Software Developers are Humans, Too!

Bogdan Vasilescu
Department of Mathematics
and Computer Science
Eindhoven University of
Technology
Eindhoven, The Netherlands
b.n.vasilescu@tue.nl

Abstract

Open-source communities can be seen as
knowledge-sharing ecosystems: participants learn from the
community and from one another, and share their
knowledge through contributions to the source code
repositories or by offering support to users. With the
emergence and growing popularity of social media sites
targeting software developers (*e.g.*, STACKOVERFLOW,
GITHUB), the paths through which knowledge flows
within open-source software knowledge-sharing ecosystems
are also beginning to change. My dissertation research
seeks to raise our understanding of these changes.

Author Keywords

Software developers; open-source; human aspects

ACM Classification Keywords

H.5.3 [Information interfaces and presentation (e.g.,
HCI)]: Computer-supported cooperative work.

Introduction

Open-source software (OSS) development is typically
characterised as a decentralised, self-directed, highly
interactive, and knowledge-intensive process [4]. In OSS,
developers, supporters, and users, likely with different skill
sets and skill levels, different personalities and different
cultural backgrounds, organise themselves in online

communities and voluntarily contribute to a collaborative software project [7]. Borrowing from ecology, OSS communities can be seen as knowledge-sharing *ecosystems* [6,8]: software developers (the organisms; biotic components) interact among themselves and with the environment (software and hardware tools; abiotic components); these biotic and abiotic components are linked together through *knowledge flows* in a symbiotic relationship, wherein "the community learns from its participants, and each individual learns from the community" [8].

Recent years have witnessed the emergence and growing popularity of social media sites targeting software developers, such as GitHub (hosting code repositories), Jira (issue tracking), or the StackExchange network of question and answer websites (*e.g.*, StackOverflow). In an ecological sense, such sites can be viewed as *disturbances* to knowledge-sharing ecosystems surrounding traditional OSS development: they may have profound immediate effects on an ecosystem and the knowledge flows therein, changing the ways in which developers collaborate, learn, and communicate among themselves and with their users [1,3,9].

Question, Methodology, Contributions

My dissertation research seeks to raise our understanding of these effects by answering the following question:

> How do software-development-related social media sites impact knowledge flows within OSS knowledge-sharing ecosystems?

To this end, I pursue a mixed-methods approach, borrowing techniques from the social sciences (*e.g.*, interviews, surveys), on the one hand, and mining software repositories (*e.g.*, merging identities used by the same individual in different repositories [5]), on the other.

The work included in this dissertation (best labelled as collaborative software engineering) sits at the intersection of two research communities, computer-supported cooperative work (CSCW) and software engineering (SE). SE could benefit, *e.g.*, from understanding whether and how social media increases the productivity of software developers, or how social aspects can inform tool design or be incorporated into models that help to predict the evolution of software systems. CSCW could benefit, *e.g.*, from the repository mining techniques or the quantitative analysis methods developed or refined as part of this work.

Preliminary Results

Within a knowledge-sharing ecosystem, *knowledge seekers* (consumers) and *knowledge providers* (producers) interact using a *medium* through which knowledge flows. Over the past two years, together with colleagues from academia I have examined two such knowledge-flow paths.

I. Knowledge sharing through contributions to source code repositories. One of the knowledge-flow paths involves developers (knowledge providers) sharing their knowledge through contributions to the source code repositories (the medium), with the goal of driving further the evolution of the software project (knowledge consumer) and ensuring the ecosystem's sustainability.

Given the natural diversity of human beings, one can also expect heterogeneity of knowledge providers within an OSS ecosystem. Indeed, some contributors are likely to: be considerably more active than others; possess different skills; be involved in different types of activities (*e.g.*, coding, translating, writing documentation); or otherwise communicate, interact and collaborate differently. The social interactions between open source contributors, as well as their degree of project participation have been

reported repeatedly to influence software quality and complexity. Using two empirical studies of GNOME [14], a popular desktop environment for UNIX-type operating systems, and EMACS, a popular text editor [15], I uncovered:

- how workload and involvement of ecosystem contributors vary across projects and across activity types [14];
- the extent to which contributors specialise in particular activity types [14];
- how the differences in skills between contributors can be modelled, and what impact such differences might have on the ecosystem's sustainability [15].

II. Knowledge sharing through questions and answers on social Q&A platforms. Perhaps the most distinctive feature of social Q&A platforms (the most visible of which is STACKOVERFLOW for programming questions and answers) is *gamification*, *i.e.*, participants compete to obtain reputation points and badges, which enable additional privileges once various thresholds are exceeded (*e.g.*, moderation rights). Reputation and badges on such sites can be seen as a measure of one's expertise by peers and potential recruiters, and are known to motivate users to contribute more. In this setting, developers can act both as knowledge seekers and as knowledge providers.

As knowledge seekers: Developers create software by standing on the shoulders of others [9]: they reuse components and libraries, and go foraging on the Web for information that will help them in their tasks [2]. Here, knowledge-flow paths are formed between developers (knowledge seekers) asking, *e.g.*, on STACKOVERFLOW (medium) for help with their code, and the experts (knowledge providers) answering their questions.

As knowledge providers: Developers also engage in online support fora, such as mailing lists or Q&A platforms (media), as experts (knowledge providers), to satisfy a demand for knowledge. Here, knowledge seekers are other developers, perhaps less experienced, or users themselves.

To study the effects associated with participating in STACKOVERFLOW for both these roles of developers (knowledge seekers and knowledge providers), I created two longitudinal data sets combining activity on STACKOVERFLOW with activity in OSS, wherein participants overlap: one data set contains changes made to GITHUB repositories [12]; the other contains user support activities on R (a popular data analysis software) mailing lists [13]. Using these data sets I found that:

- there is a positive connection between participation in STACKOVERFLOW and productivity on GITHUB [12];
- activity (expertise) on one platform is indicative of activity (expertise) on the other [12];
- more active developers are more likely to engage in STACKOVERFLOW [12, 13];
- some groups of developers benefit more from participation in STACKOVERFLOW than others [12];
- the same knowledge providers answer questions faster in the gamified STACKOVERFLOW environment than on mailing lists [13];
- STACKOVERFLOW attracts mailing list experts, who disengage from mailing lists and transition to the new social medium [13].

Social factors. While investigating the previous two knowledge-flow paths, I realised that software developers are humans, too! Different participants are likely to be attracted differently by different knowledge-flow paths within a knowledge-sharing ecosystem, depending on a

multitude of social factors like gender, educational background, or personality traits. For instance, sharing knowledge through contributions to the source code typically requires very technical skills, while sharing knowledge through answers on a user support forum may be better suited for a communicative person.

In an empirical study comparing activity on the social Q&A platform STACKOVERFLOW with that on traditional mailing lists [10, 11], I focused on one such social factor—gender. I tried to understand how and when women engage in software-development-related online communities and found, *e.g.*, that gamified platforms such as STACKOVERFLOW are less successful at retaining women than traditional mailing lists.

Expectations from CSCW doctoral colloquium

Coming from a computer science background, I feel that my research would benefit immensely from exposure to and guidance from the CSCW community. In particular, I am interested in learning more about qualitative research methods, frequently used in the social sciences and in CSCW, but less so in software engineering research.

References

[1] Begel, A., Bosch, J., and Storey, M.-A. Social networking meets software development: Perspectives from GitHub, MSDN, Stack Exchange, and TopCoder. *IEEE Software 30*, 1 (2013), 52–66.

[2] Brandt, J., Guo, P. J., Lewenstein, J., Dontcheva, M., and Klemmer, S. R. Two studies of opportunistic programming: interleaving web foraging, learning, and writing code. In *CHI*, ACM (2009), 1589–1598.

[3] Dabbish, L. A., Stuart, H. C., Tsay, J., and Herbsleb, J. D. Social coding in GitHub: transparency and collaboration in an open software repository. In *CSCW*, ACM (2012), 1277–1286.

[4] Hemetsberger, A., and Reinhardt, C. Learning and knowledge-building in open-source communities a social-experiential approach. *Management Learning 37*, 2 (2006), 187–214.

[5] Kouters, E., Vasilescu, B., Serebrenik, A., and van den Brand, M. G. Who's who in GNOME: using LSA to merge software repository identities. In *ICSM*, IEEE (2012), 592–595.

[6] Mens, T., Claes, M., Grosjean, P., and Serebrenik, A. *Evolving Software Systems*. Springer, 2014, ch. Studying Evolving Software Ecosystems based on Ecological Models.

[7] Nakakoji, K., Yamamoto, Y., Nishinaka, Y., Kishida, K., and Ye, Y. Evolution patterns of open-source software systems and communities. In *IWPSE*, ACM (2002), 76–85.

[8] Sowe, S. K., Stamelos, I., and Angelis, L. Understanding knowledge sharing activities in free/open source software projects: An empirical study. *JSS 81*, 3 (2008), 431–446.

[9] Storey, M.-A. D., Treude, C., van Deursen, A., and Cheng, L.-T. The impact of social media on software engineering practices and tools. In *FoSER*, ACM (2010), 359–364.

[10] Vasilescu, B., Capiluppi, A., and Serebrenik, A. Gender, representation and online participation: A quantitative study of StackOverflow. In *Social Informatics*, ASE/IEEE (2012), 332–338.

[11] Vasilescu, B., Capiluppi, A., and Serebrenik, A. Gender, representation and online participation: A quantitative study. *Interacting with Computers* (2013), 1–24.

[12] Vasilescu, B., Filkov, V., and Serebrenik, A. StackOverflow and GitHub: Associations between software development and crowdsourced knowledge. In *SocialCom*, ASE/IEEE (2013), 188–195.

[13] Vasilescu, B., Serebrenik, A., Devanbu, P. T., and Filkov, V. How social Q&A sites are changing knowledge sharing in open source software communities. In *CSCW*, ACM (2014).

[14] Vasilescu, B., Serebrenik, A., Goeminne, M., and Mens, T. On the variation and specialisation of workload – A case study of the Gnome ecosystem community. *Empirical Software Engineering* (2013), 1–54.

[15] Vasilescu, B., Serebrenik, A., and van den Brand, M. G. The Babel of software development: Linguistic diversity in Open Source. In *SocInfo*, Springer (2013), 391–404.