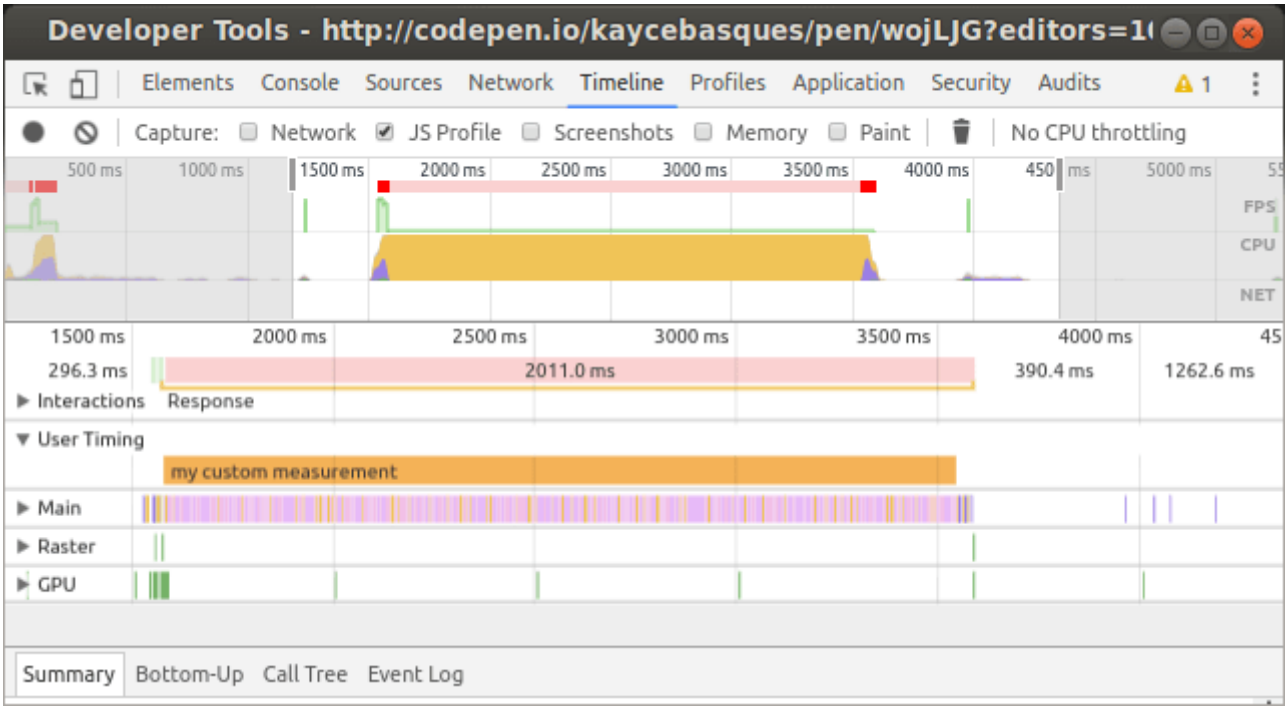
Avoids console.time() In Its Own Scripts

Overview

If you're using `console.time()` to measure your page's performance, consider using the User Timing API instead. Benefits include:

- High-resolution timestamps.
- Exportable timing data.
- Integration with the Chrome DevTools Timeline. When the User Timing function `performance.measure()` is called during a Timeline recording, DevTools automatically adds the measurement to the Timeline's results, as shown in the `my custom measurement` label in the screenshot below.



Recommendations

In your report, Lighthouse lists every instance of `console.time()` that it finds under **URLs**. Replace each of these calls with `performance.mark()`. If you want to measure the time that has elapsed between two marks, use `performance.measure()`.

See User Timing API: Understanding Your Web App to learn how to use the API.

More information

Lighthouse reports every instance of `console.time()` that it finds from scripts that are on the same host as the page. Scripts from other hosts are excluded, because Lighthouse assumes that you don't have control over these scripts. So, there may be other scripts using `console.time()` on your page, but these won't show up in your Lighthouse report.

Feedback

Was this page helpful?

YES NO

Great! Thank you for the feedback.

Sorry to hear that. Please open an issue and tell us how we can improve.

Last updated July 24, 2018.