

# 10 Quick Tweaks to Speed Up Your Website



## 1. Reduce your HTTP Requests

Every time your website requests a new ‘thing’ from a server, the webpage has to pause while it goes and gets it. Look at your page source. How many CSS files does it pull in? How many JS files does it pull in? How many images does it pull in?

Try to reduce the number of CSS and JS files you load to one each. Try to reduce the number of individual images you pull in. I’ll go deeper on these things in other tweaks below: for this tweak, just try to reduce what’s there.

## 2. Optimize your images

Not only are images http requests, but they're also typically the heaviest elements of any page. They're well worth optimizing.

For raster images (such as PNGs, JPGs, etc.) serve them at their correct size for their purpose. Measure their usage space and serve an image of that size, or twice that size for retina images (if you know how to serve variations based on screen resolution). Don't serve retina-sized images for everyone, and don't serve images larger than they need to be.

Run raster images through [ImageOptim](#), to squash their file sizes down without artifacting the images. Run vector images (SVG) through [SVGO](#) to get the same effect.

## 3. Use a CDN

A CDN (Content Delivery Network) can serve your content from a server close to your visitor geographically. Since content has less distance to travel, your visitors will notice an decrease in page load times.

[Cloudflare](#) is a great CDN which even has a free plan, so you can start using them right away, no excuses!

## 4. Preprocess Styles

Preprocessors like SASS have the ability to squash stylesheets into one file. This lets you serve multiple stylesheets from a single CSS file, which goes very nicely with tweak #1.

## 5. Grunt/Gulp your Javascript

These tools give you the same ability as tweak #4, but for your Javascript. Using a gruntfile, you can serve multiple javascript files from a single JS, which again goes great with tweak #1.

## 6. Minify & Uglify Everything

Essentially, this means “squash code as small as possible”. All whitespace, comments and unnecessary code gets removed from your files, resulting in smaller files that are faster to load. Combine this with tweak #4 for a “`sass —watch foo:bar —style compressed`” method, or include concatenation and uglify node modules with tweak #5.

## 7. Load the basics first, the rest later

Google is hot on this idea right now. Essentially, abstract the main styles that serve up the skeleton of your site first, then serve the rest after the DOM is done loading. This will make your site ‘feel’ faster, as there’s no waiting for http requests to be completed before continuing to load your page.

## 8. Remove any ‘loading’ elements on from your site

We often see elements that suggest something is ‘loading’ when things are slow. Think Microsoft Word, Adobe Photoshop, your Operating System, etc. Loading means slow. By removing any spinning ‘loading’ icons and progress bars, your site will feel faster even if it isn’t.

## 9. Go Mobile First

Mobile first means smallest devices load the least code. For phones, which are typically less powerful than laptops, and are often on pricey data connections, serving up less code on smaller devices just makes sense.

This goes against the common practice of ‘adding mobile styles’ on top of desktop code, which essentially gives the slowest devices the most work.

## 10. Lazy-Load below-the-fold content

Images that aren’t visible when you first load the page will increase load times without any perceived gain. Instead, load them just before you need them, as visitors scroll down to see them, to increase your initial page load times.

For more exclusive content and freebies like this one to help you grow your online business, join us at [FairheadCreative.com](http://FairheadCreative.com)