观察如下代码：

```
<style>a: hover,span: hover,div: hover,li: hover,p: hover,...{color: red}</style>
<a>aaaa</a><br />
<span>aaaa</span><br />
<div>aaaa</div>
<li>aaaa</li>
<p>aaaa</p>
... /<!--其他普通标记-->
```

我们将代码实际运行结果汇总如下：

:hover	IE6(QS)	IE7(Q)	IE8(Q)	IE7(S)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
A	无变化			无变化	红色			
其他普通标记	无变化			红色	红色			

":hover"属性作用在普通标记上，在IE6下是不被支持的，只有IE7之后版本与其他浏览器支持普通元素的伪类样式。因此以上写法在IE6下，鼠标置于文字上后，字体颜色无变化。

注意：IE7(S)下鼠标置上A标签后颜色无变化，由IE7(S)下的一个bug导致，本文不做说明。

4、A标签嵌套Font标签后字体和下划线的颜色

请参考分类文档 [字体样式元素对CSS的影响 第 3.3 节](#)。

三、如何避免受此问题影响

- 1、A标记的伪类书写遵循 w3c css2.1 规范中的 L-V-H-A 顺序。
- 2、注意选择器特殊性值，和IE6(S)(Q)/IE7(Q)/IE8(Q)下"A: hover"的问题，尽量避免。
- 3、尽可能不要将A标签的href属性省略掉，除非你有特殊需求，但要注意IE7(S)中存在的bug。
- 4、废弃过时的Font标记，不要使用它。