

绝对定位元素在其四个方向的偏移为"auto"时的位置

Tags: position, absolute, left, top, right, bottom, auto, 绝对, 定位

操作系统版本:

Windows 7 Ultimate [ver 6.1.7600.16399] 简体中文版

浏览器版本:

IE6

IE7

IE8

Firefox 3.6.3

Chrome 5.0.375.3 dev

Safari 4.0.5

Opera 10.51

受影响的浏览器:

IE6 (S)(Q)

IE7 (S)(Q)

IE8 (Q)

Firefox (S)(Q)

Opera (S)(Q)

1. 绝对定位元素的"top"/"right"/"bottom"/"left"特性为"auto"时	1
2. inline元素对“静态位置”的影响	1
3. block元素对“静态位置”的影响	3
4. inline-block元素对“静态位置”的影响	4
5. float元素对“静态位置”的影响	6
6. 结论	7
7. 此现象可能导致的问题	7
8. 如何避免此现象的发生	7

1. 绝对定位元素的"top"/"right"/"bottom"/"left"特性为"auto"时

当一个元素的position特性不是"static"时，这个元素被称为“定位的”，此时定位元素生成的定位框由"top"、"right"、"bottom"、"left"特性决定。

当"top"、"right"、"bottom"、"left"特性没有设置，为默认的"auto"时，绝对定位的元素可能会被放置在其“静态位置”上。

“静态位置”粗略地说是指元素如果在正常流中的位置。具体来说：“静态位置”的包含块是若一个元素的"position"为"static"、"float"为"none"时生成元素的第一个框的假想框。

相关资料:

CSS Level 2 Revision 1, [9.3 Positioning schemes](#) ;

CSS Level 2 Revision 1, [9.7 Relationships between 'display', 'position', and 'float'](#) ;

CSS Level 2 Revision 1, [10.3.7 Absolutely positioned, non-replaced elements](#) (Width) ;

CSS Level 2 Revision 1, [10.6.4 Absolutely positioned, non-replaced elements](#) (Height)。

2. inline元素对“静态位置”的影响

对于没有设置"top"/"right"/"bottom"/"left"偏移量的绝对定位元素，当其前一个兄弟元素是**行内元素**时，在IE6/7及IE8(Q)中会将其的位置放置在行内元素同一行；在IE8(S)、Chrome、Safari中会参照其“静态位置”放置；在Firefox、Opera中当绝对定位元素的display特性为“inline-block”时，将换行显示。

测试代码：

```
<style>
* { margin:0; padding:0; font:12px Tahoma; line-height:110%; }
h3 { font-size:12px; font-weight:bold; margin-bottom:5px; }
.container { background:#DDD; width:95px; height:70px; margin-
bottom:3px; }
.in { display:inline; }
.bl { display:block; width:45px; }
.ib { display:inline-block; width:45px; }
.prev { background:#BBB; }
.pos { position:absolute; width:45px; height:45px; }
.next { background:#777; }
html* .ib { display:inline; }
</style>

<div style="float:left; display:inline; margin-right:3px;">
  <h3>POSITIONED</h3>
  <div class="container">
    <span class="prev in">INLINE</span><span class="pos next
in">INLINE POS</span>
  </div>
  <div class="container">
    <span class="prev in">INLINE</span><div class="pos next bl">BLOCK
POS</div>
  </div>
  <div class="container">
    <span class="prev in">INLINE</span><div class="pos next ib">INLINE
BLOCK POS</div>
  </div>
</div>
<div style="float:left; display:inline;">
  <h3>STATIC</h3>
  <div class="container">
    <span class="prev in">INLINE</span><span class="next in">INLINE
STATIC</span>
  </div>
  <div class="container">
    <span class="prev in">INLINE</span><div class="next bl">BLOCK
STATIC</div>
  </div>
  <div class="container">
    <span class="prev in">INLINE</span><div class="next ib">INLINE
BLOCK STATIC</div>
  </div>
</div>
```

上面一组代码中，均是第一个左浮动DIV里均是一个**行内元素**之后包含一个**没有设置偏移位置**的绝对定位元素，分别测试绝对定位元素是行内、块、行内块时的情形。第二组则是去除了绝对定位特性，为默认的"position:static"，即“静态位置”。在各浏览器中的运行效果如下：

IE6/7, IE8(Q)	IE8(S), Chrome, Safari	Firefox, Opera
---------------	------------------------	----------------

POSITIONED INLINE POS	STATIC INLINE STATIC	POSITIONED INLINE POS	STATIC INLINE STATIC	POSITIONED INLINE POS	STATIC INLINE STATIC
INLINE POS	INLINE BLOCK POS	INLINE BLOCK POS	INLINE BLOCK POS	INLINE BLOCK POS	INLINE BLOCK POS
INLINE BLOCK POS	INLINE BLOCK POS	INLINE BLOCK POS	INLINE BLOCK POS	INLINE BLOCK POS	INLINE BLOCK POS

可见，各浏览器对于右侧“position:static”是显示效果一致，而对于绝对定位元素：

- 在IE6(S)(Q)、IE7(S)(Q)、IE8(Q)中，浏览器并没有根据绝对定位元素的“静态位置”为其定位，而是由于绝对定位元素之前是一个行内元素，就将绝对定位元素直接放置在该行内元素同一行之后的位置上；
- 在Firefox、Opera中，对于绝对定位元素是行内及块元素时，其位置与其“静态位置”相符，但是当绝对定位元素是“display:inline-block”时，它被放置在了行内元素的下一行上；
- 在IE8(Q)、Chrome、Safari中，绝对定位元素的位置与其“静态位置”相符，符合W3C对绝对定位元素的规定。

3. block元素对“静态位置”的影响

对于没有设置“top”/“right”/“bottom”/“left”偏移量的绝对定位元素，当其前一个兄弟元素是块元素时，所有浏览器均会将绝对定位元素显示在该块元素下方新的一行上。

测试代码：

```

<style>
  * { margin:0; padding:0; font:12px Tahoma; line-height:110%; }
  h3 { font-size:12px; font-weight:bold; margin-bottom:5px; }
  .container { background:#DDD; width:95px; height:70px; margin-bottom:3px; }
  .in { display:inline; }
  .bl { display:block; width:45px; }
  .ib { display:inline-block; width:45px; }
  .prev { background:#BBB; }
  .pos { position:absolute; width:45px; height:45px; }
  .next { background:#777; }
  html* .ib { display:inline; }
</style>

<div style="float:left; display:inline; margin-right:3px;">
  <h3>POSITIONED</h3>
  <div class="container">
    <span class="prev bl">BLOCK</span><span class="pos next in">INLINE
  POS</span>
  </div>
  <div class="container">
    <span class="prev bl">BLOCK</span><div class="pos next bl">BLOCK
  POS</div>
  </div>
  <div class="container">
    <span class="prev bl">BLOCK</span><div class="pos next ib">INLINE

```

```

BLOCK POS</div>
</div>
</div>
<div style="float:left; display:inline;">
  <h3>STATIC</h3>
  <div class="container">
    <span class="prev b1">BLOCK</span><span class="next in">INLINE
STATIC</span>
  </div>
  <div class="container">
    <span class="prev b1">BLOCK</span><div class="next b1">BLOCK
STATIC</div>
  </div>
  <div class="container">
    <span class="prev b1">BLOCK</span><div class="next ib">INLINE
BLOCK STATIC</div>
  </div>
</div>

```

上面一组代码中仅仅是将上一节中每个【container】中的第一个inline元素换为了block元素，其他保持不变。在各浏览器中的运行效果如下：

All Browsers	
POSITIONED	STATIC
BLOCK INLINE POS	BLOCK INLINE STATIC
BLOCK BLOCK POS	BLOCK BLOCK STATIC
BLOCK INLINE BLOCK POS	BLOCK INLINE BLOCK STATIC

可见，此时所有浏览器对于绝对定位元素的位置判断均相同，且符合“静态位置”。这是由于绝对定位元素之前的兄弟元素已经为块级元素，产生了换行。无论上节中浏览器如何判断位置，绝对定位的元素都将显示在新的一行上。

4. inline-block元素对“静态位置”的影响

对于没有设置“top”/“right”/“bottom”/“left”偏移量的绝对定位元素，当其前一个兄弟元素是**行内块元素**时，在IE6/7及IE8(Q)中会将其的位置放置在行内元素同一行；在IE8(S)、Chrome、Safari中会参照其“静态位置”放置；在Firefox、Opera中当绝对定位元素的display特性为“inline-block”时，将换行显示。

测试代码：

```

<style>
  * { margin:0; padding:0; font:12px Tahoma; line-height:110%; }
  h3 { font-size:12px; font-weight:bold; margin-bottom:5px; }
  .container { background:#DDD; width:95px; height:70px; margin-bottom:3px; }

```

```

.in { display:inline; }
.bl { display:block; width:45px; }
.ib { display:inline-block; width:45px; }
.prev { background:#BBB; }
.pos { position:absolute; width:45px; height:45px; }
.next { background:#777; }
html* .ib { display:inline; }
</style>

<div style="float:left; display:inline; margin-right:3px;">
  <h3>POSITIONED</h3>
  <div class="container">
    <span class="prev ib">INLINE-BLOCK</span><span class="pos next
in">INLINE POS</span>
  </div>
  <div class="container">
    <span class="prev ib">INLINE-BLOCK</span><div class="pos next
bl">BLOCK POS</div>
  </div>
  <div class="container">
    <span class="prev ib">INLINE-BLOCK</span><div class="pos next
ib">INLINE BLOCK POS</div>
  </div>
</div>
<div style="float:left; display:inline;">
  <h3>STATIC</h3>
  <div class="container">
    <span class="prev ib">INLINE-BLOCK</span><span class="next
in">INLINE STATIC</span>
  </div>
  <div class="container">
    <span class="prev ib">INLINE-BLOCK</span><div class="next
bl">BLOCK STATIC</div>
  </div>
  <div class="container">
    <span class="prev ib">INLINE-BLOCK</span><div class="next
ib">INLINE BLOCK STATIC</div>
  </div>
</div>

```

上面一组代码中仅仅是将上一节中每个【container】中的第一个inline元素换为了inline-block元素，其他保持不变。在各浏览器中的运行效果如下：

IE6/7, IE8(Q)				IE8(S), Chrome, Safari				Firefox, Opera			
POSITIONED		STATIC		POSITIONED		STATIC		POSITIONED		STATIC	
INLINE BLOCK	INLINE POS	INLINE BLOCK	INLINE STATIC	INLINE BLOCK	INLINE POS	INLINE BLOCK	INLINE STATIC	INLINE BLOCK	INLINE POS	INLINE BLOCK	INLINE STATIC
INLINE BLOCK	BLOCK POS	INLINE BLOCK	BLOCK STATIC	INLINE BLOCK	BLOCK POS	INLINE BLOCK	BLOCK STATIC	INLINE BLOCK	BLOCK POS	INLINE BLOCK	BLOCK STATIC
INLINE BLOCK	INLINE BLOCK POS	INLINE BLOCK BLOCK	INLINE BLOCK STATIC	INLINE BLOCK	INLINE BLOCK POS	INLINE BLOCK BLOCK	INLINE BLOCK STATIC	INLINE BLOCK	INLINE BLOCK POS	INLINE BLOCK BLOCK	INLINE BLOCK STATIC

可见，对于绝对定位元素之前的兄弟元素为“display:inline-block”时的情况与“display:inline”时类似。

5. float元素对“静态位置”的影响

测试代码：

```
<style>
* { margin:0; padding:0; font:12px Tahoma; line-height:110%; }
h3 { font-size:12px; font-weight:bold; margin-bottom:5px; }
.container { background:#DDD; width:95px; height:70px; margin-
bottom:3px; }
.in { display:inline; }
.bl { display:block; }
.ib { display:inline-block; width:45px; }
.prev { float:left; }
.pos { position:absolute; width:45px; height:45px; }
.opa { opacity:0.5; filter:alpha(opacity=50); }
.next { background:#777; }
html* .ib { display:inline; }
</style>

<div style="float:left; display:inline; margin-right:3px;">
  <h3>POSITIONED</h3>
  <div class="container">
    <span class="prev">FLOAT</span><span class="pos next in
opa">INLINE POS</span>
  </div>
  <div class="container">
    <span class="prev">FLOAT</span><div class="pos next bl opa">BLOCK
POS</div>
  </div>
  <div class="container">
    <span class="prev">FLOAT</span><div class="pos next ib opa">INLINE
BLOCK POS</div>
  </div>
</div>
```

上面代码中，**浮动元素**后的兄弟元素为**没有设置偏移位置**的绝对定位元素，分别测试绝对定位元素是行内、块、行内块时的情形。在各浏览器中的运行效果如下：

IE6/7, IE8(Q)	IE8(S), Chrome, Safari	Firefox	Opera	静态位置
FLOAT INLINE POS	FLOAT INLINE POS	FLOAT INLINE POS	FLOAT POS	FLOAT INLINE STATIC
FLOAT BLOCK POS	FLOAT POS	FLOAT POS	FLOAT POS	FLOAT BLOCK STATIC
FLOAT INLINE BLOCK POS	FLOAT INLINE BLOCK POS	FLOAT BLOCK POS	FLOAT BLOCK POS	FLOAT INLINE BLOCK STATIC

可见，当绝对定位元素之前的兄弟元素为浮动元素时：

- 在**IE8(S)**、**Chrome**、**Safari**中，绝对定位元素的“静态位置”与其在"position:static"时最为符合；
- 在**IE6/7**、**IE8(Q)**中，无论绝对定位元素的"display"特性为何值，均不会覆盖浮动元素；
- 在**Firefox**中，仅当绝对定位元素的"display"特性为"inline"时，会覆盖浮动元素；

- 在Opera中，无论绝对定位元素的"display"特性为何值，均会覆盖浮动元素。

6. 结论

对于绝对定位元素之前的兄弟元素为普通流内的元素：

1. IE8(S)、Chrome、Safari对于绝对定位元素的“静态位置”判断正确，且符合W3C规范要求；
2. IE6/7、IE8(S)中，绝对定位元素会完全忽略其"display"特性，即当绝对定位元素之前的兄弟元素为“inline”或“inline-block”时，绝对定位元素均会显示在兄弟元素同一行上；
3. Firefox、Opera中，当绝对定位元素的"display"特性值为“inline-block”时，其始终会显示在其兄弟元素下面的一行上。

对于绝对定位元素之前的兄弟元素为浮动元素：

1. IE8(S)、Chrome、Safari对于绝对定位元素的“静态位置”判断正确，且符合W3C规范要求；
2. IE6/7、IE8(S)中，无论绝对定位元素的"display"特性为何值，均不会覆盖浮动元素；
3. Firefox中，仅当绝对定位元素的"display"特性为“inline”时，会覆盖浮动元素；
4. Opera中，无论绝对定位元素的"display"特性为何值，均会覆盖浮动元素。

7. 此现象可能导致的问题

若不为绝对定位元素设置其“top”、“right”、“bottom”、“left”特性，则这四个特性的值为默认的“auto”，此时由于各浏览器对规范的理解差异会导致很大的布局差异。

8. 如何避免此现象的发生

首先对于绝对定位元素，应尽可能避免使其“top”、“right”、“bottom”、“left”特性的值均为“auto”。若必须这么做，则尽可能的保证绝对定位元素之前的兄弟元素为**非浮动的块级元素**。