

Performance Report for:

<https://edwardtbabinski.us/blogger/social/css-d...>

Report generated: Tue, Apr 11, 2017, 2:13 PM -0400

Test Server Region: Dallas, USA

Using: Chrome (Desktop) 54.0.2840.71, PageSpeed 1.15-gt1, YSlow 3.1.8

PageSpeed Score A (100%) ^	YSlow Score A (100%) ^	Fully Loaded Time 251ms ^	Total Page Size 15.8KB ^	Requests 2 ^
--------------------------------------	----------------------------------	-------------------------------------	------------------------------------	------------------------

How does this affect me?

Studies show that users leave a site if it hasn't loaded in 4 seconds; keep your users happy and engaged by providing a fast performing website.

As if you didn't need more incentive, **Google has announced that they are using page speed in their ranking algorithm.**

About GTmetrix

We can help you develop a faster, more efficient, and all-around improved website experience for your users. We use Google PageSpeed and Yahoo! YSlow to grade your site's performance and provide actionable recommendations to fix these issues.

About the Developer

GT.net
Performance Hosting

GTmetrix is developed by the good folks at **GT.net**, a Vancouver-based performance hosting company with over 21 years experience in web technology.

<https://gt.net/>

What do these grades mean?

This report is an analysis of your site with Google and Yahoo!'s metrics for how to best develop a site for optimized speed. The **grades you see represent** how well the scanned URL adheres to those rules.

Lower grades (C or lower) mean that the page can stand to be faster using better practices and optimizing your settings.

What's in this report?

This report covers basic to technical analyses on your page. It is categorized under many headings:

- **Executive:** Overall score information and Priority Issues
- **History:** Graphed history of past performance
- **Waterfall:** Graph of your site's loading timeline
- **Technical:** In-depth PageSpeed & YSlow information

These will provide you with a snapshot of your performance.

Waterfall Chart

The waterfall chart displays the loading behaviour of your site in your selected browser. It can be used to discover simple issues such as 404's or more complex issues such as external resources blocking page rendering.

Share This

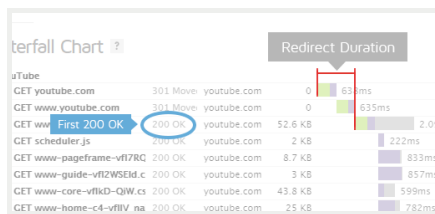


Page Load Timings

RUM Speed Index: 231

0ms Redirect	37ms Connect	160ms Backend	197ms TTFB
228ms DOM int.	228ms (0ms) DOM loaded	230ms (0ms) Onload	231ms First paint

Redirect duration



This is the time spent redirecting URLs before the final HTML page is loaded. Common redirects include:

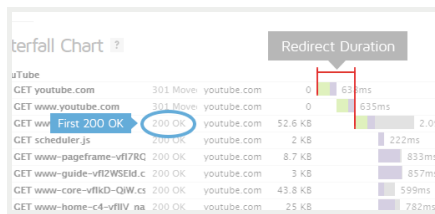
- Redirect from a non-www to www (eg. example.com to www.example.com)
- Redirect to a secure URL (eg. http:// to https://)
- Redirect to set cookies
- Redirect to a mobile version of the site

Some sites may even perform a chain of multiple redirects (eg. non-www to www, then to a secure URL). This timing is the total of all this time that's spent redirecting, or 0 if no redirects occurred.

In the Waterfall Chart, Redirect duration consists of the time from the beginning of the test until just before we start the request of the final HTML page (when we receive the first 200 OK response).

During this time, the browser screen is blank! Ensure that this duration is kept to short by [minimizing your redirects](#).

Connection duration



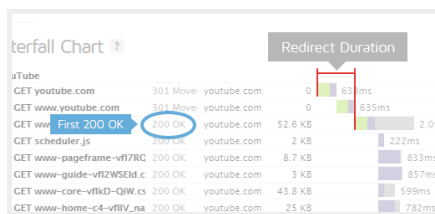
Once any redirects have completed, Connection duration is measured. This is the time spent connecting to the server to make the request to the page.

Technically speaking, this duration is a combination of the blocked time, DNS time, connect time and sending time of the request (rather than *just* connect time). We've combined those components into a single Connection duration to simplify things (as most of these times are usually small).

In the Waterfall Chart, Connection duration consists of everything up to and including the "Sending" time in the final HTML page request (the first 200 OK response).

During this time, the browser screen is still blank! Various causes could contribute to this, including a slow/problematic connection between GTmetrix and the server or slow response times from the server.

Backend duration



Once the connection is complete and the request is made, the server needs to generate a response for the page. The time it takes to generate the response is known as the Backend duration.

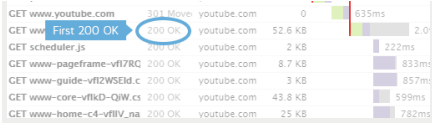
In the Waterfall Chart, Backend duration consists of purple waiting time in the page request.

There are a number of reasons why Backend duration could be slow. We cover this in our ["Why is my page slow"](#) article.

Time to First Byte (TTFB)



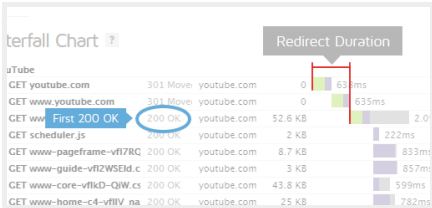
Time to First Byte (TTFB) is the total amount of time spent to receive the first byte of the response once it has been requested. It is the sum of "Redirect duration" + "Connection duration" + "Backend duration". This metric is one of the key indicators of web performance.



In the Waterfall Chart, it is calculated at the start of the test until just before receiving on the page request and represented by the orange line.

Some ways to improve the TTFB include: optimizing application code, implementing caching, fine-tuning your web server configuration, or upgrading server hardware.

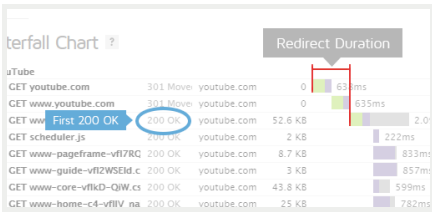
DOM interactive time



DOM interactive time is the point at which the browser has finished loading and parsing HTML, and the DOM (Document Object Model) has been built. The DOM is how the browser internally structures the HTML so that it can render it.

DOM interactive time isn't marked in the Waterfall Chart as it's usually very close in timing to DOM content loaded.

DOM content loaded time



DOM content loaded time (DOM loaded or DOM ready for short) is the point at which the DOM is ready (ie. DOM interactive) and there are no stylesheets blocking JavaScript execution.

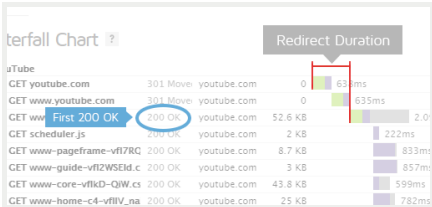
If there are no stylesheets blocking JavaScript execution and there is no parser blocking JavaScript, then this will be the same as DOM interactive time.

In the Waterfall Chart, it is represented by the blue line.

The time in brackets is the time spent executing JavaScript triggered by the DOM content loaded event. Many JavaScript frameworks use this event as a starting point to begin execution of their code.

Since this event is often used by JavaScript as the starting point and delays in this event mean delays in rendering, it's important to make sure that [style and script order is optimized](#) and that [parsing of JavaScript is deferred](#).

Onload time



Onload time occurs when the processing of the page is complete and all the resources on the page (images, CSS, etc.) have finished downloading. This is also the same time that DOM complete occurs and the JavaScript window.onload event fires.

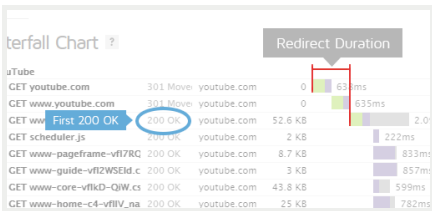
Note that there may be JavaScript that initiates subsequent requests for more resources, hence the reason why Fully loaded timing is preferred.

In the Waterfall Chart, it is represented by the red line.

The time in brackets is the time spent executing JavaScript triggered by the Onload event.

Note that Onload time was the previous default for when to stop the test for GTmetrix.

First paint time



First paint time is the first point at which the browser does any sort of rendering on the page. Depending on the structure of the page, this first paint could just be displaying the background colour (including white), or it could be a majority of the page being rendered.

In the Waterfall Chart, it is represented by the green line.

This timing is of significance because until this point, the browser will have only shown a blank page and this change gives the user an indication that the page is loading. However, we don't know how much of the page was rendered with this paint, so having an early first paint doesn't necessarily

indicate a fast loading page.

PageSpeed Recommendations

RECOMMENDATION	GRADE	RELATIVE	TYPE	PRIORITY
Avoid bad requests	A (100)	⬇️ AVG SCORE: 98%	CONTENT	HIGH
You scored 100% on this recommendation - nothing to do here!				
Avoid landing page redirects	A (100)	⬇️ AVG SCORE: 98%	SERVER	HIGH
You scored 100% on this recommendation - nothing to do here!				
Defer parsing of JavaScript	A (100)	⬆️ AVG SCORE: 70%	JS	HIGH
You scored 100% on this recommendation - nothing to do here!				
Enable gzip compression	A (100)	⬆️ AVG SCORE: 81%	SERVER	HIGH
You scored 100% on this recommendation - nothing to do here!				
Enable Keep-Alive	A (100)	⬆️ AVG SCORE: 93%	SERVER	HIGH
You scored 100% on this recommendation - nothing to do here!				
Inline small CSS	A (100)	⬇️ AVG SCORE: 96%	CSS	HIGH
You scored 100% on this recommendation - nothing to do here!				
Inline small JavaScript	A (100)	⬇️ AVG SCORE: 95%	JS	HIGH
You scored 100% on this recommendation - nothing to do here!				
Leverage browser caching	A (100)	⬆️ AVG SCORE: 58%	SERVER	HIGH
You scored 100% on this recommendation - nothing to do here!				
Minify CSS	A (100)	⬆️ AVG SCORE: 94%	CSS	HIGH
You scored 100% on this recommendation - nothing to do here!				
Minify JavaScript	A (100)	⬆️ AVG SCORE: 87%	JS	HIGH
You scored 100% on this recommendation - nothing to do here!				

Minimize redirects

A (100)

▲ AVG SCORE: 85%

CONTENT

HIGH

You scored 100% on this recommendation - nothing to do here!

Minimize request size

A (100)

◆ AVG SCORE: 98%

CONTENT

HIGH

You scored 100% on this recommendation - nothing to do here!

Optimize images

A (100)

▲ AVG SCORE: 69%

IMAGES

HIGH

You scored 100% on this recommendation - nothing to do here!

Optimize the order of styles and scripts

A (100)

◆ AVG SCORE: 96%

CSS/JS

HIGH

You scored 100% on this recommendation - nothing to do here!

Put CSS in the document head

A (100)

◆ AVG SCORE: 100%

CSS

HIGH

You scored 100% on this recommendation - nothing to do here!

Serve resources from a consistent URL

A (100)

▲ AVG SCORE: 88%

CONTENT

HIGH

You scored 100% on this recommendation - nothing to do here!

Serve scaled images

A (100)

▲ AVG SCORE: 71%

IMAGES

HIGH

You scored 100% on this recommendation - nothing to do here!

Specify a cache validator

A (100)

▲ AVG SCORE: 93%

SERVER

HIGH

You scored 100% on this recommendation - nothing to do here!

Combine images using CSS sprites

A (100)

▲ AVG SCORE: 87%

IMAGES

HIGH

You scored 100% on this recommendation - nothing to do here!

Avoid CSS @import

A (100)

◆ AVG SCORE: 98%

CSS

MEDIUM

You scored 100% on this recommendation - nothing to do here!

Prefer asynchronous resources

A (100)

◆ AVG SCORE: 99%

JS

MEDIUM

You scored 100% on this recommendation - nothing to do here!

Specify a character set early

A (100)

AVG SCORE: 100%

CONTENT

MEDIUM

You scored 100% on this recommendation - nothing to do here!

Specify image dimensions

A (100)

AVG SCORE: 98%

IMAGES

MEDIUM

You scored 100% on this recommendation - nothing to do here!

Avoid a character set in the meta tag

A (100)

AVG SCORE: 100%

CONTENT

LOW

You scored 100% on this recommendation - nothing to do here!

Minify HTML

A (100)

AVG SCORE: 98%

CONTENT

LOW

You scored 100% on this recommendation - nothing to do here!

Remove query strings from static resources

A (100)

AVG SCORE: 90%

CONTENT

LOW

You scored 100% on this recommendation - nothing to do here!

Specify a Vary: Accept-Encoding header

A (100)

AVG SCORE: 96%

SERVER

LOW

You scored 100% on this recommendation - nothing to do here!

YSlow Recommendations

RECOMMENDATION	GRADE	RELATIVE	TYPE	PRIORITY
Add Expires headers	A (100)	▲ AVG SCORE: 25%	SERVER	HIGH
You scored 100% on this recommendation - nothing to do here!				
Make fewer HTTP requests	A (100)	▲ AVG SCORE: 34%	CONTENT	HIGH
You scored 100% on this recommendation - nothing to do here!				
Compress components with gzip	A (100)	▲ AVG SCORE: 81%	SERVER	HIGH
You scored 100% on this recommendation - nothing to do here!				
Use a Content Delivery Network (CDN)	A (100)	▲ AVG SCORE: 14%	SERVER	MEDIUM
Using these CDN hostnames from your User Settings :				
<ul style="list-style-type: none"> • cleo.ns.cloudflare.com • tegan.ns.cloudflare.com 				
Minify JavaScript and CSS	A (100)	▲ AVG SCORE: 73%	CSS/JS	MEDIUM
You scored 100% on this recommendation - nothing to do here!				
Avoid URL redirects	A (100)	▲ AVG SCORE: 84%	CONTENT	MEDIUM
You scored 100% on this recommendation - nothing to do here!				
Make AJAX cacheable	A (100)	◆ AVG SCORE: 100%	JS	MEDIUM
You scored 100% on this recommendation - nothing to do here!				
Remove duplicate JavaScript and CSS	A (100)	◆ AVG SCORE: 100%	CSS/JS	MEDIUM
You scored 100% on this recommendation - nothing to do here!				
Avoid AlphasLoader filter	A (100)	◆ AVG SCORE: 98%	CSS	MEDIUM
You scored 100% on this recommendation - nothing to do here!				
Avoid HTTP 404 (Not Found) error	A (100)	◆ AVG SCORE: 99%	CONTENT	MEDIUM

You scored 100% on this recommendation - nothing to do here!

Reduce the number of DOM elements	A (100)	▲ AVG SCORE: 93%	CONTENT	LOW
--	---------	------------------	---------	-----

You scored 100% on this recommendation - nothing to do here!

Use cookie-free domains	A (100)	▲ AVG SCORE: 46%	COOKIE	LOW
--------------------------------	---------	------------------	--------	-----

You scored 100% on this recommendation - nothing to do here!

Use GET for AJAX requests	A (100)	◆ AVG SCORE: 100%	JS	LOW
----------------------------------	---------	-------------------	----	-----

You scored 100% on this recommendation - nothing to do here!

Avoid CSS expressions	A (100)	◆ AVG SCORE: 99%	CSS	LOW
------------------------------	---------	------------------	-----	-----

You scored 100% on this recommendation - nothing to do here!

Reduce DNS lookups	A (100)	▲ AVG SCORE: 68%	CONTENT	LOW
---------------------------	---------	------------------	---------	-----

- edwardtbabinski.us: 2 components, 25.6K (15.9K GZip)

Reduce cookie size	A (100)	◆ AVG SCORE: 100%	COOKIE	LOW
---------------------------	---------	-------------------	--------	-----

You scored 100% on this recommendation - nothing to do here!

Make favicon small and cacheable	A (100)	◆ AVG SCORE: 100%	IMAGES	LOW
---	---------	-------------------	--------	-----

You scored 100% on this recommendation - nothing to do here!

Configure entity tags (ETags)	A (100)	▲ AVG SCORE: 84%	SERVER	LOW
--------------------------------------	---------	------------------	--------	-----

You scored 100% on this recommendation - nothing to do here!

[Feedback Request](#)

Make JavaScript and CSS external	(n/a)		CSS/JS	MEDIUM
---	-------	--	--------	--------

Only consider this if your property is a common user home page.

- There is a total of 1 inline script