DEPARTMENT OF COMPUTER SCIENCE | LUNDS TEKNISKA HÖGSKOLA | PRESENTED 2019-08-23
MASTER THESIS Modeling and Analyzing Developer Collaboration to Guide Data Driven Decisions
STUDENTS Rasmus Hallevåg, Jesper Olsson
SUPERVISORS Lars Bendix (LTH), Bo Nyström (Praqma)
EXAMINER Ulf Asklund (LTH)

Analyzing developers collaboration through popular coordination tool, Git

POPULAR SCIENCE PAPER **Rasmus Hallevåg, Jesper Olsson**

Quantifying collaboration and relating it to the complexity of the code or the speed which work gets synchronized provides a foundation for decisions about collaboration. By examining how different ways how developers collaborate can impact performance it is possible to make intelligent decisions to how developers ought to collaborate.

In a software development context a high number of decisions have to be made. For many of these decisions project rely on the decisions of experts with high level of specific knowledge. The amount of time it takes for experts to understand the context of the problem and make a decision based on it can detract from the productivity of the expert. Even the use of experts might not always give satisfactory results, as even experts are human, and therefore subject to human biases and mistakes.

The main question is if team organization should follow the concept of collective code ownership where an entire team develops more or less together and collectively develop files for the system; or a follow a clearly divided responsibility where ownership is assigned and requests for changes are made. Creating an accurate representation of development structure and comparing it to performance metrics could help answer the debate and inform practices which would improve development. Two research questions have been formulated in this thesis to best explore this subject, namely how developer structure could be modeled and if there are some correlation between it and metrics relating to the speed of changes or complexity in the code.

Modern version control systems are used by developers to coordinate their work, one of the most popular ones is ”git”. Git stores large amounts of metadata about which developer added which change to which file, and it is possible to see when and how the code has evolved. A prototype gathering data from git, displaying representations of collaboration and relating it to the performance metrics was created. To answer the first question three models were explored but the main result came from the simplest model of just counting the amount of developers who had made any changes to a file. The other models were too complex to get a straight relationship while the problem with the amount of developers is that it does not say a lot about collaboration in the file.

To answer collaborations relationship with performance, having between 1 to 5 collaborators on a file showed similar results but having more than 5 seemed to lead to an increase in maximum complexity of a function in a file. The average complexity in a given file was similar across any amount of developers, but the most complicated parts of any file got more complex the more developers that were changing in the file. The speed of synchronizing their work was not impacted by the amount of developers or the division of labour in the file.

The end result also led to the development of a prototype that could be further developed and used for insight of a project.