Web Page Optimization

Utkarsh Rastogi

10 July 2014

Contents

1 Abstract

We propose web page optimization as necessary step in web application development. In context of virtual labs,several online experiments were designed by domain experts from various disciplines and thousands of web pages were developed.But domain expert would not know best practices for web development, therefore end user suffer from performance issues.Performance of an web application largely depends upon the content of the web page because rendering takes place on client side. And client machine may have limited resources at his side.Therefore web page optimization seems an necessary step to improve user experience.In contrast to existing work on this field,here we focus on large scale web performance visualization approach.Based on this large scale visualization,we present a utility of existing web optimization tool by experimentation on a MHRD web project "virtual labs".

2 Introduction

Earlier people while talking about web performance meant about optimizing the server side but nowadays optimization on the client side is also needed.Presently,front-end developers use a lot of javascipt ,CSS and images to make a good user interface but this all adds overhead during page rendering which means lesser user experience.Success of a website comes with a good user's experience which also includes fast response time.

Optimized web pages not only renders a web page faster but also saves network bandwidth.Along with making a good responsive web page,web developers should also focus on using a optimal number of critical resources and its size should also be minified.Critical rendering path is the chain of necessary events that occurs to make webpage appear on browser.The main critical resources on a web pages are CSS,javascript and images.For each critical resource on a wepage,browser makes a request to server. CSS and javascript are the two critical resources that blocks the rendering of the webpage.Therefore,correct order of http requests can reduce the perceived page load time which means page load time will not reduce but important components which user wants to see first is loaded first and rest resources are loaded in background.The main focus should be made to:1.minify size of crictical resources.2.minify number of critical resources.3.minify critical rendering path.

The goal of our experimentation , performed in collaboration with VLEAD lab, IIIT-Hyderabad :

- devising a technique for large scale analysis of webpages for virtual labs.

- Comparison between the web performance of virtual labs's web pages with a web optimization tool Google Pagespeed and without a web optimization tool so as to evaluate the utility of the Pagespeed which can be used for the virtuals labs.

3 Background

Yahoo yslow and Google pagespeed are well known tools that are capable of finding optimal solutions with regards to web page optimization but the latter is capable of solving it also.Both techniques follow their performance best practices to evaluate the web performance of a webpage.

Yahoo yslow.js is javascript apis which runs on phantomjs.PhantomJS is a headless browser with JavaScript API. .We used Yslow as performance measuring tool because it not only analizes a webpage but gives suggestions on how to improve it. It works on three process:

- It crawls the DOM to find each component.

- Collects information for each component and analizes each component.

- It generates scores out of 100 for each rule which produces the overall score for page.

The grades for individual rules are computed differently depending on the rule. For example, for Rule 1, three external scripts are allowed. For each script above that, four points are deducted from the grade. The code for grading each rule is found in rules.js. The overall grade is a weighted average of the individual grades for each rule, calculated in controller.js The rules are approximately in order of importance, most important first. The specific weights are in the ruleset objects in rules.js. Score computation have been discussed in the following table no. 1.

Rule	Configs	Computation
Make fewer HTTP requests	max js = 3 max images=6 max css=2	N(JS-3)*3 N(images-6)*3 N(CSS-2)*4
Use a CDN	patterns=CDN hostname	N RegExp mismatches*10
Avoid empty src or href		N empty tags*100
Add Expires headers	how far=172800s	N(expiring 2 days)*11
Compress components with GZip	min file size = 500 bytes	N(uncompressed file) * 11
Put CSS at top		1+N link tag on BODY*10
Put JavaScript at bottom		N JS on HEAD * 5
Reduce DNS lookups	max domains = 4	(N domains - 4) * 5
Minify JavaScript and CSS	types = js, css	N(unminified JS or CSS)*10
Avoid URL redirects		N redirects * 10
Remove duplicate js and CSS	types = js, css	N (duplicated JS or CSS)*5
Configure ETags	types=js,css,image,flash	N bad etag of any type*11
Reduce the number of DOM elements	range = 250,max dom = 900	
Avoid HTTP 404 (Not Found) error	types=js,css,image, favicon	N 404 * 5

Grade Computation From Score

Score	Grade
95-100	A+
90-94	A-
85-89	B+
80-84	B-
75-79	C+
70-74	C-
65-69	D+
60-64	D-
55-59	E+
50-54	E-
Below 50	F

Mod_pagespeed is the web page optimization tool developed by Google.It not only analizes a web page but also optimizes it .Based on best practices,it has certain set of filters which optimizes web page during run time.As the server gets request for the webpage, it dynamically rewrites the page using its filters and sends an highly optimized page.There are total of 40+ filters which supports optimization.These filters can be turned on or off based on requirements.

Nowadays,there are some important performace best practices that are suggested to developers to follow :

- Minimize http requests

For each critical resource in the page, browser have to make a request for it to server and then its get loaded.So,almost 80% of the response time is consumed in downloading all the resources.So to reduce number of http requests,one can combine multiple CSS into one,also multiple javascipt files can be combined into one.Other ways include image spriting ,etc.

- Use a Content Delievery Networks

Nearest server is selected for delievering the content which reduces load time.

- Add Cache Control Header

For static components,set far future expires header. For dynamic components ,use an appropriate Cache control header to help browser with conditional requests.This reduces unnecessary http requests.

- Gzip components

Compression reduces response time by reducing the size of http response. Gzip is most popular and effective compression method at this time.It reduces response time by 70%.If a web client indicates for support for compression in the http request header,server sends a compressed components.

- Stylesheets at the Top

Problem with putting style sheets not in the head tag is that it blocks progessive rendering and till the sytlesheets are not fetched users sees nothing on screen.

- Put Scripts at Bottom

Putting scripts at the top,blocks parallel downloading of resources per hostname.

- Make Inline Small CSS and javascript files

If file size is to small, then it should be made inline as it will reduce number of http requests.

- Make large css and javascript file external

CSS abd javscript files are catched by browser.So everytime it is not downloaded.First time it will take time to download but as its is cached ,it will reduce http request.

- Minify javascript and css

Unnecessary characters from code should be reduced which includes removing comments,duplicacy and removing whitespaces.

- Avoid Redirects

Connecting an old page to new one takes time and increases page load time.It should be avoided.

- Configure ETags

Entity tags is a way that browser and server use to determine whether the component in cache is same as that on server.

- Flush the buffer early

It allows to send partially ready response to browser.It should be written as early as possible in the code, preferably in the head section. In php there is function flush() to flush the buffer.

- No 404 error

Http requests made and getting a response like 404 Not found is totally useless and it increases response time.

- Make favicon small and cacheable

Favicon stays in the core of server and it is necessary as if its not there , browser will still request for it and getting 404 error will increase response time.

4 Approach

Our work is broadly divided into four major phases namely Data Collection, Data Visualization, Analysis of Data and optimizing web pages based on analysis. During data collection phase we first collected all the urls of virtual labs hosted at IIIT-Hyderabad. Then we collected yslow reports for each web page using an automated script and phantomJS. After collecting all the reports we extracted scores for each rule from reports and stored it in a csv file. During visualization phase all data is visualized using an automated script indicating performance for each rule and also overall performance of web pages. Later we did analysis for optimized pages of virtuals labs.

4.1 URls and Report Collection

To visualize the performance of all the webpages,first of all ,our need was to collect all the urls which are in deploy.virtuals-labs.ac.in hosted at IIIT-Hyderabed.Since we have access to server,to get all the urls,we wrote an automated bash script to extract all the html and php pages links from the server and stored it into the text file. To test the performance of vlab.co.in, we also collected the 5000 urls for this website.This time since we do not have the access of server ,we used an online sitemap generator to get the urls. We wrote an automated script to generate the reports for each web page using yslow.js and phantomJS .This automated script read a url from a file containing all the urls and generate report for each web page.In these script,we are making ten phantomjs call at a time.For urls which are inactive ,failed reports will be generated and we are pushing that url into the failed urls file and deleting the failed reports.These reports are input to CSV file generation.

4.2 CSV File Generation

CSV file is generated using a bash script.This CSV file will contain the overall score and scores corresponding to each rule.Script will extract all the scores for corresponding rule and will dump into the CSV file line by line.The content in the CSV file is the statistics which will be used for visualization.

4.3 Visualization

Visualization is done using python matplotlib.Somya has written an automated scripts to generate all the graphs at one shot.It will take csv file as input.

5 Experimentation

5.1 Statistics for www.vlab.co.in

This is website for virtual labs hosted outside IIIT-Hyderabad. We started with collecting statistics for vlabs.To know how much they are optimized,we collected 5000 internal links in www.vlab.co.in. and generated reports for all the links using yslow and generated CSV file for the scores.After generating graphs,we analyzed tha current status of web pages of this vlab.co.in

5.2 Statistics for deploy.virtual-labs.ac.in

This is the virtual lab website hosted at IIIT-Hyderabad.To,know how these webpages are performing,we collected all the urls inside this deploy by using our bash script.Total of 8786 urls were there in the list.Then,reports were generated for each web page and csv file was generated.After generating graphs,we analyzed the current status of web pages.

5.3 Optimization of replica of deploy.virtual-labs.ac.in using mod_pagespeed

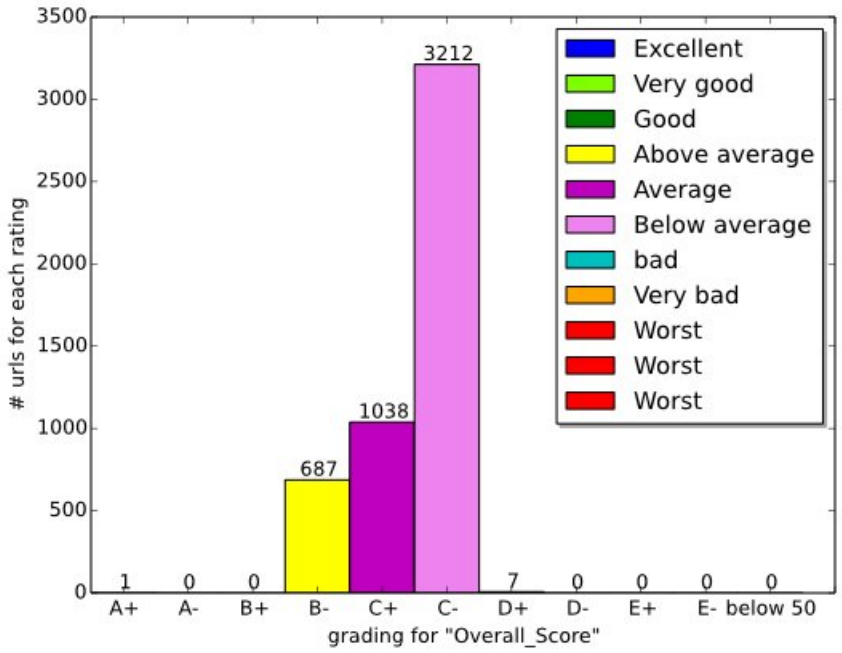
We made a replica of deploy server using a container.Its IP address is 10.4.14.31.We installed Google mod_pagespeed on it to evaluate how much it optimizes the website.Then ,we collected reports for all the web pages and generated CSV file for it.After,generating graphs we compared it with deploy server graphs to observe how much optimization was done by pagespeed.

6 Analysis

6.1 Analysis of www.vlab.co.in

We collected statistics for 4945 urls.

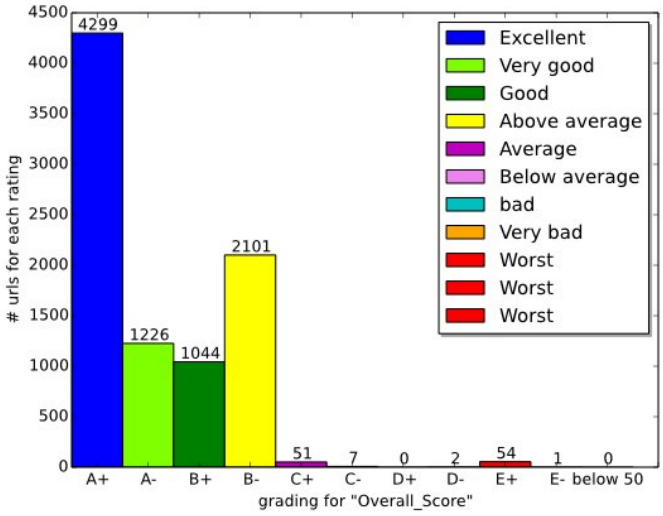
- From the graph below,we can observe that only 1 web page is having good performance.Rest web pages are not in good condition.They really need to be optimized.Out of rest,687 i.e 13% webpages are in above average conditions,1038 webpages are performing average and rest 3220 webpages are not in good condition.These all statistics are showing that these pages are not following performance best practices.



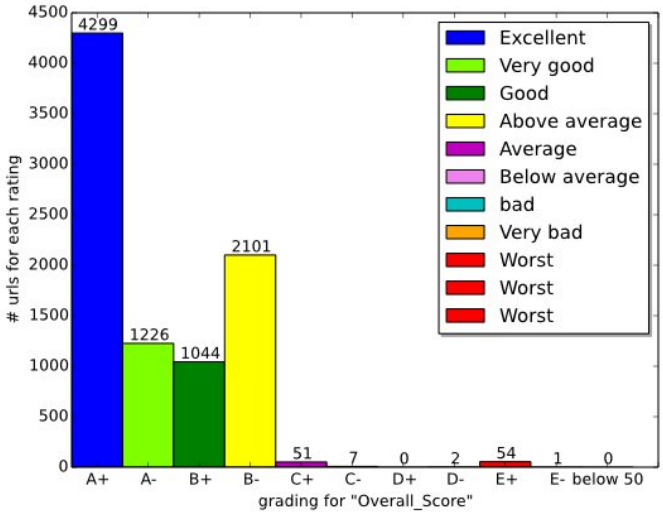
6.2 Optmization using Pagespeed.

As we have generated statistics for 8986 webpages under deploy.vitual-labs.ac.in and for replica of deploy server on which pagespeed has been installed.Here we observed how much pagespeed optimizes the webpages.

- From the graphs,we can see that for deploy.virtual-labs.ac.in 4299 are in A+ grades for **overall score** while with pagespeed this rises to 6051. It shows pagespeed is improving the performance of web pages by optimizing it.Also we can see see without pagespeed only 1226 wepages are in A- grade but with pagespeed it increased to 1667.From the graphs we can analyze that overall number of webpages is going to low to high grades.



[No Pagespeed]



[Pagespeed]

7 Conclusion

This report concerns Web Page Optimization which means fast web page rendering and using less network bandwith.This lead us to think us how

to decrease the number and size of resources and also how to decrease the perceived page load time.This wepages consists of critical resources like CSS,JavaScript and images.This optimization can be achieved by minimizing the number of critical resources,minimizing the size of critical resources and minimizing the critical path length.There are several best practices which are suggested to use in our webpages.

Our framework is mainly to visualize website performance on a large scale and it can be used to suggest lab developers to work on these best practices.Also it gives the list of inactive urls.Google pagespeed is a good web optimization tool presently available which optimize a web page on its own.It has 40+ filters which optimizesweb page and can be used according to our requirement.It is an open source and available for free.It should be used for virtual labs which will have enormous users in future.

8 Future Work

These framework can modified to give the list of web pages which are performing very badly.Generating report takes atleast 24 hrs to process 5000 urls but buy making phantomjs clusters this timing can be reduced.Also, in pagespeed many more filters can be added.For example ,there is no filter to give default favicon for a webpage if it is not there.

9 References

References

[1] Andrew B.King,2008,Website Optimization:Web performance Optimization.155-185,282-290

[2] Steve Souders,2007.High Performance Websites.10-84

[3] Steve Souders,2009.Even Faster Web Sites

[4] Yslow Official Website.Available:http://yslow.org/phantomjs

[5] Yslow Documentation Page.Available:http://yslow.org/faq

[6] Pagespeed Filters.Availbale:https://developers.google.com/speed/pagespeed/module/filte