Looking for inspection improvements through base practices

Ilkka Tervonen, Juha Iisakka and Lasse Harjumaa
Department of Information Processing Science,
University of Oulu,
P.O. Box 3000, FIN-90014 University of Oulu, Finland
Tel: +358 8 553 1908, Fax: +358 8 553 1890, E-mail: ilkka.tervonen@oulu.fi

Abstract

Inspection process improvement is typically a non-rational process, in which the improvement suggestions come from randomly arranged brainstorming meetings. The improvement ideas focus mainly on updating of checklists, adoption of new forms and definition of new metrics for measurement data. These are important and usually the most relevant improvement topics, but we should strive for a more rigid process which pays attention to case-specific improvement measures. In this paper we introduce a capability model tailored especially to inspection process evaluation that looks for relevant improvement suggestions through base practices. The first experiments reported in the paper are promising, and the evaluation process discovered improvement ideas which were also agreed on by the company concerned.

1. INTRODUCTION

Inspection process improvement is seldom a rational process in which the weak points of the inspection are really analysed. Instead, improvement suggestions typically come from randomly arranged brainstorming meetings and focus on updating of checklists, implementation of new forms and the definition of new metrics for measurement data. We do not claim that these are poor improvement topics, but we should strive for a more rigid process which pays attention to the improvement measures that are most relevant to the inspection process.

The idea of using base practices as a basis for analysis comes from the tradition of maturity and capability models as exemplified by the Capability Maturity Model (CMM) [12], SPICE [2] and Bootstrap [11] in particular. Improvement measures in the testing area are also considered in the context of models such as the Testing Maturity Model (TMM) [1], Testability Maturity Model [4] and Test Process Improvement Model (TPI) [10]. The problem is that the inspection viewpoint has not attracted much attention in these models, but has typically remained a part of a static analysis technique and/or is used as a part of a more general review process. Due to these shortcomings, and based on existing capability models, we introduce here a new capability model tailored especially to inspection process evaluation and the search for relevant improvement suggestions.

Although we focus on the inspection process, we assume that any company which aims at improving this inspection has already defined the whole software development process at an appropriate level. The need for this check comes from our own experiments as well as from reported experiences and suggestions regarding the use of capability models. It is largely accepted that plain inspection process improvement without any definition or understanding of the whole development process will not be successful.

The main structure of the tailored capability model comes from Bootstrap, in addition to which we look for the base practices of the inspection process and evaluate the grade of each practice in the company/department/project (whether it is in use totally or partially). In the first phase we do not evaluate how well the practice is carried out, as this will be addressed later, during further evaluation cycles. The evaluation is based on indicators, which can be enabling or verifying ones. The idea is that, if we find indicators of a base practice, we then have justifications for the existence of that base practice. Because the major focus in this paper is on looking for improvement suggestions, the loosely implemented base practices are regarded as guiding the improvement initiatives. Some general improvement ideas and key tips gathered from the inspection literature act as a pool of improvement suggestions which will be used as such or merely as triggers for modified ones.

In this paper we will first introduce the software inspection process with base practices, and then explain some of the base practices and indicators in more detail. After that the matrix of indicators and some general improvement ideas and key tips will be presented, and finally some evaluation experiences and case-specific improvement ideas will be reported.

2. BASE PRACTICES IN THE INSPECTION PROCESS

Inspection is traditionally defined in terms of steps such as entry, planning, kickoff meeting, individual inspection, logging (inspection) meeting, edit, follow up, exit and release [3],[6]. The structure of the ideal process of inspection which we use as a reference model in evaluation is based on these steps.

According to the Bootstrap model the ideal process is defined as a set of base practices. Our interpretation of these base practices differs from that of the Bootstrap model, however, because we also include the organisational and supporting activities among them. There are six defined goals which have guided the discovery of base practices: (1) to identify defects in an artefact, (2) to estimate the quality of an artefact, (3) to improve product quality, (4) to provide data for process improvement, (5) to provide the means for knowledge transfer, and (6) to improve the effectiveness of the development process. The base practices are classified into three sets (cf. Figure 1): supporting activities (at the bottom), which help in carrying out an instance of the inspection process, the core set of activities (one level upwards) which are the essence of the inspection process implementation, and organisational activities, which ensure continuous improvement and efficient organisation of the inspection process. The supporting activities are "Support with computer tools", "Maintain rules and checklists" and "Refine information". The core set of base practices (corresponding to the Bootstrap style base practices) include the activities "Check the preconditions for inspection", "Plan the inspection", "Find issues in the artefact", "Categorise defects", "Make corrections" and "Conclude the inspection", while the organisational set consists of the rest of the activities, such as "Organise the inspection", Train the participants" and "Establish and improve the inspection process".

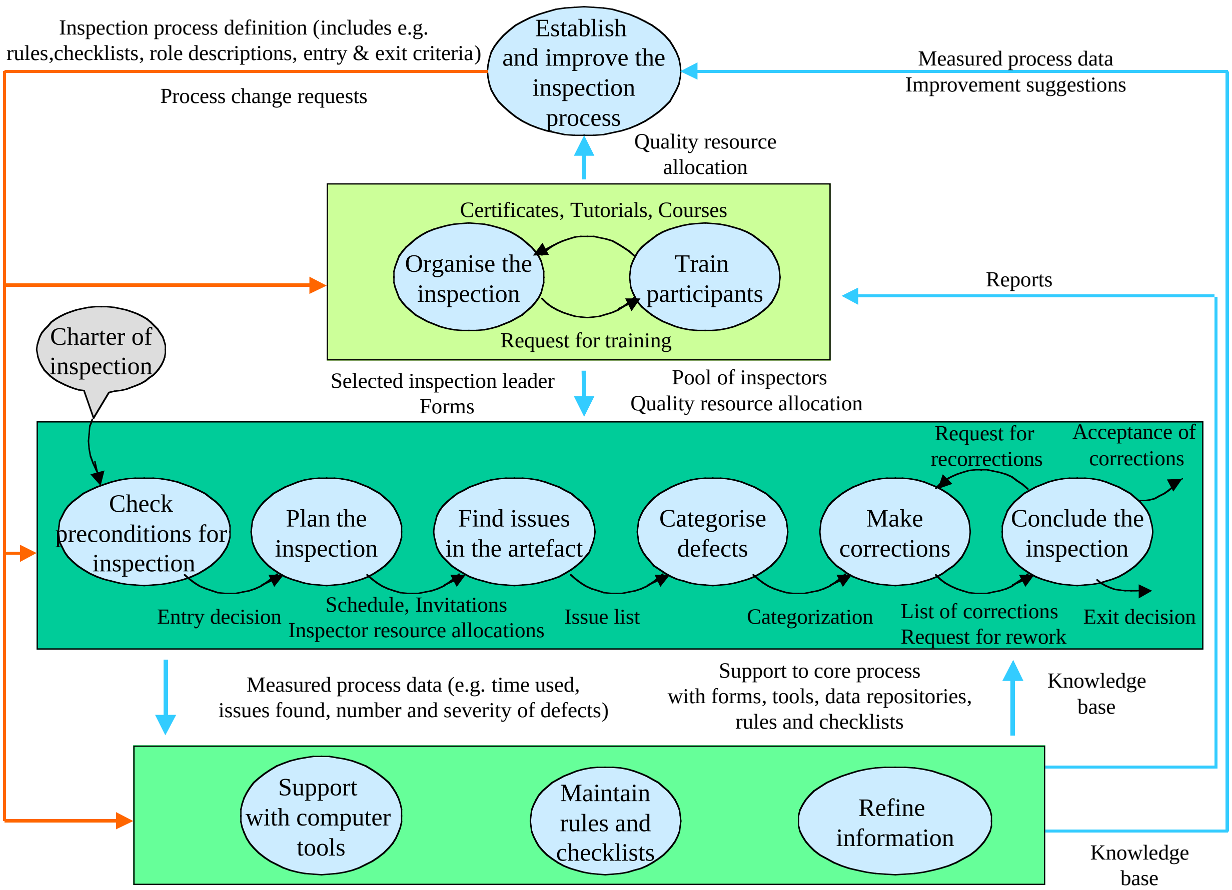

Figure 1. The base practices of the inspection process

Figure 1 also depicts work products and entities as an input to or output from the base practices. According to the Bootstrap model, these are used as enabling and verifying indicators by means of which the existence of a base practice is evaluated. The tailored capability model, base practices and indicators are defined in detail by the i3GO research group and will be available on its home page [8] after some further experiments. In this paper we introduce the principles embodied in these by means of an example. We have chosen the base practice "Find issues in the artefact" for this purpose, because it is one of the core activities and serves all six goals. The indicators of the base practice are depicted in the table below:

Table 1. Enabling and verifying indicators

Possible enabling indicators	Possible verifying indicators
Forms Rules Checklists Schedule Inspection resource allocation	Measured process data Issue list

All the indicators will be available on the home page of the i3GO group [8]. Checklists, for example, are characterised as follows:

- checklists are tools to be used by inspectors; lists of topics which should be checked in a specific artefact
- checklists are derived from rules
- checklists should be tailored to specific baselines and roles, for example
- checklists should be up to date
- focus on the discovery of major defects

In summary, we can say that base practices and indicators form the skeleton of our capability model. The indicators are evaluated and the existence (and grade) of the base practices can be analysed and justified in the light of the results. This will be explained in the following chapters.

3. HOW TO USE THE CAPABILITY MODEL

The capability evaluation is based on checking of the enabling and verifying indicators. An enabling indicator confirms that the preconditions for a specific base practice are met, and the absence of an enabling indicator strongly suggests that the corresponding base practice does not exist. A verifying indicator confirms that a specific base practice has produced appropriate and adequate results, and its existence suggests that the corresponding base practice may exist, but does not guarantee this.

The matrix for capability evaluation is presented on the next page. There are a total of 33 indicators to be walked through with the company's staff. The symbols used are the following:

& = enabling indicator
€ = verifying indicator

The existence of an indicator is evaluated with a five-grade evaluation

na = not relevant to the organisation
0 = not in existence at all
1 = exists to some degree
2 = exists fairly well
3 = always in full existence

As we can see from the matrix, one indicator has effects on a number of base practices. The white boxes of the matrix are filled with grades of existence, and the average value for each base practice is then estimated and written into the first row. Finally, the total estimate of inspection capability (0-3) is calculated.

Fulfilment of base activities													
Total value of inspection capability	Estimate	P.1 Improve the inspection process	P.2 Organise the inspection	P.3 Train participants	P.4 Support with computer tools	P.5 Maintain rules and checklists	P.6 Refine information	P.7 Check preconditions for inspection	P.8 Plan the inspection	P.9 Find issues in the artefact	P.10 Categorise defects	P.11 Make corrections	P.12 Conclude the inspection
1) Measured process data		&					&			€			
2) Improvement suggestions		&					€						
3) Process change requests		€			&	&							
4) Inspection process definition		€	&		&								
5) Reports			&				€						
6) Pool of inspectors			€										
7) Selected inspection leader			€										
8) Request for training			€	&									
9) Quality resource allocations			€										
10) Forms		€	&€		&€				&	&	&		
11) Tutorials				€									
12) Courses				€									
13) Certificates				€									
14) Tools					€								
15) Data repositories					€								
16) Rules						&€			&	&	&		&
17) Role descriptions						&							
18) Checklists						€			&	&			
19) Issue list							&			€	&	&	
20) Categorisation							&				€	&	
21) Knowledge base		&					€				&		
22) Charter for inspection								&					
23) Entry decision								€					
24) Entry criteria								&					
25) Schedule									€	&			
26) Invitations									€				
27) Inspection resource allocations									€	&			
28) Request for rework on another artefact											€		
29) List of corrections												€	
30) Acceptance of corrections													€
31) Request for recorrection													€
32) Exit decision													€
33) Exit criteria													&

Figure 2. Capability evaluation matrix

The result of a capability evaluation in an IT company is presented in Figure 3. The company has about 130 employees in Finland and our estimate represents the status of inspection in the whole company.

TOTAL: 2-3	Estimate	P.1 Improve the inspection process	P.2 Organise the inspection	P.3 Train participants	P.4 Support with computer tools	P.5 Maintain rules and checklists	P.6 Refine information	P.7 Check preconditions for inspection	P.8 Plan the inspection	P.9 Find issues in the artefact	P.10 Categorise defects	P.11 Make corrections	P.12 Conclude the inspection
Fulfilment of base activities		3	2	3	1	1	1	2	3	2	2	3	3
1) Measured process data	1	&					&			€			
2) Improvement suggestions	3	&					€						
3) Process change requests	3	€			&	&							
4) Inspection process definition	2	€	&		&								
5) Reports	0		&				€						
6) Pool of inspectors	1		€										
7) Selected inspection leader	2		€										
8) Request for training	3		€	&									
9) Quality resource allocations	2		€										
10) Forms	3	€	&/€		&/€				&	&	&		
11) Tutorials	3			€									
12) Courses	3			€									
13) Certificates	2			€									
14) Tools	1				€								
15) Data repositories	0				€								
16) Rules	3					&/€			&	&	&		&
17) Role descriptions	2					&							
18) Checklists	0					€			&	&			
19) Issue list	3						&			€	&	&	
20) Categorisation	1						&				€	&	
21) Knowledge base	1	&					€				&		
22) Charter for inspection	3							&					
23) Entry decision	3							€					
24) Entry criteria	1							&					
25) Schedule	3								€	&			
26) Invitations	3								€				
27) Inspection resource allocations	1								€	&			
28) Request for rework on another artefact	3										€		
29) List of corrections	3											€	
30) Acceptance of corrections	3												€
31) Request for recorrection	3												€
32) Exit decision	2												€
33) Exit criteria	1												&

Figure 3. Capability evaluation in company A

The grey tone indicates the degree of existence of indicators in the company's process and the fulfilment of base practices. In this case the total estimate of inspection capability was between 2 and 3.

4. SOME GENERAL SUGGESTIONS FOR INSPECTION IMPROVEMENTS

Researchers in the field of software inspection have been discussing the requirements for a reorganised inspection process [7], [9], focusing especially on reducing the number of logging meetings. We have presented in a previous paper [7] some new ways to organise an inspection meeting. A formal, strict inspection meeting is highly efficient, and unfortunately part of this efficiency may be lost when using the new approaches. However, companies must sometimes make compromises between efficiency and the costs of inspection and use virtual inspection or pair inspection, for example. *A virtual inspection* may be implemented at the same time but in different places, for example, or at different times and in different places. This makes reconciliation of the participants' timetables easier. A logging meeting may even be unnecessary when the comments from the other inspectors regarding the discussion between the author and an inspector are irrelevant. Web technology provides the base solution for place and/or time independence that virtual inspection needs in order to manage distributed inspectors and their comments. The conventional inspection has 4-8 participants. What if we cannot gather so many together? *Pair inspection* has the minimum number, that is, two members - the author and an inspector who is checking the author's document.

In addition, there are some general suggestions for improvements presented in the area of software inspection. Gilb [5], for example, introduces 24 key tips for improving the inspection process, and classifies them into eight groups: inspection strategy, entry conditions, planning, individual checking, the logging meeting, process brainstorming, exit conditions and inspection statistics. Some examples of the key tips should be mentioned in the present connection:

Inspection strategy (2 out of 5):

* Make sure there are adequate standards for identifying defective practices.
* Give the inspection team leaders proper training, coaching after the initial training, formal certification and statistical follow-up, and be prepared to withdraw their licence to inspect if necessary.

Planning phase (7 out of 10)

* Plan inspections well, using a master plan.
* Plan inspections to address the relevant purposes.
* Inspect early and often, while documents are still being written.
* Use sampling to understand the quality level of the document.
* Check against source and kin documents; check these for defects, too.
* Check significant portions of the material - avoid checking commentary.
* Allocate special defect-searching roles to people in the team.

Inspection statistics (2 out of 3):

* Build or buy an automated software tool to process basic inspection data.
* Measure the benefits gained from using inspections.

The other tips are similar and provide general improvement suggestions for other inspection phases, which we may call base practices. The next chapter considers in the light of three experiments how to merge capability evaluation and improvement suggestions.

5. PROCESS IMPROVEMENT SUGGESTIONS DERIVED FROM CAPABILITY EVALUATION

During spring 2001 we looked for inspection improvements in three companies through base practices. The work was based on capability evaluation matrices of the kind described in Figure 3 above. The primary goal was to verify the concept, to make sure that the evaluation process was reasonable from the practitioner's viewpoint and to check that improvement ideas discovered by this means were acceptable to the company.

Company A (estimate 2-3: total value of inspection capability at the company level, cf. Figure 3) managed fairly well with respect to the core base activities. The staff are well educated to inspect and follow standard processes in general. This company typically does not hire inexperienced people, and most of its employees have received proper quality education in their previous posts. The core base activities P.7 to P.12 are fulfilled fairly well because of the employees' work experience.

The company has recently started to maintain its quality processes, which is why the base activity "Improve the inspection process" is in full operation. The improvement suggestions focused on its inspection strategy and statistics (from Gilb's list) seem to be the most relevant ones. These mean that the company should "Make sure there are adequate standards for identifying defective practices" and "Build or buy an automated software tool to process inspection basic data". By means of these activities the company could improve its supporting practices, e.g. "Maintain rules and checklists" and "Support with computer tools".

Company B	P.1 Improve the inspection process	P.2 Organise the inspection	P.3 Train participannts	P.4 Support with computer tools	P.5 Maintain rules and checklists	P.6 Refine information	P.7 Check preconditions for inspection	P.8 Plan the inspection	P.9 Find issues in the artefact	P.10 Categorise defects	P.11 Make corrections	P.12 Conclude the inspection
Total: 1-2	1	1	1	1	1	0	2	2	2	1	2	2

Figure 4. Capability evaluation in company B

Company B (estimate 1-2: total value of inspection capability at the company level (about 100 employees), Figure 4) has poorer base activities overall than company A. The staff are mostly young people and the company is commonly the first established post for its employees. It has recently started to improve its quality assurance, and systematic inspecting is a new practice for it. The core activities are fairly well established, but supporting activities are almost entirely absent. The most important improvement suggestions should first be focused on core activities and after that on supporting ones,. so that the company should "Make sure there are adequate standards for identifying defective practices" and "Give the inspection team leaders proper training, coaching after the initial training, formal certification and statistical follow-up, and be prepared to withdraw their licence to inspect if necessary ". The first tip would improve activities P.5 and P.10 in particular, and the second would affect activity P.3. The company does not collect measurement data regularly nor refine such data into reports for the management, which means that it could also use later tips such as "Build or buy an automated software tool to process inspection basic data" and "Measure the benefits gained from using inspections".

Company C	P.1 Improve the inspection process	P.2 Organise the inspection	P.3 Train participants	P.4 Support with computer tools	P.5 Maintain rules and checklists	P.6 Refine information	P.7 Check preconditions for inspection	P.8 Plan the inspection	P.9 Find issues in the artefact	P.10 Categorise defects	P.11 Make corrections	P.12 Conclude the inspection
Total: 2-3	3	3	2	3	3	1	3	3	3	2	3	3

Figure 5. Capability evaluation in company C

Company C (estimate 2-3: total value of inspection capability in one division of the company (about 100 employees), Figure 5) has implemented inspections to almost the fullest possible degree. Although we have not evaluated its capability with regard to the inspection process with the CMM/Bootstrap levels, we are very sure that it would be at level 3 at least.

The most important shortcomings can be detected in the activity "Refine information", arising from the lack of a knowledge base and a haphazard collection of improvement suggestions. Gilb's tips such as "Build or buy an automated software tool to process inspection basic data" and "Measure the benefits gained from using inspections" would help in this case. The training of participants also requires some effort, and thus the tip "Give inspection team leaders proper training…" could be relevant.

6. CONCLUSIONS

The inspection-tailored capability model presented here provides a method for locating the weak points in a company's inspection process. Its main structure comes from the Bootstrap model, although our interpretation of the base practices differs from that. We look for a large set of base practices and evaluate the grade of each practice in the company/department/project (whether it is in use totally or partially). The evaluation is based on indicators, which can be enabling or verifying ones. The idea is that, if we find indicators of a base practice, we then have justifications for assuming the existence of that base practice. If some of them are at a low level or missing, they should be targets for improvement measures.

Although we focus on the inspection process, we assume that a company which aims at inspection process improvement has already defined the whole software development process at an appropriate level, and our experiments, as reported in this paper, justify this hypothesis. We can easily conclude that it is not reasonable for company B, level 1-2 inspection capability, to focus all its process improvement measures on inspection, but rather it should improve the whole development process, and inspections as a part of this. Due to this close dependence on the whole development process, our next research effort will focus on developing a totally Bootstrap-compliant alternative to the inspection capability model, by means of which we could trace the effects of inspection improvement measures on the whole development process. On the other hand, since capability evaluation applied to the whole development process is a rather laborious activity, not all companies are ready for it, and we will therefore continue adjusting the inspection-tailored capability model further (e.g. its indicators) as a lighter alternative for companies that are keen on inspection improvements.

REFERENCES

[1] Burnstein I., et al., A Testing Maturity Model for Software Test Process Assessment and Improvement, Software Quality Professional, vol 1, no 4, 1999

[2] Emam K., El J., Drouin J., Melo W., SPICE: The Theory and Practice of Software Process Improvement and Capability Determination, IEEE Computer Society, 1998

[3] Fagan M.E., Design and Code Inspection to Reduce Errors in Program Development, IBM Systems Journal, vol 15, no 3, 1976, pp.182-211

[4] Gelperin, D., and Hayashi A., How to support better software testing, Application Trends, May, 1996, pp. 42-48.

[5] Gilb T., Planning to get the most out of inspections, Software Quality Professional, vol 2, no 2, 2000

[6] Gilb T., and Graham D.: Software Inspection, Addison-Wesley, Wokingham, England, 1993

[7] Iisakka J., and Tervonen I., Painless Improvements to the Review Process, Software Quality Journal, 7, 1998, pp. 11-20

[8] i3GO, The URL of the Improved Inspection Initiative Group in Oulu is http://www.tol.oulu.fi/i3/

[9] Johnson P.M., Reengineering Inspection, Communications of the ACM, vol 41, no 2, 1998, pp. 49-52

[10] Koomen, T., and Pol M., Test Process Improvement: A practical step-by-step guide to structured testing, Addison-Wesley, 1999

[11] Kuvaja P. et al., Software Process Assessment and Improvement, The Bootstrap Approach, Oxford, Blackwell, 1994

[12] Paulk M. et al., The Capability Maturity Model: Guidelines for Improving the Software Process, Addison-Wesley, 1995