Volume 2 Issue 2

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 732253.

Quality-Aware Rapid Software Development

From Dissemination Manager's pen

I am delighted to welcome you to our latest newsletter. There has been indeed a hectic year for Q-Rapids researchers in writing the number of publications, for use case providers in deploying the developed solutions and for the consortium organising events and preparing deliverables. Enjoy reading some of our recent activities and stay tuned for the next six months to come!

Best regards,

Sanja Aaramaa
Dissemination Manager

Experiences of Q-Rapids Solutions in the Industry with Case Bittium – Requirements Engineering Course in University of Oulu

Q-Rapids solution has been now running up for a while in Bittium. The approach has been suited into the Way of Working in our company. Q-Rapids provides tooling to optimize throughput without sacrificing quality, however the applied quality model must be determined in a manner that suits company´s operating environment.

As a use case provider for Q-Rapids Bittium (www.bittium.com) has looked for enhancing software development focused data analysis and fast feedback methods on functional and non-functional requirements. In addition, the purpose has been enhancing holistic view to system or product or operations level data mining and analysis regarding non-functional requirements. As an important point for implementation and piloting of Q-Rapids dashboards and tools to support decision making practices regarding non-functional requirements was in focus as well.

These experiences were shared with the students of Requirements Engineering in the University of Oulu provided with a practical application of one of the use cases of Bittium.

Figure 1 Veikko Rytivaara and Timo Räty presenting Q-Rapids in the RE course at the University of Oulu.

Important experience of Q-Rapids solution has been that alignment of the Bittium software development process and tool chain. Based on this, Q-Rapids solution is working well. To be able to reach this stage active collaboration and input for all Q-Rapids partners has been conducted supported with help of several workshops together with Q-Rapids researchers.

The pilot case with area of production test software development has been experimented and piloted successfully and Q-Rapids solution has now been taken into use to in another use case. At the same time roadmap for scaling Q-Rapids solution as part of development chain has been planned and the first steps have been taken into that direction already.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 732253.

Dockerization of the Q-Rapids to ease adoption

A docker is an open source project that aims to simplify the development, deployment and running of applications using isolated containers. It enables building an image with the application deployed inside. A running instance of an image is called a container. The image contains all the dependencies that the applications need to be executed (e.g. operating system, runtime environment, specific system libraries).

A docker has been selected as one of the deployment options for the Q-Rapids solution, starting from its consolidated version delivered in October 2018. Now, during the development of the final version of the Q-Rapids tool, the "dockerization" of the solution allows Q-Rapids researches more effective development, adding new services and capabilities, as well as provides a consistent environment for its deployment.

Finally, the Docker images can be easily run anywhere (on the variety of host operating systems) executing the application in an isolated environment. This enhances the capabilities to ease the adoption of the final solution by the future end-users. Such dockerization approach is successfully used by ITTI in commercial software products and projects to ease the adoption by customers.

Evaluation results are being used

In our second evaluation of the Q-Rapids components (i.e. the consolidated version) we again collected feedback from practitioners.

During the evaluation sessions, our Q-Rapids implementation has been introduced to 24 practitioners out of our four companies. They perceived the opportunity to use the tool by themselves and provided for us many insights that we are now using to further improve our Q-Rapids solution. The practitioners consider the current version of Q-Rapids as relevant, understandable and usable. However, we also collected useful suggestions for improvements, e.g. increase prediction capabilities. More details will be available in deliverable D5.4 once accepted by EU.

Our further activities will increase the appropriateness of Q-Rapids being accepted in industrial settings!

Name	Last commit	Last update
..		
cluster-dbrd	fixes	3 weeks ago
node-elasticsearch	Initial version	4 weeks ago
node-kibana	Initial version	4 weeks ago
node-postgres	Initial version	4 weeks ago
node-qralert	Initial version	4 weeks ago
node-qrconnect	Initial version	4 weeks ago
node-qreval	fixes	3 weeks ago
node-rbase	Initial version	4 weeks ago
node-tomcat	Initial version	4 weeks ago
tools/elasticsearch-kibana	Initial version	4 weeks ago
.gitignore	Initial version	4 weeks ago

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 732253.

Q-Rapids Process Metrics and its Evolution

Q-Rapids paves the way to exploit the abundance of data on a project, product, and runtime performance in Agile Software Development, to provide actionable insights into a company's software development quality. Process metrics is one part of the solution, which leverages the availability of data on a software development process to provide process-related insights.

Table 1 Process Metrics operationalized in Q-Rapids. The complete list of process metrics can be found from https://bit.ly/2CgFUeS.

Process Factors	Description	Process metrics example
Testing Performance	To measure testing phase performance aspects like execution time	Error Leakage, Feedback time from CI to developer, Number of iterations in code review phase for a commit, etc.
Issues' Velocity	To measure the capability to fulfil issues planned for a sprint	Throughput time for an issue, Average speed to resolve issue, Team throughput, etc.
Development Speed	To measure daily build progress	Commit review speed, Automation speed, etc.
Quality Issues' Specification	To measure amount of issues entering a backlog in an incomplete state	Percentage of issues completely specified
Delivery Performance	To measure the capability for on-time delivery, considering resource management	Percentage of features delivered on time during development cycle, Number of commits on core component in a given period, etc.
External Quality	To measure quality of a product from customer standpoint	Feedback from end users related to issues in a given period, Number of times end user calls to support team in a given period, etc.
Issues Estimation Accuracy	To measure difference between effort estimated and actual effort invested for an issue	Estimated effort of an issue/story point, Real invested effort of an issue/story point, Percentage of correct estimations, etc.

The *Table 1* presents the process metrics, categorized under *process factors*, which we developed for the Q-Rapids partner companies. *Process factors* are attributes of parts of the process that are concrete enough to be measured. Multiple process metrics can make up a single process factor.

Four companies in Q-Rapids exhibited somewhat similar needs for process metrics, but their approach to compute them varies, to the extent dictated by their individual development practices and the tools used. For example, process metrics related to *Testing Performance* and *Issues' Velocity* process factors were common across four companies, but due to differences in development practices and tools used, the computation of certain metrics differed. For instance, the process metric *Average speed to resolve issue* was computed differently for one of the partner companies because of their distinct development practices. Therefore, data availability and compatibility to an existing development process are two primary requirements for process metrics adoption. For further maximizing the potential of process metrics and the Q-Rapids solution, in general, some companies even started to make minor tweaks to their development practices to ensure data is available for metric computation. The set of process metrics considered in Q-Rapids and the lessons learned from the implementation of process metrics in the partner companies, were presented in a paper at the PROFES'18 conference.

In addition to data availability and process compatibility, successful process metrics adoption depends on its incremental and iterative implementation, and a company champion spreading the cause of such a measurement program across the organization. It is to the credit of a champion that Q-Rapids and process metrics, by extension, is being viewed as a solution to make quality in a software development process explicit, instead of it existing as a tacit knowledge among stakeholders. This transformation can be further aided by ensuring that the process metrics used to quantify a software development process are reliable. In this sense, being able to trust metrics data is what matters the most, as it has a larger bearing on the long-term adoption of a measurement program. These lessons, among other insights, about success factors for process metrics' long-term adoption will be published in a paper at the ICSSP'19 conference.