Replication Study: Eye Tracking Study on camelCase and under_score Identifier Styles

Johannes Almroth

5th June 2019

Abstract

The purpose of this paper is to replicate the results of two earlier studies, Binkley *et al.* (2009) as well as Sharif *et al.* (2010), where the aim was to determine if the identifier naming conventions affected code comprehension. The two styles that are examined are camelCasing as well as the under_score style. The study is conducted by using an eye-tracking device to gather data through a timed responsed test.

Contents

Chapter 1

Introduction

Naming variable identifiers in a effective and efficient way is important for code comprehension. A programmer can only get so far with using variables like x and y, and to write code that encapsulates complex mental models of thinking that need to be understood by multiple people in a development team, appropriate identifier names need to be chosen. Good identifiers enables, for example, abstraction of concepts, collaboration and code preservation. This is put into perspective when reading a paper by Deissenboeck *et al.*[2] which states that 70% of the source code for a given release of the popular Java IDE program *Eclipse* consisted solely of identifiers. But assuming that the wording of the identifiers are sufficient, the question remains of *how* they can be written to further aid in code comprehension.

The two main styles of writing identifiers in code today are camelCasing (*e.g.,* coolBeans) and underscore casing (*e.g.,*cool_beans). [1]

Deciding which one to use is often a matter of the given convention within the programming language that the code is written in. As noted in a previous study on the subject [4], early programming languages such as Basic, Cobol, Fortran and Ada were case-insensitive, and thus encouraged the use either the underscore or hyphens to write their variables. When languages such as C and Java were introduced, camel casing became more common. The arguments for using camel casing is that it requires fewer keystrokes and improves typing speed,but natural language research suggests that this is the wrong approach. A paper by Epelboim *et al.*[3] found that un-spaced text lead to a 10-20% slower reading speed in subjects.

1.1 Purpose and Goals

This study will aim to provide additional data that would further aid in the judgement regarding which one of these two styles would be the most beneficiary for code comprehension. Furthermore, it will use the same data gathering equipment in the form of an eye-tracker, and hopefully achieve a more solid statistical significance through a higher subject count.

1.2 Delimitations

The study will be limited by primarily aiming to replicate the 2010 study. The same test data and variables will be gathered, as well as focusing on acquiring more

[1] Underscore casing is more commonly known as snake casing, though the previous expression will be used to be more consistent with the established literature

subjects. Additionally, a few extra variables about the subjects will be gathered pertaining to previous programming experiences as well as their identifier style of choice, as to expand the given research and to contribute data to the [...] (research area?)

Chapter 2

Background

So far there's been two studies aiming to determine if there is a significant difference between the two styles. The first study conducted by Binkley *et al.*in 2009 [1] with 135 subjects concluded that the camel casing style leads to a better all round performance, at least when the subject is trained on the style, despite taking on average 0.42 seconds longer to read. The second study conducted by Sharif *et al.*in 2010 [4] with the help of eye-tracking equipment found that camel cased words took on average 0.932 seconds longer to read, and concludes that the under score style leads to an improvement in both reading time and lessens the amount of visual effort required by the subjects. Even though the two studies differ in their conclusions, the eye-tracking study suffers from a small sample size, with a meagre 15 subjects compared to the 135 subjects in the 2009 study. However, the eye-tracking equipment used in the 2010 study lends a lot of credibility to the data presented.

Chapter 3

Methodology

3.1 Research questions

Explore design space, motivate method, consequences,

I will attempt to replicate the 2010 study using an eye tracker device from tobii. I will set up a test process as similar to the original as I can, as to gather a minimum of the same amount of data. My aim is also to gather more data about the subjects that could act as confounding variables, such as what programming languages that they've used historically and currently, age gap, and other variables TBD (to be determined).

In the original test process subjects were instructed to observe an identifier on a screen, which was then replaced with another four identifiers. One of them matched the previous identifier, whereas the rest of them were incorrect, meant of distract the subject. The data that was then measured is the accuracy of the subject to identify the correct word, as well as the time it took to reach an answer. From the tobii eye tracker additional data such as time spent on each word, amount of fixations vs saccades (eye flicker), can be extracted. Furthermore, data such as subject age, experience and background will be extracted in an interview like setting.

3.2 Eye-tracking Equipment

3.3 Material and Stimuli

3.4 Visual Effort and Areas of Interest

3.5 Study Variables

3.6 Hypotheses

3.7 Participants

3.8 Instrumentation

Chapter 4

Results

4.1 Correctness and Find Time

4.2 Visual Effort

4.3 Similarities and Differences

Chapter 5

Discussion

Chapter 6

Threats to Validity

Chapter 7

Conclusions & Future Work

7.1 Future Work

Bibliography

[1] Dave Binkley, Marcia Davis, Dawn Lawrie, and Christopher Morrel. To camelcase or under_score. IEEE, May 2009. doi: 10.1109/ICPC.2009.5090039.

[2] Florian Deissenboeck and Markus Pizka. Concise and consistent naming. *Software Qual Journal*, 14(3):261–282, 2006. doi: 10.1007/s11219-006-9219-1. eclipse.org.

[3] Julie Epelboim, James R. Booth, Rebecca Ashkenazy, Arash Taleghani, and Robert M. Steinman. Fillers and spaces in text: The importance of word recognition during reading. *Vision Research*, 37(20):2899–2914, 1997. doi: 10.1016/S0042-6989(97)00095-3.

[4] Bonita Sharif and Jonathan I. Maletic. An eye tracking study on camelcase and under_score identifier styles. IEEE, June-July 2010. doi: 10.1109/ICPC.2010.41.