

Entropy based software processes improvement

Jos J. M. Trienekens · Rob Kusters ·
Dirk Kriek · Paul Siemons

Published online: 1 October 2008

© The Author(s) 2008. This article is published with open access at Springerlink.com

Abstract Actual results of software process improvement projects show different levels of success. Although many software development organisations have adopted improvement models such as CMMI, it appears to be difficult to improve software development processes in the right way, e.g. tuned to the actual needs of the organisation. This paper presents a new approach to determine the direction of improvement for an organisation. This approach is based on an elaboration of the concept of entropy. The approach is empirically validated by carrying out interviews in 11 software development organisations in The Netherlands. The results of the research show that software development organisations can be classified and can be positioned on the basis of their internal and external entropy, c.q. the level of (dis)order in the business system and its environment. Based on a possible out-of-balance situation between the internal and external entropy, directions for software process improvement can be discussed. As such the proposed approach can support the application of current software process improvement methodologies such as the CMMI.

Keywords Software process improvement · Entropy · Empirical research results

J. J. M. Trienekens (✉) · R. Kusters
Eindhoven University of Technology, Den Dolech 2, 5600 MB,
Eindhoven, The Netherlands
e-mail: j.j.m.trienekens@tm.tue.nl

R. Kusters
e-mail: r.j.kusters@tm.tue.nl

D. Kriek
Quality House B.V., 3404 LL, IJsselstein, The Netherlands
e-mail: d.kriek@qualityhouse.nl

P. Siemons
3958 ZT Amerongen, The Netherlands
e-mail: paul.siemons@draugronth.nl

1 Introduction

To be competitive in the current economy, more and more software development organisations strive to improve their processes. This is based on convictions that an improvement of processes ultimately leads to an increased quality of software products. Over the last 10 years several improvement models have been developed that act as frameworks and tools to improve software development processes, see Balla et al. (2001). Capability Maturity Models (CMM, CMMI), see SEI (2001, 2006), Paulk et al. (1993, 1995), are being applied increasingly in the software engineering domain. The starting point for process improvements is a so-called process assessment, i.e. an analysis and subsequent determination of the current performance level of the software development process. On the basis of the outcome of an assessment, an improvement project can be initiated. However, the actual results of improvement projects are based on limited sets of data and are difficult to quantify (Ruiz et al. 2001). Case study reports and success factor investigations offer until now only a limited insight in the way software development organisations can and/or should try to improve their software development processes.

A closer look at assessment and improvement methodologies shows that they are strongly oriented to the internal processes of an organisation. They focus for example on internal organisational factors such as management commitment, on internal human factors such as training, on internal process factors such as project planning, on internal technical factors such as tool support, and on financial factors such as project budgets (Habra et al. 2007; Von Wangenheim et al. 2006). In Ngwenyama and Neilsen (2003) contradictory sets of assumptions about organisational culture in the CMM approach are revealed, however only organisation-internal change aspects are addressed. As a consequence of this internal orientation it can be concluded that current software process improvement projects pay little attention to the influence of the environment of a software development organisation when deciding on the way the software development processes could be improved (Boldyreff et al. 1997). External factors such as customer satisfaction are only very rarely taken into account in software process improvement projects, see e.g. Kuilboer and Ashrafi (2000).

This paper introduces and validates a new concept for software process improvement, i.e. the entropy balance of an organisation. The kernel of this concept is the distinction between the internal and external entropy of a software development organisation. Entropy is a cybernetic concept that stands for the level of disorder of a system (Boltzmann 2000). The main assumption in this paper is that an organisation should be aware of not only its own internal disorder but also its external disorder. Differences between the internal and external disorder, i.e. the entropy balance, should play a role in the process of choosing a direction for process improvement.

In this paper the first steps have been made to define and operationalise these theoretical concepts of internal and external entropy. This has been done in a structured process of identification and assessment of relevant internal and external factors of software development organisations. Based on a selection of these internal and external factors, the degree of internal and external entropy, and the difference between them, the so-called ‘improvement space’, has been determined in 11 software development organisations. Within its own ‘improvement space’ an organisation can strive at a ‘better’ balance between its internal and external entropy. The information about the improvement space is considered to be of importance for an efficient and an effective software process improvement program, e.g. a CMMI project.

Due to the exploratory nature of the work, the decision was taken to base the project on collaboration with experts from the field. The participation was obtained of six senior IT

consultants with a long-year experience in software process improvement, in varying industrial domains, such as the high-tech software-intensive domain, the banking and insurance domain, and the discrete manufacturing domain. During a period of 6 months the group discussed the concept of entropy, and its elaboration towards software engineering practice, in close collaboration with two senior researchers. In regular meetings, every two weeks, the concepts of internal and external entropy have been elaborated and the characteristics of organisations, which influence their entropies, have been investigated. Consequently steps have been made by the research group regarding the quantification of the factors that influence the entropies in terms of measurement mechanisms, and the development of a questionnaire in order to collect data from practice to validate the application of the elaborated entropy concept. The questionnaire has been used in 11 organisations to collect entropy data following the format of a structured interview. The ultimate goal is, on the basis of the empirical results, to further elaborate and enrich the concept of entropy in the software engineering domain, and to develop a well-founded questionnaire-based entropy instrument to support organisations in their software process improvement programs.

This paper deals in Sect. 2 with software process improvement and its limitations, and introduces the concept of entropy. Section 3 focuses on the first steps towards identification and metrification of the concepts of the internal and external entropy. In Sect. 4 the results are presented of an empirical research project in that the entropy of 11 software development organisations has been investigated. Finally, in Sect. 5 conclusions and recommendations for future research are given.

2 SPI and internal and external factors

In the software engineering domain, improvement activities are known as Software Process Improvement (SPI). SPI is being defined as:

The measurement-based improvement of the performance of the software development process aimed at delivery in due time, against agreed budgets, and with the required quality. Humphrey (2001)

A well-known and broadly accepted model for the improvement of software development processes is CMM (Capability Maturity Model). CMM prescribes a specific sequence of improvement activities that have to be followed to reach higher levels of software process maturity. CMM and comparable maturity models are used on the one hand to strive at a standardisation of processes and on the other hand to realise a decrease of product failures by eliminating their causes. However, it is questionable whether these improvement models can be applied in the same way for each and every organisation. For instance, an organisation that has to deal with an increasingly turbulent market, with changing needs, and that has improved its development processes by defining, standardising, and in some cases bureaucratising them in detail, can face difficulties in responding to the dynamic requirements of that market. As a result there could be a ‘mismatch’ between the (internal) goal of the software process improvement program of the organisation and the needs of the market. Examples of other factors in the environment of an organisation that can influence its behaviour are changing governmental and legal factors. However, these types of external factors have not until now been addressed by software process improvement methodologies, e.g. Ngwenyama and Neilsen (2003). This paper proposes that both internal and external factors should be taken into account during process assessment the subsequent determination of process improvement activities.

Table 1 Some examples of characteristics of organisations with high and low entropies

High external entropy	Low external entropy
Environment of an organisation is unstable and unpredictable	Environment of an organisation is stable and predictable
Organisation should respond with flexible and adaptable processes	Organisation should respond with standardisation and formalisation
High internal entropy	Low internal entropy
Organisation has ad-hoc and chaotic processes	Organisation has standardised and well-defined processes
Organisation can be flexible and can follow quickly changing external needs	Organisation has strong procedural performance but difficulties in responding to changing external needs

In order to make it possible to address external and internal factors in process assessment, the concept of external and internal entropy is introduced. Entropy is a concept by which the internal and external situation of a system can be characterised (Boltzmann 2000). A low entropy means a high level of order, or structure and stability, in a system. A high entropy means a low, or even chaotic, level.

Based on these characteristics, organisations can be considered as dynamic systems that are continuously moving and changing, and that have to strive at a balance between their internal and external entropy. For example, in the case of a market that is very dynamic, then a company in that market is, as a consequence, faced with a high external entropy and has to be able to respond quickly and adequately to that market, in other words, the company must be flexible. The research group of senior consultants and researchers have investigated in close collaboration the characteristics of organisations and the external and internal factors that influence the internal and external entropy. Table 1 gives some examples of characteristics of organisations with high and low internal and external entropies. These rather abstract characteristics are used as a basis for making the concept of entropy more operational.

Based on information about the internal and external entropy, an organisation can decide to follow a particular direction regarding software process improvement. For example, to increase its own flexibility, an organisation could be forced in principle to increase its internal flexibility so that it becomes able to respond to the (dynamic) market. Such an organisation might have to lower or drop certain formal standardisation rules and to increase the level of decisional freedom of employees in the development processes. This dropping of standardisation rules and an increase of decisional freedom of employees, could then become the main theme for software process improvement in this type of organisation.

3 Measuring entropy

To be able to use the concept of internal and external entropy in a practical way it is necessary to make entropy measurable. In accordance with Boltzmann (2000) entropy is based on the number of states that a system can have. In order to make entropy operational the state of a system has been linked to basic size variables. We considered organisations as business systems, consisting of interrelated components. The entropy of such a business system increases with:

- an increasing number of components;
- an increasing number of interactions between the components;
- an increasing number of changes in the environment.

From discussions on increasing and decreasing entropies, the research group derived two concepts that play a central role in the determination of entropy: *complexity* and *dynamics*. These two concepts are described as follows:

- complexity is a measure of the number of interacting components;
- dynamics is a measure of the amount of changes over time.

3.1 Entropy and its relations to complexity and dynamics

The research group then tried to make entropy measurable on the basis of these concepts of complexity and dynamics. In principle, measurement is the linking of a value to a parameter, by positioning the parameter on a measurement scale. The research group has discussed intensively the applicability of the four types of measurement scales: ratio, interval, ordinal and nominal. In this study it was finally decided to measure complexity and dynamics by using a four value measurement scale with the levels: low, average-low, average-high and high, see Table 2. Of course it was recognised that this measuring mechanism is only a first step towards the quantification of entropy. Based on further empirical investigations the applicability of the mechanism should be validated and improved.

Regarding the measurement of complexity a number of business system characteristics have been investigated such as the number of employees per business function, the number of functions per department, the number of departments in the organisation, etc. Based on numbers for these parameters, organisations can be classified. For example the number of employees can be used to classify an organisation as small (1–9), average-small (10–99), average large (100–999) or large (1000+). This particular classification is also a result of the study and based on several intensive workshop sessions of the research group.

Similarly, it was decided how the dynamics of a business system can be measured. For example a classification of an organisation regarding its dynamics, has to be determined on the basis of the number of process changes per time-frame. To determine the number of changes of a specific parameter, such as employees, information is needed on the input, the output, and the current number of the employees. Other parameters that resulted from the research process are departments, products, customers, suppliers, competitors etc. Consequently a total score for entropy has to be calculated by combining the scores for dynamics and complexity, see Table 3.

3.2 Entropy and its relations with business system aspects

The next step in the study was the development of a questionnaire to determine the internal and external entropy of a software development organisation on the basis of empirical data. For this, a questionnaire was developed as a suitable means for the collection of specific data from practitioners about their business systems. As a result of the process in this study

Table 2 Levels of complexity

Complexity	Definition
Low	Almost all parameters have a low score
Average-low	Most parameters have a low score
Average-high	Most parameters have a high score
High	Almost all parameters have a high score

Table 3 Determination of entropy on the basis of measurements of complexity and dynamics

Entropy	Complexity			
	Low	Average-low	Average-high	High
Dynamics				
Low	Low	Low	Average-low	Average-low
Average-low	Low	Average-low	Average-low	Average-high
Average-high	Average-low	Average-high	Average-high	High
High	Average-high	Average-high	High	High

Table 4 (Sub-)aspect of market, organisation and factory

	Sub aspect	Complexity	Dynamics
External entropy		see e.g. Table 5	
Market	Product		
	Design		
	Architecture		
Internal entropy			
Organisation	Process		
	Structure		
Factory	People		
	Resources		
	Technology		

the structure of the questionnaire has been based on a subdivision of the concept of business system into Market (external entropy), Organisation and Factory (internal entropy). Subsequently these aspects are further subdivided as follows: Market into Product, Design, and Architecture; Organisation into Process and Structure, and Factory into People, Resources and Technology. Table 4 summarises the basic structure of the questionnaire. The objective of the questionnaire is that for each of the three aspects and sub-aspects the complexity and dynamics can be determined. The questionnaire enables the collection of data on the sub-aspects of a business system. Based on this complexity and the dynamics, and finally the internal and external entropy of a business system can be determined.

Table 5 provides an example, addressing the complexity of the Market (external entropy), and its sub-aspect Product, to illustrate the level of detail of the questions of the questionnaire.

The initial questionnaire for carrying out the structured interviews in practice consisted of 65 questions. These questions have been elaborated, by the senior consultants in collaboration with the researchers, during several workshops. Two pilot interviews have been carried out by a researcher and a consultant to test the applicability of the questions. Subsequently some questions have been removed and some others have been reformulated.

In the questionnaire the (sub-)questions about the business system aspect Market and its sub-aspect Product, lead to scores from which the external entropy can be determined. The questions about the business system aspects Organisation and Factory lead to scores from which the internal entropy can be determined. By comparing the scores of the external entropy and the internal entropy it becomes possible to position organisations in a two-dimensional matrix, see Fig. 1.

Table 5 Example of a question about the complexity of the Market

COMPLEXITY (of market on the basis of the aspect product)

How many parties are involved in the development of the software products (e.g. as supplier of architectures, designs and/or components)

Quantification:

How large is the involvement of third parties?

 Percentage components delivered?

 Number of external parties that contributes to an architecture and/or design?

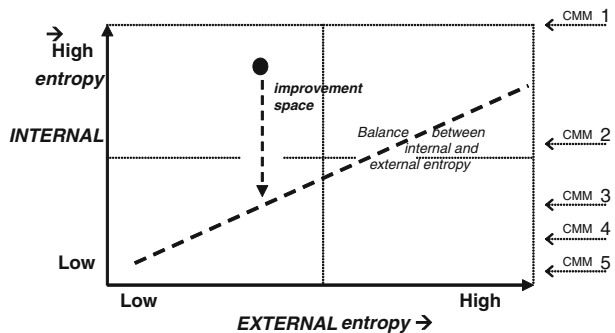
 Amount of time that external parties contribute?

What is the average effort that external parties contribute?

 Average number of employees per external party/number of employees intern

 Total number of employees of external parties/total number of employees in the project

Fig. 1 Space for improvement for software development organisations



A difference between the internal and external entropy shows, what has been called initially, the ‘improvement space’. The concept of ‘improvement space’ indicates that an organisation should strive at a balance between internal and external entropy. The larger the distance an organisation has to the balance-line, the larger the space for improvement is, see Fig. 1.

In Fig. 1 the balance-line is represented by the dotted black line. The position of this balance-line is currently based on the usage of the data collected (i.e. the 11 companies). This means that we assume that on the average these companies have a ‘certain’ balance between internal and external entropy. It has to be emphasised that this is just a first step to determine this balance-line. Further research, e.g. to improve the conceptual basis of entropy, the measurement mechanisms and the questionnaire, and the collection of more data from organisations, is needed to determine the balance-line more precisely.

In Fig. 1 the bullet represents an organisation with a particular distance to the balance-line. This organisation has a relatively high level of internal entropy and a relatively low level of external entropy. Such an organisation should discuss this out-of-balance situation internally. As a result the organisation could decide to emphasise in its software process improvement activities a decrease of its internal entropy, for example by striving at a certain level of standardisation of its processes. By doing so, the internal entropy will decrease until a balance is reached. CMM levels are shown on Fig. 1 but have been deduced somewhat intuitively. Notice that on CMM5, which is the highest maturity level, it is assumed that the internal entropy is extremely low. This means that a software development organisation has an extremely high level of internal organisational order. On

the contrary, on the lowest level of CMM, an organisation has, as also has been stated in literature, e.g. SEI (2006), quite often an ad-hoc and chaotic software development process. In that case the internal entropy, i.e. the internal level of disorder, can be assumed to be very high.

4 Results of the research: space for improvement for software development organisations

The questionnaire has been applied in an empirical research project in which 11 software development organisations in The Netherlands were involved. The questionnaire was answered during an interview so as to improve the quality of the answers. In each organisation both a person responsible for software process improvement and a representative product manager have been interviewed. The software process improvement experts, with a strong internal orientation, all had more than 5 years experience in the field. The product managers, with a strong orientation on the environment (or: Market factors) of the organisation had about 10–15 years of relevant experience. Each interview has been carried out by two persons: one senior consultant and one researcher. The interviews took about 1.5 h per interview. The ‘entropy results’, on the basis of answers to the questions and quite some clarifications, have subsequently been reviewed and validated by the interviewees.. In the introduction of the interview sessions the interviewees have been asked to answer some more general questions. One of these questions was to describe the level of experience with software process improvement of the organisation. In Sect. 4.1 we present first these experience levels. In Sect. 4.2 we will present and discuss the obtained entropy scores.

4.1 Experience with process improvement

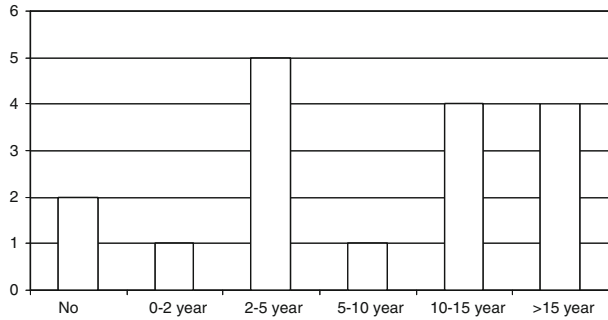
The 11 organisations that participated in this research project are all based in The Netherlands and active in various industrial domains. In the following an overview is given of the organisations and the domain in that they perform, respectively:

- one organisation from the semi-conductor domain (A)
- one organisation from the business software applications domain (B)
- one organisation from the defence system domain (C)
- one organisation from the domain of on-line photo imaging service (D)
- two organisations from the medical system domain (E and F)
- one organisation from the domain of consumer electronics (G)
- one organisation from the climate and process control domain (H)
- one organisation from a technical research application domain (I)
- one organisation from the banking and insurance domain (J)
- one organisation from the domain of copying and printing technology (K)

The identification code of each organisation will be used in Fig. 3 to clarify the entropy balance of the different organisations.

In this paper no explicit information is given on the CMM levels of the organisations mentioned above. Only some of the organisations were striving at a higher level of maturity in a formal way, i.e. based on an independent assessment of an external organisation. Other organisations tried to follow an own direction and approach to improve their level of maturity, and some organisations refused to make use of CMM or CMMI. As

Fig. 2 Experiences with software process improvement



a consequence the different situations were hardly comparable from the viewpoint of the ‘reached’ CMM level.

Figure 2 shows the experience with software process improvement in the respondent organisations. In total 17 interviewees, out of 22 (two per organisation), could answer this question and the result suggests that 80% of the respondent organisations are experienced in SPI.

4.2 Positioning organisations on the basis of their entropy scores

Based on answers from the questions regarding the complexity and the dynamics of the three aspects Market, Organisation and Factory, the internal and external entropy of the 11 organisations has been determined. Figure 3 shows the results. Each code letter refers to an individual organisation as presented in the beginning of this section. Our analysis shows that organisations can have a totally different internal/external entropy balance. As a consequence different organisations could follow rather different directions regarding the improvement of their software development processes In Tables 6–9, the characteristics of four organisations, one from each of the four quadrants of Fig. 3, denoted as organisation A, B, C and D, have been specified and their ‘space for improvement’ has been clarified.

Organisation A operates in a complex and dynamic market. But also the internal organisational characteristics show a relatively high (internal) entropy. As a consequence Figure 3 shows for this organisation a deviation regarding the balance-line of internal/external entropy. Based on this information organisation A could decide to strive at a

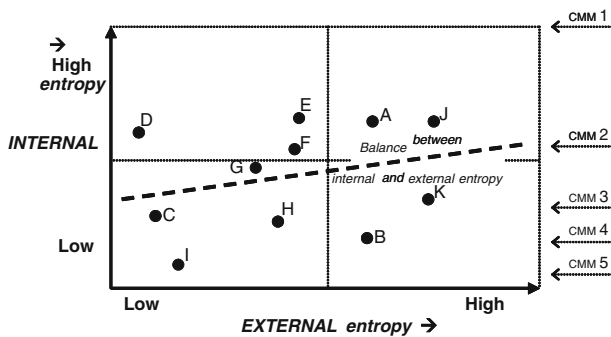


Fig. 3 Positioning of software development organisations on the basis of their internal and external entropy scores

Table 6 Characteristics of organisation A

Organisation A	Complexity	Motivation	Dynamics	Motivation
Market	H	Various products for various markets	H	Product portfolio changes rapidly
Organisation	H	Many interrelations with suppliers	H	Many changes in collaborative processes
Factory	H	Large diversity in resources and people (skills)	H	Emerging new technologies

Table 7 Characteristics of organisation B

Organisation B	Complexity	Motivation	Dynamics	Motivation
Market	H	Complex product portfolio	HH	High pressure of market which asks for lead time reduction in combination with innovative products
Organisation	L	Rather rigid formal matrix organisation	L	Stable structure and standardised processes
Factory	L	High level of standardisation	HH	Many job changes, fast emerging technologies

Table 8 Characteristics of organisation C

Organisation C	Complexity	Motivation	Dynamics	Motivation
Market	L	Stable and mature market with restricted product portfolio	L	Long lead time per product
Organisation	M	Moderate level of maturity of business processes	LL	Rather stable processes, only few changes per time interval
Factory	H	High level of standardisation of all resources is a necessity	L	Few changes in resources to be applied

Table 9 Characteristics of organisation D

Organisation D	Complexity	Motivation	Dynamics	Motivation
Market	L	Limited product portfolio, relatively low complexity of design and architecture	M	Influence of customers on product portfolio increases.
Organisation	L	Flat, formal organisation	HH	Many reorganisations necessary due to market pressure
Factory	H	Low level of standardisation	H	Allocation of (new types of) resources in the business processes is emerging

decrease of its internal entropy, e.g. by means of a further standardisation of its development processes. Of course it could also be decided that the organisation withdraws from too dynamic market-segments, in order to decrease its external entropy.

Organisation B has a relatively low level of internal entropy, which means that there exists quite some order in their development processes, but it operates in a market which is highly complex and dynamic. Based on this information organisation B could choose for more flexibility in their software development processes. This could be reached by a decrease in process standardisation, e.g. allowing the organisation to move to a lower level of maturity (of course the question is justified whether a further improvement of the level of maturity to CMM level 5, i.e. the optimized level, also could lead to more flexibility).

Table 8 shows an organisation in a quite stable market situation. Although the software development process of organisation C is quite complex, this process is stable and doesn't show many changes over time. However, to improve the identified out-of-balance entropy, organisation C could consider to allow the organisation not to strive at a higher level of maturity, or even to decrease its current level, because of the absence of pressure for further process improvement from the (stable) market situation.

Organisation D shows a rather high level of internal entropy, i.e. internal disorder, which can cause problems regarding the quality of the service they provide to the market. Although the market situation seems to be stable it could be wise to strive at a better entropy balance, e.g. by increasing the level of maturity of the software development processes.

Different software development organisations show different types of out-of-balance situations regarding their internal and external entropy. Each of these organisations should be aware of the characteristics of these out-of-balance situations during, or preferably before carrying out, software process improvement programs. The characteristics of the entropy out-of-balance situations can be of utmost importance to decide for a particular direction for improvement. The results from the study show that it is not always necessary for a software development organisation to strive at the highest level of maturity of CMM. Even a decrease in the level of maturity can be justified on the basis of the entropy data of an organisation.

5 Conclusions

This paper presents the hypothesis that software development organisations can be classified and can be positioned on the basis of their internal and external entropy. The so-called internal/external entropy balance can be used to determine directions for software process improvement. This hypothesis is elaborated in collaboration between researchers and IT-practitioners, i.e. senior consultants. The abstract concept of entropy has been made operational by means of discussions between researchers and practitioners in a series of regular workshops over a period of 6 months. Different steps in the study have been recognised, i.e. the distinction between internal and external entropy, the identification of the (sub-)aspects of business systems that can be used to determine the internal and external entropy, the measurement mechanisms, the questionnaire to collect the actual data from practice to determine finally the (eventually out-of-balance) entropy situation. Although the research group is confident about this 'first' conceptual elaboration, the group is also convinced that on the basis of the results of the application of the questionnaire, the conceptual basis should be improved further. In particular the scientific foundation of the (sub-)aspects of the internal and external entropy, i.e. Market, Organisation and Factory, needs further improvement. Also the measurability of the mentioned aspects, and in particular their complexity and dynamics, needs a further elaboration. However, the results from the application of the questionnaire in 11 organisations have also provided these organisations with valuable information regarding their software process improvement situation.

The research results show that software development organisations can have quite different internal/external entropy balances, and this can lead to quite different decisions for the direction to be followed in software process improvement.

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

- Balla, K., Bemelmans, T., Kusters, R., & Trienekens, J. (2001). Quality through Managed Improvement and Measurement (QMIM): Towards a phased development and implementation of a Quality Management System for a Software Company. *Software Quality Journal*, 9, 177–193. doi:10.1023/A:1013301503616.
- Boldyreff, C., Drost, M., Hinley, D., Morrell, M., & Symons, A. (1997). A basis for process improvement in application management. *Software Quality Journal*, 6, 99–111. doi:10.1023/A:1018547615730.
- Boltzmann, L. (2000). Entropie und Wahrscheinlichkeit, Ostwalds Klassiker, Bd 286.
- Habra, N., Alexandre, S., Desharnais, J., Laporte, C., & Renault, A. (2007). Initiating software process improvement in very small enterprises. Experience with a light assessment tool. *Information and Software Technology* (in press, corrected proof, available online September 2007).
- Humphrey, W. S. (2001). A software maturity framework. In R. Hunter & R. Thayer (Eds.), *Software process improvement* (pp. 53–63). Los Alamitos, CA: IEEE Computer Society.
- Kuilboer, J. P., & Ashrafi, N. (2000). Software process and product improvement: An empirical assessment. *Information and Software Technology*, 42, 27–34. doi:10.1016/S0950-5849(99)00054-3.
- Ngwenyama, O., & Neilsen, P. (2003). Competing values in software process improvement: An assumption analysis of CMM from an organizational culture perspective. *IEEE Transactions on Engineering Management*, 50(1), 100–112. doi:10.1109/TEM.2002.808267.
- Paulk, M., Garcia, S. M., Chrissis, M. B., & Bush, M. (1993). Key practices of the capability maturity model, Version 1.1, Technical report CMU/SEI-93-TR-25. Software Engineering Institute, Pennsylvania, Pittsburgh, Carnegie Mellon University.
- Paulk, M. C., Weber, C. V., Curtis, B., & Chrissis, M. B. (1995). The capability maturity model: Guidelines for improving the software process. The SEI Series in Software Engineering, ed. C.M.U.S.E. Institute. Reading, MA: Addison Wesley Longman, Inc.
- Ruiz, M., Ramos, I., & Toro, M. (2001). A dynamic integrated framework for software process improvement. *Software Quality Journal*, 10, 181–194.
- SEI. (2001). CMMI. Capability Maturity Model[®]. Integration (CMMI SM), Version 1.1. Technical report. <http://www.sei.cmu.edu/cmmi>.
- SEI. (2006). Carnegie Mellon Software Engineering Institute. <http://www.sei.cmu.edu/cmmi/>.
- Von Wangenheim, C. G., Anacleto, A., & Salviano, C. F. (2006). Helping small companies assess software processes. *IEEE Software*, 23(1), 91–98. doi:10.1109/MS.2006.13.

Author Biographies



Jos J. M. Trienekens (1952) is an Associate Professor at TU Eindhoven (University of Technology, Eindhoven) in the area of ICT systems development. He is responsible for a research program on ICT driven business performance and is an associate member of the research school BETA at TUE that focuses at operations management issues. Jos Trienekens published over the last 10 years various books, papers in international journals and conference proceedings in the domains of software quality and software process improvement. He joined several international conferences as PC member and member of the organisation committee. He is also an experienced project partner in various European projects.



Rob Kusters (1957) obtained his master degree in Econometrics at the Catholic University of Brabant in 1982 and his Phd. in operations management at Eindhoven University of Technology in 1988. He is professor of 'ICT and Business Processes' at the Dutch Open University in Heerlen where he is responsible for the master program 'Business process and ICT'. He is also an associate professor of 'IT Enabled Business Process Redesign' at Eindhoven University of Technology where he is responsible of a section of the program in management engineering and is an associate member of the research school BETA which focuses at operations management issues. He published over 70 papers in international journals and conference proceedings and co-authored five books. Research interests include enterprise modelling, software quality and software management.



Dirk Kriek (1953) was trained and graduated (1969) as a biologist at the Royal University of Utrecht. Since 1985 he has chosen for a career in the ICT and was for some years director and co-owner of a software development company. He has many years of experience in software development and testing and is a certified ISEB practitioner in software testing. Since 1993 he has been involved in consultancy, management and training and was co-developer of the Verification & Validation Maturity Model (V2M2). From 1999 till 2006 he was manager R&D for Qualityhouse, a company dedicated to ICT process and product quality. Currently he works as an independent consultant and is frequently involved in test process improvement projects and is advising management about test strategies.



Paul Siemons has a very broad interest in science, technology and the arts. After studying Physics, Math and Computer Science he started his career in the field of technical automation in the areas of engineering, project management and process improvement. As an improvement consultant, he is experienced in controlling process performance and he has expertise on estimation, measurement and analysis. Over the years he switched from management to consulting and founded his own company Metrific Management Consult in 2002. Since Januari 1st, 2007, Paul is operating from Draugronth BV. Paul also actively participates in scientific research performed by several university departments and is coauthoring articles on measurement and improvement related subjects.