An Empirical Analysis of the Costs of Clone- and Platform-Oriented Software Reuse

Jacob Krüger,[1] Thorsten Berger[2]

Abstract: In this extended abstract, we summarize our paper with the homonymous title published at the Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE) 2020 [KB20].

Keywords: Economics; Software reuse; Empirical study; Clone & own; Software product line; Platform engineering

Software reuse is a core practice to reduce development costs and improve the quality of software systems. Organizations typically employ one of two strategies to reuse software:

Clone & Own. When employing this strategy, developers create an independent clone of an existing system and adapt it to new customer requirements.

Platform Orientation. When employing this strategy, developers implement a code base that allows them to derive customized variants based on concepts from product-line engineering, such as variability mechanisms (e.g., preprocessors) and feature models.

Either strategy has its pros and cons, enforcing different software architectures and development activities. Clone & own is cheap and readily available, for instance, in the form of branches in version-control systems—which is why most organization start reusing software based on this strategy. However, particularly when maintaining cloned variants becomes a burden, most organizations migrate these variants towards a platform—which requires high upfront investments, but promises to substantially reduce development and maintenance costs as well as the time-to-market of new variants.

Deciding which strategy to employ is a core decision for any organization and has long-term impact in its software development. Despite this importance, the costs resulting from either strategy are not well-understood and, in fact, there is a lack of systematically elicited data to provide guidance for organizations. In our paper, we describe an empirical study with which we elicited such data, investigating the development activities, costs, cost factors, and benefits of both reuse strategies. For this purpose, we combined a systematic literature

[1] Otto-von-Guericke-University Magdeburg, Germany
jkrueger@ovgu.de

[2] Ruhr-University Bochum, Germany & Chalmers | University of Gothenburg, Sweden
bergert@chalmers.se

review of 57 publications that report quantified costs on software reuse with 26 interviews at a large organization. During the interviews, we elicited the software-reuse process employed at the organization. Furthermore, we asked interviewees to do informed, judgment-based cost estimations for a concrete variant they developed, and to assess the impact of cost factors on developing variants. We triangulated the data from both sources to confirm and refute common hypotheses on software reuse, providing concrete empirical data.

Among others, the core findings of our study are:

- The organization employs a development process that integrates clone & own with platform orientation. Such processes have recently also been identified for other organizations and open-source projects, but are still rarely considered in research.
- The success of reuse heavily depends on establishing platform orientation, with our data suggesting total median savings of 52 %. Particularly the reduced costs of variant development (-67 %) and quality assurance (-60 %) are major pros of a platform.
- A critical success factor for platform orientation is the quality of the code base, which is why fewer bugs are found (-70 %) in established platforms. Our data confirms additional benefits, such as a substantially reduced time-to-market (-63 %), and highlights the impact of various cost factors, particularly developers' knowledge regarding the software.

In total, we identified 18 pieces of confirmatory evidence for established hypotheses on software reuse, also indicating that a platform is preferable over clone & own—which does promise benefits, but on a lower level. Moreover, we identified seven inconclusive and three refuting pieces of evidence, suggesting that (1) clone & own and platform orientation are not strictly separated in practice; (2) change propagation can be more problematic in a platform compared to clone & own; and (3) platforms are essential to successfully establish innovative variants in new markets.

Our contribution is the first evidence-based body-of-knowledge on the costs of two established reuse strategies, providing important insights for research and practice. For instance, this body-of-knowledge helps practitioners understand both reuse strategies and improve their decision making, while we highlight open problems and particularly costly activities to provide guidance for further research. In future work, we will investigate our inconclusive and refuting evidence in more detail. Also, we will conduct additional studies to verify our data and derive a decision model that helps organizations decide for a reuse strategy.

Bibliography

[KB20] Krüger, Jacob; Berger, Thorsten: An Empirical Analysis of the Costs of Clone- and Platform-Oriented Software Reuse. In: Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering. ESEC/FSE. ACM, pp. 432–444, 2020.