Systematic Improvement of Technical Reviews in Large-Scale Systems Development

Oliver Laitenberger, Marek Leszak, Werner Brunck, Dieter Stoll

Abstract-- Technical reviews are a cost-effective method commonly used for the early detection of product defects. To exploit their full potential, it is necessary to constantly monitor and improve the implemented review procedure.

This paper describes a systematic improvement effort to amplify and leverage the benefits of reviews at Lucent Technologies Optical Networking Group (ONG) at Nuremberg, Germany. The motivation for the effort stems from root cause analysis results. These results reveal that defects detected in later development phases could have been found earlier by reviews. The improvement effort involved a sequence of four steps. In the first step, review data was analyzed, the current review process was observed, review participants were interviewed, and the existing review documentation was scrutinized. In a second step improvement suggestions were derived based on the collected information and incorporated in the existing approach. The third step involved the training of the participants in the revised approach. The final step consisted of the application of the revised approach in projects at Lucent/ONG.

In essence, the improvement effort provides key insights in the challenges of today's reviews. It questions existing meeting-based review processes and suggests a non-meeting based alternative. In fact, this is one of the very few efforts that implemented non-meeting based reviews in industrial projects.

Index Terms-- Technical reviews, review process, improvement effort

1. INTRODUCTION

TECHNICAL reviews[1] are a proven approach that enables the detection and correction of defects in software artifacts as soon as these artifacts are created. They not only improve the quality of the artifacts but also help software development organizations reduce their cost of producing software. This stems from the fact that reviews allow the identification of defects at a stage where they are easier and relatively inexpensive to correct, thereby causing the development process to avoid additional rework penalties associated with defect detection at later test and integration stages.

At Lucent Technologies Optical Networking Group (ONG) in Nuremberg, Germany, the review process is an essential element of the Standard Development Process (SDP). The SDP is used for all development projects. It is a managed process, i.e., its continuous evolution is driven by retrospective analysis of the projects using it. The review process has been defined based on worldwide published review processes [4] and the same review process has been applied to all major large-scale development projects since 1995.

Today, reviews at Lucent/ONG usually consume around 9% to 12% of the total development effort. These costs include gate (i.e. quality milestone) reviews as well as technical reviews on documents, software sources, and other artifacts of the development process. Still, some defect slip through the review process. A retrospective root cause analysis [8] revealed that a large amount of defects detected in late development phases, that is, system integration and test, could have been found in reviews. This finding provides the motivation for the improvement of the existing review implementation to increase its effectiveness.

In this paper we describe the activities of the improvement effort together with their major findings. The main focus of the paper is on a multi-method approach to characterize the existing review implementation. The approach as well as its findings are helpful for planning a similar effort and learning more about the large-scale use of reviews in an industrial context.

The paper is organized as follows. Section 2 elaborates upon the goals of the study and the study approach. Section 3 characterizes the review approach used until 1999 in the Lucent/ONG R&D Nuremberg organization using the multimethod approach. Section 4 elaborates upon the improvement suggestions, the revised approach, and its implementation. Section 5 concludes with a summary and directions for future work.

2. GOALS AND STUDY PROCEDURE

The starting point of the review improvement activity was given by the results of a root cause analysis [8]. These results can be summarized as follows:

- Review deficiencies were diagnosed for 66% of the defects analyzed.
- 73% of total bugfix effort was spent on defects that were related to review deficiencies.
- Defects escaped early detection due to review deficiencies (e.g., no or incomplete review or inadequate review preparation).
- Human factors have a significant influence on defect injection.

• O. Laitenberger is with the Fraunhofer Institute for Experimental Software Engineering, Sauerwiesen 6, D-67661 Kaiserslautern, Germany E-mail: Oliver.Laitenberger@iese.fhg.de

• M. Leszak, W. Brunck, D. Stoll are with the Lucent Technologies Networking Systems GmbH, Thurn-und-Taxis-Strasse 10, D-90411 Nuremberg, Germany
E-mail: {mleszak, wbrunck, dieterstoll}@lucent.com

[1] Other terms such as formal technical review or inspection could have been used here.

These findings provided the motivation for spending effort in review improvement with the goals to

- Characterize the existing review implementation in various development phases.
- Identify best practices and areas where changes to the review implementation are likely to result in a change but in an improvement.
- Revise the existing approach to implement the improvement suggestions.

To achieve the stated goals we followed a multi-method approach to characterize the current state of the practice at Lucent/ONG. The various activities are depicted in the following figure.

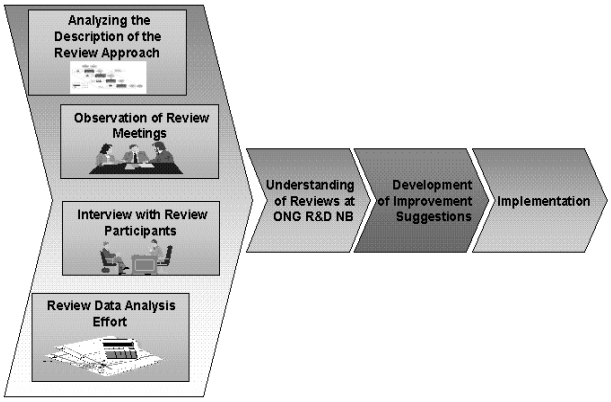

Fig.1. Multi-method Approach

The activities involved the analysis of existing review descriptions, the observation of review meetings, the interview of review participants, and the analysis of review data. Each one of these activities resulted in a unique piece of information to better understand the review approach.

The existing description allowed a first insight in the current review approach. The participation in review meetings helped examine and characterize the review process as currently performed. Notes were taken about the best practices exhibited in the review meetings. The interviews examined the individual work practices of engineers engaged in technical reviews and helped elicit their experiences. The interviews were performed according to a standardized questionnaire that ensured the comparability of answers. Finally, the analysis of review data allowed for the quantitative characterization of the review approach. The data analysis helped identify major review success factors.

The combination of the results from these activities allowed the examination of assumptions about technical reviews in this environment. It, first, provides a solid basis for understanding the existing review procedures and, second, a forum for discussion about review improvement suggestions. Some of these were finally implemented in projects at Lucent/ONG.

3. FORMER REVIEW APPROACH

3.1. Description of the Review Approach

The standard development process (SDP) requires certain reviews in certain stages of a development project. All artifacts and their quality checking activities (reviews, testing) must be planned and scheduled by the respective teamleader in charge of developing a subsystem. Once a deliverable is ready (from a developer' s point of view), the teamleader delegates review control to a qualified moderator. The moderator is responsible for the selection of the mix of experts for a review team, and for the success and performance of a review.

Since prescriptive facts on the optimal number of reviewers cannot be given, the guideline at Lucent/ONG on how many reviewers to invite to a review is based on the reviewed artifact: The following artifact characteristics are considered for the decision:

- Artifact Type
  Code artifacts, for example, require a different (usually smaller) number of reviewers than early phase documents, such as requirements specifications.
- Artifact Complexity and Dependencies
  A high-level design artifact for a certain software domain, for example, needs to be reviewed of experts from all other domains to which the domain under review interfaces. This justifies a larger number of reviewers.
- Artifact Scope
  For artifacts local to a certain development team the main review goal is to detect defects and to ensure that the artifact provides a stable and aligned basis for further development activities. However, for artifacts like higher level design documents or requirements specifications, which typically affect several teams, an additional review goal is to achieve alignment beween the teams with respect to sharing of, for example, interfaces or requirements. This additional review goal requires representatives from all affected teams, even if this would not be necessary from a pure defect detection viewpoint and, thus, a larger number of reviewers is justified.

The reviewers themselves usually have a high level of experience and can be considered experts for the reviewed artifact. The introduction of a particular reading technique for defect detection as suggested in [1] or [6] is currently under consideration.

In addition to the moderator role, the other roles in the review process are

- recorder - records defects into the defect list.
- checker - verifies correctness and completeness of the reworked artifact after the review meeting,
- reviewer - probes the artifact for defects and reports them in the review meeting. The author

or the moderator may also act as a reviewer.

The review process itself is defined in a number of standard phases, i.e., planning, kick-off and overview, preparation (of reviewers), defect logging in a group meeting, rework, and checking. The approach is illustrated in the following figure.

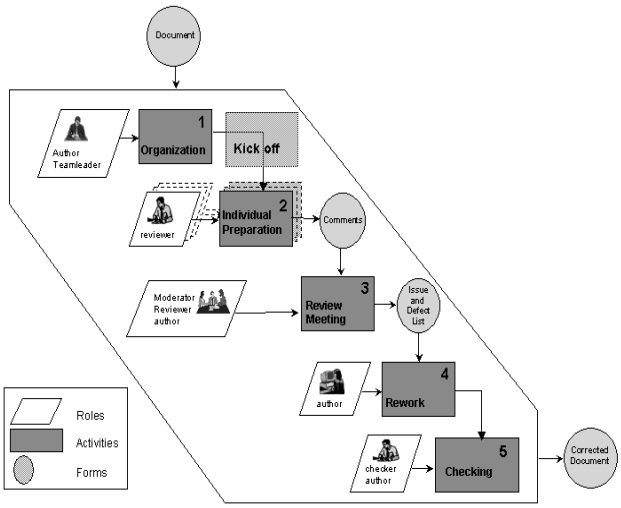

Fig.2. Review Procedure

The review process is mandatory for all newly developed and significantly changed artifacts. However, since all changes must be guarded by modification requests (MRs), not every small change triggers a review.

3.2. *Results from Observing Review Meetings*

Observation Procedure

In total, we observed eight review meetings. The primary goal of the observation was to get an idea on how these meetings were performed within Lucent/ONG. Our observation included review meetings of hardware, system, and software (i.e., specification, design, and code) reviews.

Throughout the review meetings we did not interfere in any way in the meeting procedure. We took notes about various aspects of the review meeting, such as the activities performed or the role of the participants.

From our notes, we extracted information on the purpose of the review meeting, the processing of a review meeting, and the roles assigned to the meeting participants. We discuss our observations in the context of these three categories.

The Purpose of the Review Meeting

We observed that the main purpose for a review meeting at Lucent/ONG was to assess whether an observation from an individual reviewer is really a defect. Although the meeting offers the possibility of detecting additional defects, this rarely did happen. That is, the number of defects newly detected in the observed meetings was rather low.

A follow-up objective is to decide whether the reviewed artifact needs to be re-reviewed. As the observation as well as the interviews revealed, the re-review decision is primarily a review team decision at the end of the meeting. The decision is influenced by the number and (partly) the severity of defects as well as the expected changes to the reviewed artifact.

Performing a Review Meeting

We present how the meetings were performed according to the sequence in which review meetings are usually performed: Introduction, Collection, Closure.

Introduction

Throughout the introduction phase, the moderator welcomes the review participants, presents meta information about the reviewed artifact (e.g, the name and version), and usually asks one of the reviewers to also perform the role of the recorder. In most cases, the moderator also acts as the presenter.

Some of the review meetings involved participants from other locations. They participated in the meeting via teleconference system. In this case, the moderator was also responsible for setting up the connection.

The moderator usually asked the reviewers about their preparedness for the meeting. Preparedness primarily refers to the question whether the participants have read the reviewed document(s) and how much effort they have spent on this activity. The moderator then accumulated the preparation effort and documented the accumulated effort on the defect list. Although we observed some variation in the preparation effort of reviewers, most of the reviewers in the observed meetings were prepared for the meeting. However, some reviewers did not document the defects they detected on a form. They just scribbled notes in the document or added a mark to the text. In the review meeting itself, it happened that they could not really remember the observation to be discussed. This caused some confusion and delay in the meeting. Review meetings, on the other hand, for which the reviewers already brought a documented list of observations appeared to be faster, since these lists could be used as a basis for documenting the real defects.

Collection

Throughout the collection phase, the reviewers presented the observations they had made individually and discussed the observation with the author. If there was consensus about whether an observation was a defect, then the defect was recorded. If no consensus could be reached in further discussion, the defect was added as an open issue to the defect list and the author (or even one of the reviewers) got the responsibility of clarifying it. We did not observe a separate list or form for open issues as suggested in the review process.

Since the moderator is also a reviewer, there were cases in which a long discussion took place between the author and other review participants. Since the duration of this discussion is clearly beyond the goal of achieving an effective review meeting, we set a limit of 3 minutes for discussing an observation. Afterwards, the observation is documented as a defect.

Most of the time the discussion between a reviewer and the author focused on defects that go beyond spelling and grammar mistakes. Spelling and grammar mistakes are usually not included in the defect list, which is a good practice.

In most review meetings, no decision is being made on the defect class, that is, the severity of the defect. The

classification of a defect is deferred to the author or to the recorder (which is a practice that can be followed). However, it is unclear whether the person who classifies defects applies the definition of the classes appropriately. This is doubtful since many interviewees did learn the review approach from practising it rather than getting any training. If the definition is not followed, this will introduce some inconsistency in the data and the overall quality of the data is lacking. Since defect classes are a good source of information, the defect classification issue represents an area for improvement.

In our observation of the review meetings, we observed a very open-minded atmosphere. None of the observed meetings was dominated by a single strong-willed individual that prevents others from active participation. In fact, the interviews revealed that none of the reviewers held back observations. Hence, each of the reviewers' observations was mentioned and discussed in the meeting.

Closure

The moderator asks about whether to perform a re-review of the artifact. Moreover, he or she collects the effort data from each participant, if not done at the beginning of the meeting. A good practice of some moderators was to thank the review team for their participation before actually closing the meeting.

The moderator should collect the effort information at the start of the meeting since it provides some information on whether the reviewers are well-prepared.

3.3. Interview Results

Interview Procedure

In total, we conducted interviews with 15 experts from Lucent/ONG. The experts were selected by Lucent/ONG as representing a cross-section of relevant experience within the Lucent domain and they usually participated in the observed review meetings. Contacts were made with the help of two members of the Review Process Team.

The interview questionnaire consists of three parts: Background, analysis, and comparison. The first part of the questionnaire, background information, characterizes the experience of the interviewee with software development at Lucent/ONG and his/her experience with the various review roles.

The second part of the questionnaire, analysis, includes questions that are intended to gain insight into the review practice as currently performed at Lucent/ONG. Questions in this part of the interview focus on aspects like the review process or the influential factors on review success.

The final part, comparison, lets interviewees estimate the value of reviews when compared to testing and express their suggestions for improving the current review approach.

In particular, we discuss the following issues in more detail:

- The interviewees' experience
  The primary objective behind capturing the interviewees' experience is to get some background information on the interviewees. This gives us some insight on the selected sample of review participants.
- The reasons for performing reviews
  The objective of this part of the interview was in a first step to elicit the reasons for performing reviews from the perspective of the interviewees. We identified 7 different reasons for conducting a review from the literature [12] and each interviewee was asked to perform an importance ranking of those reasons.
- The factors influencing the number of detected defects
  The objective of this part of the interview was to prioritize factors that impact the number of defects detected. We identified 10 different factors from the literature [12] and each interviewee was asked to perform a ranking of those factors according to their importance.
- The factors influencing preparation effort
  The interviewees were also requested to prioritize factors that impact the preparation effort for review. We focused on preparation effort, since we deemed adequate preparation as one of the key drivers for the number of defects detected. We identified 9 different factors from the literature [12] and asked each interviewee to perform a ranking of those factors.
- The role of review meetings
  We were interested in the interviewees' perception about the role synchronous review team meetings play in the context of the review process. Our interest results from the literature on software review [11], [12], [22], in which the tangible benefits of such meetings in terms of newly detected defects (so-called meetings gains) are questioned. However, other authors, such as the ones in [6], argue that such meetings provide intangible benefits, such as dissemination of product information, development experiences, or enhancement of team spirit. To find out the reasons for performing synchronous review team meetings at Lucent/ONG, we asked the interviewees about the importance of the various activities that are performed in the context of those meetings. In addition, we asked the participants for a ranking of the activities according to the effort they consume in the meeting.
- The estimated savings of reviews
  In the final part of the interview, we asked the interviewees how much more expensive a defect in a particular type of document is when the defect is detected in testing or, the other way round, the amount of effort saved when the defect is detected in review rather than in testing. This estimate indicates the savings that can be attributed to reviews. We have to state that most of the interviewees could not provide a clear-cut answer

to this question because they had no experience in testing.

- The improvement suggestions of the interviewees
  The goal of this question item is to give interviewees the chance to describe their own improvement suggestions. We believe this to be an important underpinning for some of our proposed improvement suggestions.

Interview Results

We present the result for the questions related to the reasons for performing reviews, the factors influencing the number of detected defects, the factors influencing preparation effort, the role of review meetings.

The reasons for performing reviews

The results of the analysis are summarized in Table I below. Each table entry contains a reason (the wording is the same as the one used during the interviews), the minimum and maximum rank for each one, its range in brackets, and its average rank. The factors are sorted according to their ranks. The meaning of the various reasons was explained to the interviewees throughout the interviews.

TABLE I
RANKING OF REASONS FOR PERFORMING SYSTEM AND SOFTWARE REVIEWS

Reason	min ➠ max (Range)	mean value
Efficiency (i.e., find defects cheaper than other defect detection activities)	1 ➠ 2 (1)	1.5
Effectiveness (i.e., find defects)	1➠4 (3)	2.3
Alignment/Coordination of the development team	1➠7 (6)	3.6
Enforce the defined standards	2 ➠ 7 (5)	3.8
Improving communication	3 ➠ 6 (3)	4.8
Education/Learning	3 ➠ 7 (4)	5.5
Team building	6 ➠ 7 (1)	6.5

The most prevalent reason for performing reviews is to detect and remove defects as early as possible (efficiency). When excluding the cost saving element, the fact of finding defects was rated second (effectiveness).

The reason "alignment/coordination of the development team" received high ratings. Hence this reason can be regarded as an intangible benefit for reviews at Lucent/ONG, which must be compensated by other means if synchronous review meetings are abandoned.

Other intangible benefits of reviews, such as education/learning, improving communication, and team building were rated at the lower end of the spectrum.

These results demonstrate that Lucent's review approach keeps defect detection as its primary objective. The lack of training, an engineer's tendency to focus on solutions, or a poorly defined process often saddle the review process with too many objectives (e.g., discuss solutions or reach a consensus regarding the implementation). The principal objective of reviews, however, is to detect defects- all other purposes are secondary and should be treated as such. This objective seems to be followed at Lucent/ONG.

The factors influencing the number of detected defects

The results of the analysis are summarized in Table II below.

TABLE II
RANKING OF INFLUENTIAL FACTORS FOR THE NUMBER OF DETECTED DEFECTS IN SYSTEM AND SOFTWARE REVIEWS

Factor	min ➠ max (Range)	mean value
System Experience	1 ➠ 7 (6)	3.6
Preparation Effort	2 ➠ 9 (7)	3.8
Experience in software development	1 ➠ 10 (9)	4.7
Domain Experience	1 ➠ 10 (9)	4.8
Initial defect-proneness of the reviewed documents	1 ➠ 10 (9)	5.2
Document characteristics (structure, complexity)	1 ➠ 8 (7)	5.4
Familiarity with the reviewed document	1 ➠ 10 (9)	5.7
Size of the reviewed documents	2 ➠10 (8)	6.8
Review Experience	4 ➠ 10 (6)	6.8
Defect Detection Support, e.g., in the form of a checklist	5 ➠ 10 (5)	8.2

A number of important observations can be made from these interview results regarding the amount of consensus that is reached by the experts in their ranking. First, the highest ranked factors are all related to experience within the domain, the system, or software development. This means that a high degree of experience is one of the most essential factors that

help reviewers detect defects. However, interviewees consider review experience not as important. This may be explained by the fact that most interviewees refer this factor primarily to the review process rather than to experiences on how to detect defects or potential sources of defects.

A second important observation is that interviewees consider the effort they can spend on preparation more important than the size of the artifact. This subjective evaluation corroborates findings from the collected review data that preparation effort is a more essential factor for explaining the number of detected defects than size that we present later on. The implication for the given project situation is that a review must provide adequate preparation effort to ensure the quality of the reviewed documents as well as of the review process.

Finally, defect detection support, e.g., in the form of a checklist, was rated the least important factor that helps detect defects. The reasons for this low ranking are two-fold. First, a reviewer is usually an expert on the reviewed artifact. Experts already have some strategies on how to scrutinize a document for defects and what to look for. In fact, interviewees at the lower end of the experience spectrum assigned higher ranks to procedural support than subjects at the higher end of the experience spectrum. Since we did not interview novices, the results of this ranking may be biased. The second reason stems from the fact that we investigated the current review procedure. Right now checklists or any other means of defect detection support are not used in a systematic fashion in system and software reviews (in contrast to the hardware ones). If something is not available or accessible, it is no surprise that it cannot have an impact on the number of detected defects.

Throughout the interview some of the interviewees stated that better support, e.g., in the form of a checklist, would be a good starting point for supporting the individual preparation phase. However, they also stated that the checklist needs to be revised and updated according to the experiences in the environment.

The factors influencing preparation effort

The results of the analysis are summarized in Table III. Each row contains the factor (the wording is the same as the one used during the interviews), the minimum and maximum rank for each one, its range, and its average rank. We sorted the factors according to their ranks.

TABLE III
RANKING OF INFLUENTIAL FACTORS FOR PREPARATION EFFORT

Factor	min ➠ max (Range)	mean value
Document characteristics (structure, complexity)	1 ➠ 7 (6)	2.9
Familiarity with the reviewed document	1 ➠ 6 (5)	3.8
Size of the reviewed documents	2 ➠ 9 (7)	4.1
System Experience	1 ➠ 8 (7))	4.3
Initial defect-proneness of the reviewed documents	1 ➠ 9 (8)	4.4
Domain Experience	1 ➠ 9 (8)	5.2
Project Management Pressure	1 ➠ 9 (8)	6.0
Review Experience	3 ➠ 9 (6)	6.6
Defect Detection Support, e.g., in the form of a checklist	6 ➠ 9 (3)	7.8

The two factors that have the biggest impact on preparation effort are system experience and document characteristics. This result suggests the hypothesis that review participants who are very familiar with the system require less preparation effort. Unfortunately, the review data at Lucent/ONG does not include information on the experience of the review participants. Neither does it include information on document characteristics apart from size. However, since these factors seem to be a major driver for preparation effort, this information may be valuable to collect in the future.

The "project management pressure" factor was not rated as important as we thought. We meant with this factor the time pressure within the project that may, for example, prevent adequate preparation of the reviewers.

We found two explanations for this result. First, although interviewees regard time pressure as a factor they do not completely attribute this to project management. Hence, a better formulation for this factor would have been "time pressure within the project". Second, once a review is initiated, reviewers take the time they need for preparation. None of the interviewees reported a situation in which they stopped looking for defects at some point in the document because they ran out of time. They all said that they at least read the document once. Some interviewees however admitted that they only performed a more rigorous check on those parts of the reviewed artifact with which they were already familiar. This may be an explanation for observed differences in the individual preparation effort.

The least important factor is again the defect detection support for reviewers. Again, this result can be attributed to the level of experience that most interviewees had and the non-availability or non-accessibility of defect detection aids in the current review implementation.

One of the interviewees mentioned that the preparation effort he or she spends also depends on the importance of the reviewed document for his or her own work. If the reviewed document is more important, the interviewee will spend more effort than if the document is less important for his or her own work. This suggests include at least the intermediate stakeholders of an artifact as reviewers in the review team.

The role of review meetings

We were interested in the interviewees' perception of the role review meetings play in the context of the review process. This interest results from the literature on software review [5], [12], in which the benefits of such meetings are questioned. Hence, we asked the interviewees about the importance of the various activities that are performed in the context of the meetings. We also asked the participants for a ranking on the question which of the activities consume the most effort. Table IV depicts the results of the interviewees' ranking. We sorted them according to the importance ranking.

TABLE IV
RANKING OF ACTIVITIES WITHIN REVIEW MEETINGS (IMPORTANCE, EFFORT)

Activity	min ➠ max (Range) of Importance	mean value (Importance)	min ➠ max (Range) of Effort	mean value (Effort)
Additional defect detection	1 ➠ 3 (2)	1.8	1 ➠ 5 (4)	2.6
Deciding which defects are really defects	1 ➠ 6 (5)	3.1	1 ➠5 (4)	2.7
Merging defect lists of individuals	1 ➠ 7 (6)	3.8	1 ➠ 7 (6)	3.6
Achieve a better understanding of the reviewed artifact	2 ➠ 7 (5)	4.6	1 ➠ 5 (4)	2.6
Ensuring adequate preparation	1 ➠ 9 (8)	4.7	4 ➠ 9 (5)	7.6
Decision about re-review	2 ➠ 9 (7)	4.7	5 ➠ 9 (4)	6.6
Group bonding/improving team spirit or communication	3 ➠ 8 (5)	6.7	4 ➠8 (4)	6.5
Acknowledgement of my own work as a reviewer/ Credible feedback on my work	5➠9 (4)	7.6	5 ➠ 9 (4)	7.6
Education of weak group members	7➠ 9 (2)	7.9	1➠ 8 (7)	5.6

Table IV shows that interviewees perceive the activity of additional defect detection as the most important one. Additional defect detection is also perceived to consume most of the effort within the review meeting. This is a somewhat surprising result, since it is not in line with our observations in the review meetings. From our observations most of the effort was spent on clarifying and discussing whether an observation was a real defect. In fact, the number of defects that was not previously detected was observed to be rather low. One possible explanation of this finding is the hypothesized synergy effect that a meeting may have. Synergy may explain why they regard the meeting as an additional opportunity for detecting defects (although this rarely happens). The question is whether additional effort for detecting defects warrants a meeting to be organized and performed.

The activities of merging defect lists and deciding whether observations are defects were ranked second and third on the list of importance. With respect to effort, only the activity of achieving a better understanding was between the two factors.

The activity of educating weak group members is not considered important. This may be explained by the fact that most of the participants are experienced and do not need to be educated. On the other hand, we did not interview developers who are new or very inexperienced. Hence, the ranking of this factor is probably biased by the selection of our reviewers.

Some of the activities, such as the decision on a re-review, were not considered as important in the meeting. But they also consume little effort on the participants' behalf.

3.4. Data Analysis Results

Data Analysis Procedure

Although we only present descriptive statistics we also performed regression analysis. These and more detailed results about the analysis effort can be found in [7].

Number of detected defects

Figure 3 depicts the number of defects that are detected in the different types of reviews. The box represents the interquartile range (i.e., 50% of all observations fall within this range), while the whiskers represent the minimum and maximum value.

As Figure 3 shows, defect distribution is consistent among review types. The median value of a specification, design, and code review is 12, 15, 14, respectively. While the interquartile range of specification reviews and design reviews is 21 defects, it is slightly lower for code reviews (17 defects).

The presented findings establish an organization-specific baseline for Lucent.]/ONG against which to compare any improvement that promises to increase the number of defects detected. However, an evaluation in the context of other review work is difficult because most studies only focus on code reviews and often do not present the number of defects found, but rather some summary statistics [11] [13].

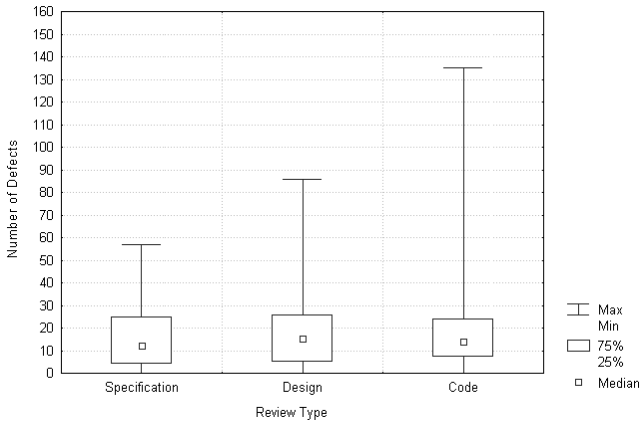

Fig.3. Number of Detected Defects

Review Effort

Figure 4 shows the preparation effort distribution as well as the distribution of the total effort spent on reviewing the various artifacts. The total effort includes the preparation as well as the meeting effort of all review participants.

Figure 4 reveals that reviewers involved in any kind of review usually spend between 2 and 8 hours for preparation (independent of the number of reviewers) and between 4 and 14 hours for the total review. The effort distribution looks similar for the different types of reviews.

The results show that the review of artifacts in early phases (i.e., specifications) does not significantly consume more effort than code artifacts. The median effort for specification, design, and code reviews (7, 6, 8 person hours) as well as the upper quartile ranges (11, 10, 13 person hours) provide a lower threshold for managers on how much effort the review of a particular artifact type may at least consume in future projects.

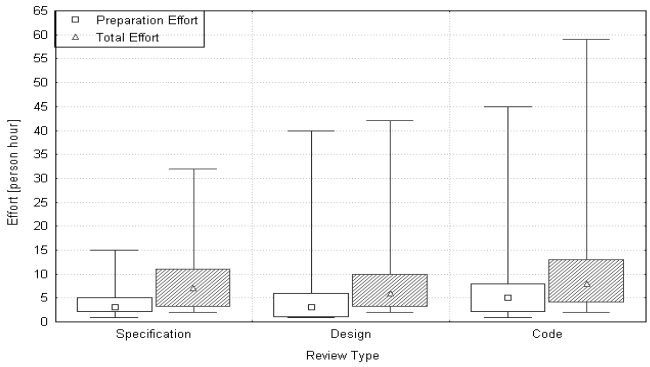

Fig.4. Review Effort

In this study we focus on preparation effort as an important parameter to optimize. Figure 5 depicts the relationship of preparation versus meeting effort. It shows that most of the reviews consume at least as much effort in preparation than in the meeting. Surprisingly, the preparation/meeting-ratio is highest for code reviews. This may be explained by the following two reasons. First, the reviewers do not spend as much effort for the preparation of design or specification reviews, which impacts the numerator of the preparation/meeting ratio. And second, the meetings for specification and design reviews are more effort consuming since documents may be larger, which affects the denominator of the preparation/effort ration. Both reasons are possible and lead to a smaller preparation/effort ratio.

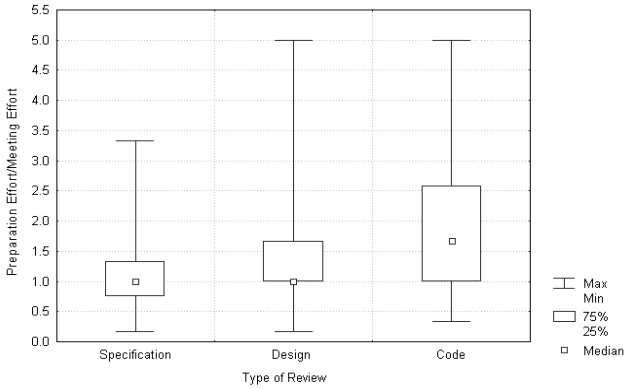

Fig.5. Relationship of Preparation Effort and Meeting Effort

Size

The unit of size for specification and design artifacts is pages whereas for code artifacts, it is noncommentary source lines of code. Since the measurement units are different for specification/design documents and code components, we present two graphs. Figure 6 exhibits the size distribution across the reviewed artifact types.

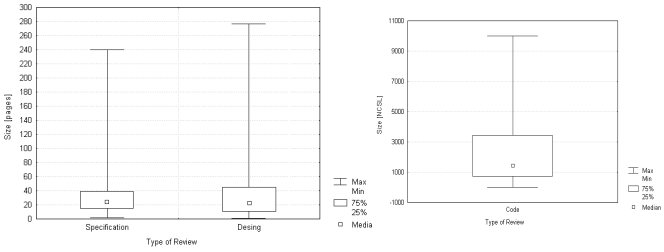

Fig.6. Size of Reviewed Artifacts

Figure 6 reveals that the median size of a specification is 24 pages, whereas it is 22.5 pages for design artifacts. Most of the reviewed documents are smaller than 50 pages. The median size for code components is 1450 NCSL and most of the reviewed code components are lower than 3420 NCSL, for components coded in ANSI-C. Artifacts of this size are neither too large nor too small for review and are within the range of the ones reported in other studies [4]. Relationship of Preparation Effort and Meeting Effort

Number of Reviewers

In addition to defect, size, and effort distribution, we also investigated the number of reviewers. Figure 7 shows how many reviews have been performed for each artifact type with a specific number of reviewers.

According to Figure 7, most of the reviews were performed with 3 reviewers. Specification reviews often involve a higher number of reviewers. This emphasizes the importance of the specification phase for design and coding.

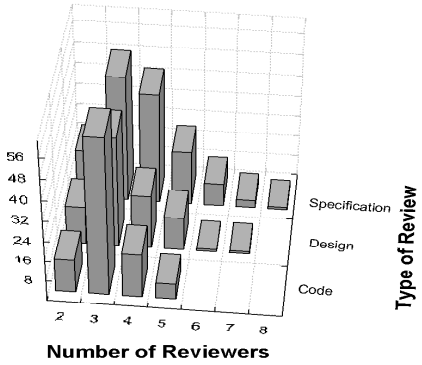

Fig.7. Histogram of the Number of Reviewers for the Different Types of Reviews

The optimal number of reviewers is still debated in the literature. This debate boils down to the question whether involving more reviewers helps detect more defects. Surprisingly, there are few consistent results so far. Weller presents some data from a field study using three to four reviewers [13]. Madachy presents data showing that the optimal size is between three and five people [9]. Bourgeois corroborates these results in a different study [2]. Porter et al.'s recent experimental results, however, suggest that reducing the number of reviewers from 4 to 2 may significantly reduce effort without increasing review interval or reducing effectiveness [10].

Our impression from the published case studies is that those do not take into account shared artifacts in large-scale product development. Reviews of such artifacts require cross-team alignment, as elaborated in section 3.1. Furthermore, while artifact type of the case study was typically code, we also studied reviews of requirements and high-level design documents. There, the number of reviewers is usually larger than for code reviews.

Defect Density

Since we assumed that the number of defects is related to the size of the document, we calculated the defect density defined as defects per unit of size. Figure 8 shows the result of this calculation.

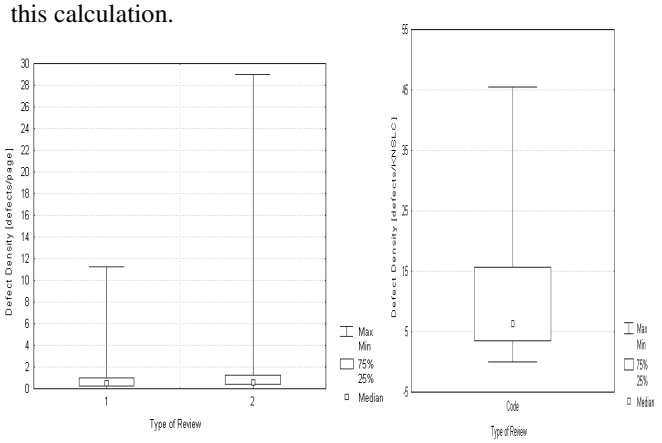

Fig.8. Defect Density

Reviews exhibit on average 0.53 defects/page when looking at specification reviews. Design reviews exhibit 0.58 defects/page. Finally, code reviews find 7.0 defects/KNCSL. The variation seems to be small.

For specification and design artifacts, we did not find comparable figures in the literature. For code artifacts, however, the defect densities are within the reported range of other telecom organizations. Ebert et. al. describe some results from Alcatel Telecom [3]. There, code components exhibit an average defect density of 9 review-found defects/KNCSL. This result supports the initial statement that the review process for code components at Lucent/ONG belongs to the state of the practice that can be found in the software industry.

4. IMPROVEMENT SUGGESTIONS AND REVISED APPROACH

4.1. Improvement Suggestions

Based on the collected information, we derived suggestions on how to improve the review approach. We classified them according to the technical and measurement dimension of reviews.

Technical Dimension of Reviews

The Role of Review Meetings

- Track the number of defects that are detected in the meeting as well as the number of observations that turned out to be no defects for a subset of review meetings. The result helps determine the synergy effect to be expected from meetings.
- Depending on the observed synergy effects , decide upon the following alternatives:
  If synergy effects are generally low and it is impossible or not cost-effective to increase them (e.g., by including additional reviewers), the use of the review meetings should be reduced to save the meeting expenditures.
  If synergy levels are generally low but can reasonably be raised to cost-effective levels (e.g, by including additional reviewers or by providing training in defect detection), the meeting effort may be justified.

Best Practices for Running a Review Meeting

This is a list of best practices and issues identified while observing the review meetings. They were already presented in more detail in the last section. We repeat them here in a condensed form, although we do not elaborate upon them in detail in each and every case.

Introduction

- It is important for the moderator to ensure entry criteria for the meeting at the beginning of it.
- If other reviewers participate in the meeting via a

video- or teleconference system, the moderator should establish the connection before the meeting actually starts to avoid delays.
- Since there is some variation in the preparation effort of individual reviewers, the preparation effort should not be accumulated but collected for each single reviewer. The additional data collection does not add any extra effort since the moderator asks for these data anyway and the data helps to build better effort prediction models.

Collection
- A form for documenting the observations throughout the individual preparation phase should be introduced. This form may have a similar format as the defect list. An electronic elaboration and exchange of this list has two benefits: First, it avoids the problem of unreadable handwriting. And second, observations that turn out to be real defects can easily be marked (e.g., with a check box) and do not need to be documented again.
- Tool support is beneficial for defect documentation.
- Examine for a small subset of reviews whether the exchange of defect information before the meeting significantly lowers the meeting effort or provides other (intangible) benefits.
- Exchange of defect information before the meeting can take the following form: The moderator gets the results of the individual preparation phase from each of the reviewers in electronic form not later than one day before the meeting is scheduled. He or she produces a consolidated defect list including the observations thus far detected. The consolidated defect list can be sent to the reviewers and the author as input for discussion in the meeting. If consensus is reached on an observation during the review meeting, the observation can be marked as defect.
- Other reviewers should pay attention to the amount of discussion and intervene in cases where they have the impression that no progress is being made.
- Moderator and recorder need to coordinate themselves throughout the meeting since the success of the meeting depends on both, adequate amount of discussion (moderator responsibility) and documentation of defects (recorder responsibility).
- Editorial comments, like spelling and grammar mistakes, should not be included in the defect list unless they have major consequences on artifact quality.
- A training for selected moderators increases the moderator's awareness to prevent solution discussions in the review meeting.

Closure
- The moderator should collect the effort information at the start of the meeting, since it provides some information on whether the reviewers are well-prepared. Moreover, it represents the basis for any kind of systematic analysis and evaluation of review metrics.

Defect Detection Support for Reviewers

The quantitative findings revealed that some reviewers did not spend as much effort for the review preparation as others. To ensure that the preparation effort they spend is well invested, defect detection support in the form of reading techniques should be used for the defect scrutiny. To implement this idea, the following guidelines are helpful:
- Create checklists for the various document types and make those available for reviewers. Criteria for the development of an approprate checklist are for example, adequate level of abstraction or tailoring to the company specific needs and problems.
- Consider more procedural reading techniques for a more systematic and rigorous scrutiny.
- Develop tool support for comment creation, discussion, assessment, and communication.

Measurement Dimension of Reviews

Collection of Review Data

Refine and stream line the data collection procedure. This includes answering the following questions:
- why to collect which measure, i.e., what is the concrete purpose?
- when to collect which measure?
- what needs to be collected for the stated purpose?
- how to collect the measures?
- how to support the documentation of defects by a tool?.

Feedback on the Success of Reviews for Participants and Managers
- Perform a regular analysis of collected review data.
- Feed back the analysis results to data providers. The feedback for a single review may be, for example, performed on an individual basis via e-mail or for a set of reviews on a team basis (e.g., in a project team meeting).
- Use analysis results to consider review effort in project plans.
- In a pilot study, determine the cost/benefit relationship of reviews and testing.

4.2. Revised Approach

Based on the improvement suggestions, the existing review approach was revised. A major change was the introduction of an asynchronous review, i.e. without a (face-to-face) meeting to collect and discuss defects. There, defects are typically

collected and reported via e-mail to the author and all other reviewers. To achieve alignment and synergy among the participants, this type of review is performed as a so-called 2-round e-mail broadcast. After a first consolidation activity, the author sends the comments back to all reviewers. This leads to another round of aligning the comments for acceptance or rejection.

The process variation, which is denoted “desk review”, is depicted in Figure 9.

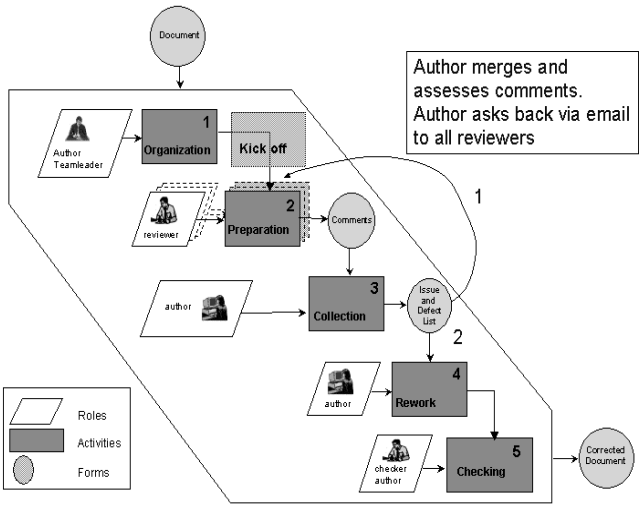

Fig.9. Non-Meeting based Review

The use of reviews without a meeting is restricted to artifacts that

- are not newly developed
- and contain only small changes
- and contain only non-critical changes.

This constraint is based on the assumption that all other artifacts require not only the detection and correction of defects, but also an alignment among reviewers. The latter can be best accomplished in a meeting.

After the introduction of the non-meeting based approach, the review process has been changed for meeting reviews as well. To increase the effectiveness of a review meeting, it is now highly recommended for reviewers to deliver comments prior to the meeting in an electronic format. The more the reviewers follow this recommendation, the more cost-effective the meeting can be conducted. These advantages can be accomplished by the additional request on the author to pre-assess all early delivered comments for his/her intended acceptance or rejection. All pre-accepted comments do not need to be discussed and assessed during the meeting any more (given that no reviewer participating in the meeting objects against their acceptance). As a consequence, this practice reduces discussion and recording time during the meeting.

An important effect of using both review approaches in projects is the use of a hybrid review approach, i.e., a mix of synchronous and asynchronous aspects. Some reviewers tend to deliver comments prior to the meeting and then do not appear at the meeting, while some others do not provide early comments but rather join the meeting and report their findings there. This practice ensures maximal flexibility especially in high-pressure project situations for which it is difficult to find a meeting appointement so that all critical reviewers can participate. This flexibility leads to easier guaranteed review meeting schedules even if some critical reviewers cannot join the meeting. To our best knowledge, this hybrid approach has not been evaluated qualitatively or quantitatively in the review literature yet. So we cannot compare our experiences to those of others.

To convey the knowledge of the various review approaches, a tutorial was performed for review participants. The revised approach has been introduced for all projects of Lucent/ONG at Nuremberg.

5. CONCLUSION

Technical reviews are considered one of the most effective methods for software quality improvement and defect cost reduction. To exploit their full potential, they need to be constantly monitored and optimized. In this paper, we presented an extensive improvement effort performed at Lucent/ONG. The effort started with a extensive characterization of the former review implementation. A multi-method approach was followed to collect the major information. Using multiple collection methods turned out to be very helpful since they allow for a complete picture of the review implementation. Moreover, the different method results can be cross-validated.

Based on the collected information improvement suggestions were derived. One suggestion involved a major change in the review process. Previously, the review meeting at Lucent/ONG has been conducted by default as a so-called `face-to-face meeting'. However, due to the increasing integration of Lucent/ONG into international development activities and due to budget constraints, a non-meeting based, asynchronous procedure called “desk review” has been added. No cost information about this process change is currently available Hence, we are not yet in the position to compare the cost-effectiveness of both approaches and package our experiences. We can only hypothesize that abandoning the review meeting improves the review cost-effectiveness, since meeting exenditures are saved. However, this is an hypothesis that needs to be examined once quantitative information become available.

REFERENCES

[1] Victor R. Basili. Evolving and Packaging Reading Technologies. Journal of Systems and Software, 38(1), July 1997.

[2] Karen V. Bourgeois. Process Insights from a Large-Scale Software Inspections Data Analysis. Cross Talk, The Journal of Defense Software Engineering, pages 17–23, oct. 1996.

[3] Christof Ebert and Thomas Liedtke and Ekkehard Baisch. Software Measurement - Current Trends in Research and Practice, chapter Improving Reliability of Large Software Systems, pages 209–228. Deutscher Universitaets Verlag, Gabler Edition Wissenschaft Edition, 1999.

[4] Tom Gilb and Dorothy Graham. Software Inspection. Addison-Wesley Publishing Company, 1993.
[5] Philip M. Johnson. Reengineering Inspection. Communications of the ACM, 41(2):49–52, 1998.
[6] Oliver Laitenberger. Cost-Effective Detection of Software Defects through Perspective-based Inspection. PhD thesis, University of Kaiserslautern, 2000.
[7] Oliver Laitenberger, Marek Leszak, Dieter Stoll and Khaled El-Emam. Causal Analysis of Review Success Factors in an Industrial Setting. 6th IEEE International Symposium on Software Metrics (METRICS 99). West Palm Beach FL, Nov. 1999.
[8] Marek Leszak, Dewayne Perry and Dieter Stoll. A Case Study in Root Cause Defect Analysis. Proc. of IEEE Int. Conf. on Software Engineering (ICSE-22), Limerick/Ireland, 7-9 June 2000
[9] Ray Madachy, Linda Little, and Sylvia Fan. Analysis of a successful Inspection Program. In 18th Ann. NASA Software Eng. Laboratory Workshop, pages 176–198. NASA, November 1993.
[10] Adam A. Porter, Harvey P. Siy, Carl A. Toman, and Lawrence G. Votta. An Experiment to Assess the Cost-Benefits of Code Inspections in Large Scale Software Development. IEEE Transactions on Software Engineering, 23(6):329–346, June 1997.
[11] Glen W. Russell. Experience with Inspection in Ultralarge-Scale Developments. IEEE Software, 8(1):25–31, January 1991.
[12] Lawrence G. Votta. Does Every Inspection Need a Meeting? ACM Software Eng. Notes, 18(5):107–114, December 1993.
[13] Edward F. Weller. Lessons from Three Years of Inspection Data. IEEE Software, 10(5):38–45, September 1993.