2016 7th International Workshop on Empirical Software Engineering in Practice

How do GitHub Users Feel with Pull-Based Development?

Yusuke Saito*, Kenji Fujiwara*, Hiroshi Igaki†, Norihiro Yoshida‡ and Hajimu Iida*
*Graduate School of Information Science, Nara Institute of Science and Technology
Takayama-cho 8916-5, Ikoma-shi, Nara, 630-0192 Japan
†Graduate School of Information Science and Technology, Osaka Institute of Technology
Kitayama 1-79-1, Hirakata-shi, Osaka, 573-0196 Japan
‡Graduate School of Information Science, Nagoya University
Furo-cho, Chikusa-ku, Nagoya, 464-8601, Japan

***Abstract*—Modern OSS projects have adopted Git to manage versions of their source code and GitHub for hosting their Git repositories. GitHub provides a characteristic feature notably pull request, and many projects adopt pull-based development model by using it. This development model offers an opportunity to review the source code before merging it into the mainstream. Getting acceptance of the pull request, any developer should strictly follow the flow of pull-based development. To follow the flow, she/he needs to use correctly Git commands. However, using these commands is complicated and requires further knowledge of them. In this paper, we conducted a large-scale survey of 1552 developers on GitHub to investigate the difficulty of the Git from the aspect of the pull request. The result shows developers struggle with 'git rebase -i' command tends not to perform re-ordering, compressing and dividing their commits.**

I. Introduction

As distributed version control systems (DVCS) have been widespread [1], more projects, both proprietary and especially open source, adopt DVCS for managing the source code history. Emerging of the source code hosting services such as GitHub[1] and Bitbucket[2] have impacted the DVCS to be widely used. Up to November 2015, more than 29.9 million projects have been created in GitHub, and more than 3 million developers have been registered with Bitbucket. These hosting services provide features that support software development in various ways such as pull request and issues. In particular, the pull request, one of the features to create contributions [2], [3], support pull-based software development model [4], [5]. In pull-based development, developers pull them from other repositories and merge them locally instead of pushing changes to a central repository [4].

However in pull-based development, developers need to understand the flow of development and correctly use the DVCS commands, which is stated as complicated [6]. Our goal is to support these developers in pull-based development by proposing a tool to support the development and workflow. As a preliminary study, in this paper, we investigated how developers feel with pull-based development. We conducted a large-scale survey of 1,552 developers on GitHub to investigate how the developers feel with pull-based development. The survey was created based on prior research and our experience of pull-based development. To understand the habit of developers in pull-based development, the questions we are trying to answer are as follow.

- RQ1: *How do developers feel with pull-based development?*
- RQ2: *How unique practices are used in pull-based development?*
- RQ3: *How is the command usage different for Git users who use unique practice?*

II. Related Work

In recent work, several researchers investigated the characteristic of the pull request on GitHub and its acceptance factor by quantitative study [4], [7], [8], [9]. Also, Gousios et al. [10] conducted a more in-depth qualitative study by survey and interview to integrators in GitHub. They indicate accepted pull requests contain changes for the hot area in the project, consist of high-quality code and follow the project policy and coding style. Their studies suggest the necessity of supporting developers on integrating with a pull request. However, to support integrating by what developers wish to be, understanding of how developers act and feel in pull-based development is needed.

Git Wiki community conducted a survey to investigate how developers use Git on 2012 [3]. This survey had been opened for 20 days in 2012, and anyone who knew about the survey could have answered. The main focus of this survey is to understand the habit of Git users and what kind of tools they use to operate their Git repositories. Since this survey was conducted to entire Git user, examine about developers in pull-based development for further understanding is essential.

As mentioned before, following the project policy increases the acceptance rate of the pull request [10]. From this findings, developers to follow the flow of pull-based development will also increase the acceptance rate. To follow the flow, using the Git and GitHub correctly for developers is essential. However, DVCS, especially Git, is stated as complicated because of its design concept [6] which makes developers struggle during

[1] https://github.com

[2] https://bitbucket.org

[3] https://git.wiki.kernel.org/index.php/GitSurvey2012

978-1-5090-1851-2/16 $31.00 © 2016 IEEE
DOI 10.1109/IWESEP.2016.19

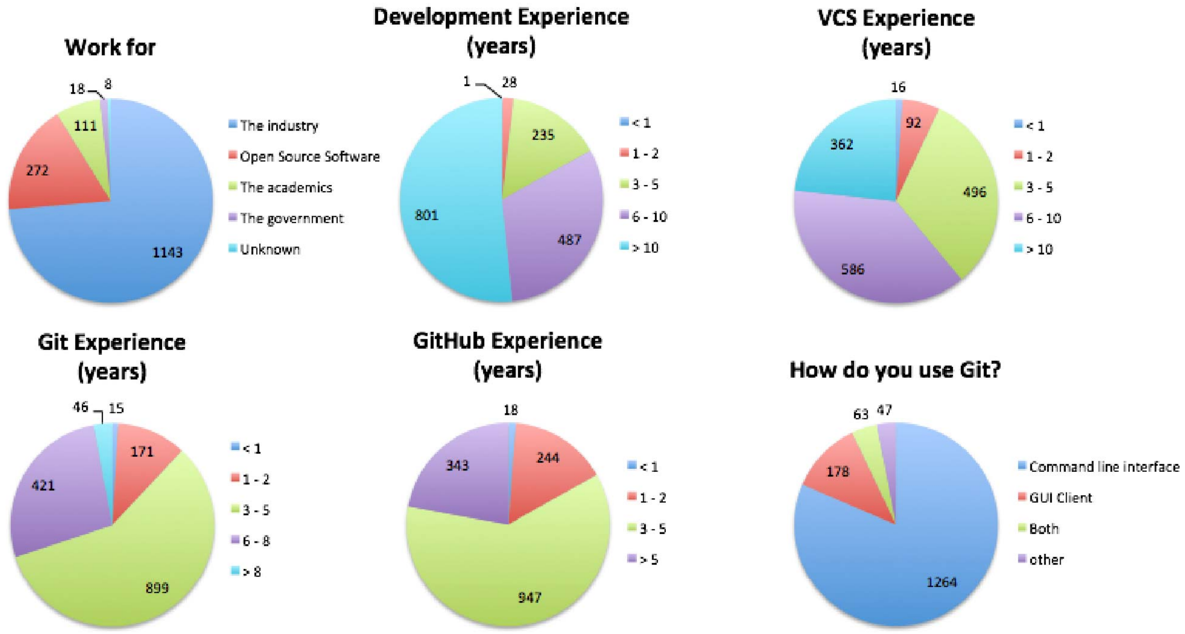

Fig. 1. Respondents profiles

pull-based development. In this paper, we conducted a qualitative study based on what Rosso et al. research [6] indicates.

III. Research Questions

The main focus on this study is to reveal how developers feel with pull-based development. As pull-based development popularity has been increasing, supporting the developers in pull-based development is in demand. To support those developers, it is important to know the characteristics and difficulties of using DVCS and the repository hosting services, used by developers during pull-based development. Our first question explores how developers feels with using Git and GitHub.

RQ1: *How do developers feel with pull-based development?*

To make the analysis easier, we further refine RQ1 in the following subquestions.

- RQ1.1: *How do developers feel with using Git?*
- RQ1.2: *How do developers feel with using GitHub and pull request?*

In the pull-based development model, some unique practices are performed. One of the practice is commit refactoring by using ‘git rebase -i’ command [11], [12]. Commit refactoring, such as reordering, compressing and dividing commits, are done to make neat commits contained in the pull request. When integrators of the project review the source code before merging the pull request, refactored commits increase the readability and acceptability of the pull request [10]. Another practice is Work-in-Progress (WIP) pull request[4], a pull request created in the middle of implementation[5]. WIP pull request is used to get a review on current source code and have a conversation with core team member on how to implement or solve the assigned issue. These practices are not indispensable, but it affects the pull request acceptance. To examine the usage of these unique practices, the second research question is as follow.

RQ2: *How unique practices are used in pull-based development?*

While performing unique practices, using uncommon Git commands is required. Because of the inconsistency of Git commands and options [6], using uncommon commands is troublesome for beginner developers with little experience in pull-based development. For the further understanding of those uncommon commands, it is beneficial to clear which commands are used and struggles for beginner developers. This leads to the last research question.

RQ3: *How is the command usage different for Git users who use unique practice?*

IV. Survey Target

We conducted a large-scale survey in order to investigate a large number of developers from a wide range of beginner to expert. In this section, we describe the data collection method and the characteristic of the survey respondents.

[4] https://github.com/blog/1943-how-to-write-the-perfect-pull-request

[5] https://www.atlassian.com/git/tutorials/making-a-pull-request/how-it-works

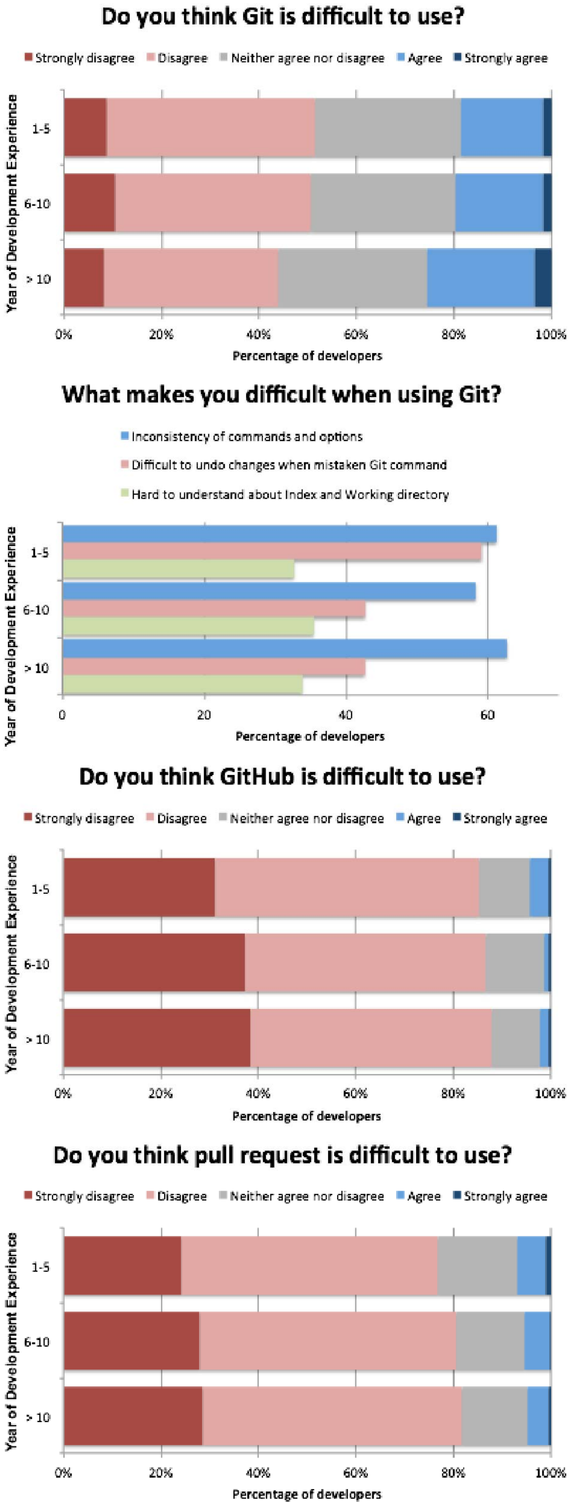

Fig. 2. Developers feeling with Git, GitHub and pull request

A. Pull-based Development Project and Its Developers

We used GHTorrent dataset [13] to select the pull-based development projects and its developers. Since most of the repositories on GitHub are inactive and not all repositories use the pull request [4], it is necessary to select effective sample [14]. In order to ensure our selected repositories are active and apply pull-based development, we have selected the 333 repositories that have received at least 5 pull request for each month in the year 2014. For every repository, we extracted the 13,152 developers on the following conditions.

- She/he created at least one pull request at the selected repositories in 2014.
- Her/his email address is registered in GitHub[6].

B. Respondents

To all developers extracted in the previous section, we sent an email for the survey request. Figure 1 shows an overview of the 1,552 respondents (11.8% answer rate). Most of the respondents work for industries (74%) and have more than 6 years of development experience (82%). For VCS experience, the percentage was mostly same among 1-5, 6-10 and more than 10 years. On the other hand, the majority of experience in Git and GitHub was 3-5 years (58% and 61%).

V. Results

In this section, we present our survey results per respondents question. For open-ended questions, we present the answers in *slanted*.

A. RQ1: How do developers feel with pull-based development?

To understand how developers feel with pull-based development, we asked developers how they think of using Git, GitHub, and pull request. We refined RQ1 into two subquestions, how developers feels with Git and GitHub. Overall results for RQ1.1 and RQ1.2 are shown in Figure 2.

1) RQ1.1: How do developers feel with using Git?: We asked developers question with Likert scale if they feel Git as difficult. The results show that about 20% of developers feel Git as difficult regardless of their development experience.

For those who felt Git as difficult, we investigated what factor influence the difficulty of Git. To find the influence factor of difficulties, we asked a multiple choice question based on the findings by Rosso et al. [6]. The respondents also had the opportunity to report the difficulties with the open-ended question. From the Figure 2, most of the developer feels inconsistency of Git commands and options as troublesome. Also, developer with little experience feels difficulties undo/redo Git commands more than the developers with high experience. Other than the provided choice, "*Merging and resolve conflict*" and "*Steep learning curve*" was also reported.

From this findings, amount of the developer who feels Git as difficult does not relate with their developers experience,

[6] Invalid address such as twitter account or with the wrong format was excluded

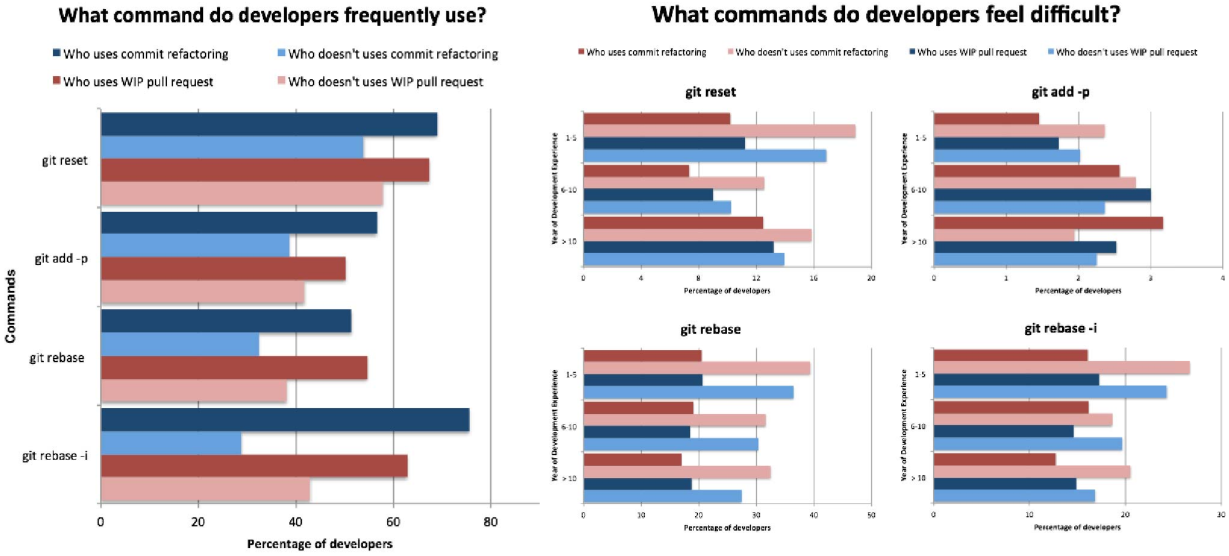

Fig. 4. Specific Git commands used by developers according to the use of unique practice

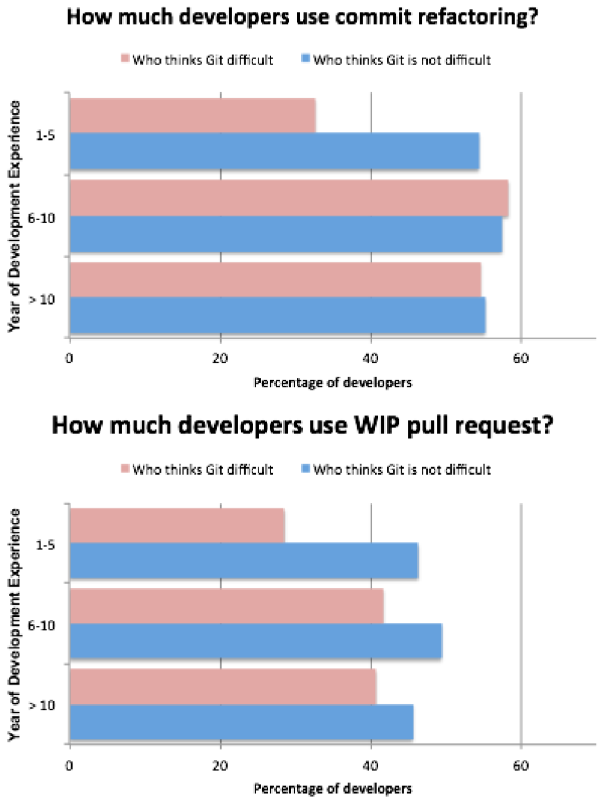

Fig. 3. Number of developers who use unique practices

but the influence factor of difficulties changes with their experience.

2) RQ1.2: How do developers feel with using GitHub and pull request?: Same as RQ1.1, we have asked the same question but this time about GitHub and pull request. Unlike from the result from RQ1.1, almost none of the developers felt GitHub and pull request as troublesome. From the open-ended question, "*Not all Git function are available in GitHub*" and "*Want to create a pull request from command*" was reported for the GitHub and "*Rebasing and resolve conflict*" for the pull request. These results show GitHub and pull request are user-friendly enough to developers. For more improvement to its usability, supporting GitHub on command line interface or unifying Git and GitHub could be conceivable.

B. RQ2: How unique practices are used in pull-based development?

The second research question is to examine how unique practices are used during pull-based development. Figure 3 shows that the results of differences on how much developers use WIP pull request and perform commit refactoring between who feel Git as difficult and who does not. The results show that both practices are used by 50% of the developers. Both have a similar result that who feel Git difficult and development experience with 1-5 years tend not to use these unique practices. For other year of experience, a developer who feels Git difficult performs commit refactoring as who feel not, but the number of using WIP pull request is slightly lower than the other. These suggest that encouraging the beginner to use WIP pull request and commit refactoring by supporting these unique practices. To support it, it is necessary to understand the usage and difficulty using Git on unique practices. For further investigation on what Git commands developers use and what affects the difficulties on unique practices are carried out in next research question.

C. RQ3: How is the command usage different for Git users who use unique practice?

In order to clear what Git commands are used and struggles developers for unique practices in pull-based development, we asked respondents a multiple choice question and open-ended question. Choices for the multiple choice question was selected from commands which should be used in those unique practices. Frequently used Git commands and commands which struggle developers are shown in Figure 4.

As we expected, developers who used unique practices tend to use selected commands more than who does not. Especially 'git rebase -i', the command to rewrite code history, had great differences between them. The right side of Figure 4 shows that the the number of developers who feel each command as troublesome. Except the 'git add -p' command, a developer who does not use unique practices feel difficulties and decreases as development experience increases. Supporting these commands to help performing unique practices is essential to leads developer to work efficiently in pull-based development.

VI. Conclusion

In this study, we examined how developers feel with pull-based development by conducting a large-scale survey of 1,552 developers on GitHub. Our main findings along with research questions are as follows:

- RQ1: *How do developers feel with pull-based development?*
  Developers feels GitHub and pull request more user-friendly than Git. From further examination, the inconsistency of Git commands and options and undo/redo Git commands are the influence factors of difficulties to developers.
- RQ2: *How unique practices are used in pull-based development?*
  Unique practices, such as WIP pull request and commit refactoring, were used by the half of respondents. A beginner who feels Git as difficult tends not to use both WIP pull request and commit refactoring. Support those beginners to use these practices is needed.
- RQ3: *How is the command usage different for Git users who use unique practice?*
  Commands selected for multiple choice question, especially 'git rebase -i', were used by the developers who use unique practices. Most of the commands, except 'git add -p', feel troublesome for developers who does not use unique practices. By supporting these commands, it may lead more developers to perform unique practices.

Researchers who design tools for supporting developers may use our findings to understand how developers feel with pull-based development. Specifically, tools for developers to support following the flow of pull-based development are needed. Since developers with little experience feel the WIP pull request and the commit refactoring difficult, supporting to simply perform those unique practices is important.

We are currently developing the supporting tool for pull-based development which reflects our findings. Our developing tool supports the WIP pull request, the commit refactoring and a smooth transition to operating Git in command line interface which are used for almost all developers.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Numbers 24700030 and 26730036.

References

[1] K. Muşlu, C. Bird, N. Nagappan, and J. Czerwonka, "Transition from centralized to decentralized version control systems: A case study on reasons, barriers, and outcomes," in *Proceedings of the 36th International Conference on Software Engineering*, ser. ICSE 2014, 2014, pp. 334–344.

[2] Y. Yu, H. Wang, G. Yin, and C. Ling, "Reviewer recommender of pull-requests in github," in *Software Maintenance and Evolution (ICSME), 2014 IEEE International Conference*, 2014, pp. 609–612.

[3] Y. Zhang, G. Yin, Y. Yu, and H. Wang, "A exploratory study of @-mention in github's pull-requests," in *Proceedings of the 2014 21st Asia-Pacific Software Engineering Conference - Volume 01*, ser. APSEC '14, 2014, pp. 343–350.

[4] G. Gousios, M. Pinzger, and A. v. Deursen, "An exploratory study of the pull-based software development model," in *Proceedings of the 36th International Conference on Software Engineering*, ser. ICSE 2014, 2014, pp. 345–355.

[5] E. T. Barr, C. Bird, P. C. Rigby, A. Hindle, D. M. German, and P. Devanbu, "Cohesive and isolated development with branches," in *Proceedings of the 15th International Conference on Fundamental Approaches to Software Engineering*, ser. FASE'12, 2012, pp. 316–331.

[6] S. Perez De Rosso and D. Jackson, "What's wrong with git?: A conceptual design analysis," in *Proceedings of the 2013 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming & Software*, ser. Onward! 2013, 2013, pp. 37–52.

[7] J. Tsay, L. Dabbish, and J. Herbsleb, "Influence of social and technical factors for evaluating contribution in github," in *Proceedings of the 36th International Conference on Software Engineering*, ser. ICSE 2014. New York, NY, USA: ACM, 2014, pp. 356–366.

[8] D. M. Soares, M. L. de Lima Júnior, L. Murta, and A. Plastino, "Acceptance factors of pull requests in open-source projects," in *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, ser. SAC '15, 2015, pp. 1541–1546.

[9] M. M. Rahman and C. K. Roy, "An insight into the pull requests of github," in *Proceedings of the 11th Working Conference on Mining Software Repositories*, ser. MSR 2014, 2014, pp. 364–367.

[10] G. Gousios, A. Zaidman, M.-A. Storey, and A. van Deursen, "Work practices and challenges in pull-based development: The integrators perspective," in *Proceedings of the 37th International Conference on Software Engineering*, ser. ICSE, 2015, pp. 358–368.

[11] M. Dias, A. Bacchelli, G. Gousios, D. Cassou, and S. Ducasse, "Untangling fine-grained code changes," *CoRR*, vol. abs/1502.06757, 2015.

[12] H. Kirinuki, Y. Higo, K. Hotta, and S. Kusumoto, "Hey! are you committing tangled changes?" in *Proceedings of the 22Nd International Conference on Program Comprehension*, ser. ICPC 2014, 2014, pp. 262–265.

[13] G. Gousios, "The ghtorrent dataset and tool suite," in *Proceedings of the 10th Working Conference on Mining Software Repositories*, ser. MSR '13, 2013, pp. 233–236.

[14] E. Kalliamvakou, G. Gousios, K. Blincoe, L. Singer, D. M. German, and D. Damian, "The promises and perils of mining github," in *Proceedings of the 11th Working Conference on Mining Software Repositories*, ser. MSR 2014, 2014, pp. 92–101.