

Chapter 5.2

Beyond Development: A Research Agenda for Investigating Open Source Software User Communities

Leigh Jin

San Francisco State University, USA

Daniel Robey

Georgia State University, USA

Marie-Claude Boudreau

University of Georgia, USA

ABSTRACT

Open source software has rapidly become a popular area of study within the information systems research community. Most of the research conducted so far has focused on the phenomenon of open source software development, rather than use. We argue for the importance of studying open source software use and propose a framework to guide research in this area. The framework describes four main areas of investigation: the creation of OSS user communities, their characteristics, their contributions and how they change. For each area of the framework, we suggest several research questions that deserve attention.

INTRODUCTION

In recent years, the open source software (OSS) development movement has captured the attention of both information systems practitioners and researchers. The “open community model” is one that involves the development and support of software by volunteers with no or limited commercial interest. This model differs from proprietary software development, and with other open source business models such as corporate distribution, sponsored open source and second-generation open source (Watson, Boudreau, York, Greiner, & Wynn, in press). The open community model is appealing to many because of its application of community principles of governance over commercial activities (Markus, Manville, & Agres, 2000; von Hippel & von Krogh, 2003). By describing open source as a “movement,” we

reflect the broader excitement about the implications of community governance processes in a knowledge economy (Adler, 2001).

Open source has rapidly become a popular area of study within the information systems (IS) research community, as evidenced by the appearance of special tracks for OSS within conferences and special issues of journals. For example, the Americas Conference on Information Systems sponsored an “Open Source Adoption and Use” minitrack, and the Hawaii International Conference on System Sciences offered a minitrack on “Open Source Software Development.” On the journal side, *Management Science* invited submissions to a special issue on “Open Source Software” in 2004. Also, *Journal of Database Management* announced a special issue on “Open Source Software.”

Although these calls for OSS research do not limit contributions, the vast majority of the research conducted so far has focused on OSS development rather than use (Fitzgerald & Kenny, 2003). The interest in open community development reflects a desire to explain the counterintuitive practice of treating commercially valuable products as public goods rather than proprietary products for sale. Likewise, the development and maintenance of complex software products by communities of expert volunteers has piqued interest into the incentives for developers. As a consequence of the primary focus on OSS development, little research has yet been conducted on OSS use, especially by non-technical users.

The neglect of OSS use may be attributed to two false assumptions about OSS projects. First, it is known that people often become OSS developers because they intend to use the product being developed. To echo Raymond's (2001) frequently quoted expression, OSS developers are users with an “itch to scratch,” so they are willing to devote time and expertise to develop software solutions to their own problems as users. Thus, it is commonly assumed that there is no distinction between OSS developers and users (Feller & Fitzgerald,

2000). Following this line of argument, we might conclude that no special research agenda for OSS use is needed because OSS use is redundant with OSS development.

However, this argument and its underlying assumption can be challenged on the grounds that OSS use by technically experienced developers differs from OSS use by technically naïve users. Verma, Jin, and Negi (2005) argued that technical developers and non-technical users have very different interpretations of OSS's ease of use. Non-technical users may experience difficulty in using OSS products because OSS developers are motivated to improve functionality rather than usability (Nichols & Twidale, 2003). Thus, even though all OSS developers are likely also to be users, the distinction between developers and non-technical users remains important.

The assumption can also be challenged by statistics showing the rapid rise in the number of OSS users, the vast majority of whom have no interest or capability to contribute to modifications of the source code (Fitzgerald & Kenny, 2003). For widely distributed OSS such as Linux, it makes no sense to assume that more than a small percentage of users could possibly become developers (von Hippel & von Krogh, 2003). Clearly, users vastly outnumber developers in larger OSS projects. As OSS development becomes increasingly targeted toward productivity and entertainment applications, the relative proportion of non-technical users can be expected to increase.

The second assumption dissuading research on OSS use is that the OSS movement is unique solely because of the way software is developed, but that its use is similar to any other type of software. Given the abundance of IS research that is focused on the adoption and use of software applications, one might assume that no special research program is needed for OSS use.

This assumption can be challenged by examining some differences between OSS and proprietary software. Users of OSS are typically confronted by a fundamentally different type of technical

11 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/beyond-development-research-agenda-investigating/29480

Related Content

Importance of Systems Engineering in the Development of Information Systems

Miroljub Kljajic and John V. Farr (2010). *Emerging Systems Approaches in Information Technologies: Concepts, Theories, and Applications* (pp. 51-66).

www.irma-international.org/chapter/importance-systems-engineering-development-information/38173

Security Evaluation of Service-Oriented Systems Using the SiSOA Method

Christian Jung, Manuel Rudolph and Reinhard Schwarz (2013). *Developing and Evaluating Security-Aware Software Systems* (pp. 20-35).

www.irma-international.org/chapter/security-evaluation-service-oriented-systems/72196

Model Driven Integration of Heterogeneous Software Artifacts in Service Oriented Computing

Eric Simon and Jacky Estublier (2013). *Migrating Legacy Applications: Challenges in Service Oriented Architecture and Cloud Computing Environments* (pp. 332-360).

www.irma-international.org/chapter/model-driven-integration-heterogeneous-software/72223

Milestone-Driven Agile Execution

Eduardo Miranda (2021). *Balancing Agile and Disciplined Engineering and Management Approaches for IT Services and Software Products* (pp. 1-27).

www.irma-international.org/chapter/milestone-driven-agile-execution/259169

A Survey of Digital Image Watermarking Techniques in Spatial, Transform, and Hybrid Domains

K. Prabha and I. Shatheesh Sam (2022). *International Journal of Software Innovation* (pp. 1-21).

www.irma-international.org/article/a-survey-of-digital-image-watermarking-techniques-in-spatial-transform-and-hybrid-domains/309113