Chapter V

A Quantitative Study of the Adoption of Design Patterns by Open Source Software Developers

Michael Hahsler
Vienna University of Economics and Business Administration, Austria

ABSTRACT
Several successful projects (Linux, Free-BSD, BIND, Apache, etc.) showed that the collaborative and self-organizing process of developing open source software produces reliable, high quality software. Without doubt, the open source software development process differs in many ways from the traditional development process in a commercial environment. An interesting research question is how these differences influence the adoption of traditional software engineering practices. In this chapter we investigate how design patterns, a widely accepted software engineering practice, are adopted by open source developers for documenting changes. We analyze the development process of almost 1,000 open source software projects using version control information and explore differences in pattern adoption using characteristics of projects and developers. By analyzing these differences, we provide evidence that design patterns are an important practice in open source projects and that there exist significant differences between developers who use design patterns and who do not.
INTRODUCTION

The growing need for reliable software has made software engineering an important industry in the last few decades. The steady progress produced an enormous number of different approaches, concepts, and techniques: structured analysis and design, the object-oriented paradigm, agile software development, component-based systems, frameworks, and design patterns, just to mention a few. These techniques were developed with traditional software development in a commercial environment in mind. Recently, a new organizational form of collaborative software development, the open source movement (Raymond, 1999), gained popularity. Open source software (OSS) development differs from traditional forms in many respects. For example, the source code is publicly shared and therefore rigorously peer reviewed, the development teams are often geographically dispersed, and massive system-level testing by large user communities is conducted instead of extensive unit tests (Dempsey, Weiss, Jones & Greenberg, 2002; Jorgensen, 2001; Perpich, Perry, Porter, Votta, & Wade, 1997; Vixie, 1999). Quantitative research is the key to understanding how these differences influence the adoption of existing software engineering practices or how software engineering practices are adapted to the needs of open source development.

In this chapter we study how design patterns are used in open source software development teams. We are interested in the question “Are design patterns are useful for open source development and, if so, are there factors that influence their adoption?” To gain an insight into the application of design patterns, we analyze historic data of the development process of OSS projects.

This chapter is organized as follows: As the starting point for the chapter we review literature about design patterns to identify how patterns are used for traditional software development. Next, we describe the research method employed by the study. We present the used data set and its main characteristics followed by the analysis of the data set and the discussion of the results. We conclude the chapter with the main findings and point out directions for further research.

BACKGROUND AND RELATED LITERATURE

Design patterns describe non-obvious solutions in a standard written form for recurring software design problems in a certain context. They are normally developed by experts from their experiences with many existing systems and represent good and flexible solutions.

Since the introduction of the first software pattern catalog containing 23 design patterns by Gamma, Helm, Johnson, and Vlissides (1995), design patterns were rapidly accepted by the software engineering community, and their use is now strongly facilitated by the Unified Modeling Language (UML), the standardized notation for object-oriented analysis and design. The number of publications about design patterns has soared, and even several conference series on the topic were initiated.
Related Content

A Perspective on Software Engineering Education with Open Source Software
[www.igi-global.com/article/a-perspective-on-software-engineering-education-with-open-source-software/101203?camid=4v1a](www.igi-global.com/article/a-perspective-on-software-engineering-education-with-open-source-software/101203?camid=4v1a)

Open Source Approach for Mitigating Misinformation Risk in Complementary and Alternative Medicine Practices
[www.igi-global.com/chapter/open-source-approach-for-mitigating-misinformation-risk-in-complementary-and-alternative-medicine-practices/121008?camid=4v1a](www.igi-global.com/chapter/open-source-approach-for-mitigating-misinformation-risk-in-complementary-and-alternative-medicine-practices/121008?camid=4v1a)
Aligning Practice and Philosophy: Opening up Options for School Leaders
www.igi-global.com/chapter/aligning-practice-and-philosophy/120997?camid=4v1a

Issues and Aspects of Open Source Software Usage and Adoption in the Public Sector
www.igi-global.com/chapter/issues-aspects-open-source-software/21207?camid=4v1a