

IE对一些CSS2.1标准的支持缺陷

标签：!important, display:inline-block, display:inline-table, position:fixed, min-width/min-height, max-width/max-height, height/width:inherit, font-weight: inherit, 伪类, 伪元素

操作系统版本：

Windows 7 Ultimate build 7600

浏览器版本：

IE6

IE7

IE8

Firefox 3.6

Safari 4.0.4

Chrome 5.0.356.2 dev

受影响的浏览器：

IE6

IE7

IE8

IE对一些CSS2.1标准的支持缺陷	1
一、概述	1
IE浏览器对一些CSS标准支持的不够完善，因此造成了很多兼容性问题。	1
二、IE浏览器对CSS标准的支持缺陷	2
1. 层叠(Cascade)	2
!important规则	2
2. 继承(Inherit)	2
特性值"inherit"	2
3. 选择器	3
子选择器	3
普通兄弟选择器	4
相邻兄弟选择器	4
属性选择器	5
4. 伪类和伪元素	5
:first-child伪类	5
非锚点元素的:hover伪类	6
非锚点元素的:active伪类	6
:focus伪类	7
:before伪元素和:after伪元素	7
5. 属性支持	8
min-height 和 min-width	8
max-height 和 max-width	9
position: fixed固定定位元素	9
display特性的替代值	10
outline特性	10
三、如何避免受此问题影响	11

一、概述

IE浏览器对一些CSS标准支持的不够完善，因此造成了很多兼容性问题。

本文针对IE对某些特性的支持缺陷进行了一个汇总。因此，不对文中出现的CSS标准做过多的介绍和说明。

其中包括：

- IE6/IE7/IE8都不支持的情况
- IE6/IE7/IE8中有一个或两个版本支持的其他版本不支持的情况

二、IE浏览器对CSS标准的支持缺陷

1. 层叠(Cascade)

!important规则

简介：

!important可以提高样式的优先级。

更多说明见W3C CSS2.1规范[6.4.2](#)

实例代码：

```
<style type="text/css">
div{
width:100px;
height:30px;
background-color:gold !important;
background-color:silver;
}
</style>
<div></div>
```

以上代码中，金色的背景后使用了!important，因此支持!important的浏览器中最终DIV的背景应该是金色。

各浏览器截图：



IE6(QS)/IE7(Q)/IE8(Q)



其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	Y	N	Y	Y	Y	Y

2. 继承(Inherit)

特性值"inherit"

简介：

每一个特性(property)可以有一个指定值为'inherit'，即，对于给定的元素，该属性和它父元素相对属性的计算值取一样的值。继承值通常只用作后备值，它可以通过显式地设置'inherit'而得到加强。

更多说明见W3C CSS2.1规范[6.2.1](#)

实例代码：

```
<div>
  <span style="display:inherit;">Aloha!</span>
  Hello!
</div>
```

以上代码中，SPAN元素使用了display:inherit，而它的包含元素的display属性值为"block"，所以最终该SPAN元素会是一个块级元素，因此，Aloha和Hello不应该在一行显示。

各浏览器截图：

Aloha! Hello!	Aloha!
	Hello!
IE6(QS)/IE7(Q)(S)/IE8(Q)	其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	N	N	Y	Y	Y	Y

3. 选择器

子选择器

简介：

选择一个特定父级元素的所有直接子级元素。

更多说明见W3C CSS2.1规范[5.6](#)

实例代码：

```
<style type="text/css">
  div>p {
    color: red;
  }
</style>
<div>
  <p>Aloha!</p>
</div>
```

以上代码中，所有DIV元素下的P都会被选中，P中的文本颜色会变成红色。

各浏览器截图：

Aloha!	Aloha!
IE6(QS)/IE7(Q)/IE8(Q)	其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	Y	N	Y	Y	Y	Y

Bugs:

IE7中，如果在父级元素和子级元素之间有一个HTML注释，子选择器将不会工作。

普通兄弟选择器

简介：

选择指定元素后面的所有兄弟元素。

实例代码：

```
<style type="text/css">
  div~p {
    color: red;
  }
</style>
<div></div>
<p>Aloha!</p>
```

以上代码中，DIV元素后的P元素会被选中，其中文本变成红色字体。

各浏览器截图：

Aloha!	Aloha!
IE6(QS)/IE7(Q)/IE8(Q)	其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	Y	N	Y	Y	Y	Y

相邻兄弟选择器

简介：

选择紧邻指定元素的兄弟元素。

更多说明见W3C CSS2.1规范[5.7](#)

实例代码：

```
<style type="text/css">
  div+p {
    color: red;
  }
</style>
<div></div>
<p>Aloha!</p>
<p>Hello!</p>
```

以上代码中，紧邻DIV的第一个P会被选中，其内容文本变红，而第二个P的内容不会有变化，还是黑色。

各浏览器截图：

Aloha!	Aloha!
Hello!	Hello!
IE6(QS)/IE7(Q)/IE8(Q)	其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
---------	--------	--------	--------	--------	-------------	------------	------------

N	N	Y	N	Y	Y	Y	Y
---	---	---	---	---	---	---	---

Bug :

在IE7中，如果在兄弟之间有一个HTML注释，临近兄弟选择器将无效。

属性选择器

简介：

选择拥有指定属性的元素。

更多说明见W3C CSS2.1规范[5.8](#)

实例代码：

```
<style type="text/css">
  p[id] {
    color: red;
  }
</style>
<p id="aloha">Aloha!</p>
```

以上代码中，含有id属性的P元素会被选中，其内容文本会变成红色。

各浏览器截图：

IE6(QS)/IE7(Q)/IE8(Q) Aloha! Aloha!

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	Y	N	Y	Y	Y	Y

4. 伪类和伪元素

:first-child伪类

简介：

:first-child伪类匹配其他元素的第一个子元素。

更多说明见W3C CSS2.1规范[5.11.1](#)

实例代码：

```
<style type="text/css">
  div p:first-child {
    color: red;
  }
</style>
<div>
  <p>Aloha!</p>
  <p>Hello!</p>
</div>
```

以上代码中，DIV元素的第一个子元素P被匹配，所以字体被设置成红色。

各浏览器截图：

Aloha!	Aloha!
Hello!	Hello!
IE6(QS)/IE7(Q)/IE8(Q)	其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	Y	N	Y	Y	Y	Y

Bugs：

IE7中，如果要定位的第一个子元素之前有HTML注释，first-child伪类将会无效。

非锚点元素的:hover伪类

简介：

:hover伪类是指元素的鼠标悬停状态，可以应用到所有元素，不只是锚点元素。

更多说明见W3C CSS2.1规范[5.11.3](#)

实例代码：

```
<style type="text/css">
p:hover {
    color: red;
}
</style>
<p>Aloha!</p>
```

以上代码中，当鼠标悬停在P元素上时，P中的字体会变成红色。

各浏览器截图：（以下是鼠标悬停状态下的截图）

Aloha!	Aloha!
IE6(QS)/IE7(Q)/IE8(Q)	其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	Y	N	Y	Y	Y	Y

非锚点元素的:active伪类

简介：

:active伪类适用于一个元素被用户激活的时候。如鼠标在元素范围区域中按下但还没有释放的时候。

更多说明见W3C CSS2.1规范[5.11.3](#)

实例代码：

```
<style type="text/css">
```

```
p:active {
    color: red;
}
</style>
<p>Aloha!</p>
```

以上代码中，当鼠标在P元素上按下时，P中的字体会变成红色。

各浏览器截图：（以下是鼠标按下状态下的截图）

Aloha! Aloha!

IE6(QS)/IE7(S)(Q)/IE8(Q)

其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	N	N	Y	Y	Y	Y

:focus伪类

简介：

:focus伪类定位获取了键盘焦点的所有元素。

更多说明见W3C CSS2.1规范[5.11.3](#)

实例代码：

```
<style type="text/css">
    input:focus {
        color: red;
    }
</style>
<input value="Aloha!">
```

以上代码中，当INPUT获得焦点后，被选择，所以字体被设置成红色。

各浏览器截图：（以下是INPUT获得焦点后的截图）

Aloha! Aloha!

IE6(QS)/IE7(Q)(S)/IE8(Q)

其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	N	N	Y	Y	Y	Y

【注】关于:hover和:focus的更多信息，请参考《[动态伪类在各浏览器中的支持情况](#)》一文。

:before伪元素和:after伪元素

简介：

':before'和':after'伪元素用来在一个元素的内容之前或之后插入生成的内容。

更多说明见W3C CSS2.1规范[5.12.3](#)

实例代码：

```

<style type="text/css">
  p:before {
    content: "before";
  }
  p:after {
    content: "after";
  }
</style>
<p> Aloha! </p>

```

以上代码中，在P元素中“Aloha！”的前面和后面都用伪类添加了新的文本。

各浏览器截图：

Aloha! before Aloha! after
 IE6(QS)/IE7(Q)(S)/IE8(Q) 其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	N	N	Y	Y	Y	Y

5. 属性支持

min-height 和 min-width

简介：

这两个属性分别指定元素的高和宽的最小值，不允许一个盒子比指定的最小值小。

更多说明见W3C CSS2.1规范[10.4](#), [10.7](#)

实例代码：

```

<style type="text/css">
  div {
    min-width: 100px;
    min-height: 100px;
    width: 50px;
    height: 50px;
    background-color: silver;
  }
</style>
<div></div>

```

以上代码中，DIV元素的宽度和高度的最小值都是100px，后面设置的高度宽度值都是50px，所以最终DIV的高度和宽度都会是100px。

各浏览器截图：



IE6(QS)/IE7(Q)/IE8(Q)

其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	Y	N	Y	Y	Y	Y

max-height 和 max-width

简介：

这两个属性分别指定元素的高和宽的最大值，不允许一个盒子比指定的最大值大。

更多说明见W3C CSS2.1规范[10.4](#), [10.7](#)

实例代码：

```
<style type="text/css">
div {
    max-width: 50px;
    max-height: 50px;
    width: 100px;
    height: 100px;
    background-color: silver;
}
</style>
<div></div>
```

以上代码中，DIV元素的宽度和高度的最大值都是50px，后面设置的高度宽度值都是100px，所以最终DIV的高度和宽度都会是50px。

各浏览器截图：



IE6(QS)/IE7(Q)/IE8(Q)



其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	Y	N	Y	Y	Y	Y

position: fixed固定定位元素

简介：

position属性的fixed值允许一个元素绝对的相对于浏览器窗口定位。

更多说明见W3C CSS2.1规范[9.6.1](#)

实例代码：

```
<div id="DIV1" style="float:left; height:3000px;"></div>
<div id="DIV2" style="position:fixed; top:300px;">fixed div</div>
```

以上代码中，DIV1的高度为3000px，使浏览器窗口出现滚动条，DIV2是固定定位元素，它的位置不会随页面滚动而变化。

各浏览器截图：
无

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	Y	N	Y	Y	Y	Y

display特性的替代值

简介：

display 属性通常设置为block, inline, 或none。替代值包括：

- inline-block
- inline-table
- list-item
- run-in
- table
- table-caption
- table-cell
- table-column
- table-column-group
- table-footer-group
- table-header-group
- table-row
- table-row-group

更多说明见W3C CSS2.1规范[9.2.4](#)

实例代码：(仅以inline-block为例)

```
TEXT
<div style="display:inline-block;">
  Hello!
</div>
```

以上代码中，DIV的display特性值为inline-block，所以TEXT应该与Hello在同一行显示。

各浏览器截图：

TEXT
Hello!

TEXT Hello!

IE6(QS)/IE7(Q)(S)/IE8(Q)

其他浏览器

支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	N	N	Y	Y	Y	Y

outline特性

简介：

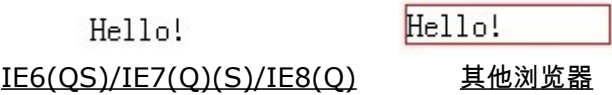
outline 是outline-style, outline-width, 和outline-color的缩写。该属性要优于border属性，因为它不会影响文档流。
更多说明见W3C CSS2.1规范[18.4](#)

实例代码：(仅以inline-block为例)

```
<div style="outline:1px solid red;">
  Hello!
</div>
```

以上代码中，DIV的outline被设置为1像素的红色框。

各浏览器截图：



支持情况：

IE6(QS)	IE7(Q)	IE7(S)	IE8(Q)	IE8(S)	Firefox(QS)	Chrome(QS)	Safari(QS)
N	N	N	N	Y	Y	Y	Y

三、如何避免受此问题影响

- 1. 使用浏览器的标准模式
- 2. 可用JS模仿一些低版本IE不支持的CSS标准。