



# Quality-Aware Rapid Software Development



## From Project Coordinator's pen

Dear colleagues,

Almost three years back in time, we were starting the Q-Rapids project, and I was writing the foreword for the first project newsletter. Time flies and Q-Rapids is approaching its end.

During this time, we have experienced the typical vicissitudes that characterise this type of projects. We needed to match complementary research views, industry and academy partners needed to understand each other's priorities, and techniques didn't always behave as expected. Nevertheless, here we are, delivering an integrated Q-Rapids solution embracing software, processes, models, catalogues and so. As a coordinator, it has been rewarding to witness its increasing adoption and integration into the way of working in the consortium companies, uncovering new challenges on its way. We hope to be able to address some remaining challenges in future endeavours!

Best regards,

**Xavier Franch**  
Project Coordinator

## Q-Rapids Summative Evaluation

As part of the technology transfer activities in the project, the integrated Q-Rapids solution has been incrementally developed between academic and industrial partners to address their software quality problems. The summative evaluation has focused on exploring the value provided by the integrated Q-Rapids solution in pilot projects.

Practitioners emphasized major benefits including the improvement of product quality and process performance and an increased awareness of product readiness. They especially perceived the semi-automated functionality of generating quality requirements by the integrated Q-Rapids solution as the benefit with the highest impact and the most novel value for them. Practitioners can benefit from modern software analytics platforms, especially if they have time to adopt such a platform carefully and integrate it into their quality assurance activities.

If you are interested, you can read the preliminary results of the summative evaluation.<sup>1</sup>



*Figure 1 Conducting evaluation in case companies.*

<sup>1</sup> A. M. Vollmer, S. Martínez-Fernández, A. Bagnato, J. Partanen, L. López, and P. Rodríguez, Practical experiences and value of applying software analytics to manage quality, ACM/IEEE International

Symposium on Empirical Software Engineering and Measurement (ESEM). Industry Track, 2019.

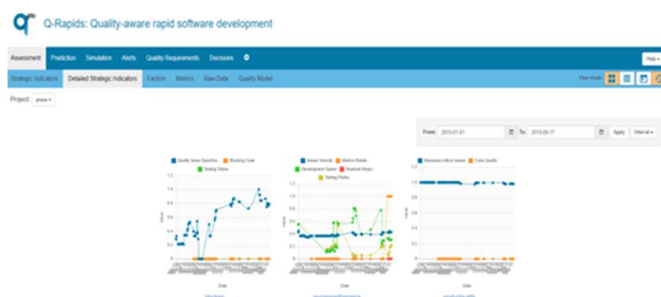


## Benefits of Q-Rapids solution highlighted in Bittium Prose use case

In the previous Q-Rapids newsletter Bittium told about second use case Prose, which was established with a larger set-up with altogether five implementations of Q-Rapids solution in the context of product information systems development. Based on the use case Bittium has been looking for enhancing system implementation to be able to do roll-out and exploit Q-Rapids for the use of products and projects in the company in a wide scale. As a result, third use case has been started in the area of Medical technologies of Bittium.

Bittium Product Services (PROSE) is a solution developed for the Device Life Cycle Management and it is used for example for managing devices and their software, handling maintenance and diagnostic activities, controlling manufacturing and production testing and test events in repair operations. PROSE also handles business analytics and intelligence information and operative level reporting of the solution. The solution contains user management for information authorization and for example the first secure deployment of devices, commissioning, is possible with PROSE solution.

The whole team in the pilot 2 consists of a product owner, project manager and some 15 developers. In addition, there have been several stakeholders following up the success of the implementation. The PROSE use case is using Q-Rapids dashboards in the daily scrum meetings where e.g. new alerts are checked and root causes are clarified. During the sprint planning days needed actions are decided and quality stories are taken to sprint. With Q-rapids Prose, is following and analysing the project quality standards including aspects of implementation quality, data capture in various aspects and for example how testing is working.



As a reminder the following factors are used for the integrated Q-Rapids solution for the follow-up and prediction of the performance:

- *Feature throughput; the number of features that meet a time to market target with the targeted levels of quality.*
- *Release frequency; intervals and number of the releases as target increased amount of releases per time unit.*
- *Realized requirements; the number of overall realized requirements referring to quality requirements that are used in actual features and releases.*
- *Product quality; product quality aspects referring to the "ilities" like the maintainability, reliability, and functional suitability of the product being developed.*
- *Process performance; process performance referring to the efficiency and quality of the used software development life-cycle approach.*

The Prose use case results with the Q-Rapids show the possibilities based on using Q-Rapids solution to find out the seeds for improved performance. "We are happy users of Q-Rapids solution to find out the performance gaps in our several development targets in the Prose development. I am delighted to recommend to use Q-Rapids for similar purposes, says Milla Ahola, SW project manager of Bittium Prose solutions".



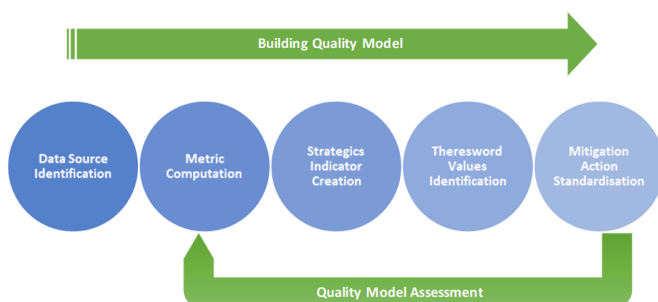
Members from Prose Use Case, together with Product Owner in the left and Project Manager Milla Ahola in the right together



## Building an accurate Quality Model in Five Steps, an essential stage in tooling quality management processes with Q-Rapids

To standardise the way how the quality is managed in its development process by integrating quality managements tools like Q-Rapids, Softeam has built an actionable and accurate Quality Model. The goal of this Quality Model is to determines which quality characteristics, among those that we were able to collect in our organization, will be considered when evaluating the properties of our software product.

The approach followed by Softeam to build its Quality Model required five steps: the identification of available data sources, the normalisation of data, the computation of metrics, the specification of high-level strategic indicators, the identification of Threshold Values for this metrics and indicators and finally the standardisation of mitigation action to implement to solve quality issues.



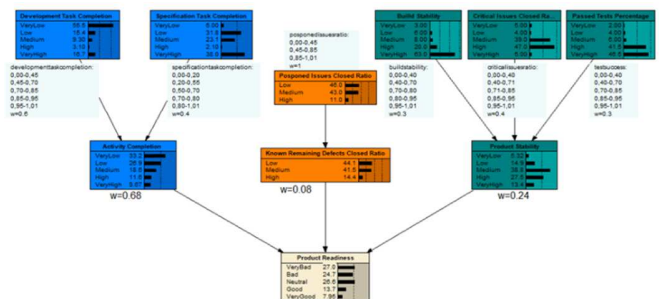
**Data Source Identification:** Specify a theoretical Quality Model based on data which are not available on your organisation is a waste of time. Thus, our first task was to make a list of available data. We started by making the inventory of all the applications used in a development process. A method to proceed to this identification could follow a Model Based Enterprise Architecture approach.

**Metric Computation:** Next we worked on the specifications of metrics which will be computed from this identified row data. This step required collecting requirements from the future end-users of the platforms (project manager, managers, quality engineers) to define the metrics which are useful in the organization.

**High Level Indicator Creation:** The metrics specified in previous step is easy to understand and trustworthy by a specific employee (they are close to her daily work) but

not very useful for management purpose and are specific to a business domain. Defining high level indicators which aggregate various low-level metrics is a way to provide an overview of the status of a product or process for managers.

**Threshold Values Identification:** We have now specified how to compute low level metrics and high-level indicators, but these measurements are difficult to understand for the end-user. In order to facilitate the adoption of the quality model, it is required to translate the numerical values of these indicators into an easy to understand format. Our solution was to create a Bayesian Network which leads to a much more adapted assessment model compared with a weighted average for the strategic indicator.



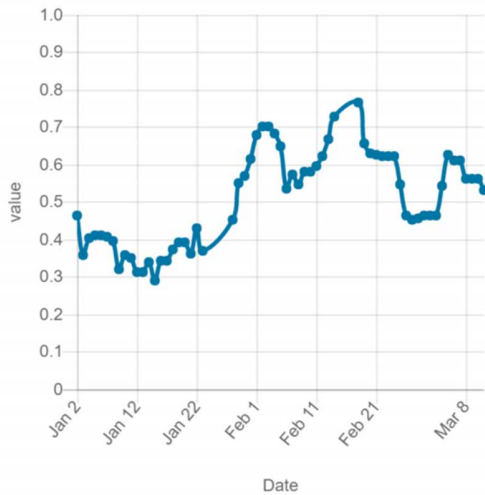
**Mitigation Action Specification:** Finally, it would be required to anticipate in advance the action to be taken in the case of identification of quality issues, characterized by the crossing of the threshold values. This requires the creation and implementation of a mitigating action catalogue by the company when these quality problems are detected.

## The practical use of Q-Rapids process metrics: ITTI example

The results and progress in the process metrics are discussed weekly in the Scrum Team meetings. Those meetings are always held on Tuesdays (the sprint in ITTI is from Tuesday to Tuesday) after lunch/brunch. Each Scrum Team meeting consists of the following parts: review, retrospective and planning (the next sprint). The process metrics is being displayed and discussed at retrospective. The process metrics is a great fit since this part of the meeting is devoted to people, processes, tools, lessons learnt and how to improve the way of working. At each Scrum Team meeting, from 15 to 20



minutes can be spent on the analysis of the process metrics.



"Density of estimated dev. tickets [last 7 days]"

Figure 2 Figure 1: Density of dev. tickets with estimated effort (time spent).

Of course, the role of the Product Owner and Scrum Master is to make those discussions and displays interesting, but developers usually like statistics and some trends/ graphs, such as those used at ITTI.

Those results, trends and metrics values are used to motivate the team and improve the process, and to find the problems in order to resolve them.

## Experiences of the Q-Rapids solution in the Telecommunications industry with Case Nokia

During the past three years, Nokia initiated two separate use cases where the integrated Q-rapids software solution has been deployed. Third year of the project, Nokia has focused on setting up the Q-Rapids solution for the second use case (UC2) – Productization Tribe. The UC2 has been completely different to the first one as the focus has not only been in improving software quality, but gaining visibility to the overall quality of products, which include hardware and software elements. Giving the distinct nature of the UC2, basically nothing could be reused from the UC1 deployment for the UC2 context. Even though UC2 team included two use case champions (involved in all the activities during the project), the team faced several challenges during the implementation.

Addressing the challenges has required collaboration across business units and functions internally as well as with Q-Rapids project organisations.

The deployment of the integrated Q-Rapids solution in the UC2 started by defining relevant strategic indicators, factors and metrics using the Quamoco method. However, the method turned out to be practically impossible to apply without external facilitators who were not available at this time of the project. Thus, the scientifically proven approach was replaced with a “trial and error” type of expert knowledge approach. In practice this led to situation where the factors, metrics and computations had to be adjusted several times along the way. In addition, many of the challenges were related to the inflexible architecture of the Q-Rapids solution, as e.g. hard coded connectors are used instead of configurable ones. Also, the vast expert knowledge required from various areas, such as Java coding, Kafka, and RD tools used in different parts of the development process slowed down the deployment process.

Even though the deployment of the Q-Rapids solution in UC2 has advanced to a level where the dashboard and other supporting features are “up and running”, a lot of work is to be done to reach the maturity of the solution so that it could be used in decision making for example in regards quality or business. Several aspects of the Q-Rapids solution, such as simulation and prediction capabilities, still need to be further developed and validated through various pilot projects. Finally, the advanced capabilities, such as Machine Learning (ML) and Artificial Intelligence (AI) yet remains to be seen in the Q-Rapids solution. These types of capabilities are the ones that decision-makers across industries seek to provide support for advised decision making in the future.

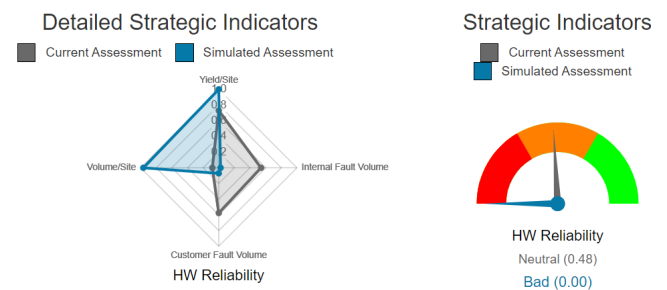


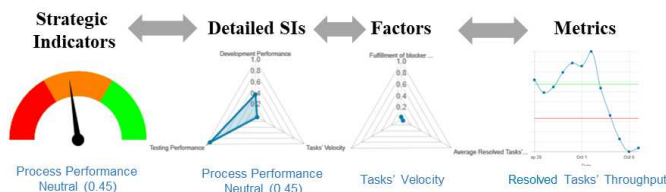
Figure 3 Simulation feature on indicator level.



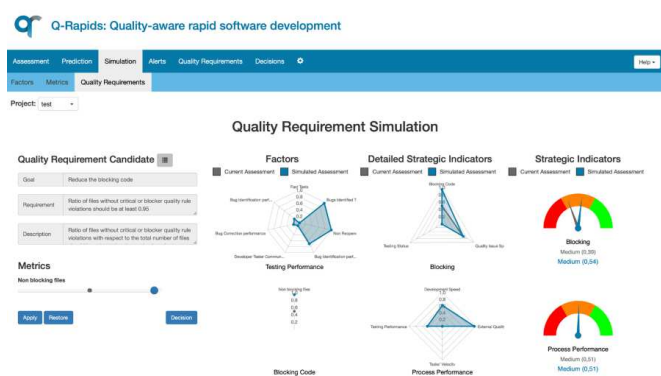


## QaSD: Quality-aware Strategic Dashboard

As part of the Q-Rapids Tool, we provide Quality-aware Strategic Dashboard tool (QaSD). QaSD is, a modular, configurable and extensible dashboard supporting decision makers in ASD to improve their software development processes and product.



QaSD includes the following features: (a) *configuring and assessing* strategic indicators (SI) to evaluate high-level characteristics that provide data-driven evidence to support decisions; (b) *SI assessment visualisation*, QaSD includes 16 views (graphical and textual) offering different visualisations for the assessment of SI and quality factors used for their evaluation, these include views in metrics used to compute the quality factors; (c) *navigation capabilities to drill-down from SI to metrics* to provide a decision-maker the rationale behind the SI assessment; (d) *SI assessment forecasting*, to visualise predicted assessment to evaluate the need for taking some quality-related decision; (e) *what-if analysis*, allowing the analysis of scenarios to explore possible improvements before taking a decision; and (f) *quality requirements (QR) semi-automatic generation*, QaSD suggests to a decision-maker a QR based on low assessment values.



One of the distinguishing features of QaSD is the mentioned QR generation, this feature includes the

simulation of the impact of a concrete QR as part of the development backlog. As a result of this simulation, a decision-maker can see which metrics are impacted by the QR, and how this impact is propagated to the SIs assessment.

QaSD software components are released as Open Source available in GitHub<sup>2</sup> as well as the Users' Guide<sup>3</sup> and including links to video tutorials<sup>4</sup> describing the project assessment visualization).

## The Q-Rapids software development process

The work towards the Q-Rapids software development process started in WP2 at the beginning of the project by studying the software development processes of the use case projects. Those processes together formed the baseline of the Q-Rapids software development process.

It turned out that all use case projects utilised agile methodology – not necessarily by the book but adjusted to the case company's business model, products, size, customers, and many other company specific characteristics. The diversity of the companies, which at the beginning looked like an obstacle to defining a generic process model, turned out to be a source of possibilities.

As a result, three layered process model was defined. The work done in sprints at team level was covered by the Sprints layer. The Releases layer contains a planning frontend and a validating backend serves as an umbrella for the Sprints layer. The top most planning and validating level, the Products layer, expands the model up to full products. All the process layers implement continuous, agile, rapid, and incremental ways of working – though with different paces.

The findings from the actual usage of Q-Rapids in the case companies were considered by broadening the deployment scope of Q-Rapids from the Sprints layer to the Releases and Products layers – the Q-Rapids software development process encourages the deployment of agile methods and ways of working throughout the whole organization.

<sup>2</sup> <https://github.com/q-rapids/grapids-dashboard>

<sup>3</sup> <https://github.com/q-rapids/grapids-dashboard/wiki/User-Guide>

<sup>4</sup> [https://www.youtube.com/watch?v=zKxr-FLfG\\_k](https://www.youtube.com/watch?v=zKxr-FLfG_k)