

論文の内容の要旨

論文題目 An Empirical Study and Code-Generation Techniques for Fluent Interfaces
(Fluent Interface のための実証研究とコード生成技術)

氏 名 中丸 智貴

This dissertation presents our research to improve the user experiences of software libraries with *fluent interfaces*, interfaces designed to be used by chaining method invocations. The dissertation includes three studies: (1) An empirical study of the use of fluent interfaces in the real world, (2) an empirical study to discover desirable language designs for the use of fluent interfaces, and (3) the development of code-generation techniques to enable quick construction of safe fluent interfaces.

Study (1) is a background study to quantitatively reveal the significance of studying fluent interfaces. In previous studies on fluent interfaces, the significance has been claimed only qualitatively based on the abundance of existing fluent interfaces. To the best of our knowledge, no quantitative evidence has been provided to support the widespread use of fluent interfaces. For our goal, we conducted repository mining of numerous git repositories. Specifically, we collected 2,814 Java repositories on GitHub and analyzed historical trends in the use of fluent interfaces in those repositories.

Study (2) aims to help language developers to design their language appropriately for using fluent interfaces. To find such language designs, we analyzed Java code snippets in the real world and mined problematic code patterns for fluent interfaces. Our results are summarized as a list of desirable language designs and the statistically-estimated values of how effective it would be to introduce each language design into Java. The information we made is beneficial for language developers who attempt to improve the user experiences of fluent interfaces in their language because they can use the list and estimated impact values to smoothly discuss what language design to be adopted.

Study (3) aims to enable library developers to quickly give misuse-detection capabilities to their fluent interfaces, i. e., to quickly create safe fluent interfaces. Although safe fluent interfaces have been known to benefit their users, they are not widely developed in the real world due to their high development cost. While several code-generation techniques have been proposed to reduce the cost, those existing techniques lack two essential features for practical use: generics support and sub-chaining support. In Study (3), we propose two novel code-generation techniques to address those problems.

All the studies presented in the dissertation include artifacts that offer values to society, not only to academia. In Study (1) and Study (2), we analyzed real-world Java source code to benefit real-world programmers and built a publicly available dataset to further investigate the real-world use of fluent interfaces. The code-generation techniques in Study (3) are demonstrated in tools named PROTOCOOL and Silverchain, which allow real-world library developers to test our techniques in real-world settings. Those tools are openly available at GitHub.