

UiO : Institutt for informatikk
Det matematisk-naturvitenskapelige fakultet

How social networks influence open source software development

John Rongved , Lars Tveito , Kristian A. Hiorth
November 14, 2014



Contents

1	Introduction	1
2	Related works and opinions	3
3	Discussion	6
3.1	The Emacs community	6
4	Concluding remarks	7

1 Introduction

Open-source software development is often a highly collaborative activity, and effective communication between members and coordination of tasks is critical for a successful project. Historically, mailing lists have been the preferred medium for coordinating development and user support activities[4]. In recent years however, developers are moving away from mailing lists as the preferred means of communication. Replacing the mailing lists are a number of websites founded in the recent years such as GitHub and StackOverflow. Social media has changed the landscape of the internet in several ways, and developers are taking advantage of this with employing new methods in the development. In this report we will describe some of the most common websites used for open-source development. We will also briefly present relevant litterature on the subject, and discuss advantages and challenges presented by the new paradigm. Social media has changed the internet communities, since Commons Based Peer Production is very much community driven, we want to investigate how social media has impacted open source development as a whole.

During the previous decade, a number of websites have been founded that provide developers with multiple specialized channels of communications. Websites such as Reddit*, StackOverflow†, GitHub‡ and HackerNews§ have become widely popular.

Reddit

Reddit is a social networking and news site that allow users to submit content such as links or text content. The site does not itself host multimedia content,

*<http://www.reddit.com>

†<http://www.stackoverflow.com>

‡<http://www.github.com>

§<https://news.ycombinator.com>

but users provide links with interesting content from sites like YouTube*, Imgur† and SoundCloud‡. Reddit is organized by a collection of categories or ‘subreddits’. The subreddit for programming([r/programming](#)) consists of about 500 000 subscribers. Reddit works by a voting system that prioritizes ‘good’ content, and ‘bad’ content is displaced. This way, if a website is prominently displayed it means that many people have voted on it as a interesting piece of content.

Reddit has a high number of users, around 175 million unique users every month. Because of this huge number of users it has the potential to generate huge traffic spikes, this is sometimes referred Reddit also functions as sort of a funnel for traffic, where the ‘good’ content quickly rises to the top of the page, resulting in more traffic. This in turn spikes traffic on the site that hosts this content, sometimes resulting in what is know as the ‘Reddit hug of death’, where a small website hosting popular content crashes due to the spike in traffic generated by Reddit.

We will not do any investigation into non-technical subreddits, nor discuss any of them. We do think technical subreddits have an effect on CBPP, but we have no reason to believe that less technical subreddits (like birds-with-arms) have any impact on CBPP.

Hacker News

The news aggregator Hacker News is a social news website that caters to programmers and entrepreneurs. In a similar in structure to Reddit, Hacker News uses ‘upvoting’ as a way to ensure quality content is displayed most prominently on the site. The absence of a ‘downvote’, along with more comprehensive guidelines§, makes the community perhaps more stable, compared to Reddit.

StackOverflow

The social Q&A website for programmers, hobbyists, students or anyone seeking technical advice. StackOverflow has quickly become the number one place to seek information related to programming related problems. On the website users submit a question and ‘tags’ it with relevant categories. Similarly to Reddit posts are up- or down-voted by other users. If a submitted question is too similar to an already answered thread, it may be merged, or

*<http://www.youtube.com>

†<https://imgur.com>

‡www.soundcloud.com

§<https://news.ycombinator.com/newsguidelines.html>

if the question is badly worded or too banal, it may be deleted. If a question avoids these pitfalls, it may be answered by anyone. Quality responses that many users find useful will be upvoted to the top, and the user who submitted that answer will be awarded ‘rep points’ to indicate his/hers level of expertise. This reward system is frequently used in similar social websites, and is a form of ‘gamification’.

GitHub

Github is a Git repository web-based hosting service which offers its users repositories for revision control and source control management. The site has gained popularity by providing powerful tools, whilst staying easy to use. It revolves around Git which is a free and open source distributed source control management system, and plays a large role in teaching how to use the system. Since it’s no less powerful than it’s predecessors, but is a lot easier to use, we suspect it has lowered the bar to start making contributions to open sourced projects. For all public repositories the service is free, and in this way it pays a large contribution to open source development. One can also keep a private repository on their server, but this is in exchange for a fee. The site has by 2014 grown to be the largest code host in the world, and has over 3,4 million users. In 2013 Github announced that they had reached 10 million public repositories, which accumulates to a very large contribution to the open source community.

2 Related works and opinions

Traditionally, Free and Open Source Software (FOSS) projects have relied upon mailing lists as the primary means of communication in their communities.

Over the last several years, however, there has been a shift in the way communication happens on the Internet, with the advent of the Social Networks such as Facebook, Twitter and reddit. The latter, and other special interest social media sites like Hacker News, have seen particularly high adoption rates amongst software developers and power users, also for relatively technical discussions.

Additionally, several services which one would consider part of Social Media have been launched that are specifically aimed at software development, such as GitHub and StackOverflow, amongst many others.

There has been quite a bit of research into FOSS projects and their organizational behaviour, but much of this work may be starting to show its age and

has very much been centered around mailing lists (such as [2, 3]) and centralized (source) version control systems. Whilst mailing lists undoubtedly remain a very important communication channel for most projects, they are being challenged by social media and community behaviour may be changing because of it.

While research into social media is quite hot, its intersection with software development, and indeed FOSS projects has not seen very much published research. However, there have been a handful of interesting works during the last years.

In a recent case study of the R statistical programming language's community [4], Vasilescu et al. demonstrate how that project has seen a marked migration from traditional FOSS community support in the form of a mailing list, to community support through the Social Q&A site StackOverflow. StackOverflow is built around the concept of gamification[1] which appears to actually have changed the behaviour of community members. Through correlating email addresses, the authors showed that a significant number of informing providers (community members who answer questions) participated in both the mailing list and on StackOverflow, and that the same individuals would tend to answer much faster on the latter site. This can be explained by the fact that on StackOverflow, good answerers are rewarded with points, prominently displayed to other community members. Thusly, they may feel more motivated to provide timely responses to questions posed there, since the reward is higher.

StackOverflow has a very defined communication form, questions and answers. Therefore it is very suited to typical support/knowledge sharing activities. It is harder to imagine such benefits of gamification applied to less didactic communication, such as technical decision making, design issues, etc. These discussions will tend to be conducted among peers, whilst support usually is between knowledge seekers and providers, who are not equal. Furthermore, there may be no "right" answer, so awarding points and such would be very difficult. For these reasons, it is likely that sites such as StackOverflow will merely serve as a supplement to traditional mailing lists (or similar fora).

The increasing popularity and migration to distributed version control systems (dCVS) is also, in some cases, changing the way code actually makes it way into projects. Examples of dCVS are Git, Mercurial (colloquially known as Hg) and Bazaar. These have in common that they are less reliant on a single point of coordination during active development, which can make it much easier for small time contributors to work on code changes, since they can work locally under source control, or even set up third party repositories synchronized with the upstream.

Github not only offers gratis* hosting of Git dCVS repositories, but also provides a number of other code collaboration tools, like a graphical web interface to Git, issue tracking and code review tools. This has the potential to ease the burden of starting new projects, as well as contributing to existing ones that use Github's platform - project instigators need not spend time on setting up complicated infrastructure, and newcomers get access to good tools off the bat.

It should be noted that while Github revolves around Git, which itself is Free Software[†], the Github platform, including all the tools and data generated by them, is actually proprietary. This poses a dilemma for FOSS projects, since they may often not wish to become reliant on proprietary software (and indeed, a whole proprietary third party hosted platform).

FOSS projects, like all CBPP projects, are nothing without people forming a community around it. It is from a community, generally mostly users, that the project can recruit active contributors that help keep it moving forward and stay alive. In order to foster a community, projects need to be able to get word out about their existence, and about why they are interesting.

There have been some sites dedicated to this, such as (the now shuttered) FreshMeat, or SourceForge, in addition to the various technical news venues. However, social media present new opportunities in this domain. Programming/IT related subreddits, and especially Hacker News (and other similar social media) gather a large number of individuals with the interest and skills needed to participate in FOSS projects, not to mention people responsible to fulfill computing needs, either for themselves or for their employers. On social media, project members or anyone else are free to just post about interesting projects, forgoing the need for editorial approval and filtering. Usually, there is some form of crowdsourced moderation or promotion of posts, so that content deemed interesting by many gains more exposure. This might be an advantage for smaller projections, especially, that do not have any brand name recognition to allure tech journalists.

One especially interesting point is that, according to Singh et al. [3], inter-project relationships between developers may be quite beneficial to their success. If social media and coding platforms can help foster such relationships, which they would seem suited to do, this would be a big plus for the FOSS community.

*Github's services are free of charge for publically available repositories

[†]Git is licensed under the GNU General Public License version 2.0

3 Discussion

Knowing that social media is having an effect on FOSS projects in some ways, we would have liked to be able to investigate whether this effect stretched into the realm of development. In particular, it's known in the community that receiving attention on certain of the most popular software development social medias like (parts of) reddit and Hacker News tends to lead to heavy traffic spikes for the exposed project. Beyond this, there is little knowledge about longer term effects. Do projects really gain more users, and do these spikes lead to recruitment of new contributors?

In order to attempt to answer these questions, we contacted GitHub to inquire about the possibility of receiving some general traffic statistics, so we would be able to detect and document the occurrence of the aforementioned social media spikes. Unfortunately, this turned out not to be possible because of privacy issues.

3.1 The Emacs community

We wanted to look at a few existing projects and see how Social Media has influenced the community and project development. We were already to some extent a part of the Emacs community, and we thought this would make it easier to reach out to it.

Emacs, originally developed by Richard Stallman in 1976, currently has 191 active project members*. Since 2012 Emacs has featured its own package manager, making external packages more easily available. This has extended Emacs' ecosystem beyond the development of the Emacs core, and people are increasingly depending on external packages.

The most popular repository for external packages is Melpa[†] and currently contains 2076 external packages; of which 1762 are mirrored from GitHub and 220 are mirrored from EmacsWiki[‡]. The total number of libraries in EmacsWiki is 1952, which is relatively few compared to the swarming number of GitHub repository's, which is 21 152[§].

The presence of Github seems to have made a serious impact to the number of openly available projects for this particular community. We want to argue that this development also has an indirect effect on the development of the core of Emacs.

*<http://savannah.gnu.org/project/memberlist.php?group=emacs>

[†]<http://melpa.org> - at 11. November 2014

[‡]<http://www.emacswiki.org/>

[§]<https://github.com/search> - at 11. November 2014

Richard Stallman has spoke of the development of external packages several times on the Emacs mailing list. His general opinion is that libraries that other packages depend on should be included in the Emacs core*, but also shows skepticism toward this development†.

One consequence of external packages is that it makes changing the interfaces in Emacs more difficult, because the maintainers has no control over these packages. Because the copyright is held by whomever submitted the package, the packages can't simply be included in Emacs. He also expresses some concern that these external packages speaks against the GNU projects values, such as recommending use of non-free software.

4 Concluding remarks

References

- [1] Sebastian Deterding et al. “From game design elements to gamefulness: defining gamification”. In: *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*. ACM, 2011, pp. 9–15.
- [2] Christopher Oezbek, Lutz Prechelt, and Florian Thiel. “The Onion Has Cancer: Some Social Network Analysis Visualizations of Open Source Project Communication”. In: *Proceedings of the 3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development*. FLOSS ’10. Cape Town, South Africa: ACM, 2010, pp. 5–10. ISBN: 978-1-60558-978-7. DOI: 10.1145/1833272.1833274. URL: <http://doi.acm.org/10.1145/1833272.1833274>.
- [3] Param Vir Singh, Yong Tan, and Vijay S Mookerjee. “Network Effects: The Influence of Structural Capital on Open Source Project Success.” In: *MIS Quarterly* 35.4 (2011), pp. 813–829.
- [4] Bogdan Vasilescu et al. “How Social Q&A Sites Are Changing Knowledge Sharing in Open Source Software Communities”. In: *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing*. CSCW ’14. Baltimore, Maryland, USA: ACM, 2014, pp. 342–354. ISBN: 978-1-4503-2540-0. DOI: 10.1145/2531602.2531659. URL: <http://doi.acm.org/10.1145/2531602.2531659>.

*<https://lists.gnu.org/archive/html/emacs-devel/2014-09/msg00630.html>

†<https://lists.gnu.org/archive/html/emacs-devel/2014-09/msg00582.html>