



## Close Gaps Next to Floated Images in Internet Explorer

By: [Zoe Gillenwater](#)

One of the most common tasks when laying out the content of a web page is floating images to the right or left so that text flows around them. This is dead simple to do with CSS:

```
img {  
  float: right;  
  margin: 0 0 1em 1.5em;  
}
```

A problem can arise if the text beside an image is not long enough to wrap around it. In this case, the next image could potentially float up into this space beneath the text, instead of sitting directly beneath the previous floated image. Once again, this is simple to fix:

```
img {  
  float: right;  
  clear: right;  
  margin: 0 0 1em 1.5em;  
}
```

The addition of the clear to the floated image ensures that each one will always sit below the previous one. However, placing the float and clear properties on the same element can cause large gaps to appear in Internet Explorer (IE) — gaps that take more complicated CSS to fix than what we've used so far.

To see the problem, download the support files for this article and open **IE\_adjacent\_float\_gap.html** in Dreamweaver. Preview it in IE. Unless you have a very narrow screen so that the text of the first paragraph is long enough to wrap around the first image, what you will see should resemble Image 1.



Image 1

What seems to be happening is this: IE floats the first image to the right. The text of the paragraph that follows it wraps around it. Next up in the XHTML source is the second image, which is also floated to the right and cleared below the first image. But instead of allowing the text that follows the second image to flow up beside the first image, as it ought to, the text acts as if it, too, had a clear applied to it.

When an IE bug is present, the first line of defense is to see if it falls to either of the two "magic bullets" described in [How To Attack An Internet Explorer \(Win\) Display Bug](http://www.communitymx.com/abstract.cfm?cid=C37E0) (<http://www.communitymx.com/abstract.cfm?cid=C37E0>) by Holly Bergevin and John Gallant. I'll save you some trouble — this one doesn't. No matter which elements you apply `position: relative` or `height: 1%` to, the gap remains.

What to do now? [Georg Sørtun](http://www.gunlaug.no/) (<http://www.gunlaug.no/>), a helpful and inventive member of [css-discuss](http://www.css-discuss.org/) (<http://www.css-discuss.org/>), came up with the idea to make the content adjacent to the floats inline. For some reason, perhaps because the clear property can only be applied to block-level elements, the content no longer acts like it is inheriting the clear, and the gap is gone.

Go to the Code View of `IE_adjacent_float_gap.html`. Add the conditional comment block below that includes special styles for IE only to the head of the document, below the closing `style` tag.

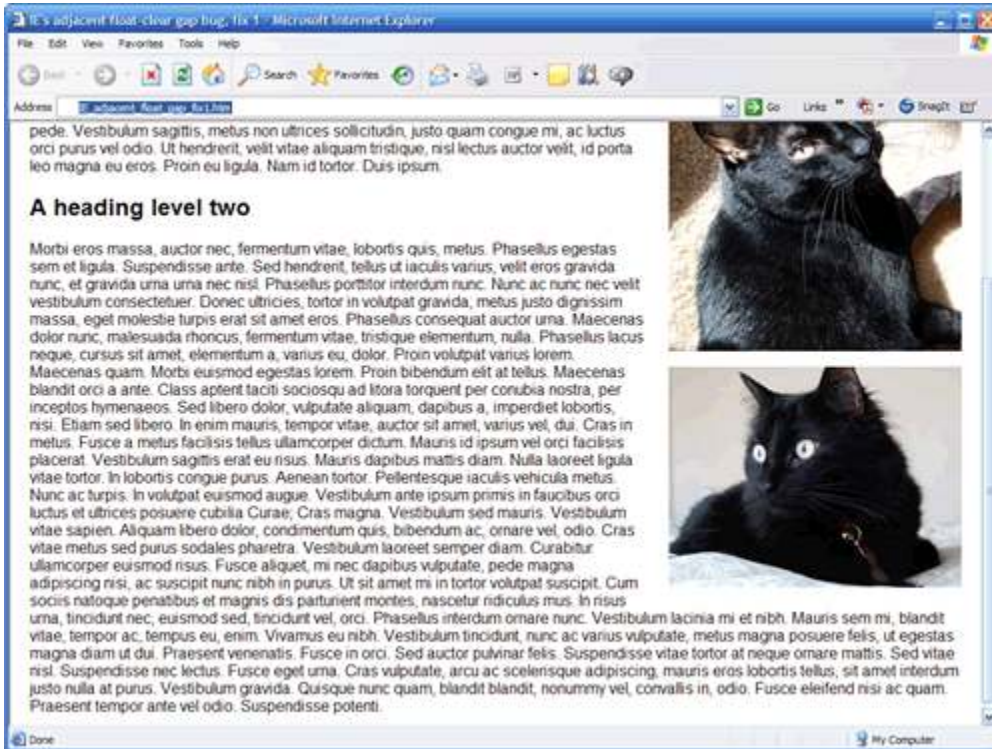
In a real page, the styles for all browsers would be placed in one style sheet, which would be linked or imported from the head of the page, while the styles for IE would be linked or imported from within the conditional comments. Here, the styles are listed within the head of the page itself for your ease of editing.

```

<!--[if IE]>
<style type="text/css">
p {
display: inline;
}
</style>
<![endif]>

```

If you preview in IE after adding the above code, you will see that the gap is now gone. Of course, since the paragraphs are all inline now, they all run together as one big lump of text. This is even worse than before!



**Image 2:** The gap is gone, but all of the paragraphs are run together.

Luckily, we can make IE treat the paragraphs like blocks again, without making the gap return, by giving them "layout."

"hasLayout" is an IE-only property that makes certain IE bugs disappear and changes how elements are laid out and relate to each other. If you want to dig into how hasLayout works, read this excellent article by Ingo Chao, [On having layout \(http://www.satzansatz.de/cssd/onhavinglayout.html\)](http://www.satzansatz.de/cssd/onhavinglayout.html).

The Holly Hack, described in the article [How To Attack An Internet Explorer \(Win\) Display Bug \(http://www.communitymx.com/abstract.cfm?cid=C37E0\)](http://www.communitymx.com/abstract.cfm?cid=C37E0) referenced earlier, is one way to give an element layout because it sets a dimension on the element. But the Holly Hack doesn't work here. Georg Sørtun had the answer again, though: use `zoom` instead. This is another IE-only property that induces layout, and if we set it to "1" it will not change the appearance of the text (if set to 2, it would zoom in to make the text twice as big, for instance). When it is applied to the paragraphs, they split apart again. With some padding added between them, they look like normal paragraphs once more (see Image 3). Modify your IE-only CSS to match the block below:

```
p {
display: inline;
zoom: 1;
padding: .5em 0;
}
```



Image 3

Although an h2 element, not just paragraphs, is also adjacent to the floated and cleared images, applying this fix to the paragraphs only was good enough in this case — IE is a mysterious browser! However, on real pages that are more complicated, you may need to apply the fix to all content that is adjacent to gap-inducing floats. Tinkering is always present when IE needs to be supported!

The only other thing we may want to tweak at this point is the spacing of the the headings in relationship to the paragraphs. Headings have default top and bottom margins. These margins are butting up against and adding to the padding we've applied to the paragraphs to space them out from one another. So, add these additional rules within the conditional comment:

```
h1 {
margin: .3em 0;
}
h2 {
margin: .5em 0;
}
```

Now the spaces between the headings and paragraphs are consistent with how they looked before.

There's one niggling side effect of this method. If you scroll down the page, you will see that the paragraph that is adjacent to the bottom of the second photo no longer wraps around that photo's bottom (see Image 4). This new gap cannot be prevented or fixed. However, it will occur much less frequently



than the first gap, usually be smaller in size, and is generally easy to overlook, so this solution is best for most situations.



**Image 4:** Notice the gap beneath the second image that occurs because the text of an individual paragraph can no longer wrap around a float.

If, however, you have a very picky client who uses IE and cannot stand the gap underneath some images, there is another method that you can use, courtesy of [Ingo Chao \(http://www.satzansatz.de/\)](http://www.satzansatz.de/), another creative css-discuss member.

Go back to your XHTML file and remove the entire conditional comment — it's no longer needed. Instead, add the following class:

```
.fixgap {
float: right;
height: 0;
}
```

Then, add this line of HTML immediately before the second floated image:

```
<div class="fixgap"><!-- --></div>
```

Preview the file in IE. Both the gap above the text and below the photo will be gone.

You can add this empty div above every float-clear combination element. If you go with this method, keep in mind that you are adding unneeded bulk to your pages and that there's a chance (especially if your pages are maintained by someone else) that the empty divs could be accidentally removed down the road. For these reasons, the first method is recommended in most cases.

## Summary

### Recommended Fix

Add this rule to an IE-only style sheet:

```
p {  
  display: inline;  
  zoom: 1;  
  padding: .5em 0;  
}
```

Also, adjust margins on any content adjacent to the paragraphs.

See the file **IE\_adjacent\_float\_gap\_fix1.html**.

### Alternate Fix

Add this rule to the main style sheet:

```
.fixgap {  
  float: right;  
  height: 0;  
}
```

Add this line of HTML immediately before the float-clear combination element:

```
<div class="fixgap"><!-- --></div>
```

See the file **IE\_adjacent\_float\_gap\_fix2.html**.

Thanks to [Georg Sørtun](http://www.gunlaug.no/) (<http://www.gunlaug.no/>), [Ingo Chao](http://www.satzansatz.de/) (<http://www.satzansatz.de/>), and [Bruno Fassino](http://www.brunildo.org/) (<http://www.brunildo.org/>) for their creative fixes to yet another IE bug!

Approximate download size: 285k

#### Keywords

CSS, cascading style sheets, stylesheets, Windows Internet Explorer, IE, float, floats, floated image, clear, gaps, bugs, hacks, hasLayout, zoom, inline