

Inner Source in Platform-based Product Engineering

Dirk Riehle¹, Maximilian Capraro², Detlef Kips³, Lars Horn⁴

Abstract: Inner source is the use of open source best practices within an organization. As such, it is an approach to collaboration across intra-organizational boundaries for the creation of shared reusable assets. Prior project reports on inner source suggest improved code reuse and better knowledge sharing. Using a multiple-case case study research approach, we analyze the problems that three major software development organizations were facing in their product line engineering efforts. We find that a root cause, the separation of product units as profit centers from a platform organization as a cost center, leads to delayed deliveries, increased defect rates, and redundant software components. This article is a not-so-much extended abstract of [R+16].

Keywords: Inner source, product line engineering, product families, platform-based product engineering, open source, open collaboration, case study research.

1 Research Question and Approach

This article presents case study research on the situation of three major software development organizations which were trying to apply inner source to platform-based product engineering.

Our case study companies expected inner source to help them overcome problems with lack of resources, lack of pertinent skills, and unclear requirements. Yet, they had problems putting inner source into practice. To this end, this article addresses the following research questions:

- *RQ1: What are current problems in platform-based product engineering?*
- *RQ2: What benefits do organizations expect from adopting inner source?*
- *RQ3: What problems did they experience when adopting inner source?*

The research method employed is *multiple-case case study research*. Data gathering and analysis was performed using workshops, formal interviews, and materials review. The process was incremental with learnings being provided back to the case study

¹ Computer Science Department, Friedrich-Alexander University Erlangen-Nürnberg, 91058 Erlangen, Germany. E-mail: dirk.riehle@fau.de.

² Computer Science Department, Friedrich-Alexander University Erlangen-Nürnberg, 91058 Erlangen, Germany. E-mail: maximilian.capraro@fau.de.

³ Develop Group, 91058 Erlangen, Germany. E-mail: kips@develop-group.de.

⁴ E-solutions, 91058 Erlangen, Germany. E-mail: lars.horn@esolutions.de.

participants to receive validating feedback as to the theories being built (“member checking”).

2 Contributions and Results

Similar to open source, which often evolved from a volunteer-based (“free-for-all”) development process to a foundation-based (“managed”) software development process, we find that for our case study organizations, inner source should move to a governed process beyond the definition given in the beginning of this section.

The theories we present are only as good as the hypotheses that they generate and that can be validated in future work. Such confirmatory research will also allow for generalized conclusions that are not possible from pure case study research.

H1 Resistance and misunderstandings (like expected lower code quality of inner source components) can be addressed by way of education and active participation in the practice of inner source software development.

H2 Psychological openness or resistance to inner source (i.e. desire or fear to work under quasi-public scrutiny) depends on manager and developer personalities and is not a function of organizational structure or process.

H3 As long as open source does not come natural to an organization, inner source will not come easy to it either. Until this has changed, an organization will need an explicitly governed inner source process.

H4 Inner source and open source draw on the same competencies of people and a person who is good at one is likely to be good at the other.

H5 While there is no doubt about the need of platform software and shared reusable assets, a platform development organization may not be needed any longer. It can be replaced by an inner source program.

The last hypothesis is an interesting though probably controversial hypothesis: If large companies can work together in an open source foundation to develop shared infrastructure components, why can’t product units within an organization work together to create a platform of shared reusable assets without the need for a dedicated organizational unit that maintains this platform?

References

- [R+16] Riehle, D., Capraro, M., Kips, D., & Horn, L. (2016). Inner source in platform-based product engineering. *IEEE Transactions on Software Engineering*, 42(12), 1162-1177.