

EASY WAYS TO BOOST YOUR SALES



WEB PERFORMANCE RECIPES

By Fabio Cicerchia

**WHILE MAKING YOUR USERS AND
THE SEARCH ENGINES HAPPIER**

Web Performance Recipes

Easy ways to boost your sales while making your users and the search engines happier

Fabio Cicerchia

This book is for sale at <http://leanpub.com/webperformances>

This version was published on 2020-02-07



Leanpub

This is a [Leanpub](#) book. Leanpub empowers authors and publishers with the Lean Publishing process. [Lean Publishing](#) is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

© 2019 - 2020 Fabio Cicerchia

Contents

Intro	i
Book's Topic	i
Web Performances	ii
Is all about money	iii
Measuring	1
How?	2
Random Sampling	2
Central Tendency	3
Percentiles	4
Conclusions	5
References	6

Intro

Hi, I'm Fabio Cicerchia, I'm an Italian Developer, Full-Stack Web Developer. I've been doing some coding over the past 15 years. I'm mainly a PHP developer, but I enjoy the front-end side as the back-end side, especially the sys-admin part. So, in the last 4 years, I've been doing some performance optimisation of several websites, the websites were quite slow, we are talking about something like 20 seconds to load the whole page, and I decrease it to half a second. So I going to show you how I did it, so practically what I've done to achieve such a good loading time. So you can trust my experience of making a website faster, and I want to pass this to you. At the moment I'm the CTO of Skuola.net.

Book's Topic

Hi guys, welcome to my first book. I decided to talk about web performances, how to improve the speed of the website, how to make it faster, how to climb up the SERP of Google by improving the speed, which is improving as well the SEO score, the page rank. It's going to improve the satisfaction of the users, the users will be happier to use your website and less keen to abandon it. More revenues for you, the faster, the better.

The book is, based on the speech that I had, a few months ago. Which it had many attendees on the talk, I was expecting 20 people at the most, it was during the PHP User Group in Rome, and there were 50 people, more than expected. It was a tough one, it was the first one. It took me a month just to put up the slides. The slides are showing you some facts about the performance impact, on the business side, user side, search engine.

The book shows you as well, how to improve it, using some

techniques, how they're called, and what you can do with it.

I decided to split the lessons into several levels. Each level contains a bunch of suggestions based on the level itself. The beginner will have simple stuff, the expert level needs to implement all the previous level to achieve it, you cannot just stick to the level itself, you need to do all of them.

After that, there is a section to explain which tools you can use, and another one with a list of suggested books to expand your knowledge even more.

So basically this book is not a comprehensive one, yet. It'll be very soon because I'm going to update it over time, if something new will come up I will update it and add the lessons. If you need any help don't hesitate to contact me, I'll be happy to help you; send me a message, drop me an email, send me a tweet. If you have any suggestions, they are more than welcome.

I hope you'll enjoy the book, stay tuned. Thank you again.

Web Performances

First things first, let's define what web optimisation is.

Web optimisation aims at enhancing a website's performance to make a more satisfying user experience. It generally deals with improving the design of a website to increase the speed of loading and page functionalities. Efficient and optimised websites increase the duration of user interaction, increase the traffic volumes and improve the search engine scores.

So let's talk about performance. Performance is not a project, it's a process, kind of. So it cannot be considered a project because you cannot just build a team to do performance optimisation. Optimise after you build everything is going to take a significant impact on the delivery time, it's going to have a massive impact on the cost because some part of the optimisation process needs to be thought before the actual implementation. Some can happen after, some need to happen before.

So when someone says the premature optimisation is the root of all evil, he's right, but the over-optimisation is the real root of all evil. So, as said, performance is a process, not a project, in every layer of the development cycle, you need to think about it. You need to think about the UI, the UX, you need to think about performance when you start writing PHP, ASP, .NET code, JavaScript code.

You need to think about performance even when you write the CSS. And you need to think about performance when you configure the web server, database server, the actual virtual machine, or, physical machine.

It's quite a horizontal process. It's not a project, definitely not. So because it's a process, it involves every engineer in the building to do their part about performances, to think about performance. If everyone thinks how to make it better, a faster, lighter, the result is going to be faster.

Is all about money

So, Time is Money; I don't need to say much about it, everyone would agree on this. Wasting time is wasting money, it's not a good idea. You cannot buy time, so you need to make good use of it. Please don't waste it. Don't do the wrong thing. Just, try to make good use of it.

Measuring

How?

If you cannot measure you can't optimise it, so we need to measure first.

How can we do it? We can do it in several ways. We need to measure page performance accurately.

You need to get the right loading time.

We need to try to be unobtrusive because otherwise, we might create some disruptions to our customers.

So let's think about you went to launch a JMeter test, or an AB test the performance of the website we will dramatically decrease, so let's try to be unobtrusive. We also need to have a good sampling, we can't just benchmark everything; we need to sample what we need to test.

And also we need to focus on useful metrics, we cannot just optimise everything, let's focus on matters most.

Random Sampling

Sampling is about selecting units from a large population of interest, to make generalisations of the whole population. A small group of individuals or subjects taken from a larger population is called a sample.

Random sampling methods are directly used, so the sample is actually representing the population in terms of style, quality and nature. If this is not done, then the sample will be biased, and wrong conclusions will be drawn from the results.

When applying a random sampling, every individual in the population has the same chance of being selected. The whole method of sampling is carried out in a one-step procedure where every unit is selected uniquely from every other member of the population. How can we actually randomise? We can randomise it in several ways. Two easy ways, just you can pick a percentage, just stick to

that 10%, 15%. Otherwise, you can actually try to select the user at runtime.

You can check if a user can be eligible for a testing phase. So, this if he is in the bucket of the testable user, then they should be tested. This means that users will always be the same.

Central Tendency

We need to try to get the central tendency of our distribution of samplings.

We can use three values, the mean, the median and the mode. Usually, we can focus on the median.

Mean

The mean is generally known as average, that is the sum of all the numbers divided by the total number of occurrences, but it is different in statistical terms. There are different ways of averaging and the mean is just one of them. Every measure of the central tendencies is just a simple average. The basic formula is:

$$\text{Mean} = \text{Sum of data} / \text{Number of data}.$$

So we add up all the unordered data and divide it by their amount. The mean, unfortunately, is affected by outliers. So if you have very different values, then is not the right one.

Mode

The mode is the number with the highest frequency, it is usually seen as the most straightforward measure of central tendency to

calculate. In order to get our mode out of raw data, we have to get the frequencies of all the numbers. After the sorting is done, we can now declare our mode which to be the number with the highest frequency. In our example of numbers, the mode is 9 because it occurs most (three times). A set of numbers or data having too many similar frequencies can be said to have no mode, but two or three numbers with the same frequencies are generally accepted.

Median

Median is also easy to calculate, the element corresponding to the average or central position of the data. In determining the mode, start with arranging the numbers in numerical order (ascending or descending) so as make the middle position correspond to the right element. Then the total number of data has to be known. This is, again, easier when having frequency data because the sum of frequencies is actually the total number of data. The total number of occurrences gives us the total number of data. Another metric which is useful to focus on is the percentile.

Percentiles

Are perfect for automatic baselining. This is because the averages can be misleading. So, not all the user will load your website at the same time. Even if your website loads in half a second, not all the user will, unfortunately. There are many factors, and I'm gonna cover this later on. But I'm gonna tell you it cannot be the same time every time.

A percentile is a statistical measure that tells us the value below which a given percentage in a group of observations would fall. A simple example will be the 25th percentile which is the value below which 25% of the population is found in the observation.

So percentiles are actually useful in this case because they allow us

to understand more about the real distribution of the page loading times.

The percentiles to focus on are the 50th (i.e. the median), 95th, and 98th, these percentiles can be used for comparison against the median to see the average and worst scenarios.

Conclusions

So this is because you cannot improve what you cannot measure, how can you do it? So what's the reference to say, this is faster, this is slower?

If it's precisely the same amount of time, if you cannot measure it, it cannot be appropriately optimised.

References

- The statistics of performance measurement - Random Sampling¹
- Why Averages Suck and Percentiles are Great²

¹<http://tech.bluesmoon.info/2010/01/statistics-of-performance-measurement.html>

²<http://apmblog.dynatrace.com/2012/11/14/why-averages-suck-and-percentiles-are-great/>