Empirical investigation of selected software review process variables, results from a case study

Empirical investigation of selected software review process variables: results from a case study

Rob Kusters, Jos Trienekens

Abstract

This paper presents the results of a statistical analysis on real-life data from an industrial software development company in The Netherlands. The data have been collected from the actual review processes which have been carried out in the context of quality assurance regarding the software products and software processes. The goal of this research project is to get more insight in the efficiency and effectiveness of the review processes. Based on questions from the software management in the company, and on findings as reported in literature, three review process variables have been defined, respectively the review size, the review speed and the review team size. Based on a statistical analysis of the data interesting insights have been gained in the review process, some selected statements from literature have been validated, and recommendations have been given to improve the review process.

1. Introduction

Software engineering is the application of a systematic, disciplined and quantifiable approach to the development, operation, and maintenance of software. It surrounds techniques and procedures, often regulated by a software development process, with the purpose of improving the quality, e.g. the reliability and maintainability of software systems. The effort is required by the potential complexity of the systems, which may contain hundred thousands of lines of code (DeMarco, 1999). One of the most important quality assurance activities regarding the software development process and the product quality is the software review process. Review processes are aimed at a prevention and (preferably early) detection of defects in both the intermediate products (e.g. specifications, design) and the final software products (e.g. code).

The goal of this research project is to get more insight in the effectiveness of review processes, and to improve or optimise them. Based on questions from software management in the software development company, and on findings on review effectiveness as reported in literature, three review variables have been identified, respectively the review size, the review speed and the review team size.

The structure of the paper is as follows. Section 2 gives some background information regarding the software development process, software defects and the review process. Section 3 addresses data collection in software development and Section 4 describes the selected review process variables and accompanying research questions. In section 5 selected results from the statistical analysis are presented and discussed. Section 6 finalises this paper with conclusions and future work to be done.

2. Background

In this section we will give some information about the context of the review process, i.e. the software development process, and the need for reviewing. I.e. the detection of software defects.

2.1. Software development

The V-Model shows the different phases of the development of software products and the associated phases of verification and validation. Figure 1 shows that verification and validation is carried out in the right part of the V-model and in each phase of it, respectively in the conceptual phase, the requirements and architecture phase and in the detailed design phase. The software review process is one of the most well-known verification and validation processes (Softdevteam, 2006).

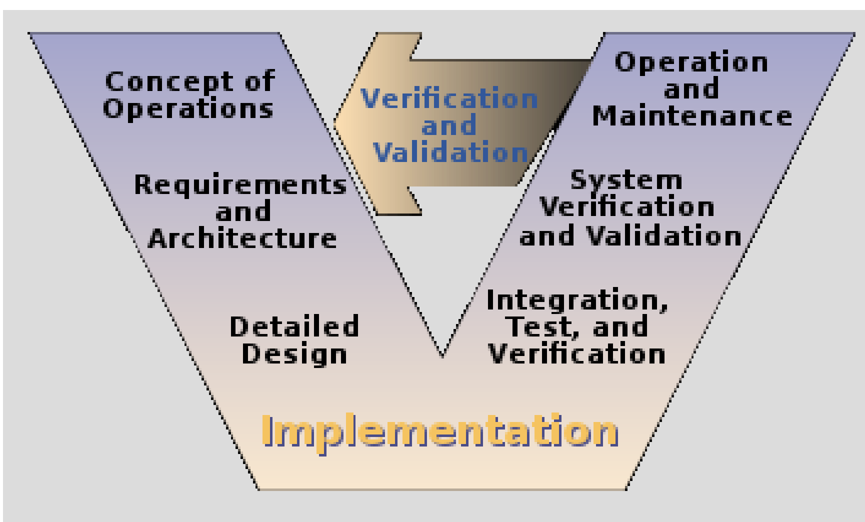

Figure 1: The V-Shape model (Softdevteam, 2006)

The quality of software products is difficult to be measured quantitatively. Measurement is very labour-intensive, in particular at the end of the software development process. Therefore the objective of software management is to monitor and control software product quality during the whole life cycle. Monitoring and control is often carried out on the basis of software reviews processes.

2.2. Software defects and the importance of reviews

A defect is any unintended characteristic that impairs the utility or worth of an item, or any kind of shortcoming, imperfection, or deficiency. We will define a software defect to be any flaw or imperfection in a software work product. These definitions have been transferred to the field of software engineering by various authors (Kanem et al, 2001) and (Ghezzit et al, 2003). In (DeMarco, 1999) a difference between bugs and defects is discussed. "A bug is called something that crawls of its own volition into your code and maliciously messes things up, and even he thinks that: to a certain extent the bugs are cute for developers. Bugs could happen to anyone, on the other hand a defect is your own damned fault".

Software defects include all defects that have been encountered or discovered by reviewing intermediate products during the development or during the operation of a final software product. We mention here three types of defects. *Requirements defect*: A mistake made in the definition or specification of the customer needs, i.e. the concept of operations phase, for a software product. This includes defects found in functional specifications; interface, design, and test requirements; and specified standards. *Design defect*: A mistake made in the design of a software product, i.e. in the requirements and architecture phase. This includes defects found in functional descriptions, interfaces, control logic, data structures, error checking, and standards. *Code defect*: A mistake made in the implementation or coding of a program, i.e. in the detailed design phase. This includes defects found in program logic, interface handling, data definitions, computation, and standards.

In (Gilb, 2004) a review is defined as: "any process of human examination of ideas with defined purposed and defined standards of inquiry". Review is an effective but also expensive quality assurance activity with the aim of finding defects during the process to avoid becoming defects after the delivery of the software. The benefit of these inspections is to discover defects as soon as possible so that they do not spread to the next phase in the software development life cycle, because fixing defects when a product is delivered can be 50-100 times more expensive than during a development process (Biffl and Halling, 2003). In various research publications different variables of review processes have been discussed, and questions about their optimal values. Some examples of these variables and questions are:

- The review size, e.g. what is an optimal size of a review document?
- The speed of a review, e.g. how fast / slow should a review team carry out the work?
- The team size (number of review participants/time invested), e.g. what is the optimal number of reviewers in a team?

This type of review variables have a considerable impact on the review effectiveness and the cost-benefit balance of review activities (Aurum et al, 2002). However, only few studies have been carried out on the basis of a statistical analysis of well-collected real-life data of the review process.

3. Review process data

Our research is based on data collected from the software development processes of a large industrial company in The Netherlands, that develops and produces electronic consumer products. The collected data is presented in the form of so-called 'snapshots' taken weekly from a database with well-defined and well–structured data from the review process. A snapshot is a copy of the actual data collection at a particular point in time. Regarding the snapshot data a well-structured database has been developed. The data have already been used successfully in different types of research projects, see e.g. (Ahogado, 2009), (Samalikova et al, 2010).

3.1. Review process variables

The data comes from review processes which are carried out in each of the phases of the software development cycle. Every review process is carried out by a review team, e.g. in accordance with (Gilb and Graham, 1993). The data contains all kinds of information (data elements) about the review process, such as the review initiation date, the closure date, the number of participants (in a review team), the major defects found, the minor defects found, the review document unit (lines or pages), etc.

In this research some additional 'compound' variables have been defined on the basis of the review data elements mentioned, in order to analyze more aspects of the review process. We will address this further in Section 5, in the analysis and results section. These additional 'compound' variables are respectively:

- Total defects → The sum of the major defects and the minor defects.
- Time or inspection duration → The subtraction between the closure date and the initiation date. (time spent in a review).
- Inspection rate → The division between the number of pages of a document reviewed (the review size) and the time spent on the review (also called the speed of the review).
- Preparation rate → The average effort spent per page on the preparation of a review
- Defect Density → The division between the number of defects found and the number of pages reviewed.
- Effectiveness → The number of defects found per staff time. In a formula:

$$Effectiveness = \frac{Total\ defects}{(number\ of\ participants * time)}$$

3.2. Selected review process variables and research questions

In this research project the focus will be on three variables of the review process, respectively the review document size, the review speed and the review team size.

The review size

"Large documents appear to have been inspected at a high number of pages per hour. The defect density (defects per page) for such documents appears to be lower. While a number of explanations could be put forward, my favoured one is that people find it hard to check and Inspect large documents, which leads to a 'let's get on with it' attitude" (Gilb and Graham, 1993). Based on this statement a number of questions can be formulated, such as:

- Q1: What is the relation between the number of pages reviewed (per time unit) and the size of the document?
- Q2: What is the relation between the number of defects found per page (i.e. the defect density) and the size of the document?
- Q3: What is the relation between the cost of the review and the size of the document?

The review speed (or inspection rate)

"The defect density drops sharply with the increase in material coverage, reinforcing the message 'go slow'", (Gilb and Graham, 1993). So, reviewing long documents can lead to decreasing numbers of defects found. Based on this statement the following research questions can be derived:

- Q4: What is the relation between the review speed (inspection rate) and the number of defects found?
- Q5: What is the difference between major and minor defect detection, in relation to the review speed?

The review team size

In (Gilb and Graham, 1993) it is suggested that four people could form a good-sized inspection team. This is confirmed by (Shiwei, 2009) who state that four person teams are more effective and efficient than three person teams. In (Biffl and Halling, 2003) the following information is given: "For a given inspection duration there is a team size that maximises the net gain of the inspection. The diminishing added defect detection effectiveness per added inspector will at some point let the associated benefit fall to the level of added cost for this inspector". On the basis of these findings and statements the following research questions can be formulated:

- Q6: what is the relation between the number of defects found and the team size?
- Q7: what is the optimal team size for an effective review team?

4. Review data analysis and results

In this section the results will be presented of the statistical analysis on the collected data regarding the review size, the review speed and the review team size.

4.1. Review variable 1: the review size

The first relation to be analyzed is the relation between the review time and the document size (Research question Q1, see section 3.2). The aim is to determine whether reviewers go faster / slower when they have to review shorter / larger documents. For this analysis we have created the compound variable 'Inspection rate' which consists of the number of pages reviewed per time unit.

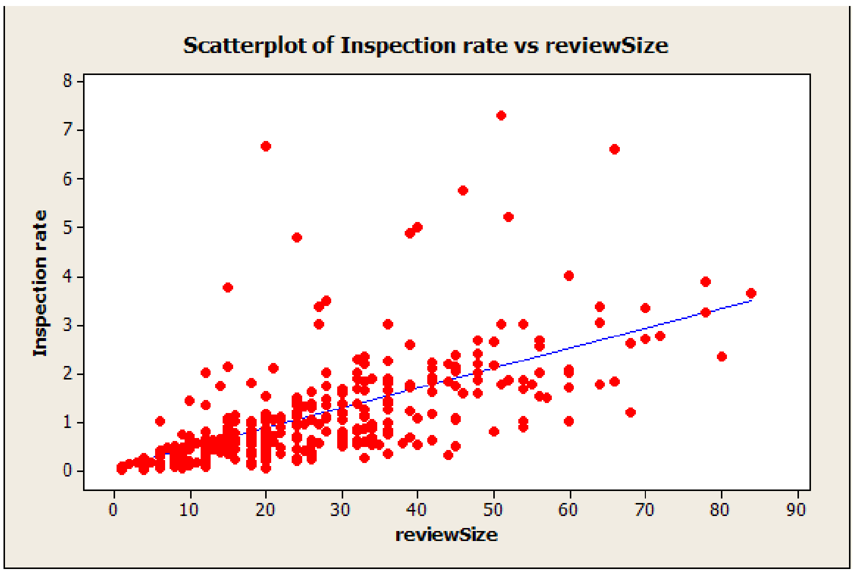

Figure 2: Inspection Rate (# pages/ time unit) against the review Size (# pages)

Figure 2 shows a positive correlation between the two variables. However, there is not a proportional increase of the time allocated in a review when the size of a document increases. The results of the regression analysis also confirm that when the size of a document increases, the time allocated per page decreases. The regression is significant (p-value<0) with a high correlation between variables (R-Sq(adj.)=40,4%).

Next we have analyzed the relation between the size of a review document and the number of defects found per page (Research question Q2, see section 4). To analyze this relation the compound variable 'Defect Density' has been defined as the number of defects found divided by the unit size (in number of pages). Figure 3 shows the result.

Figure 3 shows a decrease of the 'Defect Density' when the size of a review document increases, until it reaches a certain point (i.e. around 40 pages) and then stays rather constant.

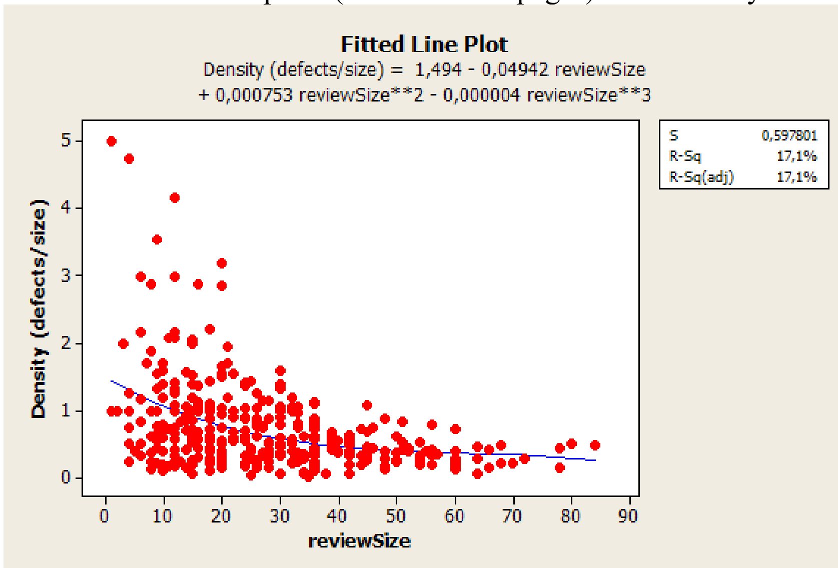

Figure 3: Defect Density (# defects / # pages) - document Size (# pages)

This means that when the number of pages in a review increases, the quality of the result (the review document and/or the software product) will decrease (because less defects per page are found). The statistical analysis details are: a good correlation coefficient (R-sq(adj)=14,5%) and significance. For the next analysis (research question Q3, see section 3.2) we defined the compound variable 'Preparation rate'. The 'Preparation rate' shows the average effort spent per page on the preparation of the review (which can be considered as an important cost aspect of a review). The aim is to get insight into how the cost of a review evolves in the case that more pages are reviewed. Figure 4 shows the result.

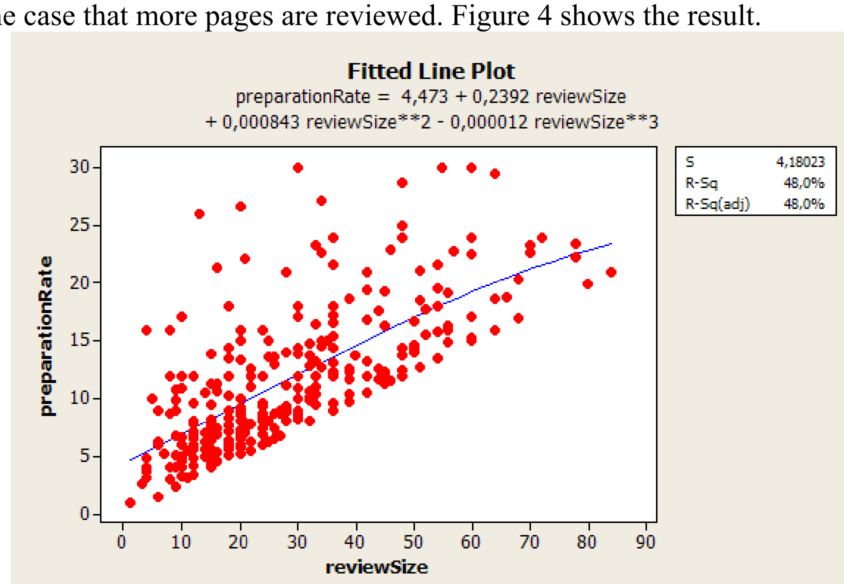

Figure 4: Preparation Rate (effort spend in the preparation / # pages) - the Size (# pages)

Figure 4 shows a clearly increase of the effort spent in the preparation of a review when the size of a document increases. This can be interpreted as that an increase in the number of pages to be reviewed will lead to higher costs per page (i.e. a higher preparation rate). The statistical analysis details are: a good correlation between the variables Preparation time and Review Size (R-Sq(adj)=48% and P-value<0,05).

Summarizing we can conclude in this subsection that it is beneficial, from different perspectives, to review short documents. From the perspective of the time spend on a review we conclude that an increasing review size leads to less time spent on a review (which influences negatively the number of defects found, and the quality of a software document). From the perspective of review costs we conclude that a larger size leads (proportionally) to higher review costs.

4.2. Variable 2: the review speed (inspection rate)

To check the relation between the Inspection rate and the number of defects found (research question Q4, see section 3.2), we made use of the numbers of defects found per page (the Defect Density). Figure 5 shows that the faster a page is reviewed less defects per page are found. The Defect Density decreases faster in the beginning until we arrive at an Inspection rate of 1-2 pages per time unit. At an Inspection rate between 2 and 3 pages a more constant Defect Density occurs. Statistical analysis details: a significance of the regression analysis with a p-value =0,00, R-Sq(adj.)= 16,3%).

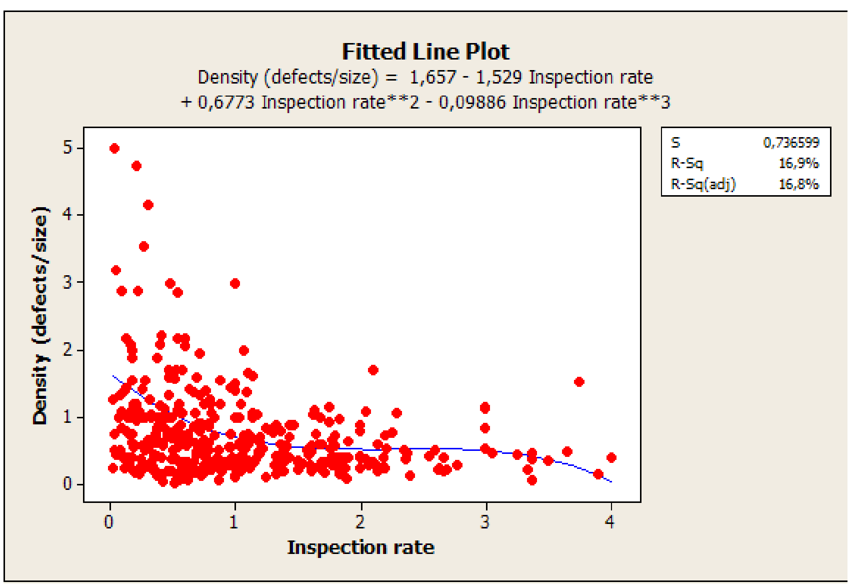

Figure 5: Defect Density (# defects/# pages) - Inspection Rate (# pages/time unit)

This analysis result confirms the statement of (Gilb and Graham, 1993) regarding the review speed. Although more pages or lines have to be reviewed: 'go slow', or do not allocate less time to a document. The higher the Inspection rate the less defects will be found (which will decrease the quality of the document and finally the quality of the software product).

To investigate the curve in Figure 5 further we carried out the following analysis (to answer research question Q5, see section 3.2). We divided Total defects into Minor and Major defects, respectively the less important defects and the most important defects that are found in a review. The aim of this analysis is to determine whether the review speed influences the finding of different types (i.e. minor and major) defects in a review. In the following Figures we show some interesting differences between the minor defects and the major defects found in relation to the Inspection rate.

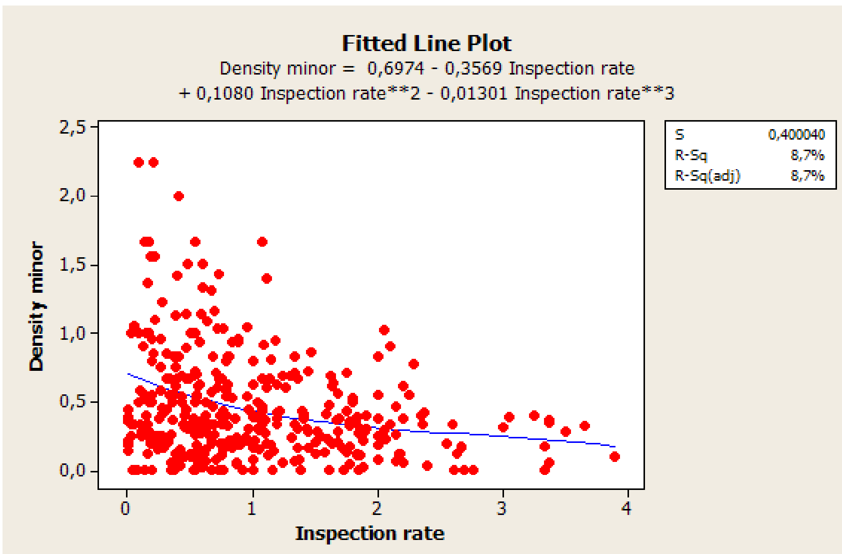

Figure 6: Density of the Minor defects (# minor defects/# pages) - the Inspection rate (# pages/time unit)

Figure 6 shows that the Minor Defect Density decreases fast with an increase of the Inspection rate (from 0 to 1,5 pages per time unit). This can be interpreted as follows. When reviewers have to review faster, they will pay less attention to minor defects reducing the number by 43%. This percentage is calculated on the basis of the inspection rate (=0) and the Minor Defect Density =0,7, and an inspection rate (=1,5-2) and a Minor Defect Density =0,4 (so: (0,7-0,4)/0,7).

In the next Figure 7 we carried out the same analysis for the major defects.

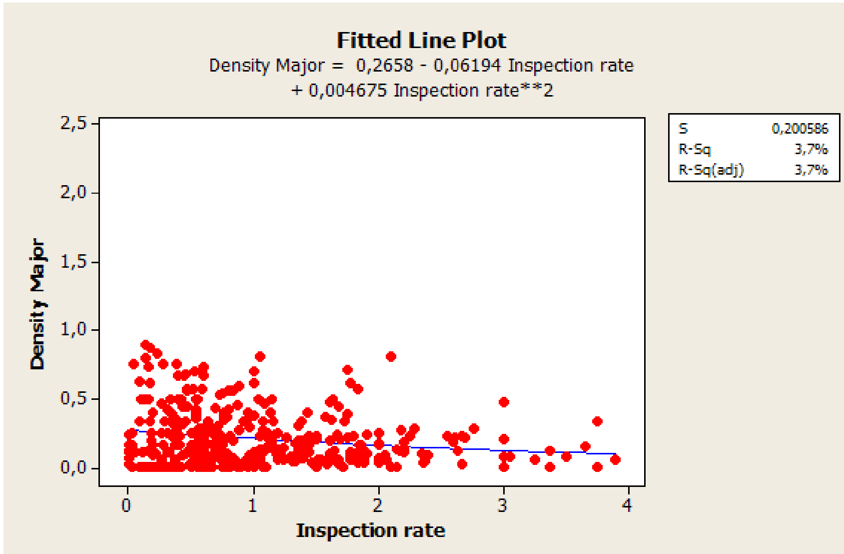

Figure 7: Density of the Major defects (# major defects / # pages) - the Inspection Rate (# pages/unit time).

The relation between the Major Defect density and the Inspection rate follows is a quite constant one. With respect to the curve from Figure 7, the strong decrease in the beginning and later the constant values can also be explained by Figure 7. The explanation is that in the beginning the curve in Figure 6 drops very fast because reviewers are much more focused on finding major defects than on finding the minors.

Summarizing it has become clear that the review speed has a significant effect on the number of defects found. So the time allocated per page in a review has to be determined carefully. Further we found that when reviewers are reviewing faster they are more focused on finding the major defects (probably taking less care of finding the minor defects).

4.3. Review variable 3: the number of participants in a review

To answer research question Q6 (see section 3.2) we analyzed the relation between the number of defects found and the team size. The result is shown in Figure 8. The regression curve first increases till 10-12 people, and then decreases. This confirms earlier research results that adding more people to a review team does not mean that more defects will be found (Biffl and Halling, 2003).

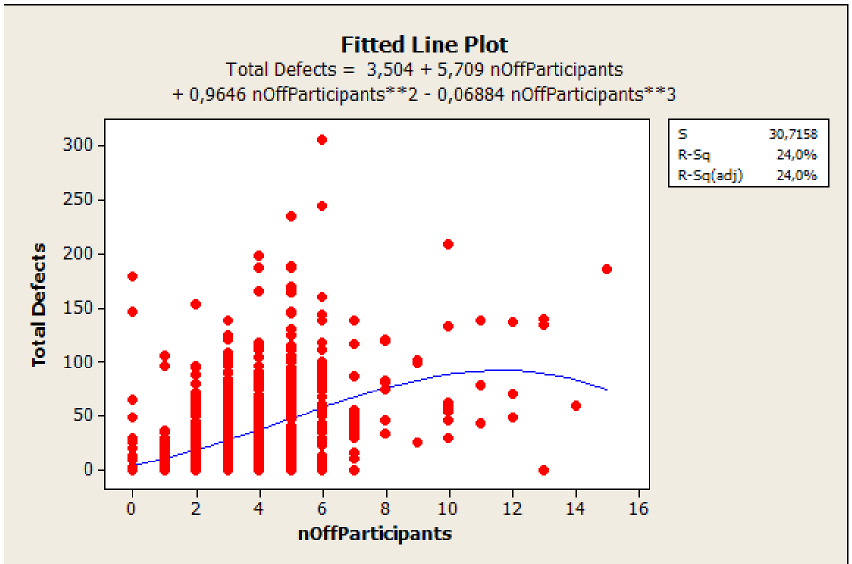

Figure 8: Total Defects (# defects) - the Team Size (# participants) (cubic approximation)

Interesting in Figure 8 is that there exist a broad range of values for the number of participants in review teams in the industrial company. Normally, the recommended team size is between 4-6 (see section 2). From the data from our company the team size can increase till 15! Statistical analysis details: significance (p-value<0), and a high correlation between the variables (R-sq(adj)=24%).

To answer question Q7, see section 3.2, and to find the optimum number of participants (i.e. the review team size) we have to focus on the balance between benefits and costs of a review. Therefore we investigated the relation between the Defect detection effectiveness (benefits) and the team size (costs). The Defect detection effectiveness has been calculated by dividing the total number of defects found (i.e. the benefits) by the (number of participants multiplied by the inspection time (i.e. the costs)). In Figure 9 we have calculated the mean Defect detection effectiveness of each team size.

On the basis of the list in Figure 9 the graph in Figure 10 could be made. Figure 10 shows the relation between the Mean effectiveness and the Number of participants.

Descriptive Statistics: Effectiveness

Variable	nOffParticipants	N	N*	Mean	SE Mean	StDev	Minimum
Effectiveness	1	2467	0	0,29535	0,00914	0,45377	0,00000
	2	7204	31	0,31567	0,00557	0,47241	0,00000
	3	9419	62	0,37698	0,00588	0,57042	0,00000
	4	8011	93	0,39843	0,00723	0,64750	0,00000
	5	3371	62	0,4859	0,0171	0,9939	0,0000
	6	1759	31	0,4137	0,0166	0,6950	0,0000
	7	558	0	0,3030	0,0131	0,3101	0,0000
	8	279	0	0,3501	0,0209	0,3492	0,0708
	9	93	0	0,2571	0,0224	0,2158	0,0415
	10	248	0	0,2383	0,0153	0,2413	0,0063
	11	93	0	0,17111	0,00779	0,07509	0,09534
	12	124	0	0,2123	0,0170	0,1896	0,0186
	13	124	0	0,1308	0,0120	0,1340	0,0000
	14	31	0	0,19156	0,000000	0,000000	0,19156
	15	31	0	0,11651	0,000000	0,000000	0,11651

*Figure 9: The Mean effectiveness (# defects/(# participants*time unit)) of each Team Size (# participants)*

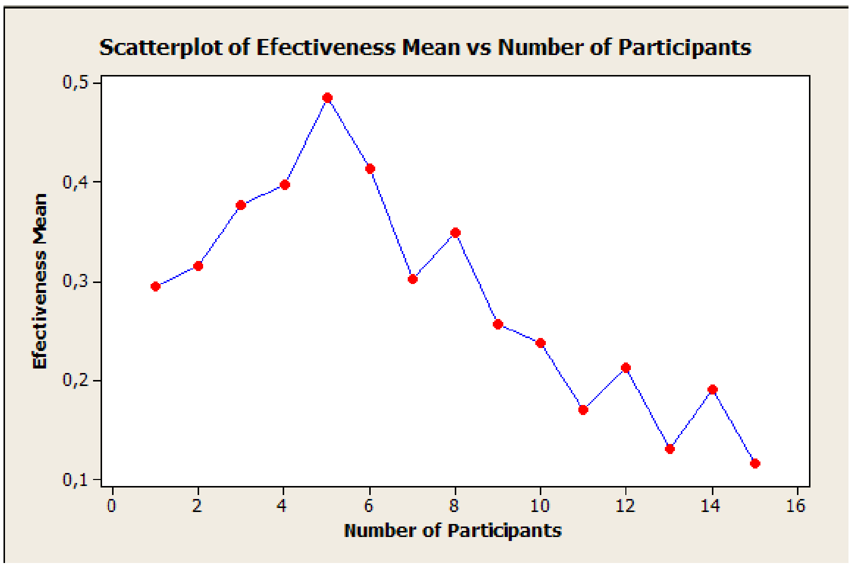

*Figure 10: Mean effectiveness (# defect found/(# participants*time unit)) - Team Size (# participants)*

Figure 10 shows an optimum team size (with a higher effectiveness). The optimum team size is 5, but also the effectiveness of teams from 4 to 6 participants is quite good. This is in conformance with (Gilb and Graham, 1993). Interesting is that the effectiveness of review teams with more than 8 participants drops sharply.

5. Conclusion

In this research we showed that a statistical analysis on real-life data from a software development company can provide interesting insights in an important quality assurance process, i.e. the review process. The three review process variables, respectively the review size, the review speed and the review team size appeared to be interesting focus points in our research to get more insight in reviews as well as to validate some (often quite intuitive) statements about review effectiveness in literature.

Regarding the review variable review size we conclude that it is beneficial to review short documents. On the one hand we found that an increasing review size doesn't lead to a proportional increase in review time, but to a lesser review time, and that consequently influences the number of defects found. On the other hand we found that the costs of a review don't decrease in case that the review size increases.

With respect to the review speed a significant effect has been discovered regarding the number of defects found. We concluded that the time allocated per page in a review has to be determined carefully. Further we found also that when reviewers are reviewing faster they are more focused on finding major defects, taking less care of finding minor defects.

Regarding the team size we determined for our company a clear optimum for a team size (with a higher effectiveness). This optimum team size is 5, however the effectiveness of teams from 4 to 6 participants is also quite good. This is more or less in conformance with literature. Interesting is that the effectiveness of review teams with more than 8 participants drops sharply.

Summarizing we can conclude that a statistical analysis on well-collected data form review processes can provide interesting insights into the actual review process of an industrial company. However, this will not lead to a 'one big step' improvement of the review process in that company. The results have to be discussed subsequently with the different stake-holders, such as the responsible software management, the software engineers, and the quality assurance staff. On the basis of these discussions the applied review variables and the found insights have to be validated, and eventually actions can be defined to improve the review process further, e.g. with respect to the review data to be collected, the review variables to be defined and new research questions identified.

6. References

[1] Ahogado D.C., 2009. Setting up quality database for analysis of Software Development Process Information. Internal report, University of Technology Eindhoven.

[2] Aurum A., Petersson H., Wohlin C., 2002. "State-of-the-Art: Software Inspections after 25 Years", Software Testing Verification and Reliability, Wiley Online Library, Vol. 12, No. 3, pp. 133-154.

[3] Biffl S., Halling M., 2003. Investigating the Defect Detection Effectiveness and Cost Benefit of Nominal Inspection Teams, IEEE Transactions on Software Engineering, Vol. 29, no. 5.

[4] DeMarco T. and M. Lister, 1999. Peopleware: Productive Projects and Teams (Management Can Make Quality (Im)possible). Dorset House Publishing company.

[5] Ghezzi C, Jazayeri M, Mandrioli D., 2003. Fundamentals of Software Engineering. Pearson Education, ACM Digital Library.

[6] Gilb T., 2004. Rule-Based Design Reviews, in: Fundamental Concepts for the Software Quality Engineer, S. Carroll, T. Daughtrey. Volume 2, ASQ Quality Press, 2007.

[7] Gilb T. and Graham D., 1993. Software Inspection. Addison-Wesley.

[8] Kanem C., Bach J., Pettichord B., 2001. Lessons Learned in Software Testing: A Context-Driven Approach, John Wiley and Sons.

[9] Putman L.H., Myers W., 1992. Measures For Excellence: Reliable Software On Time, Within budget. Prentice Hall.

[10] Samalikova J., Kusters R., Trienekens J., Weijters T., Siemons P., 2010 Towards objective software process information; experiences from a case study, Software Quality Journal.

[11] Shiwei D., 2009. Defect Prevention and Detection of DSP-Software, World Academy of Science, Engineering and Technology, Volume 58.

[12] Softdev Team, 2006. www.softdevteam.com/V-shaped-lifecycle.asp (accessed in June-2010).