**hasLayout && Block Formatting Contexts**

1. **hasLayout**

概念说明：

* ‘Layout’ 可以被某些 CSS property（特性）**不可逆的触发**，而某些 HTML 元素本身就具有 layout 。
* ‘Layout’ 在 IE 中可以通过 hasLayout 属性来判断一个元素是否拥有 layout ，如 object.currentStyle.hasLayout 。
* ‘Layout’ 是 IE 浏览器渲染引擎的一个内部组成部分。在 IE 浏览器中，一个元素要么自己对自身的内容进行组织和计算大小， 要么依赖于包含块来计算尺寸和组织内容。为了协调这两种方式的矛盾，渲染引擎采用了 ‘hasLayout’ 属性，属性值可以为 true 或 false。 **当一个元素的 ‘hasLayout’ 属性值为 true 时，我们说这个元素拥有了一个布局（layout），即成功触发hasLayout**

触发方式：

* 默认拥有布局的元素：



可触发 hasLayout 的 CSS 特性：

****

1. **Block Formatting Contexts（BFC）**

IE有它自己的hasLayout属性，那么非IE浏览器呢？非IE浏览器采用的就是BFC（块格式化上下文）

* 概念说明：
  + **BFC**是 W3C CSS 2.1 规范中的一个概念，它决定了元素如何对其内容进行定位，以及与其他元素的关系和相互作用。
  + 在创建了 BFC的元素中，其子元素会一个接一个地放置。垂直方向上他们的起点是一个包含块的顶部，两个相邻的元素之间的垂直距离取决于 ‘margin’ 特性。在BFC中相邻的块级元素的垂直边距会折叠（collapse）。
  + 在BFC 中，每一个元素左外边与包含块的左边相接触（对于从右到左的格式化，右外边接触右边）， 即使存在浮动也是如此（尽管一个元素的内容区域会由于浮动而压缩），除非这个元素也创建了一个新的BFC。



*Tips:我们有时会用overflow:hidden的方法去清除浮动，就是因为触发了元素的块格式化上下文(IE6 7要申明zoom为1)，这个方法的确简单，但很暴力*

1. **hasLayout 和 BFC 的特点**
2. **在触发 hasLayout 的元素和创建了 BFC的元素中，浮动元素参与高度的计算**
3. **与浮动元素相邻的、触发了 hasLayout 的元素或创建了 BFC 的元素，都不能与浮动元素相互覆盖**
4. **触发 hasLayout 的元素和创建了 BFC的元素不会与它们的子元素发生外边距折叠**

**hasLayout和BFC 的异同及产生的问题**

共同点

* 两者都是决定了对内容如何定位及大小计算的规则。
* 两者都决定了与其他元素的相互作用的规则。
* 浮动元素，绝对定位元素，inline-block 元素以及除 ‘visible’ 外任意值的 overflow(IE7) 在 IE 中可以触发 hasLayout，同时在标准中，又可以创建BFC。

**产生折叠的必备条件：margin必须是邻接的!**

而根据w3c规范，两个margin是邻接的必须满足以下条件：

* 必须是处于常规文档流（非float和绝对定位）的[块级盒子](http://www.w3.org/TR/CSS2/visuren.html#block-boxes),并且处于同一个[BFC](http://www.w3.org/TR/CSS2/visuren.html#block-formatting)当中。
* 没有线盒，没有空隙（[clearance](http://www.w3.org/TR/CSS2/visuren.html#clearance)，下面会讲到），没有padding和border将他们分隔开
* **以上的条件意味着下列的规则：**
* 创建了新的BFC的元素（例如浮动元素或者'overflow'值为'visible'以外的元素）与它的子元素的外边距不会折叠
* [浮动](http://www.w3.org/TR/CSS2/visuren.html#floats)元素不与任何元素的外边距产生折叠（包括其父元素和子元素）
* 绝对定位元素不与任何元素的外边距产生折叠
* inline-block元素不与任何元素的外边距产生折叠