

# Web Page Load Time: A Quality of Experience perspective.

Aruna Balasubramanian, Jihoon Ryoo, Samir Das, Stony Brook University

## Introduction

Browsers and Web pages are increasingly becoming one of the main portals for e-commerce, news, educational activities, and other sources of information on the Internet. For example, studies show that over 40% of the cellular load carries Web page traffic [1]. There are also numerous studies that show the importance of Web page loads for companies: For example, Amazon increased revenue by 1% for every 0.1 second reduction in PLT, and Shopzilla experienced a 12% increase in revenue by reducing PLT from 6 seconds to 1.2 seconds [2].

Not surprisingly, researchers are designing new techniques to reduce page load time. But one critical problem is in the definition of page load time. When can we say that a page is loaded? Is it the time until which all objects are loaded? This is not the best definition because the user may not be interested in loading certain objects such as an advertisement. Is it the time taken for the content to stop changing? This is hard to measure because for certain Javascript enabled pages, some content such as banners constantly change.

It is crucial to accurately measure the page load time for optimization. How can we optimize the page load time metric, if we cannot accurately measure the page load time?

## Traditional Page Load Time Metrics

Typically OnLoad, DomComplete, and DomContentLoaded are the three most common metrics that browsers use to track page load time. The DOMContentLoaded is triggered when the HTML document has been completely loaded and parsed, but before stylesheets, images, and subframes have finished loading. The DomComplete is triggered when all the objects in the page are loaded, but not necessarily rendered. Finally, the OnLoad event, the most commonly used metric, is triggered when the browser has rendered all objects in the page.

In the past few years, there is general agreement that these metrics are insufficient to capture page load time [3]. For example, Figure 1 shows a video capture of a Webpage loading and illustrates the problem with current metrics. The video was captured using tests on Webpagetest.org.

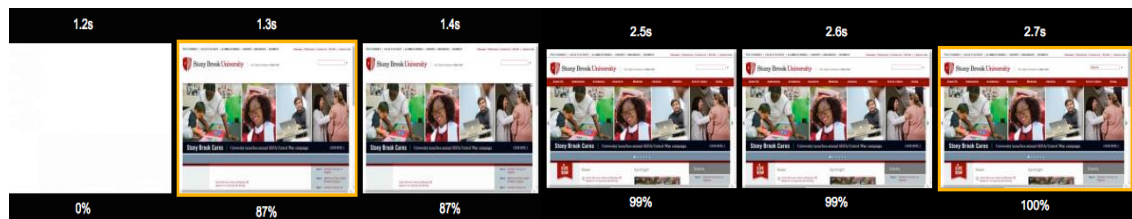


Figure 1: Video capture of the Webpage

As can be seen, most of the page content is loaded at 1.3 seconds. A user may perceive the page to be loaded at 1.3 seconds. But the OnLoad event is fired only 2.7 seconds, when all content on the page is loaded and rendered.

There are several reasons why the user-perceived page load time can differ from the OnLoad metric. For example, a user may perceive a page to be loaded even if some objects such as ads are not loaded. In general, the user perceives a page to be loaded as soon as all the data that the user is *interested in* has been loaded. Unsurprisingly, it is not easy to capture this metric because it is subjective.

## **Page Load time Metrics to measure user-perceived page load time**

Over the years, two main metrics have emerged to compute the user-perceived page load time: Above the Fold metric and Speed Index. Unfortunately, neither of these metrics can completely capture the user-perceived page load time.

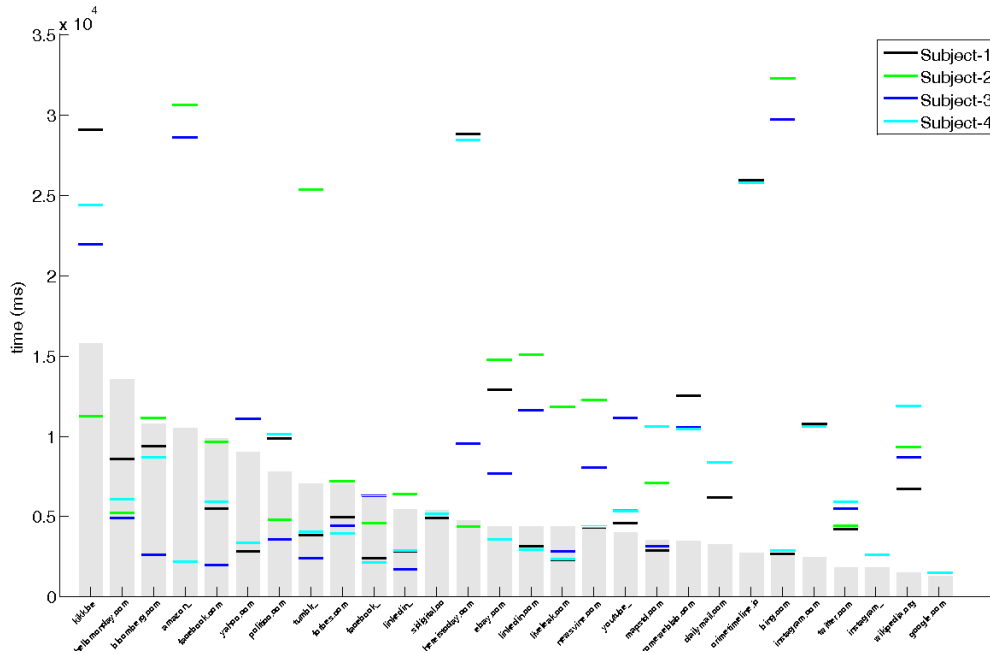
The Above the Fold Time (AFT) [4] is the time at which the content on the web page stops changing. The change is estimated based on changes in the rendered pixels. Since some pixels such as video and dynamic content change constantly, the AFT metric only takes into account the pixel changes in static content. This is a problem because many Web pages have a large percentage of dynamic content. It is also not easy to determine when the content is static versus dynamic. As a result, the Above the Fold metric does not accurately measure the user-perceived page load time.

Speed Index [5] measures the page load time after the fact, different from the Above the fold metric. It works by measuring the difference in pixel changes from when the page is completely rendered to when it is partially rendered. The Speed Index is the time when the page is not completely rendered, but most of the pixels are unchanged from the completely rendered page. This definition is problematic, particularly around new elements triggering a layout that shifts other elements around. For example, if a photo was positioned even a single pixel off its final position, the Speed Index metric will not measure the page load time accurately.

## **Our Preliminary Work**

The objective of our work is to reduce the user-perceived page load time. As a first step, we are conducting a user study to capture the page load time perceived by the user. To this end, we built a user study tool to record important timing information when a webpage is loading, while simultaneously recording a video of the page load process. We present the video to the users and record when the user perceives the page to be loaded.

Figure 2 shows the user-perceived page load time for 4 users, compared to the browser's page load time measurement (the height of each bar represents the browser's page load measure). The user study was done over 27 Web pages of varying types: news pages, Javascript heavy pages, portal pages, etc.



In many cases, the user-perceived page load time is less than the browser's page load time measure. But in other cases, it is more than twice of the browser's page load time measure.

We are continuing the user study to collect data from a large number of users. In parallel, we are also collecting the user's gaze point when they are looking at the Web page. Our next step is to use the user's gaze to predict and improve user-perceived page load time.

## References

[1] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang. Characterizing Geospatial Dynamics of Application Usage in a 3G Cellular Data Network . In INFOCOM, 2012.

[2] Shopzilla: faster page load time = 12% revenue increase. <http://www.strangeloopnetworks.com/resources/infographics/web-performance-and-ecommerce/shopzilla-faster-pages-12-revenue-increase/>

[3] Moving Beyond Windows.OnLoad(). Steve Souders <http://www.stevesouders.com/blog/2013/05/13/moving-beyond-window-onload/>

[4] Michael Butkiewicz,, Daimeng Wan, Zhe Wu, Harsha V. Madhyastha and Vyas Sekar KLOTSKI: Reprioritizing Web Content to Improve User Experience on Mobile Devices, NSDI 2015

[5] Speed Index. <https://sites.google.com/a/webpagetest.org/docs/using-webpagetest/metrics/speed-index>