

# Getting Started with Cufon

Posted At : April 9, 2010 11:02 PM | Posted By : Jon Hartmann

Related Categories: Web Trends, Examples & Tutorials, jQuery



I've been curious for a while about techniques for font replacement on the web. We've probably all heard about the idea of embedding fonts, but looking around at the standards right now, it doesn't sound like any one browser has things right, so I decided to explore a third-party option. Read more to learn about the basics of Cufon, an easy to use font "embedding" system.

One of the primary reasons that I took a look at **Cufon** is that, unlike some other alternatives like **sIFR**, it doesn't required Flash. Cufon uses Canvas to render the font directly in the browser and uses VML for Internet Explorer. Even better, it's fairly simple to implement.

[See Cufon in action in the completed example.](#)

## Keep it Legal

First things first, **always** make sure that you have a legal license to use the fonts on the web. Just because you downloaded it for free from a website or it came with Windows doesn't mean you've got license to use it on your site. Luckily a license for most fonts you can think of can be obtained for a reasonable price, and there are a fair number of fonts out there that have open license for you to embed in your site, such as **Titillium** and **Merge** (both which I will be using for this example).

## Step 1: Get Cufon

The first step to getting Cufon up and running is to download the YUI compressed Cufon script from **Cufon**. We're also going to need to include some scripts that describe your fonts, but that's a whole step in and of itself, and we'll get to it soon enough. If you're a jQuery fan though, this is where you'll also want to include it as well. jQuery you say? Yes! Cufon doesn't need jQuery at all, but it can pick up on the presence of jQuery and extend its built in selector engine to let you target your text replacement with jQuery selectors. So, if you do the jQuery thing, include it now:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <script http-equiv="Content-Type" content="text/html; charset=utf-8">
    <script src="cufon-yui.js" type="text/javascript"></script>
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js" type="text/javascript"></script>
  </head>
  <body>

  </body>
</html>
```

## Step 2: Create Font Scripts

Now we need to use the **Cufon Generator** to create scripts based on your fonts. What's this doing? Basically its converting your font file into Javascript instructions to create the Canvas and VML elements you need to recreate your font in the browser. You'll need to download the actual to .ttf or .otf files to convert first, so if you're following along go ahead and grab **Titillium** and **Merge**.

Now, let's generate a script for each font. The Cufon generator has a lot of options, and I'd recommend that you explore them thoroughly for production projects, but the only one that I'd pay attention to right now is the option "Use the following value as the font-family of the generated font": this option can be very useful if you've got multiple fonts being used on a page. For this simple demonstration, I just checked that the EULAs were ok, that I wanted All glyphs to be converted, and that I acknowledge and accept the terms and conditions At the end you should have generated two .js files (Merge\_400.font.js and TitilliumText22L\_400.font.js for me). We'll want to include these into the header like so:

```
...
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <script src="cufon-yui.js" type="text/javascript"></script>
  <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js" type="text/javascript"></script>
  <script src="Merge_400.font.js" type="text/javascript"></script>
```

```

<script src="TitilliumText22L_400.font.js" type="text/javascript"></script>
</head>
...

```

### Step 3: Create HTML

We're going to need some HTML to replace, so let's create some simple text that we want to upgrade with Cufon.

```

...
<body>
  <h1>I love the Merge font</h1>
  <p>
    The font above is Merge, and its cool.
  </p>
  <h2>I also like Titillium.</h2>
  <p>
    Its a little different, but it looks cool, too.
  </p>
  <p class="fancy">
    This text will be shown in Merge.
  </p>
  <p class="fancy2">
    This text will be shown in Titillium.
  </p>
</body>
...

```

### Step 4: Time for Cufon!

Easy so far, right? Now let's replace some text with Cufon. We'll need a new script tag in the header, and we'll target the H1 tag like so:

```

...
<script type="text/javascript">
  Cufon.replace('h1');
</script>
...

```

If you've been following along, you might notice that something looks wrong once you've added this in. Cufon's definitely replaced the H1, but it's applied Titillium! This isn't really surprising though: we've included two scripts in the page, but we've not told Cufon which one we want to apply to H1. To remedy this, all we need to do is indicate a fontFamily for our replace function like so:

```

...
<script type="text/javascript">
  Cufon.replace('h1', { fontFamily: 'Merge' });
</script>
...

```

If you're only going to use a single Cufon font on your site, you can get away with dropping fontFamily, but I'd recommend it as good practice, especially because if you add another font later, Cufon breaks if *any* replacement doesn't specify the font family.

One gotcha of Cufon is that sometimes in IE, the replacement can be a little sluggish. To get around this problem, they recommend adding a call to Cufon.now() at the end of your body tag like so:

```

...
<body>
  ...
  <script type="text/javascript">
    Cufon.now();
  </script>

```

```

</body>
...

```

Easy enough, and it keeps IE in line, so don't forget it!

### Step 5: Kicking it up a Notch

Ok, we've replaced one font which means that technically we're done with the "steps", but let's kick it up a notch! Lets replace the h2 with Titillium, and because I want to assure you that Cufon works with CSS declarations, lets adjust the h2 with a style as well:

```

...
<script type="text/javascript">
    Cufon.replace('h1', { fontFamily: 'Merge' });
    Cufon.replace('h2', { fontFamily: 'TitilliumText22L' });
</script>
<style>
    h2 {
        color: blue;
        font-style: italic;
        font-size: 1em;
    }
</style>
...

```

Run it again, and you'll find that H2 renders in Titillium, is blue, and is 1em in size, but you'll find it is **not** going to be italic. This is because we didn't include an italic version of the font when we generated the .font.js file for Titillium. If you want bold or italic versions, you're going to have to make sure you include that font when generating the scripts.

### Step 6 (optional): Bam! jQuery!

Lastly, remember how I told you jQuery fans to go ahead and plug it in the header? Well, we'll go ahead and use jQuery to make Cufon replace the font in our two special CSS classes, fancy and fancy2:

```

...
<script type="text/javascript">
    // jQuery Not Required
    Cufon.replace('h1', { fontFamily: 'Merge' });
    // jQuery Not Required
    Cufon.replace('h2', { fontFamily: 'TitilliumText22L' });
    // jQuery Required
    Cufon.replace('p.fancy', { fontFamily: 'Merge' });
    // jQuery Required
    Cufon.replace('p.fancy2', { fontFamily: 'TitilliumText22L' });
</script>
...

```

Run it all again and you should see the text in the last two paragraphs rendered in Merge and Titillium, respectively.

### You're Finished!

Congrats, you just replaced text with Cufon! Not that hard is it? So far I like how easy it is to create a font for Cufon and integrate it into a site. I'm not 100% crazy about its rendering (some things looked a little blurry), but I think that that might be rectified by using a different font size that better matches the font. Hopefully with this example you can start exploring an internet without font boundaries!

[Download the completed code for this example.](#)