

Pull Request Latency Explained: An Empirical Overview

An Extended Abstract of a Paper Published in Empirical Software Engineering

Xunhui Zhang
National University of Defense
Technology
Changsha, China
zhangxunhui@nudt.edu.cn

Yue Yu
National University of Defense
Technology
Changsha, China
yuyue@nudt.edu.cn

Tao Wang
National University of Defense
Technology
Changsha, China
taowang2005@nudt.edu.cn

Ayushi Rastogi
University of Groningen
The Netherlands
a.rastogi@rug.nl

Huaimin Wang
National University of Defense
Technology
Changsha, China
whm_w@163.com

1 EXTENDED ABSTRACT

Pull based development is a widely used paradigm of distributed software development that includes social coding communities including GitHub and GitLab. Because of the temporal and spatial asynchrony of the project participants in this model, the latency of pull requests is an important issue. For reviewers, understanding the latency of pull requests leads to a predictable development process and helps managers make plans and decisions. For contributors, the latency prediction can remind developers about the remaining time and accelerate the completion of pull requests, which can minimize the abandonment behavior from contributors. For the pull-based paradigm, latency covers the entire pull request lifecycle and is an important research area to grasp the pull-based development model as a whole.

There is a lack of work that systematically organizes the factors that affect pull request latency. Also, there is no related work discussing the differences and variations in characteristics in different scenarios and contexts. Building on a large-scale and diverse dataset, this paper conducts an empirical study on the impact of factors in different situations and contexts on the latency of pull requests. Notably, we explore the following two research questions:

RQ1 *How do factors influence pull request latency?*

RQ2 *How do the factors influencing pull request latency change with a change in context?*

Through systematic literature review (SLR), we found 17 papers presenting the factors related to the pull request latency. Then we extract 45 features from the studies together with the description of measurement, the influence direction and significance. Based on GHTorrent MySQL, Mongo data dump, GitHub API and cloned repository, we collected the extracted factors. After data preprocessing, we built mixed-effect linear regression models to explain the influence of factors in models and their relative relevance.

Our findings indicate that when submitting a pull request, the length of description and the size of source code change are the most influential factors. However, at close time, process related factors (e.g., *has_comments*) take over. For pull requests with comments or using CI tools, the comment related factors (e.g., *first_response_time*) and CI related factors (e.g., *ci_latency*) becomes important, respectively. The effect of factors on pull request latency is subject to

change with context. For example, the project's maturity is found to have a better explanatory role as it evolves.

Our work have implications for research and practice as follows:

- I1** We provide a list of recommended control factors when conducting research on pull request latency.
- I2** We considered the difference of factors in different states (submission and close time of pull requests), which provides opportunities for future research.
- I3** The delay of the first comment of the pull request has a very strong correlation with the delay of the overall latency. We think appropriate mechanisms should be devised to allow reviewers to focus on pull requests that have not been followed for a long time.
- I4** For pull requests reviewed by the same person, if the review time is much lower than the average time and no one else is involved in the review process of a self-approved pull request, project managers can be alerted to the quality of the merged code by adding a self-approved tag.
- I5** As the length of description strongly influences pull request latency, we argue that it's appropriate to reduce the complexity of a single pull request, and use associated pull request to refine the contribution granularity.

By summarizing the pull request latency related papers and factors, our work is helpful to follow-up research and practice. Our actionable suggestions for pull request contributors and reviewers can facilitate the review efficiency and the merge of contribution.

2 ORIGINAL JOURNAL PAPER

The original paper [1] has been accepted by **Empirical Software Engineering (ESE)** on 9th March 2022, and can be accessed at <https://link.springer.com/article/10.1007/s10664-022-10143-4>. The paper is not an extension of prior work, and has not been presented at other conferences.

REFERENCES

- [1] Xunhui Zhang, Yue Yu, Tao Wang, Ayushi Rastogi, and Huaimin Wang. 2022. Pull Request Latency Explained: An Empirical Overview. *Empirical Software Engineering* 27, 6 (July 2022), 126. <https://doi.org/10.1007/s10664-022-10143-4>