

Lighthouse January 2017 update



By Brendan Kenny.

Lighthouse Bulb Installer



By Eric Bidelman

Engineer @ Google working on web tooling: Headless Chrome, Puppeteer, Lighthouse

[Lighthouse](#) is an [open-source](#), automated tool for improving the quality of your web apps. You can install it as a [Chrome Extension](#) or run it as a Node command line tool. When you give Lighthouse a URL, it runs a barrage of tests against the page and then generates a report explaining how well the page did and indicating areas for improvement.



Today, we're happy to announce the [1.5 release](#) of Lighthouse, a huge release, with over **128 PRs**. Lighthouse 1.5 includes a bunch of big new features, audits, and the usual bug fixes. You can install it from npm (`npm i -g lighthouse`) or [download the extension](#) from the Chrome Web Store.

New Audits

The **CSS Usage Audit** reports the number of unused style rules in your page and the cost/time savings of removing them:

✖ Uses 90% of its CSS rules: **106KB (~530ms) potential savings** ?







▼ More information

URL	Unused Rules	Original (KB)	Potential Savings (%)
/css	0	3 KB	0%
css/devsite-google-blue.css	756	18 KB	80%
inline nav.devsite-page-nav,	0	0 KB	0%
inline article.hentry:first- <	0	0 KB	0%
embed/P95ZDIzjg0Q	1	0 KB	100%
cssbin/www-embed-player-s	1870	93 KB	97%

The **Optimized Images Audit** reports images that are unoptimized and the cost/time savings of optimizing them:

✖ Has optimized images: **359KB (~1780ms) potential savings** ?






▼ More information

	URL	Original (KB)	WebP Savings (%)
	images/web-fundamentals-icon192x192.png	10 KB	47%
	images/ic_home_black_48d	13 KB	60%
	images/ic_thumb_up_black_	13 KB	54%
	images/ic_https_black_48d	17 KB	56%
	images/ic_important_device	16 KB	57%
	common/push-notification-16x9.png	348 KB	92%

The **Responsive Images Audit** reports images that are too big and the potential cost/time savings of sizing them correctly for the given device:

✖ Has appropriately sized images: **436KB (~2160ms) potential savings** ⓘ

▼ More information

	URL	Original (KB)	Potential Savings (%)
	images/web-fundamentals-icon192x192.png	10 KB	100%
	common/push-notification-16x9.png	348 KB	94%
	images/ic_important_devices_black_4	16 KB	94%
	images/ic_thumb_up_black_48dp.png	13 KB	94%
	images/ic_home_black_48dp.png	13 KB	94%

The **Deprecations and Interventions Audit** lists console warnings from Chrome if your page is using deprecated APIs or features that have interventions:

✖ Avoids deprecated APIs: **2 warnings found** ⓘ

▼ URLs

<https://docs-05-dot-polymer-project.appspot.com/0.5/components/core-header-panel/core-header-panel.html> (line: 258)

`::shadow pseudo-element is deprecated. See
https://www.chromestatus.com/features/6750456638341120 for more details.`

<https://docs-05-dot-polymer-project.appspot.com/0.5/components/polymer/polymer.js> (line: 13)

`Calling Element.createShadowRoot() for an element which already hosts a
shadow root is deprecated. See
https://www.chromestatus.com/features/4668884095336448 for more details.`

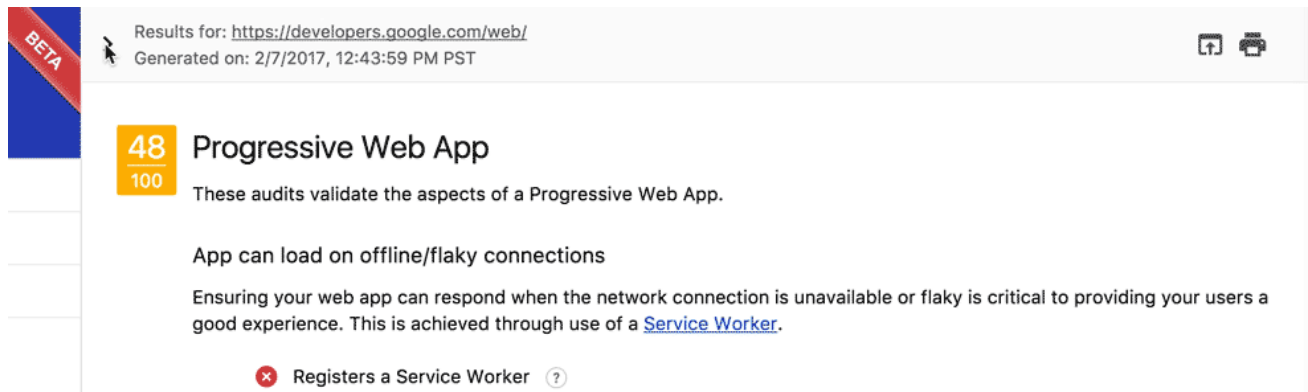
Report changes

As you've seen, we've focused on making the report less visually cluttered by adding tabular data, hiding extraneous help text, and adding new features to make it easier to navigate the data.

Emulation settings

It's easy to forget what throttling and emulation settings were used for a particular Lighthouse run. Lighthouse reports now include the **runtime emulation settings** that were

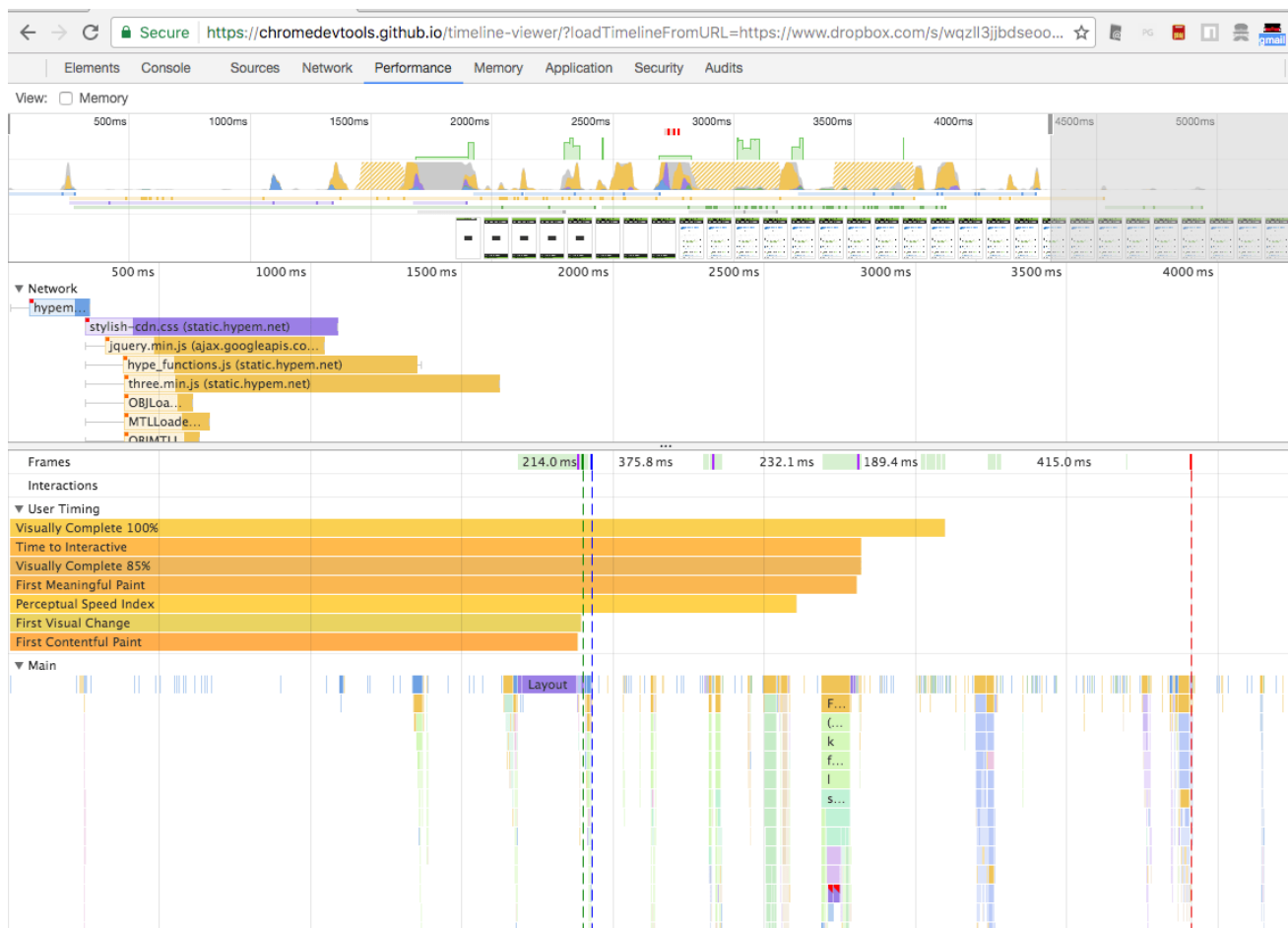
used to create the report; a long time feature request:



More useful trace data

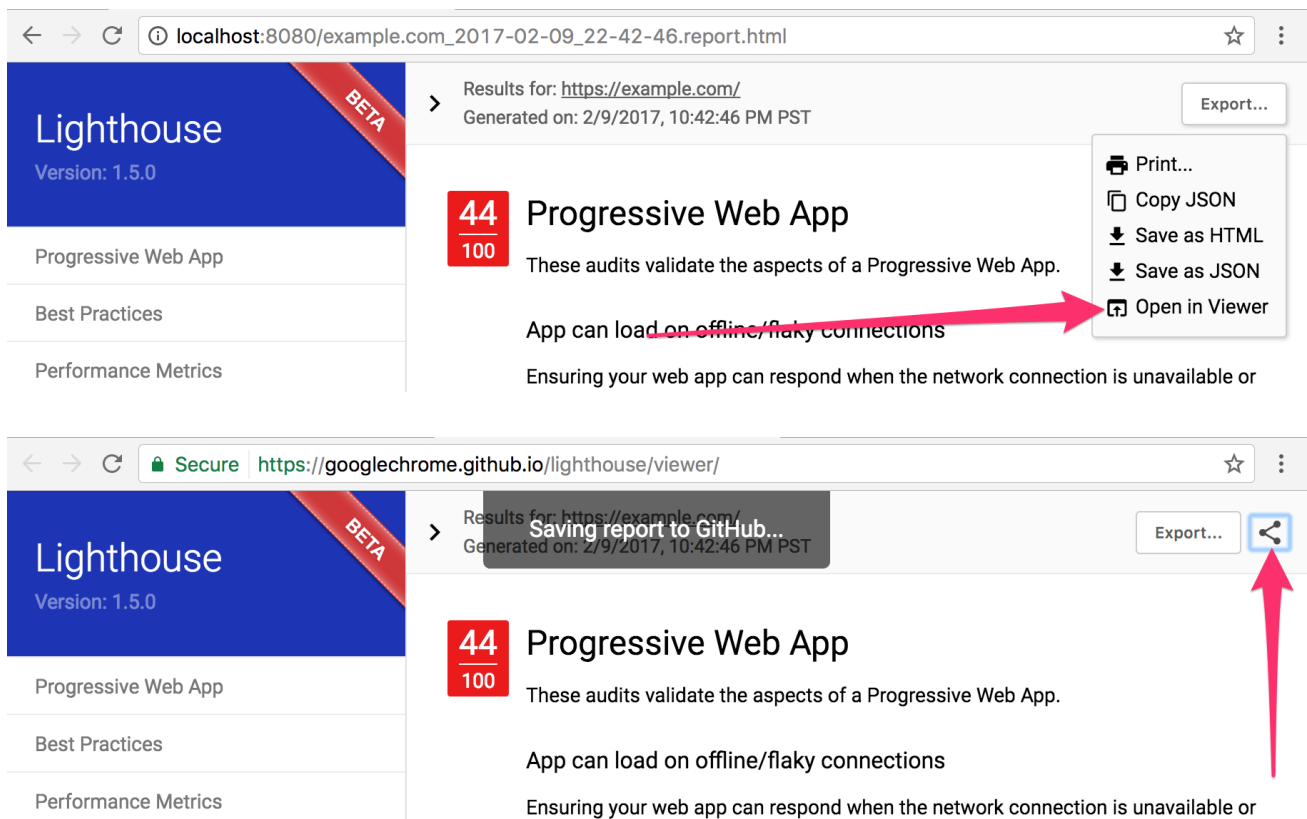
Lighthouse metrics like "First meaningful paint", "Time to Interactive", etc are mocked out as User Timing measures and injected back into the trace data saved with the `--save-assets` flag.

If you use the `--save-assets` flag, you can now drop the trace into DevTools or open it in the DevTools Timeline Viewer. You'll be able to see your key metrics in context with the full trace of the page load.



Lighthouse Viewer

In HTML reports, you'll notice a new button with options for exporting the report in different formats. One of those options is "Open in Viewer". Clicking this button will send the report to the online [Viewer](#), where you can further share the report with GitHub users.



Behind the scenes, Viewer gets your permission via OAuth to create a GitHub secret gist and saves the report there. Since it's done as your Gist, you maintain full control over the sharing of the report and you can delete it at any time. You can revoke the Viewer's permission to create gists under your [GitHub settings](#).

Performance Experiment

The first version of the [Performance Experiment](#) project has landed in 1.5.0. This lets you experiment with your page load performance, interactively testing the effects of blocking or delaying assets in your critical path **during development**.

When Lighthouse is run with the `--interactive` flag, a special report is generated that allows interactive selection of costly page resources. The experiment server then reruns Lighthouse on that page with those resources blocked.

Page load performance is fast

Users notice if sites and apps don't perform well. These top-level metrics capture the most important perceived performance concerns.

62 First meaningful paint: **3436.8ms** (target: 1,600ms) ?

73 Perceptual Speed Index: **3550** (target: 1,250) ?

First Visual Change: **2279ms**

Last Visual Change: **5937ms**

100 Estimated Input Latency: **16ms** (target: 50ms) ?

49 Time To Interactive (alpha): **5080.1ms** (target: 5,000ms) ?

List View Tree View


- ☐ js/jquery-ui-bundle.js (developers.google.com) - **2952.32ms, 78.08KB**
- ☐ v21/2fcrYFNATjcS6g4U3t-Y5UEw0IE80llgEseQY3FEmqw.woff2 (fonts.gstatic.com) - **3408.95ms, 60.0KB**
- ☐ js/script_foot.js (developers.google.com) - **3423.51ms, 100.91KB**
- ☐ js/script_foot_closure.js (developers.google.com)
- ☒ embed/P95ZDIzjg0Q (www.youtube.com)
 - ☒ www-embed-player-vflgCTvK8/www-embed-player.js (www.youtube.com) - **5798.76ms, 190.1KB**
 - ☒ cssbin/www-embed-player-sprite-mode-vflozhumh.css (www.youtube.com) - **5985.33ms, 20.0KB**
 - ☒ player-en_US-vflkRUE82/base.js (www.youtube.com) - **7671.57ms, 549.52KB**
- ☐ common/push-notification-16x9.png (developers.google.com) - **5920.39ms, 362.97KB**
- ☐ images/favicon.png (developers.google.com) - **8194.18ms, 0.99KB**

^ Added URL Blocking Pattern: <https://www.youtube.com/embed/P95ZDIzjg0Q?autoplay=1&showinfo=1> Rerun


[Learn more](#) about the Performance Experiment in Lighthouse.

New Documentation

Last but not least, we've modernized the documentation at developers.google.com/web/tools/lighthouse/ and added new audit references.

 Web

FundamentalsToolsUpdatesCase Studies

 Search

All ProductsSIGN IN

Tools

Getting Started

Chrome DevTools

Lighthouse

Overview

Audit References

Service Worker Libraries


Web Starter Kit


Polymer Starter Kit

Lighthouse

Lighthouse is an [open-source](#), automated tool for improving the quality of your web apps.

Lighthouse can be run as a Chrome Extension, from the command line, or used programmatically as a Node module. You give Lighthouse a URL that you want to audit, it runs a barrage of tests against the page, and then it generates a report on how well the page did. From here you can use the failing tests as indicators on what you can do to improve your app.

 INSTALL LIGHTHOUSE

 FILE AN ISSUE

★ **Note:** Lighthouse currently has a large focus on Progressive Web App features, such as [Add to Homescreen](#) and offline support. However, the overarching goal of the project is to provide an end-to-end audit of all aspects of web app quality.

Contents

Setting up Lighthouse

Install the Chrome Extension

Install the Node Command line tool

Get started

Chrome Extension

Online Report Viewer

Contributing

Setting up Lighthouse

There are two primary ways to run Lighthouse, as a Chrome Extension, or as a command line

That's it for now!

For all the details on the latest in Lighthouse, see the [full release notes](#) over on Github. As always, [hit us up](#) to [report bugs](#), file feature requests, or brainstorm [ideas](#) on what you'd like to see next.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated July 2, 2018.