PLTSpeed: Suggesting Personalized Web Optimizations for Developers

Javad Nejati, Aruna Balasubramanian
jnejati@cs.stonybrook.edu, arunab@cs.stonybrook.edu

1. Introduction

Page load performance on mobile devices is known to be orders of magnitude poorer compared to desktops. However, improving mobile Web performance is not a trivial task: several Web optimizations have been designed, but their effect on improving mobile page load times have been mixed.

Among others, Google PageSpeed Insights [2] and Yahoo's YSlow [4] are two notable current web optimization tools. These two tools suggest a set of optimizations for any page, based on analyzing the page. However, the two tools use an indirect scoring mechanism to score the effect of an optimization on a page. As a result, it is not clear how much page load benefit can be expected from each optimization. Further, both techniques do not take into account the effect of an optimization on the critical path.

We looked at 4 websites for which Google PageSpeed Insight suggested *inlining* as an optimization. Inlining is a technique where the scripts, usually Javascript and CSS, are included as part of the original html file, rather than as an external link. We observed that out of four websites, inlining helped only two of the Websites as shown in Figure 1. This mismatch is because, existing tools do not take into account the effect of the optimizations on the *critical* path.

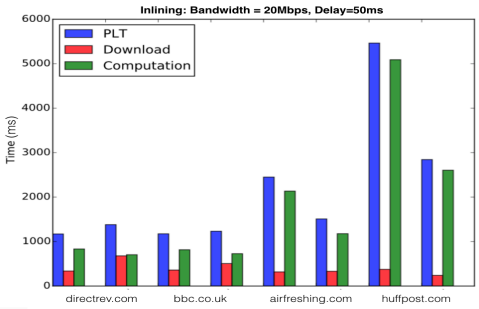

Figure 1: Inlining scripts does not help always.

2. PLTSpeed

In this poster, we introduce PLTSpeed, a platform that estimates the effect of each optimization on the page load time, and suggest the optimizations to a Web developer.

The key intuition in PLTSpeed is as follows: the optimization affects different objects during the page load process differently. For instance, compression algorithms may improve the download time for a large object, but not provide benefits for smaller objects. The objects themselves do not contribute to the page load time equally. Therefore, rather than estimate the effect of an optimization on the entire page, PLTSpeed breaks down the page load process into various objects. PLTSpeed applies the optimization to each object, and then reconstructs the page load with the *optimized* objects.

We apply the PLTSpeed technique to *minification*, an optimization that removes white spaces and comments from the files. Figure 2 shows the dependency graph for *www.cbs.com* for both the original Web page (gray bars) and our *estimated* minified version. The page load time result of the minification as estimated by PLTSpeed is consistent with the page load time when the page was actually minified. We performed the experiments on our controlled testbed. The testbed uses WProf [1] to break down the page load process into various objects. The testbed also uses a traffic controller to control network variations. The testbed is described in detail in our upcoming paper to be published at the WWW conference [3].

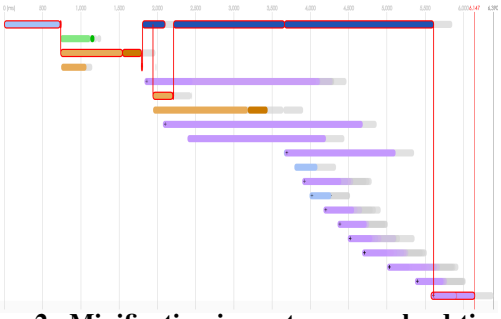

Figure 2: Minification impact on page load time for *www.cbs.com*

3. Future work

As the next step, we will leverage PLTSpeed to model the effect of various optimizations, as well as combinations of optimizations. Our goal is to build a visualization tool that Web developers can use to get suggestions and to visualize the effect of different optimizations.

4. References

[1] X. W. et al. Demystify page load performance with wprof. In *Proc. of the USENIX conference on Networked Systems Design and Implementation (NSDI)*, 2013.

[2] Google. Google Pagespeed Insights. `https://developers.google.com/speed/pagespeed/insights`.

[3] J. Nejati and A. Balasubramanian. An in-depth study of mobile browser performance. *Proceedings of the 25th International Conference on World Wide Web*, 2016. To appear.

[4] Yahoo. YSlow. `http://www.yslow.org`.