

# Qualitative Model of Open Source Code Review Practices

## Abstract

Code review is a core quality assurance practice in open source. However, it is unclear what code review practices are used in open source projects. In this thesis, we use a multi-case case study approach to study code review practices exercised in  $n$  open source projects that are of different age, use different review tooling, and provide software for different domains. We found distinctive variation points for code review and were able to distill the core characteristics of the functional open source code review practice. We induct a  $m$ -dimensional classification model of code review practices. {Despite these variation points, the same review process underlies all those open source projects. | Our theory presents  $l$  different code review practices. | Other results.}

## Related Work

### Method

- Yin, R. K. (2003). Introduction and designing case study. *Case Study Research Design and Methods*, 28. <https://doi.org/10.1017/CBO9780511803123.001>
- Mayring, P. (2014). Qualitative Content Analysis. *Empirical Methods for Bioethics: A Primer*, 11, 39–62. [https://doi.org/10.1016/S1479-3709\(07\)11003-7](https://doi.org/10.1016/S1479-3709(07)11003-7)

### Topic

- Rigby, P. C. (2011). Understanding Open Source Software Peer Review: Review Processes, Parameters and Statistical Models, and Underlying Behaviours and Mechanisms. *ProQuest Dissertations and Theses*, 194. Retrieved from [http://search.proquest.com.proxy1.ncu.edu/docview/898609390?accountid=28180%5Cnhttp://xt6nc6eu9q.search.serialssolutions.com/?ctx\\_ver=Z39.88-2004&ctx\\_enc=info:ofi/enc:UTF-8&rft\\_id=info:sid/ProQuest+Dissertations+%26+Theses+Global&rft\\_val\\_fmt=info:ofi/fmt:](http://search.proquest.com.proxy1.ncu.edu/docview/898609390?accountid=28180%5Cnhttp://xt6nc6eu9q.search.serialssolutions.com/?ctx_ver=Z39.88-2004&ctx_enc=info:ofi/enc:UTF-8&rft_id=info:sid/ProQuest+Dissertations+%26+Theses+Global&rft_val_fmt=info:ofi/fmt:)

- Rigby, P. C., & Bird, C. (2013). Convergent contemporary software peer review practices. *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2013*, 202. <https://doi.org/10.1145/2491411.2491444>
- Rigby, P. C., German, D. M., & Storey, M.-A. (2008). Open source software peer review practices. *Proceedings of the 13th International Conference on Software Engineering - ICSE '08*, 541. <https://doi.org/10.1145/1368088.1368162>
- McIntosh, S., Kamei, Y., Adams, B., & Hassan, A. E. (2014). The Impact of Code Review Coverage and Code Review Participation on Software Quality Categories and Subject Descriptors. *Proceedings of the 11th Working Conference on Mining Software Repositories (MSR 2014)*, 192–201. <https://doi.org/10.1145/2597073.2597076>
- Sadowski, C., Söderberg, E., Church, L., Sipko, M., & Bacchelli, A. (2018). Modern code review. *Proceedings of the 40th International Conference on Software Engineering Software Engineering in Practice - ICSE-SEIP '18*, 181–190. <https://doi.org/10.1145/3183519.3183525>
- Rigby, P. C., & Storey, M.-A. (2011). Understanding broadcast based peer review on open source software projects. *Proceeding of the 33rd International Conference on Software Engineering - ICSE '11*, 541. <https://doi.org/10.1145/1985793.1985867>
- Bahamdain, S. S. (2015). Open source software (OSS) quality assurance: A survey paper. *Procedia Computer Science*, 56(1), 459–464. <https://doi.org/10.1016/j.procs.2015.07.236>

## Sampling

- Tooling: Mail (Linux), Gerrit (Chromium, OpenStack), Phabricator (LLVM, FreeBSD), Pull request (React)
- Age: Linux Kernel (1992), Chromium (2008), OpenStack (2010), LLVM (2000), FreeBSD (1993), and React (2013).
- Domain: OS, infrastructure, browser, compiler, GUI framework