Recommendations to Achieve Software Product Quality and Process Improvements

Ž. Antolić
R&D Center
Ericsson Nikola Tesla d.d.
Complete Address: Krapinska 45, Zagreb, HR-10000, Croatia
Phone: +385 1 365 4584 Fax: +385 1 365 4082 E-mail: zeljko.antolic@ericsson.com

Abstract - This paper gives an overview of 15 recommendations that can be taken within software development organization in order to achieve software product quality and process improvements. Each recommendation is described and discussed according to activities performed in Ericsson R&D Center in Croatia. In addition, few examples of process improvement deployment activities performed in software development projects, including achieved results, are also described in the paper. Some proposals and future directions in the area of software product quality and improvements visibility are given.

I. INTRODUCTION

In the area of software development, despite achieved results with various quality improvement approaches, the IT industry is still far from the goal of zero-defect software. The software product quality is very complex issue. It is commonly recognized today that building quality software is fundamentally hard work. Software development usually takes longer that it was originally planned. There is always more work that remains to be done. Waiting for the high quality software usually means later appearance on the market, and for development company it means not to be a leader. Oppositely, to be fast on market, but at a low product quality, means lower customer satisfaction.

Managing improvements is the key to achieve operational excellence and product quality. The improvement management process, a support process within the business process framework, is organizational way of managing improvements in a consistent, effective and uniform way and governed by the requirements of the company strategies and targets.

In this article, the software quality assurance definition and basic activities are described. In addition, the role of Operational Development and Quality function within one software development organization is described. Operational Development and Quality management are parallel, and mutually supporting activities within software design organization. Based on experiences from Ericsson R&D Croatia, the 15 recommendation for product quality and process improvements are described. Some successful implementations of improvements (Block RCA process, Code review tool) are described as examples of quality and improvements management. At the end of the paper, visibility of improvements (including improvements handling tool), current status and goals for the future are discussed.

II. Software Quality Assurance - Definition

Quality assurance comprises all planned and systematic activities implemented within the quality system. A clearly documented and fully established quality framework will increase confidence regarding the project's ability to perform and deliver result according to expectations and relevant quality standards. The purpose of quality assurance is to achieve high quality on both project outcome by assuring high-quality project performance, and to ensure continuous learning and enhanced performance within the organization and its ongoing and future projects.

Quality assurance is about implementing the quality assurance activities. All operative processes and support activities defined in the project quality plan should be adhered to. To ensure that the project quality framework provides the support needed in the project, it should be continuously evaluated, and suggestions for improvements made [1].

Quality assurance activities are performed throughout the project and include reviews, audits and assessments, and monitoring of both results achieved in the project and of the project performance. All planned quality assurance activities should be specified in the Project Specification. Important quality assurance activities are:

- Project milestone reviews;
- Tollgate assessments;
- Project audits;
- Project performance monitoring and measurements, such as reporting of project performance indicators;
- Project evaluation.

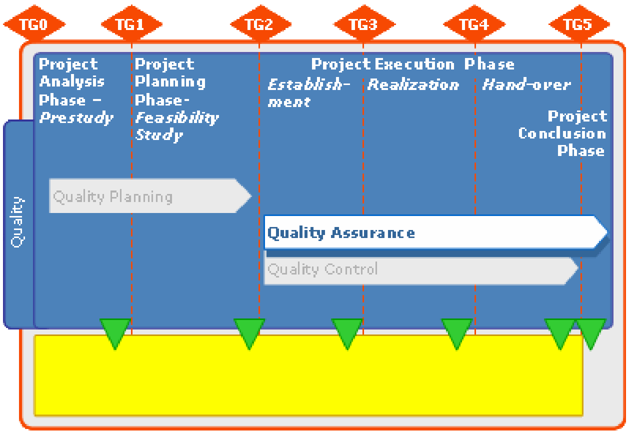

Fig. 1. Quality Assurance in Projects

III. Operational Development and Quality Function

Operational Development and Quality (OD&Q) function within software development organization has responsibility for continuous improvement of the management system, software product quality, and development processes [2].

The OD&Q management activities are the following:

- Driving the software development organization towards operational excellence in the defined subject areas of Operational Development, Quality Management, and Security & Risk Management;
- Responsibility for Management System, Assessment, Certification, Target setting & Performance management, Process Management, Improvements, Efficiency and Quality within organization;
- Defining organizational targets and ensuring that targets will be met;
- Analysis of business environment and stakeholders' requirements, definition of targets, performance indicators, and measurements;
- Check and review of target status, preventive and corrective actions to meet defined target, review target specifications when needed;
- Driving the selected subject areas for organization with a common approach towards specified targets;
- Consolidation of critical issues and risks from reviews, assessments, measurements and own observations for decisions by the management;
- Driving of improvements and ensuring the alignment of improvement programs to strategies inside the organization;
- Guiding managers and employees in the Organization with respect to OD, Quality, and Security & Risk management in the daily operations;
- Active participation in the quality and improvements networks;
- Sharing of knowledge, experiences, and good practices.

The OD&Q subject areas are shown on the Fig. 2.

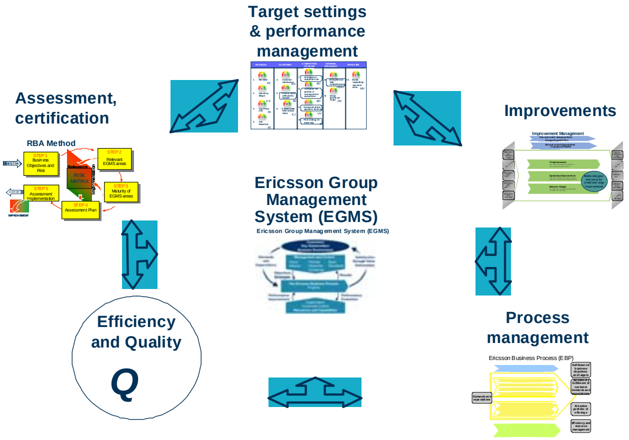
Target settings & performance management
Assessment, certification
RBA Method
Improvements
Ericsson Group Management System (EGMS)
Ericsson Group Management System (EGMS)
Efficiency and Quality
Q
Process management
Ericsson Business Process (EBP)

Fig. 2. OD&Q Subject Areas

IV. Quality and Process Improvements Steps

A. Management Commitment

Organizational commitment to quality goals starts always with management commitment. Management provides vision and organizational goals, including quality goals as mandatory. They are based on the previously achieved results and wanted position. During that process, management should take care that goals are realistic, and about required improvements that should be implemented in order to achieve the goals. Possible quality goals for development projects are: quality rank, fault density, fault slip through, review and inspection efficiency, code review coverage, number of re-deliveries, etc.

Project goals and product quality improvements are possible to achieve with development team that has good competencies, and authority given by the management to perform quality assurance activities according to organizational quality policy. This is very important because serious improvements are not possible to achieve if active management support does not exists.

Management has to be continuously involved in quality issues in the organization. It is responsible for continuous tracking of development processes usage and product quality parameters. If process or product quality deviations occur, management is responsible to define set of preventive and corrective actions in order to improve the situation. Management is also responsible for the quality assessment at the end of the project in order to check if product is compliant to quality goals set at the beginning of the project. Development project can not be finished if only functionality is implemented within required time and budget. Product quality is one of the key parameters to judge if development project has successfully finished [3].

B. Quality Goals and Business Objectives

Business objectives for software development organization usually are defined as goals related to functional content of delivery, project budget, and lead time for delivery. The quality is very often skipped during the goal setting process, taken as granted, something that is already built into product. In some projects, when quality goals are not defined explicitly as business goals, when optimization regarding project execution time and budget is needed, project is asked to skip some quality assurance activities directly connected to the product quality, wrongly assuming that these activities are not value adding. In any case, quality assurance activities in the projects should not be skipped or cancelled without management approval. In cases when management approves such exemptions, project must prepare action plan when and who will finish postponed quality assurance activities. Management involvement must be secured for tracking of this action plan and fulfillment of quality assurance activities [3].

C. Quality and Customer Requirements

When software development project defines customer requirements, it usually defines only functional requirements. Quality related customer requirements, if exist, usually are analyzed separately from functional requirements. But from customer point of view, quality is satisfying the requirements and showing that the product will work in the intended environment by the intended users [3].

This should be taken into consideration in the project when doing test strategy preparation. Beside function test with goal to verify product functionality, and system test with goal to check system performances, testing phase should include simulation of end user environment, and check user guides in a way how customer will use them.

D. Competence and Development Process

Quality policy deployment and performing of quality assurance activities in the project are impossible without competent resources. Only well educated project members can perform product development tasks, and have awareness about importance of product quality at the same time. In order that project executes development tasks according to project plan, project members should have required competencies. If specific or missing competencies are required, project should prepare education plan for its members, and reserve time and budget for training.

Continuous competence building is organizational management task that has to be planned and performed if development organization wants to achieve expected product quality level. In addition, investments in competence building when organization and projects have problems with time plans and budgets are extremely important for long-term survival. When the good times come back, software developers will be competent and ready for the new challenges.

Each software development organization has to define competence development plans for its members, based on long-term objectives and strategies. These plans should be realized with high precision, in order to eliminate chaotic approach and secure needed competence level of the organization right on time.

E. Deviation Analysis

When software development organization has quality problems during project execution and after product delivery, very often mistake is blaming everyone else for poor performance or low quality. It is important during project execution to collect data about performance, compare them with similar projects within own organization, and perform external benchmarking. Project should be encouraged to exchange and analyze data together with colleagues from other projects, and make conclusions.

When problems are recognized in the project, Root Cause Analysis (RCA) has to be performed. The common mistake during analysis could be looking for explanations, excuses, and blaming someone else for problems. Focus of RCA should be finding of corrective and improvement actions. By performing these actions, project will mitigate damage, and prevent the same mistakes in the future. Management is responsible to initiate RCAs, give all needed support, and follow up corrective actions in the projects. Results of RCAs are also learning points. They should be implemented in all future projects in order not to repeat the same mistakes again.

F. Project Audits

Project audits are very often considered as external activities with intention to find something wrong in the project, and source of unplanned and unwanted actions. Projects, sometimes even management, have negative attitude towards project audits. The fact is that project audits should be positive and constructive process. The goal of project audit is to help project management to regain control or to ensure it is maintained. In addition, project audits are very efficient management tool to check quality management system in the organization and ongoing projects. Project audits can point out problems in process deployment or product development. In any case, project audits help management to find possibilities and implement improvements in the projects that will help to finish project according to the project plan.

The common management mistake is that complete responsibility for project audits is delegated to quality assurance manager or QA group in the organization. Management should be the most interested in project audits and its results because management plan and order audits in the projects, and use results to control quality assurance activities, product quality, and work on process improvements.

G. Reviews and Inspections

The experience shows that most efficient defect removal is in the early phases of development. The sooner we find defect, the cost of removal is lower. The most effective verification method in the early phases of development is code review.

The common mistake that happen in development projects with execution time and budget problems is cost and time reduction of all activities which are not coding or function testing. Reviews and inspections are skipped first. Initially, project can save some time and budget, but at the same time project lose possibility to find faults before start of the next project phase. Finally, it can result with faults slipped through the customer and low product quality.

Inspection plan in the project should not be changed (reduced) without management approval and clear plan how and when remaining activities will be performed. Experience from the finished projects and from the industry shows that most successful projects were those with consistently performed reviews and inspections.

Analysis of the project plans from the past shows that in general time reserved for code reviews is not sufficient. In addition, time reserved for testing is about the same as time reserved for design. Knowing that fault removal cost is much lower in design phase than in testing, we have proposed one improvement to the project plan in MicroCPP project recently. Testing phase was reduced by 10% and this time was reserved for code reviews and design documentation inspections. The final results are not available yet, but at the moment (project is in function testing phase) we can notice that number of faults found in testing is significantly lower (approx. 30%), compared to previous similar project.

H. Built-in Quality

Testing is probably the most important and critical quality assurance activity, but it is not only one. It is not enough to put focus only on testing in the project. Testing as project phase comes after product specification, design, and integration. Many quality assurance activities have to be finished before that point in the project, because testing only has to confirm what have been done in previous phases. The common mistake in software development organizations when they have product with quality problems is that they are trying to increase quality only with increased testing scope. Although that method can give some results, in most of the cases it is not enough. Additional testing can not increase the quality of bad designed product. When RCA discovers that low quality is result of bad design, the solution of the problem is to improve software architecture and implementation. Better results regarding product quality can be achieved if product is redesigned and code is completely re-written. After that, additional testing can confirm product quality increase. In order to minimize testing and related costs, project should put more focus on early quality assurance activities (built-in quality) in the project like design documentation inspections, code reviews, solution presentations to other project members, design base analysis, etc. Also, before product in its life-cycle is transferred from design to maintenance, maintainability analysis and product quality assessment has to be performed by system or product responsible person in the organization.

I. Cost of Rework

Cost of rework is any cost related to the product maintenance after design project is finished, including cost of people, equipment, integration, packaging, and delivery. After any fault correction in the product or analysis of problem reported by customer, feedback to other design projects must be secured. In that way organization can minimize the risk of repeating the same faults again in the future. It can significantly minimize the cost of rework in the next period. Another problem that can be noticed in the organizations working on software products maintenance is introduction of the new faults while correcting others. Some experiences from the organizations with low competence level of the maintenance teams, and with limited quality assurance activities show that introduction of the new faults rate can reach 50%. It means that on each two corrected faults, one new fault is introduced. If maintenance team performance indicators show that trend, RCA has to be performed. In the most cases, the solution is to increase team members' competence, and perform additional code reviews in order to discover new faults and side-effects before delivery to the customer. Software faults introduced in the product maintenance process are the most expensive faults because they are widely deployed, and they are resulting with customer dissatisfaction.

J. Continuous Improvements

Competitive software development organization should never be happy with existing situation and results. Business performance indicators may show that goals are achieved, but everything can be improved. Organization should constantly challenge way of thinking and its ideas, based on experience, information, and environment where exists today [4]. It should constantly try to improve even if those improvements appear to be very small to others. Many small improvements can result in big cost savings. But, the most important achievement is that organization becomes innovative environment, where all members continuously work on improvements. That kind of organization has secured competitive position on the market.

K. Measurements as Support to Business Objectives

One approach to support business objectives by measurement is Balanced Scorecard (BSC) with financial, customer, competitive position, internal efficiency, and employee perspectives. Each of the perspectives has the goals with measurable key performance indicators. Each goal has responsible person with authority to define and perform preventive and corrective actions in the organization that will help to achieve the goal. Goal responsible is usually member of the management, directly influencing on the results and improvements. Balanced Scorecard gives possibility of simple presentation of the achieved results, giving the situation overview for each target perspective.

Management sets BSC goals based on organization vision, mission, and business objectives. When goals are defined, goal driven operational plan can be prepared. Management also performs analysis of the results and defines corrective actions if needed.

L. Search for Alternative Solutions

Projects with focus only on getting functionality developed usually have problems with budget overrun, missed schedules, low quality, and unsatisfied customers. It is always recommended to explore alternative solutions in case of problems that include balance between project budget, time schedule, product quality, and product performances. If requirements are unrealistic, project should communicate on time that expectations are impossible. In many cases is better to delay delivery instead of insisting on delivery of not finished product with low quality and performances, causing customer dissatisfaction. If quality improvements are necessary for customer satisfaction, project should be re-planned with additional quality assurance activities.

M. Development Process Improvements

Software development process improvements require coordination and involvement on all levels of management, designers, and testers. It is not the job only of the QA group or the Software Engineering Process Group (SEPG) to make quality and process improvements. The most useful suggestions and proposals are coming from the people who are actually using the processes in their daily work. The role of SEPG is to collect ideas, analyze them from technical and business perspective, prepare for introduction, piloting and results analysis, and in case of success deployment in all projects in organization.

All organization members must be encouraged to be innovative and propose improvements. Many small ideas could be more beneficial than one big. To support innovativeness in the organization, appropriate rewarding system has to be implemented [4].

N. Manage and Reward Cooperation

Cooperation between team members in project is crucial for success and product quality. While individual excellence is still needed and can be rewarded, it should be offered to support the teams. Many projects have focus on ‘firefighters’, people capable to jump in when problem occurs, investigate cause of the problem, and find solution. Although these people are very important and irreplaceable, often very well rewarded, organization can lose the focus on good performing teams, members who correctly manage their tasks without problems for project execution. We have to promote and reward the best performing teams with no delays and no ‘firefighting’ actions. The experience shows that this kind of teams usually delivers the complete functionality on time with highest quality.

O. Promote Process Improvements

Process improvement is not just about improving a technical processes, it is also about improving the company’s culture and the people’s processes. Processes help to establish business culture in the company [4].

Processes should be promoted as soon as they are proposed. Software development organization should try to involve all members in process evaluation and work on final solution. People should be encouraged to discuss about improvements proposals and give their contribution to process deployment. While promoting new or improved processes, organization should use all possibilities for communication like, web, e-mail, meetings, and other collaboration tools.

V. Improvements Examples

A. Block RCA Process

Based on recommendations (C, E, H, I, M) from chapter IV, we have developed new process to improve product quality for finished products – Block RCA Process [5].

The purpose of the process is to measure number of faults per SW Block, select products that need quality improvements, perform RCA on selected products, and plan and execute product quality improvements.

The sub-processes/activities from which the process is consisted of are the following (see Fig. 3):

- Measurement is performed by product responsible each quarter. The result is reported to Subsystem responsible and line manager who is keeping measurement record for the subsystem. Measurement records on the subsystem level are analyzed by OD&Q manager and line manager.
- If number of faults exceeded limit defined by Subsystem responsible, RCA has to be performed by product responsible, and result sent to Subsystem responsible and line manager.
- Subsystem responsible and product responsible perform RCA Review and decide if product has quality problems or not. The report is sent to line manager.
- If product has quality problems, Subsystem responsible and products responsible prepare Improvement Plan proposal and send it to Improvement Forum for decision (GO / No GO decision).
- Actions from the Improvement Plan are implemented with resources and within the budget and time frame approved by Unit Management Team.
- Improvement implementation is verified with resources and within the budget and time frame approved by Unit Management Team.

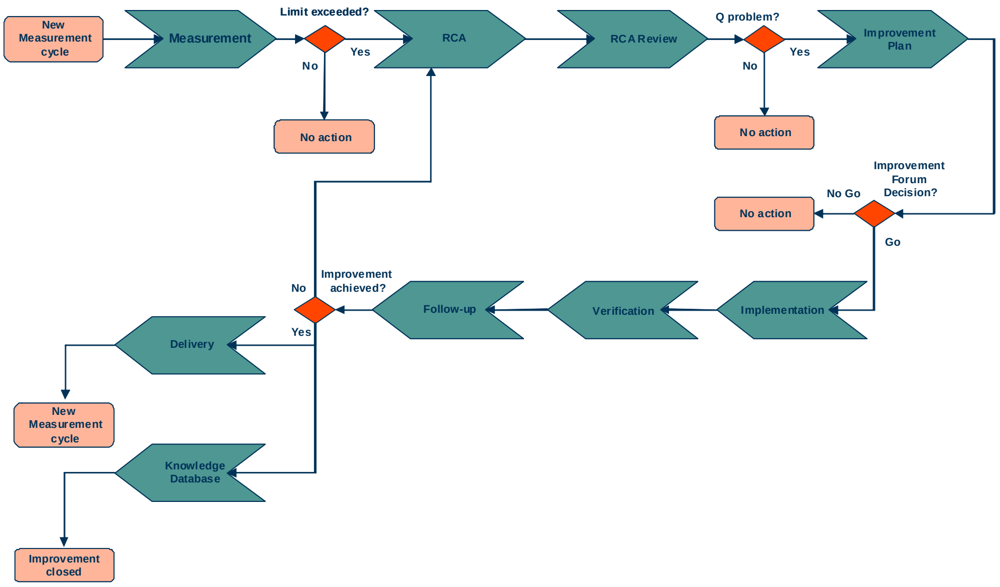

Fig. 3. Block RCA Process

- Product responsible is following up closely product quality, and report to Subsystem responsible and line manager if defined improvement is achieved or not.
- Finished improvement is delivered as updated product, and new measurement cycle can start.
- Lessons learned (type of the faults, code improvements, test improvements, configuration improvements, and competence improvements) are documented by Subsystem responsible and stored for future use.

B. Code Review Improvements

Recent project audit in one of the development projects in Ericsson R&D Croatia showed that designers currently perform code and model reviews in a rather inefficient way. There is unclear review baseline, little support for identifying portions of code/model that should be reviewed, most reviews are conducted on paper, and review findings are typically not logged consistently in proper tools. The code reviews are not visible because project does not track review activities, when schedules are aggressive, reviews are skipped, and there is no way to measure the efficiency of reviews.

Based on these results of project audit, and recommendations (F, G, H, I, J) from chapter IV, we have decided to introduce new tool for code review support.

Some of the achieved benefits are:

- Reviews are performed using the electronic version of the artifact;
- Statistical reports on review data;
- The tool automatically keeps track via meta-data;
- Anomalies found outside of reviews and unfixed anomalies found in previous reviews are automatically included when creating a new review of the artifact;
- Wizards help users find review items, e.g. find all lines of code that have not been reviewed for a component;
- Automated meeting request creation. Scheduling data is always in synch between tool and Outlook Calendar;
- Review item/anomaly location (e.g. file and line number) and version (e.g. from the file version tree) are automatically managed by the tool;
- Review status and suggested/postponed anomalies are shown in the file version tree.

VI. Visibility of Improvements

Analysis of the current status of improvement work in our organization resulted with following findings:

- Process improvements and audits are deployed thorough project without clear management involvement and responsibility;
- System Improvement Board not fully working in all units;
- Identified problems not always covered by implemented corrective actions;
- People are focused on their area of responsibility, defending mode regarding improvements;
- Improvement work is not visible for stakeholders;
- Pro-activity in SW engineering process improvements not visible.

Based on analysis results, we have defined desired state that we want to achieve in 1-2 years period:

- Improvement Program established with purpose to create awareness of improvement programs that will change way of working in most disciplines;
- System Improvement Board established and working in each unit;
- The Process Requirement Specification and implementation plan prepared and deployed;
- R&D Quality Door Board established;
- SW engineering process improvement roadmap for 1-2 years in place.

In addition, we will start with implementation of the new improvements handling tool (IdeaBox) based on EriColl (Ericsson Collaboration Platform). All organization members are tool users. Users will have opportunity to submit ideas, and give comments and ratings for ideas submitted by other users. Domain experts will receive alert when idea comes related to their domain of expertise, and they have obligation to review and give feedback about idea. Innovation managers are responsible for idea deployment in the organization through local improvement processes, and rewarding of idea submitters.

VIII. Conclusion

Software development organizations that want to stay competitive on the market should have strong focus on quality issues and process improvements. This paper gives some recommendations how to achieve quality and process improvements, as well as some examples from the practice. Only high quality products and organizational ability to work on continuous improvements can guarantee customer satisfaction and loyalty.

References

[1] ***, “iPROPS – Ericsson Project management method for R&D projects, rev. R3B”, Ericsson, Stockholm, 2007.

[2] ***, “R&D Center Croatia Operational Description – Operational Development & Quality Management”, Ericsson, Zagreb, 2008.

[3] J. Evans, “How Quality Assurance Helps with the Delivery of Projects”, SEPG Conference, Munchen, 2008.

[4] R. Bechtold, “12 Proven Principles for Process Improvement & Organizational Success”, SEPG Conference, Munchen, 2008.

[5] Ž. Antolić, “Block RCA Process”, Internal Ericsson documentation, Zagreb, 2009.