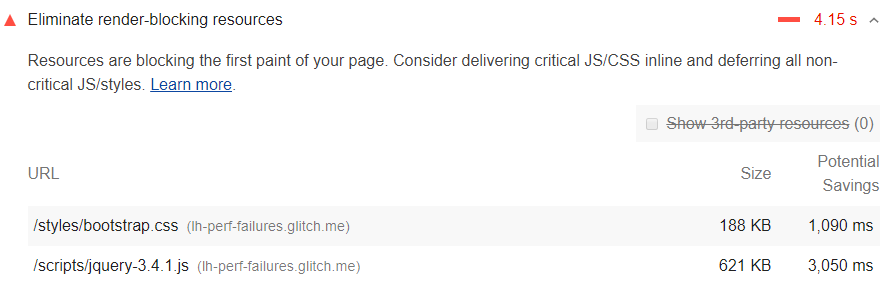
**Eliminate render-blocking resources**

May 2, 2019 • Updated Aug 11, 2020

Appears in: [Performance audits](https://web.dev/lighthouse-performance)

The Opportunities section of your Lighthouse report lists all URLs blocking the first paint of your page. The goal is to reduce the impact of these render-blocking URLs by inlining critical resources, deferring non-critical resources, and removing anything unused.



**Which URLs get flagged as render-blocking resources?** [**#**](https://web.dev/render-blocking-resources/?utm_source=lighthouse&utm_medium=devtools#which-urls-get-flagged-as-render-blocking-resources)

[Lighthouse](https://developers.google.com/web/tools/lighthouse/) flags two types of render-blocking URLs: scripts and stylesheets.

A <script> tag that:

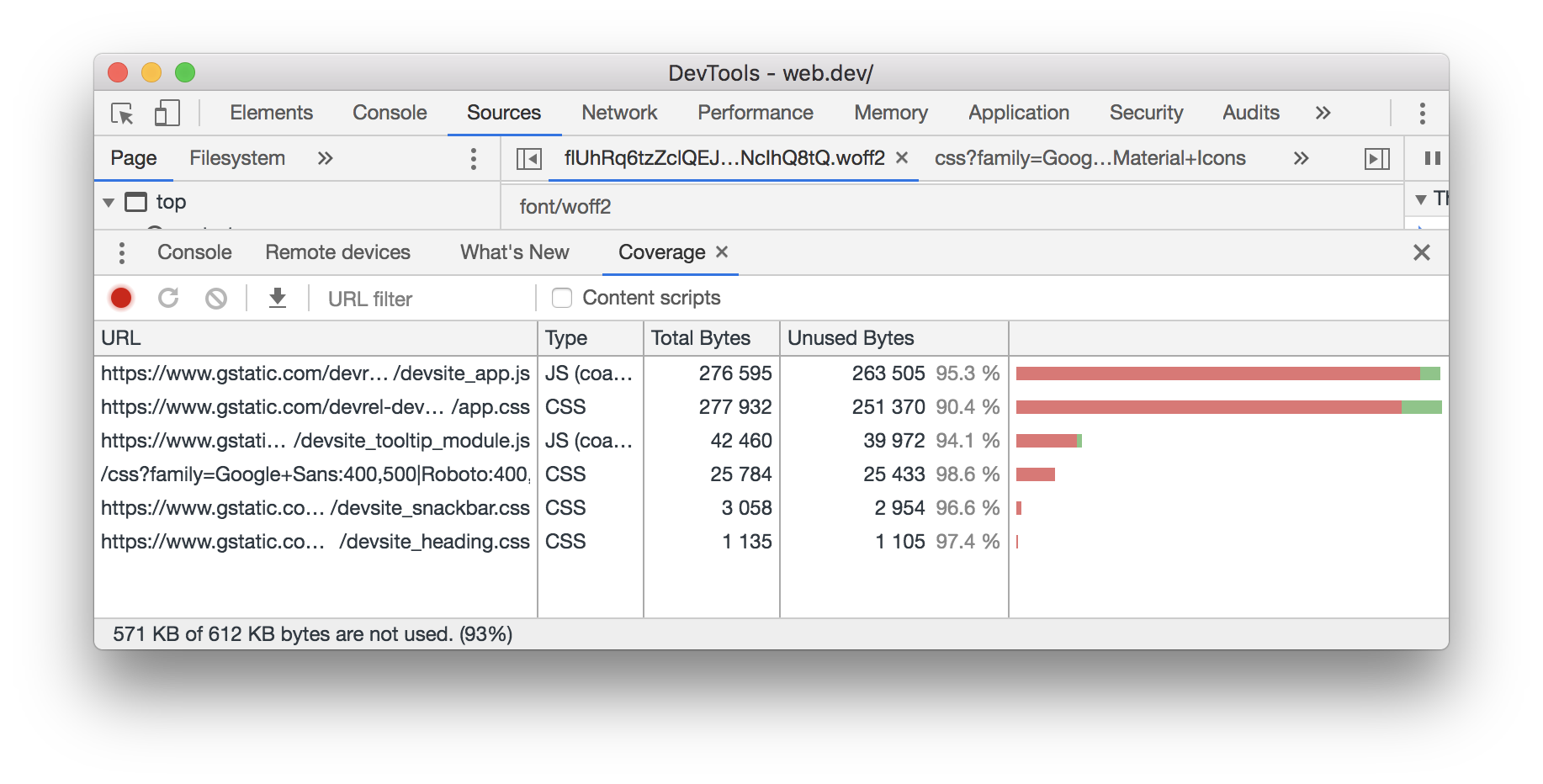
* Is in the <head> of the document.
* Does not have a defer attribute.
* Does not have an async attribute.

A <link rel="stylesheet"> tag that:

* Does not have a disabled attribute. When this attribute is present, the browser does not download the stylesheet.
* Does not have a media attribute that matches the user's device specifically. media="all" is considered render-blocking.

**How to identify critical resources** [**#**](https://web.dev/render-blocking-resources/?utm_source=lighthouse&utm_medium=devtools#how-to-identify-critical-resources)

The first step to reducing the impact of render-blocking resources, is to identify what's critical and what's not. Use the [Coverage tab](https://developer.chrome.com/docs/devtools/coverage/) in Chrome DevTools to identify non-critical CSS and JS. When you load or run a page, the tab tells you how much code was used, versus how much was loaded:

Chrome DevTools: Coverage tab.

You can reduce the size of your pages by only shipping the code and styles that you need. Click on a URL to inspect that file in the Sources panel. Styles in CSS files and code in JavaScript files are marked in two colors:

* **Green (critical):** Styles that are required for first paint; code that's critical to the page's core functionality.
* **Red (non-critical):** Styles that apply to content not immediately visible; code not being used in page's core functionality.

**How to eliminate render-blocking scripts** [**#**](https://web.dev/render-blocking-resources/?utm_source=lighthouse&utm_medium=devtools#how-to-eliminate-render-blocking-scripts)

Once you've identified critical code, move that code from the render-blocking URL to an inline script tag in your HTML page. When the page loads, it will have what it needs to handle the page's core functionality.

If there's code in a render-blocking URL that's not critical, you can keep it in the URL, and then mark the URL with async or defer attributes (see also [Adding Interactivity with JavaScript](https://developers.google.com/web/fundamentals/performance/critical-rendering-path/adding-interactivity-with-javascript)).

Code that isn't being used at all should be removed (see [Remove unused code](https://web.dev/remove-unused-code)).

**How to eliminate render-blocking stylesheets** [**#**](https://web.dev/render-blocking-resources/?utm_source=lighthouse&utm_medium=devtools#how-to-eliminate-render-blocking-stylesheets)

Similar to inlining code in a <script> tag, inline critical styles required for the first paint inside a <style> block at the head of the HTML page. Then load the rest of the styles asynchronously using the preload link (see [Defer unused CSS](https://web.dev/defer-non-critical-css)).

Consider automating the process of extracting and inlining "Above the Fold" CSS using the [Critical tool](https://github.com/addyosmani/critical/blob/master/README.md).

Another approach to eliminating render-blocking styles is to split up those styles into different files, organized by media query. Then add a media attribute to each stylesheet link. When loading a page, the browser only blocks the first paint to retrieve the stylesheets that match the user's device (see [Render-Blocking CSS](https://developers.google.com/web/fundamentals/performance/critical-rendering-path/render-blocking-css)).

Finally, you'll want to minify your CSS to remove any extra whitespace or characters (see [Minify CSS](https://web.dev/minify-css)). This ensures that you're sending the smallest possible bundle to your users.

**Stack-specific guidance** [**#**](https://web.dev/render-blocking-resources/?utm_source=lighthouse&utm_medium=devtools#stack-specific-guidance)

**AMP** [**#**](https://web.dev/render-blocking-resources/?utm_source=lighthouse&utm_medium=devtools#amp)

Use tools such as [AMP Optimizer](https://github.com/ampproject/amp-toolbox/tree/master/packages/optimizer) to [server-side render AMP layouts](https://amp.dev/documentation/guides-and-tutorials/optimize-and-measure/server-side-rendering/).

**Drupal** [**#**](https://web.dev/render-blocking-resources/?utm_source=lighthouse&utm_medium=devtools#drupal)

Consider using a module to inline critical CSS and JavaScript, or potentially load assets asynchronously via JavaScript such as the [Advanced CSS/JS Aggregation](https://www.drupal.org/project/advagg) module.

**Joomla** [**#**](https://web.dev/render-blocking-resources/?utm_source=lighthouse&utm_medium=devtools#joomla)

There are a number of Joomla plugins that can help you [inline critical assets](https://extensions.joomla.org/instant-search/?jed_live%5Bquery%5D=performance) or [defer less important resources](https://extensions.joomla.org/instant-search/?jed_live%5Bquery%5D=performance).

**WordPress** [**#**](https://web.dev/render-blocking-resources/?utm_source=lighthouse&utm_medium=devtools#wordpress)

There are a number of WordPress plugins that can help you [inline critical assets](https://wordpress.org/plugins/search/critical+css/) or [defer less important resources](https://wordpress.org/plugins/search/defer+css+javascript/).

## Description

**Hummingbird makes your website faster and optimizes site performance by adding new ways to boost Google PageSpeed Insights with fine-tuned controls over file compression, deferring CSS and JavaScript styles and scripts, minify for CSS and JS, Lazy Load integration, and world-class caching.**

Hummingbird is brought to you by the WordPress speed specialists that created [Smush](https://wordpress.org/plugins/wp-smushit/) image optimization, now active on more than +1 million websites. Get the complete speed boost with Hummingbird and Smush.

### If PageSpeed Insights is making these speed recommendations Hummingbird can help:

* Enable text compression – Use gzip to make your site fly.
* Preconnect to required origins – Use Hummingbird to establish earlier connections.
* Preload key requests – Prioritize resources based on order.
* Avoid enormous network payloads – Consider Lazy Load for comments or breaking up smaller posts.
* Use efficient cache – The Hummingbird Cache suite offers effective browser cache for caching any site.
* Fix your JavaScript execution time – Deliver smaller JS payloads, preload JS, and defer JS.
* Minify CSS – Strip unused code from your CSS.
* Minify JavaScript – Speed up the time it takes to parse your JS files.
* Eliminate render-blocking resources – Move critical CSS and JS inline and defer all non-critical JS/CSS.
* Defer unused CSS – Defer the loading of CSS not used for above-the-fold content.
* Lazy Load offscreen images ([Smush free](https://wordpress.org/plugins/wp-smushit/) integration).

Hummingbird scans your site and provides one-click fixes to speed up WordPress in a flash.

You’ll get faster loading pages, higher search rankings (SERP) and PageSpeed scores, and happier visitors with Hummingbird’s WordPress speed optimization. Optimizing the speed of your site has never been easier!

### Features Available in Hummingbird Include:

* Scan and Fix – Get a scan of your site, find out what’s slowing it down, and use one-click performance improvements to make critical speed improvements.
* World-class caching – A full caching suite to load pages faster with full-page, Gravatar, and browser cache tool.
* Performance Reports – Pro tips for running your site at super speed.
* Asset Optimization – Position, minify and combine Javascript, CSS, and Google Font files for top performance.
* Better Rankings – Improve scores on Google PageSpeed Insights (SEO ranking factor), YSlow, Pingdom, and GTmetrix.
* Increase Your Conversion Rate – Don’t keep visitors waiting: faster sites convert better.
* GZIP Compression – Blazing-fast HTML, JavaScript, and stylesheet (CSS) transfer.
* Configs – Set your preferred performance settings, save them as config, and instantly upload to any other site.

### Learn The Ropes With These Hands-On Hummingbird Tutorials

* [How To Optimize WordPress For Speed With Hummingbird](https://wpmudev.com/blog/optimize-wordpress-speed-hummingbird/)
* [How To Optimize Elementor for Free Using Smush and Hummingbird](https://wpmudev.com/blog/optimize-elementor-wordpress-smush-hummingbird/)
* [How To Optimize WPBakery Sites Using Smush And Hummingbird](https://wpmudev.com/blog/optimize-wordpress-wpbakery-free-plugins/)
* [Speed Up and Optimize Avada for Free Using Smush and Hummingbird](https://wpmudev.com/blog/speed-up-optimize-avada-theme-smush-hummingbird-plugins/)
* [Optimize Divi for Free Using Smush and Hummingbird](https://wpmudev.com/blog/optimize-divi-wordpress-smush-hummingbird/)

#### Hummingbird Features to Speed Up WordPress

### Scan and One-Click Fix

Hummingbird is a WordPress speed optimization plugin. It will scan your site, find files that are slowing it down, and provide tips and fixes for making your site run at top speed.

Hummingbird even has one-click improvements like a full cache suite, one-click minify for styles and scripts, and deferring CSS and JS for quickly optimizing performance. What could be easier!

### World-Class Caching

You’ll get a world-class caching suite, including full-page, browser, and Gravatar cache.

Make your site load even faster with Hummingbird’s complete set of cache tools that give your visitors a faster browsing experience. Including full-page, browser and Gravatar caching.

### Asset Optimization

Did you know that the more files you add to your site’s header, the slower it will load? With Hummingbird, you can easily customize the load position of your CSS, JavaScript (defer CSS and Javascript), Google Fonts, and other files to increase your page speed.

Quickly reorder, compress, and reposition files with Hummingbird’s minification feature. But play carefully – minification is a powerful tool (though if you need it, you can reset any changes you make in one click).

### Transfer Data at Top Speed With GZIP

Hummingbird has GZIP powers to make sharing your site more efficient.

Sending zipped files is faster and can save you money on hosting. And don’t worry about setup, send Hummingbird instructions with the click of a button and she’ll handle the rest.

### Built-in Cloudflare Integration

Hummingbird can be used to control your [Cloudflare](https://www.cloudflare.com/) browser cache and Automatic Platform Optimizations (APO) settings as well! Simply add your Cloudflare API key and configure away.

### Fully Compatible With Smush Image Optimization

You can complement Hummingbird’s WordPress speed optimization features with our award-winning sister-plugin [Smush image optimization](https://wordpress.org/plugins/wp-smushit/). Smush compresses your images, giving your site less to load – and thus a faster load time. Hummingbird + Smush integrate perfectly together, and are the perfect match to speed up WordPress.

Compress, optimize (optimise), and fix PageSpeed performance with properly sized images, lazy load, next-gen WebP convert, image formatting, and more.

### Save time with Hummingbird Configs

Configs allow you to save your prefered Hummingbird configuration settings and apply them to your other sites in a few clicks. You can create unlimited configs.

#### Faster Websites Rank Higher, Convert Better

Every millisecond counts: your visitors expect an ever-faster website, with a page load time of under two seconds expected – and the norm. If visitors don’t get that on your site, they will leave.

If you’re running a business website or eCommerce store, that means **if your website does not load quickly, you will lose sales.**

Hummingbird is here to help you; it’s a one of a kind WordPress performance optimization plugin that can make your site run at superspeed, for free!

You get our WordPress performance optimization suite, which includes minification and GZIP for small page sizes, full caching for faster loading, and integration with Cloudflare’s APO / browser cache, and our sister-plugin [Smush image optimization](https://wordpress.org/plugins/wp-smushit/).

Hummingbird is built with ease-of-use in mind; it makes your WordPress site faster, but it’s also fast to set up. You can scan your site and implement recommended changes in one-click, getting a fast site in mere minutes.

All the above is free and will speed up WordPress for you. **If you need the very fastest WordPress site,** [**you should get a WPMU DEV Membership**](https://wpmudev.com/?utm_source=wordpress.org&utm_medium=readme&utm_campaign=hummingbird-readme&utm_content=you_should_get_wpmudev_membership#trial)**.**

Our Membership gives you access to Hummingbird Pro – which features automated scanning, uptime monitoring, enhanced minify compression (with 2x the regular optimization), CDN hosted minification – alongside Smush Pro image optimization, all our premium WordPress plugins, and 24/7 WordPress support.