Dynamics of communities and code in open source software

Author(s):
Geipel, Markus Michael

Publication Date:
2009

Permanent Link:

https://doi.org/10.3929/ethz-a-005901107 →

Diss. ETH No. 18480

Dynamics of Communities and Code in Open Source Software

A dissertation submitted to the
ETH ZURICH

for the degree of
Dr. sc. ETH Zürich

presented by
MARKUS MICHAEL GEIPEL
Dipl.-Inf. Univ., Technische Universität München
born 29. August 1980
citizen of the Federal Republic of Germany

accepted on the recommendation of
Prof. Dr. Dr. Frank Schweitzer, examiner
Prof. Dr. Georg von Krogh, co-examiner

2009

Abstract

Abstract in English

The term Open Source implies that the source code, defining the software, is open. This means that everyone possesses the right to read the code, modify it, copy it, compile and use it free of charge. From an economic perspective this seems paradoxal: A complicated information good is produced without direct monetary compensation. And yet, Open Source software became a multi-million dollar business, with numerous highly successful applications.

The counterintuitive success of Open Source software as well as its distributed and open organization spawned considerable research interest across diverse scientific disciplines. Yet, interdisciplinary research is still scarce. Moreover, a majority of the work is either case study based or theoretic. Quantitative analyses are rare. In this thesis, existing work is thus complemented in two ways: First, the research is based on an extensive database which allows for large-scale quantitative analyses. Second, to process this data, methods and knowledge originating from different disciplines are employed: physics of complex systems, computer science and economics & management.

The topical focus of this thesis is twofold. First, the dynamics of the social communities around the software are investigated. Second, the dynamics of the source code itself. To unveil the dynamics of Open Source communities, an empirical analysis of communication in 100 Open Source projects was conducted. The results show a high involvement of users and strong interaction between users and developers. Moreover, there is an association between community turnover and segregation between users and developers. Finally, a highly skewed distribution of communication effort is found. This last finding in combination with the fact that most correlations in the data were weak to medium in strength, entail that Open Source projects can be quite diverse in dynamics. Thus, case study based research and quantitative approaches need to complement each other to paint a complete picture.

The analysis of the dynamics of the code focusses on the effect and evolution of dependencies between software modules. An analysis of 30 large and active Java projects shows that the majority of the dependencies are irrelevant for the co-change characteristics of the modules. Furthermore, dependencies may foster flexibility. This counterintuitive result suggests that the negative role of dependencies is overestimated in literature. Thus, new light is shed on the conflict between module reuse and cutting dependencies. Apart from the change dynamics, an analysis of the time-evolution of the software, revealed several novel features of the growth of the dependency network. Among them, power-law distributed initial degrees of newly added modules, and a density growth in the dependency network. Furthermore, mathematical models are proposed and verified. The findings contribute to a better understanding of software evolution and set the stage for new software engineering and management tools. They also show the importance of verifying models against different aggregate representations of the data.

In a broader context, this thesis shows how in interdisciplinary research the different fields of knowledge complement each other. Implications as well as perspectives of the interdisciplinary data-driven approach are discussed.

Kurzfassung auf Deutsch

Der Begriff Open Source impliziert, dass der Quellcode, welcher die Software definiert, offen ist. Das heisst, dass jeder das Recht hat, den Code kostenlos zu lesen, zu verändern, zu kopieren, zu kompilieren und zu benützen. Vom ökonomischen Standpunkt betrachtet erscheint dies paradox: Ein komplexes Informationsgut wird ohne direkte Vergütung produziert. Und dennoch hat sich Open Source Software zu einem Multi-Millionen-Dollar-Geschäft entwickelt, mit zahlreichen ausgesprochen erfolgreichen Anwendungen.

Der überraschende Erfolg von Open-Source-Software, wie auch die verteilte und offene Organisationsstruktur von Open-Source-Projekten haben beträchtliches Forschungsinteresse in verschiedenen wissenschaftlichen Disziplinen geweckt. Und dennoch ist interdisziplinäre Forschung die Ausnahme. Des Weiteren ist ein grosser Teil der Literatur entweder fallstudienbasiert oder theoretisch. Quantitative Analysen hingegen sind selten. In dieser Arbeit wird daher die existierende Literatur in zweierlei Hinsicht ergänzt: Erstens, die vorliegende Forschung gründet sich auf eine umfangreiche Datenbasis, welche gross angelegte quantitative Analysen ermöglicht. Zweitens, um diese Daten zu verarbeiten, wird auf Wissen und Methodik aus verschiedenen Disziplinen zurückgegriffen: Die Physik der komplexen Systeme, Informatik und Ökonomie & Management. Zwei thematische Schwerpunkte werden behandelt: Erstens, die soziale Dynamik in Open-Source-Projekten. Und zweitens, die

Dynamik des Quellcodes selbst. Um die soziale Dynamik offen zu legen, wurde die Kommunikation in 100 Open-Source-Projekten empirisch untersucht. Die Ergebnisse zeigen eine hohe Beteiligung von Benutzern sowie eine starke Interaktion zwischen Benutzern und Entwicklern. Ausserdem korrelieren Mitgliederfluktuation und Segregation zwischen Benutzern und Entwicklern. Schliesslich lässt sich noch eine stark ungleiche Verteilung von Kommunikationsaufwand nachweisen. Diese Erkenntnis und die Tatsache, dass die meisten Korrelationen in den Daten lediglich schwach bis mittelstark sind, implizieren, dass Open-Source-Projekte sich in ihren Eigenschaften stark unterscheiden. Daher müssen sich Forschungsergebnisse aus Fallstudien und quantitative Ansätze gegenseitig ergänzen um ein vollständiges Bild zu zeichnen.

In der Analyse der Codedynamik werden die Wirkung und Evolution von Abhängigkeiten zwischen Softwaremodulen untersucht. Die Auswertung von 35 grossen und aktiven Java-Projekten zeigt, dass die Mehrheit der Abhängigkeiten keinerlei Bedeutung für die Modifikationscharakteristik der Module hat. Des Weiteren können Abhängigkeiten die Flexibilität der Software begünstigen. Abgesehen von der Modifikationsdynamik, offenbart die Analyse der zeitlichen Entwicklung der Software neue Aspekte des Wachstums des Abhängigkeitsnetzwerks. Unter anderem eine Power-Law-Verteilung der Knotengrade von neu hinzugefügten Modulen und eine mit der Zeit wachsende Dichte des Abhängigkeitsnetzwerks. Zudem werden mathematische Modelle für die beobachtete Dynamik vorgeschlagen und verifiziert. Diese Ergebnisse tragen zum besseren Verständnis der Software-Evolution bei und schaffen die Voraussetzungen für neue Software-Engineering und Management-Werkzeuge. Des Weiteren unterstreichen sie die Wichtigkeit, Modelle gegen verschiedene aggregierte Darstellungen der Daten zu verifizieren. In einem weiteren Kontext zeigt die vorliegende Arbeit, wie sich verschiedene Wissensbereiche in fachübergreifender Forschung ergänzen. Die Implikationen als auch Perspektiven des datenbasierten, interdisziplinären Ansatzes werden diskutiert.