IMPROVEMENT PRODUCTION PROCESS BY INSPECTIONS CMMI

Florina-Cristina FILIP
Transilvania University of Brasov, Romania

Abstract. Peer reviews are important instances of verification and quality assurance activities and they shall be conducted early in the development life cycle to identify defects and problems as early as possible in order to prevent them from being delivered to the customers. An inspection is a formal variant of conducting peer reviews on work products. Inspections are conducted on work products by the author's peers to identify defects for removal and to recommend other changes that are needed. These can be conducted via "formal inspection", "lean inspection" or "pass-around-review", which are all collegial review methods. Alternatively, it is possible to conduct peer reviews in a less formal way as "walk-through" process.

Keywords: inspection, process, review, decision, method

1. Introduction

In recent years, software development companies and departments follow software process methodology (SPM) including ISO9000, Capability Maturity Model (CMM) and Capability Maturity Model Integrated (CMMI) for improving competitiveness and quality [1]. Moreover, many researches investigate the impact of software process improvement on organizations performance and the situations [1, 2, 3].

Capability Maturity Model Integration (CMMI) is a process improvement framework for the development of products and services [4, 5]. It consists of practices that address development and maintenance activities for the entire product life cycle, from conception through delivery and maintenance [5, 6]. It also contains practices to be implemented at the organizational and project levels for purposes of institutionalization and improvement [6].

The essence of the CMMI movement is the idea of a Capability Maturity Model (CMM). The basic idea is that companies that are more systematic and process oriented do better than those that don't have any underlying or repeatable processes [4]. CMMI is the successor to CMM and is consistent with the international standard ISO/IEC 15504 [7].

CMM frameworks describe a set of practices or stages that companies work through as they become more proficient. The CMMI framework attempts to combine and become the integrated successor for three earlier CMM frameworks: the Capability Maturity Model for Software (SW-CMM), the Systems Engineering Capability Model (SECM), and the Integrated Product Development Capability Maturity Model (IPD-CMM) [4].

The CMMI product suite consists of process improvement models, appraisal methods, and training materials. The CMMI process improvement framework integrated three source models into a single framework. The current CMMI models accommodate multiple disciplines to support enterprise-wide process improvement and yield the benefits of an integrated model [8]. CMMI models are not processes or process descriptions [9].

CMMI was developed to provide a single model to be used by organizations pursuing enterprise-wide process improvement. It provides needed guidance for integrating systems and software development activities. It supports the coordination of multidiscipline activities that are or may be required to successfully build a project [10].

The CMMI describes the steps an organization has to take to advance its systems and software engineering processes from an initial maturity level to a managed and eventually quantitatively managed level. Thus, among the CMMI goals were these: reduce the cost of implementing process improvements when several disciplines have to be considered, enhance understanding, eliminate duplication and inconsistencies among the models, develop common components, and ensure consistency with the emerging ISO TR 15504 standard [11].

Today, CMMI is an application of the principles introduced almost a century ago to this never-ending cycle of process improvement. The value of this process improvement approach has been confirmed over time. Organizations have experienced increased productivity and quality, improved cycle time, and more accurate and predictable schedules and budgets [9, 12].

2. Process description

2.1. Process roles and stakeholders

Within this process the following roles are involved:

- author - the originator (developer) of the material to be inspected, must not be the review leader;
- review leader - plans, coordinates and controls all reviewers' activities, decides in case of conflicts (e.g. classification of an issue as a defect), responsible for a target-oriented and successful management/ moderation of the review, must not be the author;
- reviewer - a person conducting examination of the review object;
- recorder - person who records or logs the issues and other items during the review meeting;
- project leader - responsible for managing a dedicated project.

The process "inspections" can be invoked at virtually any point in the development process. Generally speaking, any work product can undergo a review.

As far as effort and resources for reviews have to be provided within the project, this has to be regarded in the project planning process.

2.2. Inputs and outputs process, entry and exit criteria

Inputs process includes planning documentation (effort and resources), work product to be reviewed and organizational review strategy (as part of the verification strategy).

Outputs process include review schedule and agenda, review comments from each reviewer, inspection / review protocol including list of issues (with closure information), documented decision on acceptance / non acceptance of review object, reworked review object.

Entry criteria: review method is selected and objectives for review are defined, work product is available for review, a review leader has been assigned for the review.

Exit criteria: the reviewers have performed an appropriately prepared review meeting, the reviewers have agreed on issues, the issues have been processed as agreed, and the reworked review object is accepted/not accepted.

3. Process activities and flowchart

3.1. Review preparation

Entry criteria: review method is selected and objectives for review are defined, work product is available for review, a review leader has been assigned for the review.

Inputs: planning (effort and resources) and review object.

Activity steps: check entry criteria for the review object (e.g. completeness, required maturity, no obvious defects), partition review object to manageable parts, decide on necessity for kick-off meeting, unless otherwise specified by local review strategy, nominate the reviewers and schedule review session(s), distribute the review object to the reviewers.

Templates/checklists: no templates/checklists defined.

Responsible: review leader.

Stakeholder: project leader, reviewer and author.

Outputs: review schedule and agenda.

Exit criteria: reviewers are nominated and available, review object is distributed.

3.2. Kick-off (team preparation)

Entry criteria: review process is scheduled, reviewers are nominated and available, review object is distributed, kick-off meeting considered necessary.

Inputs: review schedule and agenda.

Activity steps: recapitulate review schedule including use of applicable checklists, assign specific roles and responsibilities (e.g. for perspective based reading), the review leader or the author gives an introduction to the contents of the work product.

Templates/checklists: inspection/review protocol.

Responsible: review leader.

Stakeholder: reviewer and author.

Outputs: partially filled inspection/review protocol.

Exit criteria: reviewers are prepared appropriately.

3.3. Checking (individual preparation)

Entry criteria: review object is distributed.

Inputs: review object, organisational review strategy, inspection/review protocol.

Activity steps: conduct checking according to organisational review strategy, project specifications or review-specific definitions given in the kick-off-meeting, mandatory for "formal inspection" and "pass-around": document issues and comments detected during checking.

Templates/checklists: checklist for review object.

Responsible: reviewer.

Stakeholder: review leader.

Outputs: review comments from each reviewer, complemented inspection / review protocol.

Exit criteria: individual checking completed as planned.

3.4. Review meeting

Entry criteria: reviewers are appropriately prepared for the review meeting.

Inputs: review object, review comments from each reviewer, inspection/review protocol.

Activity steps: in case of "formal inspection" and "pass-around": analyse and clarify issues, in case of "lean inspection": systematically examine the review object (e.g. line-by-line) and identify issues, agree on one formulation of the issue. The reviewers have to decide on the review result (accepted, not accepted, accepted conditionally) and in case of "not accepted" initiate new review session, and in case of "accepted conditionally" define follow-up proceeding for verification of rework and closing issues; document the issues and complete the review protocol.

Templates/checklists: no templates/checklists defined.

Responsible: review leader.

Stakeholder: author, recorder, reviewer, project leader.

Outputs: completed inspection/review protocol, decision of the reviewers

Exit criteria: review object has been reviewed and agreed documentation of issues.

3.5. Rework

Entry criteria: decision for rework (not accepted or accepted conditionally).

Inputs: inspection / review protocol.

Activity steps: rework the review object as documented and agreed on during the review meeting, document additional changes or rework deviating from meeting decisions.

Templates/checklists: no templates/checklists defined.

Responsible: author.

Stakeholder: review leader.

Outputs: reworked review object.

Exit criteria: all issues have been processed.

3.6. Verification

Entry criteria: all issues have been processed.

Inputs: inspection / review protocol, reworked review object.

Activity steps: verify that the issues assigned in the meeting have been processed, decide on acceptance/non acceptance of rework and document the decision, update review protocol.

Templates/checklists: no templates/checklists defined.

Responsible: review leader.

Stakeholder: reviewer, author and project leader.

Outputs: updated inspection/review protocol, documented decision on acceptance/non acceptance of review object

Exit criteria: reworked review object is accepted/not accepted.

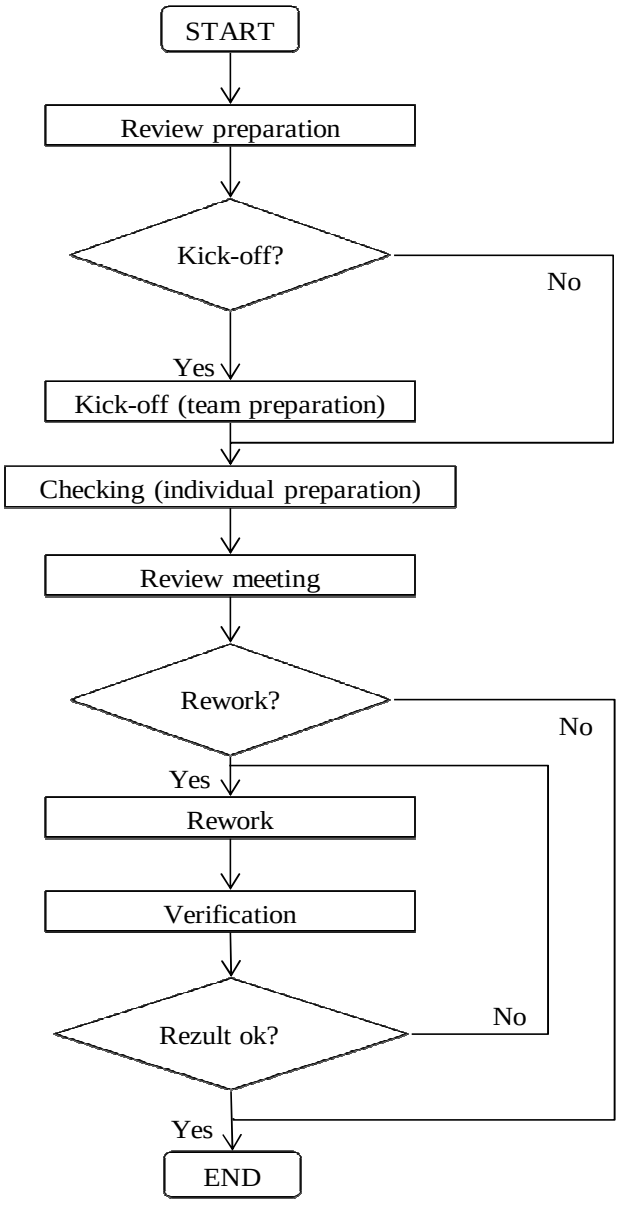

Figure 1. Process activities flowchart

4. Conclusion

As far as effort and resources for reviews have to be provided within the project, this has to be regarded in the project planning process.

There are the following variants of the inspection process:

- formal inspection, characterized by 3 to 6 peers (minimum author of the work product and 2 reviewers) participate in a formal inspection, a separate kick-off meeting is optional but recommended and author must not act as review leader;
- lean inspection, characterized by at minimum 1 reviewer apart from the author required (recommended: author of the work product and 2 reviewers), a separate kick-off meeting is optional and usually dispensable, individual checking of the work product to be inspected is always recommended, but not enforced by the lean inspection method (in this case, the activity "review meeting" is used for the joint checking of the review object);
- pass-around review, characterized by pass-around is used for non-joint checking and evaluation of a review object (typically to conduct reviews with remote developers or reviewers), the review object is sent by the author to the reviewers via e-mail or published in a shared directory on a fileserver, minimum 2 reviewers (author and one additional reviewer), maximum number in theory unlimited (recommended: not more than 10 participants), method formality can be regulated by prescriptions (time, effort, perspective, standards and checklists).

The lean inspection is derived from formal inspection by application of the following rules:

- activity "checking": the activity step "document issues and comments detected during checking" is dropped;
- activity "review meeting": the activity step "analyse and clarify issues" is replaced by "systematically examine the review object (e.g. line-by-line) and identify issues".

As far as the rules for pass-around reviews are allowed by local definitions and policies, the following rules can be applied:

- the author of the review object might also be review leader;
- the activity "review meeting" might be replaced by consolidation of review feedback and further enquiries of the author/review Leader; the respective decisions on the review object (accepted/not accepted) is part of each reviewers feedback and has to be consolidated by the author/review leader, too;
- there might be no further "verification" activity.

Especially the first and last options differ from conventional definitions of inspection methodology.

Generally speaking, any work product can undergo a review and the inspection process can be invoked at virtually any point in the development process.

References

1. Cheng, C.H., Chang, J.R., Kuo, C.Y. (2011) *A CMMI appraisal support system based on a fuzzy quantitative benchmarks model.* Expert Systems with Applications, Vol. 38, No. 4, p. 4550–4558, ISSN 0957-4174
2. Ashrafi, N. (2003) *The impact of software process improvement on quality: In theory and practice.* Information & Management, Vol. 40, no. 7, p. 677–690
3. Li Eldon, Y., Chen, H.G., Lee, T.S. (2002) *Software process management of top companies in Taiwan: A comparative study*. Total Quality Management, Vol. 13, No. 5, p. 701–713
4. Chrissis, M.B., Konrad, M., Shrum, S. (2003) *CMMI: Guidelines for Process Integration and Product Improvement*, Publisher Addison-Wesley Professional, ISBN 0-321-71150-5
5. Wangenheim, C.G., Silva, D.A., Buglione, L., Scheidt, R., Prikladnicki, R. (2010) *Best practice fusion of CMMI-DEV v1.2 (PP, PMC, SAM) and PMBOK 2008*. Information and Software Technology, Vol. 52, No. 7, p. 749–757, ISSN 0950-5849
6. Tamura, S. (2009) *Integrating CMMI and TSP/PSP: Using TSP Data to Create Process Performance Models*, Publisher Carnegie Mellon University, Software Engineering Institute, ISBN 0-321-71150-5
7. Staples, M., Niazi, M., Jeffery, R., Abrahams, A., Byatt, P., Murphy, R. (2006) *An exploratory study of why organizations do not adopt CMMI*. The Journal of Systems and Software, Vol. 80, No. 6, p. 883–895, ISSN 0164-1212
8. Huang, S.J., Han, W.M. (2006) *Selection priority of process areas based on CMMI continuous representation*. Information & Management, Vol. 43, No. 3, p. 297–307
9. CMMI Product Team (2010) *CMMI for Development. Improving processes for developing better products and services*, Publisher Carnegie Mellon University, Software Engineering Institute
10. Kasse, T. (2004) *Practical Insight into CMMI*, Publisher Artech House, ISBN 1-58053-625-5, Norwood, MA, USA
11. Mutafelija, B., Stromberg, H. (2003) *Systematic Process Improvement using ISO 9001:2000 and the CMMI*, Publisher Artech House, ISBN 1-58053-487-2, Norwood, MA, USA
12. Gibson, D.L., Goldenson, D.R., Kost, K. (2006) *Performance Results of CMMI-Based Process Improvement*. Publisher Carnegie Mellon University, Software Engineering Institute

Received in January 2013