The Role of GQM in the PROFES Improvement Methodology

Dirk Hamann, Dietmar Pfahl, Janne Järvinen, and Rini van Solingen

ABSTRACT

In the ESPRIT project PROFES (PROduct-Focused Improvement of Embedded Software processes) a goal-driven process improvement methodology has been developed that combines and enhances methods like goal-oriented measurement, product assessment, process assessment, and process modelling. So far, the PROFES improvement methodology has been applied in multiple projects at three industrial software development organisations. In all three organisations, considerable product quality and process improvements have been achieved.

A fundamental element of the PROFES improvement methodology is goal-oriented measurement conducted according to the principles of the Goal/Question/Metric (GQM) paradigm. In the PROFES improvement methodology, GQM is used for several purposes: (1) characterisation and evaluation of product quality, (2) characterisation and evaluation of process performance, (3) modelling and evaluation of product-process dependencies, and (4) facilitation of continuous assessment. In the course of the PROFES project, GQM was also used to evaluate the PROFES improvement methodology in all three industrial software development organisations.

This paper outlines the PROFES improvement methodology and reports experience with its application in three software development organisations. The different roles of GQM in the PROFES improvement methodology are presented in detail.

KEYWORDS: Goal-Oriented Measurement, Process Improvement, GQM, Software Process Assessment, Embedded Software, Product-Process Dependency, Continuous Assessment, PROFES

1 INTRODUCTION

GQM (Goal/Question/Metric) is a well-known and widely used method for defining and executing goal oriented measurement programmes. It originated from the work lead by Prof. Victor Basili at the University of Maryland and NASA Software Engineering Laboratory in 1980's [3]. It has since been further formalised and developed into a practical methodology ([1], [8], [10], [19]).

GQM has been used in several European projects ([4], [6], [9], [15]). The ESPRIT project AMI has integrated its own version of the GQM paradigm with CMM style assessment, and developed tools and manuals to support the AMI method. The ESSI CEMP project introduced GQM in three industrial companies and cost benefit analysis of introducing GQM was one of the key topics of the project. The results were very positive and resulted in many improvements in the applications [15]. Especially Schlumberger Retail Petroleum Systems (RPS) (in 1998 RPS was sold to Tokheim) has widely reported GQM experience ([7], [15], [19]).

GQM has been used by Nokia Mobile Phones in collaboration with VTT Electronics in the DSP-ACTION project for improving DSP (digital signal processing) software development [18], which is a rapidly growing area in the domain of deeply embedded real-time software. The focus of the DSP-ACTION project was to support reuse of not only source code but also on every level of design and project level documentation. GQM played a key role in the project and measurements indicated significant improvements. Consequently, a sound basis for continuous improvement and measurement has been formed.

In another type of telecommunication domain Nokia Telecommunications and VTT Electronics have been applying GQM in improving the software processes of mobile base stations which are large software systems involving large development projects [17]. Large telecommunication applications place great demands on the quality and timely development of software. By measuring the software development process, quantitative information is gained for software project control and process improvement. Nokia Telecommunication experiences are based on measurement data collection for a period of two years in industrial pilot projects. The use of GQM has been successful. The defined measurement programme is expandable and reusable to future projects.

Successful applications of GQM in the banking and insurance industries have also been reported. In co-operation with Fraunhofer IESE, at Societa Interbancaria per l'Automazione (SIA) GQM was successfully applied to monitor the improvement of the configuration management (CM) process, and to gain experience for future redeployment of the CM process into other projects and departments of SIA [9]. At Allianz Life, market leader of life insurers in Germany, GQM was applied to help establish the infrastructure for a learning organisation. For this purpose, goal-oriented measurement was fruitfully combined with data mining [12].

The above mentioned projects are just a few examples of applying GQM in industrial projects. Companies like Ericsson, Daimler Chrysler, Motorola, ABB, Bosch and Siemens have also reported experiences from using GQM in software process improvement (SPI). However, despite the positive experiences GQM is yet to be integrated with other SPI methods. The harsh reality is that companies are often using different SPI approaches in isolation and sometimes even in a way where the methods compete with each other. The PROFES[1] project, a European SPI research project, tries to overcome these problems by integrating GQM in a practical product focused process improvement methodology.

The PROFES project consortium consists of methodology providers and practitioners with comprehensive expertise in process improvement: Dräger Medical Technology (The Netherlands), Ericsson (Finland), Etnoteam S.P.A. (Italy), Fraunhofer IESE (Germany), Tokheim (The Netherlands), University of Oulu (Finland), and VTT Electronics (Finland).

Based on the fundamental concepts of the Quality Improvement Paradigm (QIP) [2] the PROFES improvement methodology offers means to implement a systematic and cost-effective approach to product quality focused process improvement. PROFES integrates existing methods such as software process assessment, product and process modelling, goal-oriented measurement and the experience factory concept, supported with operational guidelines and tools. An essential element of the PROFES improvement approach is the explicit modelling of relationships and dependencies between process and product quality. Product-process dependency (PPD) models enable a software organisation to focus improvement actions precisely to those parts and characteristics of the development processes that are critical to achieve the planned product quality.

In the scope of the PROFES improvement methodology, goal-oriented measurement according to the GQM approach is used for several purposes: (1) characterisation and evaluation of product quality, (2) characterisation and evaluation of process performance, (3) modelling and evaluation of product-process dependencies, and (4) facilitation of continuous assessment. Furthermore, the PROFES project also used GQM to evaluate the PROFES improvement methodology itself in all three industrial software development organisations. This included the development of detailed cost models.

The structure of this paper is as follows. The next chapter briefly outlines the PROFES improvement methodology, characterises the application of the PROFES improvement methodology in three different embedded software development organisations, and summarises achieved benefits. Chapter 3 gives a short introduction into the principles of goal-oriented measurement and the GQM method. Chapter 4 provides descriptions of the different purposes for which GQM is used in the PROFES improvement methodology. Chapter 5 summarises how GQM was used during the PROFES project to validate the PROFES improvement methodology. The paper concludes with a brief discussion of the different roles of GQM in PROFES and an outlook to potential future work after finalisation of the PROFES project.

[1] PROduct Focused improvement of Embedded Software processes. PROFES is an applied research and technology transfer project supported by European Commission under ESPRIT grant no. 23239.

2 The PROFES Improvement Methodology and its Application

Most improvement methodologies in software engineering focus on improvement of the software *process*. Their underlying assumption is that improved software engineering processes result in better product quality. However, none of these improvement methodologies address this link between process and product quality explicitly and none of them deploys such links to steer improvement actions.

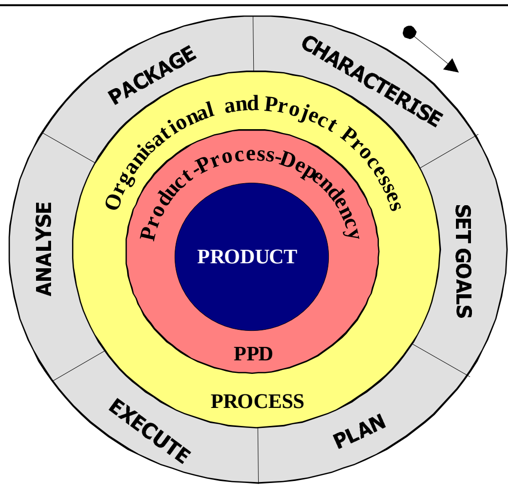
The PROFES improvement methodology [5] guides improvement efforts based on customer-driven product quality goals. It builds on state-of-the-art knowledge about the product quality impact of critical software development processes and integrates the strengths of widely applied improvement techniques.

The main building blocks of the PROFES improvement methodology are:

- Systematic improvement planning starting with the identification of an organisation's product quality goals and proceeding with the determination of appropriate process improvement actions according to their expected impact on product quality.
- Product-process dependency models (PPD models) that describe the impact of software processes and development practices on software quality [11]. PROFES offers a PPD repository that can be customised and integrated to suit the needs of individual software organisations or projects.
- The integration of well-established improvement techniques such as process assessment (e.g., ISO 15504 / SPICE [13] and BOOTSTRAP [14]), goal-oriented measurement following the Goal/Question/Metric (GQM) approach [1] [3] [19], process modelling, and product assessment. The PROFES improvement methodology is modular and supports the integration of improvement techniques in a way that is most beneficial for each individual software organisation.
- A continuous improvement cycle according to the Quality Improvement Paradigm (QIP) / Experience Factory (EF) [2] approach that allows for well-customised, gradual improvement steps and fosters the accumulation and deployment of relevant software development expertise throughout a software organisation.

2.1 Outline of PROFES improvement methodology

The PROFES improvement methodology uses a modified version of the Quality Improvement Paradigm. To illustrate and emphasise the importance of the product as a driver for process improvement, it is placed in the heart of the PROFES improvement circle (see Figure 1). The product is the starting point for any improvement activity, starting with the identification of the product quality needs and the determination of the preliminary product quality goals. Product-Process Dependencies (PPD) form the linking element between the product and the product development processes. PPD models are used to find and determine the required process changes such that stated product quality improvement goals are achieved. The PROFES improvement methodology consists of six phases. These phases, which are further refined into 12 steps, are described briefly in the following.



PROFES Phases	PROFES Activities
Characterise	1. Gain commitment
	2. Identify product quality needs
	3. Identify current product quality
	4. Identify current process status
Set Goals	5. Set product improvement goals
	6. Select appropriate process changes
Plan	7. Describe process changes
	8. Set metrics for process and product
	9. Prepare improvement implementation
Execute	10. Implement and monitor improvements
Analyse	11. Evaluate results
Package	12. Update experience base

Figure 1: PROFES improvement cycle with phases and activities

Phase 1: Characterise

A general requirement for any improvement program is to gain the commitment of all involved management levels and development team members. Therefore, “gaining commitment” is the first step in the PROFES improvement methodology (step 1). On a more technical level, product-driven process improvement starts with an identification of current or future product quality needs (step 2). This information can be derived from customer surveys, market research or from other sources. To be able to define product quality improvement goals in a later phase, it is necessary to identify the current status of product quality (step 3). This can be done, for example, based on measurement data from past projects or carrying out an independent third party product assessment. In addition to the product quality status, it is necessary to identify the current process status (step 4). Otherwise, it is hard to identify process changes that are suited to achieve the product improvement goals in a later phase. The current process status can be identified based on measurement data or on process assessments.

Phase 2: Set goals

Based on product quality needs and current status of product quality, the product quality improvement goals are defined (step 5). Of course, the product quality improvement goals have to be aligned with the overall business goals of the organisation. Based on the product quality improvement goals and using implicit (expert knowledge) or explicit (PPD models) suggestions for appropriate process changes are made (step 6). One way in which such PPD models could have been constructed is by applying goal-oriented measurement.

Phase 3: Plan

The suggested process changes are described in detail using a process modelling notation (step 7). This is necessary to define adequate process metrics for the purpose of monitoring and control during execution of the development project. In addition to the process metrics, product quality metrics are defined. Again, these metrics have the purpose to monitor and control the achievement of the product quality improvement goals (step 8). The planning concludes with preparing the implementation of the suggested process changes, and with implementing the measurement infrastructure (step 9).

Phase 4: Execute

The suggested process changes are implemented in the development process and followed during the development project. Process and product quality measures are collected and used for monitoring and control (step 10).

Phase 5: Analyse

The purpose of the analysis phase is to analyse if product quality has improved as assumed with the changes made to the process. Based on measurements the process conformance and the achievement of the product quality goals are analysed and root causes of deviations are identified. The analysis includes an evaluation of the PPD models used step 6, and the collection of lessons learnt with improvement actions during the development project (step 11).

Phase 6: Package

All results of the improvement cycle are stored in an experience base, and all new or re-used models are consolidated (step 12).

2.2 Characterisation of PROFES applications and summary of achieved benefits

The following sections briefly introduce three industrial software organisations that used the PROFES improvement methodology in the PROFES project. In each case, results and experiences from the application of the PROFES improvement methodology are summarised.

2.2.1 Dräger Medical Technology

Dräger Medical Technology (MT) develops patient monitors and other medical devices for application in anaesthesia, intensive care, neonatal care, and emergency care. Development projects typically deal with software as well as with hardware and mechanics. A strong trend is toward connection of devices through networks and their integration into health care information systems.

Important product quality attributes for Dräger MT are reliability, fitness for use, and predictability of quality, time, and cost. Based on experience with current product prototypes in field test, several important product quality goals could be achieved by using PROFES. Examples are on-schedule delivery, functionality very well in accordance with user needs, and very low number of defects in field tests. In addition, a wide spectrum of process-related improvements were accomplished, such as fast process capability increase to level 3 on the BOOTSTRAP scale and meeting the ISO 9001 certification criteria.

2.2.2 Ericsson Telecom R&D

Ericsson Telecom R&D in Finland applied PROFES in the context of software development for the AXE telecommunication exchange, which is Ericsson's core product. This project involved multiple, globally distributed development sites. Another project has developed charging functionality for operators to handle interoperator tariff account settlement.

Ericsson Finland focused mainly on two product quality goals within the PROFES improvement program: reliability and maintainability. Important quality improvements were achieved with regard to design quality in terms of fault density. This is measured by defect data from the first six months after delivery of the product. The improvements were attributed to significantly more careful preparation for software inspections and more intense desk-checking. In addition, a web-based inspection tool (webIR) was applied. Two BOOTSTRAP assessments have indicated capability level improvements from below level 2 to nearly level 3.

2.2.3 Tokheim Dispenser Electronics

Tokheim is world-wide market leader in manufacturing equipment and providing services for self-service petrol stations. Products are fuel dispensers, point of sales systems, electronic funds transfer equipment and others. All products contain both hardware and software, where the importance of software is increasing rapidly. In some cases, even 80 per cent of the product development effort is spent on software. The Tokheim site that applied PROFES is certified according to TickIT and ISO 9002 and has been assessed at level 2 of the SEI's Capability Maturity Model.

The product quality goals for Tokheim within PROFES were focusing on reliability with additional strict cost and time targets. Related achievements were well-structured product architecture, better traceability and analysability of the product, as well as very low number of defects during the first field-tests with the product. At the same time, the targeted cost reductions were achieved and product delivery was within the planning limits. In addition, several process improvements were achieved, for instance better design and testing practices and better integration of quality assurance with the project team.

3 Goal-oriented Measurement (GQM)

Measurement is a technique that supports in understanding, controlling, predicting, and improving the software development processes and products. Goal-oriented measurement according to the Goal/Question/ Metric (GQM) paradigm represents a systematic approach for tailoring and integrating the objectives of an organisation into measurement goals and their stepwise refinement into measurable values (metrics). The GQM method was chosen as an element of the PROFES improvement methodology as it is the most mature and widely used measurement approach available today.

3.1 GQM Principles

GQM represents a systematic approach to tailoring and integrating goals with: models of the software processes, software products, and with particular quality perspectives of interest. GQM focuses on the specific needs of the software project and of the development organisation. Measurement goals are defined on the basis of high-level corporate goals, and refined into metrics. In other words, GQM defines a certain goal, refines this goal into questions, and defines metrics that must provide the information to answer these questions. The GQM paradigm provides a method for top-down metric *definition* and bottom-up data *interpretation*.

The principles of GQM measurement are:

- A measurement programme must reflect interests of data providers and must be based on the knowledge of the people who are the real experts on the measurement goals. In this paper, these are members of the software *project team*.

- Since the design of the measurement programme is based on the knowledge of the project team, only they can give valid interpretations of the collected data. Therefore, they are the only ones who are allowed to interpret measurement data.
- Due to the limited amount of time of project members, and their commitments to project planning, conflicts of interest may occur when all improvement efforts are also assigned to the project team. Therefore a separate team, a *GQM team,* should be created that facilitates the collection and analysis of measurement data by performing all operational activities not necessarily to be executed by the project team.

These principles imply that the members of the GQM team offer a service to the software project team by doing most of the technical work, related to setting up and performing the measurement programme. Essentially, during execution of the measurement programme, the GQM team provides a data validation and analysis service, by organising 'feedback sessions' in which graphical measurement data is presented to the project teams.

3.2 GQM Process

The GQM process is divided into several stages. After the pre-study, the next stage is to identify a set of measurable quality goals. After the goals have been set, questions that define the goals are derived as completely as possible. The next step consists of specifying the metrics that need to be collected in order to answer the questions defined, and to track the conformance of products and processes to the defined measurable quality goals. Defined goals, questions and metrics are described in the GQM plan. The three layers (goals, questions, and metrics) of the GQM plan correspond to the following three levels:

- *Conceptual level (Goal)*: The definition of the measurement goal specifies the object of measurement, the purpose of measurement, the quality model of interest, the role for whom the measurement results are of interest (viewpoint), and the environment in which the measurement programme takes place.
- *Operational level (Question)*: A set of questions is used to define in a quantitative way the goal and to characterise the way the data will be interpreted. Questions try to characterise the object of measurement with respect to a selected quality issue and to describe either this quality issue from the selected point of view or the factors that may affect the quality issues.
- *Quantitative level (Metric)*: A set of metrics - combined into a model - is associated with every question in order to answer the question in a quantitative way.

The definition of the questions and metrics contained in a GQM plan is usually done with the help of so-called abstraction sheets. Basically, an abstraction sheet is a means for acquiring, structuring, and documenting all the relevant information provided by participants in the measurement programme. An abstraction sheet contains information about the measurement object and its associated attributes representing the quality focus (as specified by the measurement goal), and information about factors that have an impact on the quality focus (so-called variation factors). In addition, hypotheses about the performance of the quality focus attributes and the way in which the variation factors influence the performance of the quality focus attributes are documented. Based on this information, for each measurement goal, a set of questions, metrics, and models can be defined (for details see [3] and [8]).

After the measurements have been specified, a mechanism for collecting measurement data is developed. This is described in the measurement plan and in the associated data collection forms. The data is then collected and validated during the software development project according to the measurement plan.

The collected data is analysed and discussed in feedback sessions. Feedback sessions are organised meetings involving members of the project team and the measurement team. It is an essential mechanism supporting analysis and interpretation of the measurement results. The main objective of feedback sessions is to discuss the preliminary findings and results of the measurement programme and derive interpretations by the project team from the data collected so far with the GQM experts.

After the end of the software development project all relevant information gathered during the project has to be packaged and stored for later retrieval and reuse. This is especially important for continuous learning and improvement.

Practical guidelines, examples and procedures for the GQM process in practice can be found in [19].

4 The Roles of GQM in the PROFES Improvement Methodology

In the scope of the PROFES improvement methodology, goal-oriented measurement according to GQM is used for the following purposes:

- characterisation and evaluation of product quality,
- characterisation and evaluation of process performance,
- modelling and evaluation of product-process dependencies, and
- facilitation of continuous assessment.

In addition, during the PROFES project, GQM was used to validate the PROFES improvement methodology. The following five subsections describe the different roles that GQM plays in the scope of the PROFES improvement methodology. In the last subsection, the results of the PROFES validation are briefly presented.

4.1 GQM to characterise and evaluate product quality and process performance

For the PROFES application partners, three product quality characteristics were in the focus of interest: reliability (all partners), maintainability (Ericsson Finland only), and fitness for use (Dräger Medical Technology only). GQM was used to measure these product quality characteristics for the purpose of a) baselining, and b) validation of product quality improvement (cf. Table 1).

	Dräger Medical Technology	**Ericsson Finland**	**Tokheim RPS**
Product qualities of interest for improvement	Reliability Fitness for use	Reliability Maintainability	Reliability
Processes of interest for improvement	Inspections (wrt. Reliability) Testing (wrt. Reliability) Config. Mgmt. (wrt. Reliability) Customer Needs Management (wrt. Fitness for use)	Architectural and functional design Testing Project management (and others; each of these is investigated wrt. Reliability and Maintainability)	Testing Configuration Management Quality Assurance
Measured Product and Process Attributes	Reliability of system product Effectiveness and efficiency of inspections	Reliability of software product and process impact on reliability Maintainability of software product and process impact on maintainability	Reliability of system product Performance of the testing process Acceptance and stop criteria for testing

Table 1: The main improvement and measurement concerns of the PROFES applications.

In all applications reliability was expressed in terms of field defect density. Field defect density was anticipated based on defects detected in the product before delivery. For instance, Figure 2 presents the design quality improvements that were achieved in two subsequent projects at one of the PROFES applications, based on the defect density at the end of function test. The diagram compares the baseline data with the quality goals (defined for function test) and actually measured defect density at the end of function test. It can be seen that the defect density (measured in kilo non-commented source statements) in project B is much lower than in the preceding project A. Although both projects met their quality goals, it was surprising that outcome of project B was much better than the outcome of project A, if compared to the baseline and goals set. The analysis to explain this discrepancy was done based on measurements of process performance (cf. sub-section below).

In the meanwhile, project A has passed more than 6 months at the customer. Only 3 major faults influencing fatally on operational performance were detected. Fault analysis has also shown that only one fault detected by the customer was received.

Changes in product quality can only be achieved by changing the development processes. To be able, to judge if an intended process change has really been implemented, a process performance baseline has to be established (characterization) and the current performance has to be observed on a regular base (monitoring). This can be done based on measurement. As more knowledge is acquired by analyzing the relationships between process performance variations (due to process changes) and changes in product quality, models can be built that directly relate product quality and process performance. These models help control ongoing and guide future improvement activities.

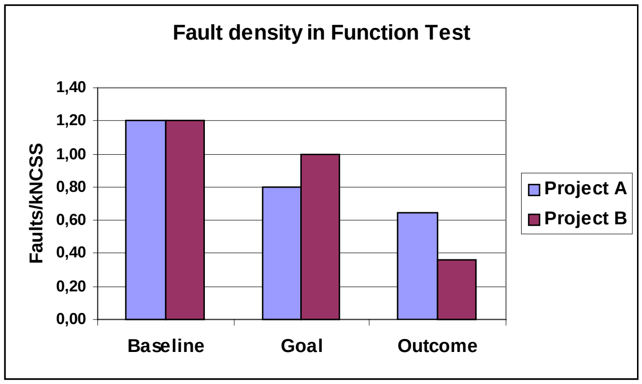


Figure 2: Measurements of design quality characteristics

Typical process metrics were effort and time used for development, verification (inspections) and validation (test). Based on these metrics, and in combination with product size and defect data, the effectiveness and efficiency of verification and validation activities could be monitored over time. Using these results, the effects of process changes on product quality could be investigated. A major project change that occurred in one of the projects at Dräger Medical Technology was the introduction of incremental development. The new organisation of the development process shortened cycle-time considerably, and thus made it possible to achieve a shortened time-to-market for the new product with compromising quality. Similar measurement-based analyses were conducted in the projects of all three PROFES application providers. Moreover, in the Dräger case, a considerable increase in process capability (from 1 to 3 on the SPICE scale) could be achieved.

4.2 GQM to model and evaluate product-process dependencies

The PROFES project investigated the impact of software development processes on product quality. Identified product/process dependencies (PPDs) are modelled and packaged for reuse (PPD models). The PROFES improvement methodology uses PPD models to identify candidate process changes that are appropriate to yield a required product quality.

A PPD model is shown in Figure 3. It states that a certain software engineering technology (e.g., software inspections) has a significant impact on achieving a high level of a certain product quality (e.g., reliability), when applied in a certain software engineering process (e.g., software requirements analysis) and context (e.g., under low or average time pressure).

PPD Model 1.3.1		
	Technology Application Goal	
Technology	**Software Inspections**	
Product Quality	**Reliability**	
Process	**ENG.3 Software Requirements Analysis**	
	Technology Application Context	
CF.1	Experience of inspection team	low average ***high***
CF.2	Management commitment	low ***high***
CF.3	Overall time pressure	***low average*** high
CF.4	Module affected by new hardware	old_hw ***new_hw***
CF.5	Module developed externally	internally ***externally***

Figure 3: Example PPD model

GQM can be used to identify and validate PPD models. Information sources for defining PPD models are experiential knowledge of software professionals and empirical investigations of software projects. The structure of a GQM goal is suitable to express PPD-related information needs. An example of such a GQM goal can be defined as follows. It is associated with the example PPD model of Figure 3.

Goal: Analyse the *sensor software* (final software product) for the purpose of *PPD evaluation* with respect to *reliability* from the viewpoint of the *software engineers* in the context of *PROFES-A*.

In the GQM goal, the product quality slot of the PPD model is mapped onto the so-called *quality focus* (i.e.,. *reliability* in the example). It is the dependent variable of the product/process dependency. A coherent collection of questions and metrics in the GQM plan, which is usually referred to as *quality definition*, should be used to define the product quality in terms of appropriate measurable indicators.

The independent variables contained in a PPD model are mapped onto a collection of GQM questions and metrics called *variation factors*. These independent variables are technology, process, and the context factors. Each of them must be defined in terms of questions and metrics. For instance, a technology like software inspections should be defined in terms of process conformance metrics (e.g., inspection effort or percentage of inspected documents). In addition, the expected impact of the independent variables on the dependent variable (i.e., the product quality) must be defined. In a GQM plan, this information is associated with the questions of the respective variation factors.

To derive a PPD model using GQM, the appropriate GQM goal is defined (see above) and GQM interviews are conducted. Their purpose is to acquire the needed information from the personnel of the project in which the PPD is to be investigated. A GQM plan documents the results. It can be translated into an initial and hypothetical PPD model. GQM feedback sessions validate the hypothesised PPD using the measurement data. If needed, they suggest refinements and corrections of the PPD model.

During a PPD-related feedback session the following issues must be addressed: (1) Has the desired product quality been achieved (e.g., a certain level of product reliability)? (2) Has the observed practice had impact on the product quality (e.g., did requirements inspections improve reliability?) (3) Are there other impacting factors that have not been measured but that have had impact on the achieved product quality (e.g., code inspections or module testing instead of requirements inspections)?

To validate an already existing PPD model, it must be translated into a GQM plan. Interviews are only needed to validate the translation and to acquire possibly missing information. The actual validation is performed by the aid of GQM feedback sessions.

In some cases measurement programmes can not provide all data needed for deriving or validating a PPD model. Then the collected data must be complemented with past measurement data, qualitative information, and experiential knowledge of project personnel. GQM feedback sessions provide an appropriate framework to access and integrate additional information in a systematic and well-organised manner.

4.3 Integration of GQM with Assessment

GQM is a very flexible tool for detailed in-depth quantitative and qualitative analyses. Common tools for more general in-breadth process analyses are process assessment methods. To combine both types of process analyses, possibilities for integrating GQM with SPICE conformant assessment methods was investigated.

The integration of assessment and measurement is focusing on two major aspects (cf. Figure 5):

1. Integration of GQM planning with process assessment planning and execution.
2. Use of empirical data from GQM programmes for capability measurement during process assessments.

During GQM planning, three process steps have to be passed: a) identification of business goals and characterisation of the environment in which measurement takes place, b) definition of measurement goals and set up of GQM plan (design of measurement programme), and c) detailed definition of measurement execution, i.e. measurement plan, including set up of the measurement infrastructure (data collection forms and tool support). Through reuse of information gathered during assessment planning (e.g., document review) and execution (assessment interviews), valuable information for GQM planning can be gained. In particular, information about the environment (characterisation) and business goals can be easily reused for GQM planning and thus reduce effort for GQM interviews. The most effective way of integrating GQM planning with assessment planning and execution is for the GQM team to adopt the assessment schedule and participate in assessment interviews and feedback session. During the PROFES project, the perceived benefits from the integration of BOOTSTRAP interviews with GQM planning were that the responsible for the measurement programme:

1. gained deeper understanding and familiarity with relevant software development and improvement issues,

2. received a broader spectrum of information than they would have gained from GQM interviews alone,
3. limited the effort overhead for the project team, and
4. shortened the time needed for conducting process assessment and GQM measurement planning together.

The integration of GQM execution with process assessments is mainly based on the use of measurement data during assessments. If validated data and its interpretation (provided during GQM feedback sessions on a regular base) can directly be used for assessing processes, then process assessments can be accelerated and conducted more often, hence gradually converging to the idea of "continuous assessment". It is, however, not a trivial research issue to investigate under which circumstances a particular measure is suited to help automate the assessment of a specific process. Although first experience has been gained during the PROFES project, this is still an ongoing work. More details on the topic are presented in the next chapter.

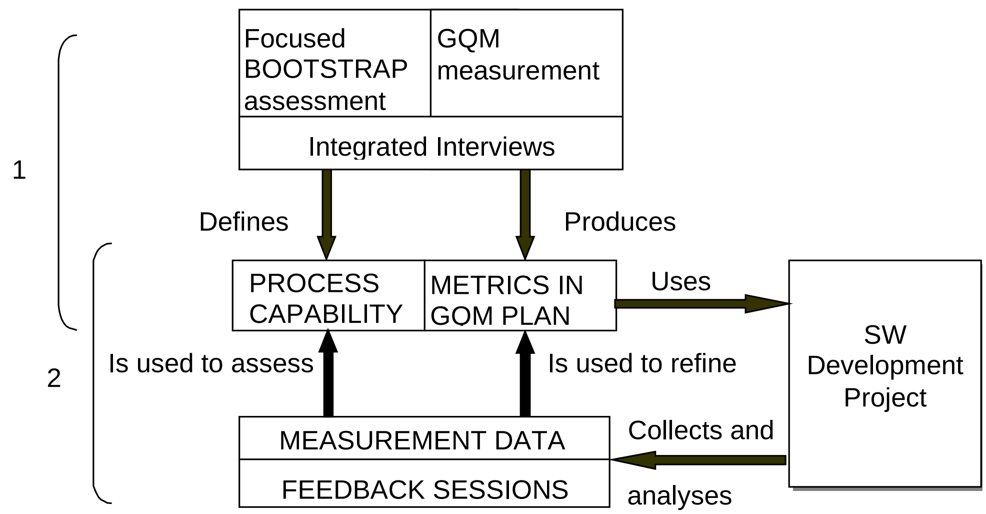


Figure 4: Aspects of integrating GQM with software process assessment

4.4 GQM to facilitate continuous assessment

Typically, an assessment is an annual or biannual snapshot of the software development activities, and is conducted as a self-assessment or using an external assessment team. Information gathering is done manually through document reviews and interviews. Use of supporting tools is minimal.

The basic idea of continuous software process assessment is to collect relevant information from the software process as it becomes available. This information can then be consolidated and used to help an assessor to make judgement of the process status.

There is a paradigm shift with continuous assessment. Information is continuously gathered using existing data from the development process where possible. While the act of assessment is done in a traditional sense by a competent assessor or team using available information, the continual manner how the assessment is done changes the role of assessment within process improvement.

The degree of continuity and automation determines how embedded the assessment is in the software development process. If majority of assessment information is gathered (automatically) via a measurement programme, the notion of Measurement bAsed Assessment (MAA) clarifies this special instance of continuous assessment.

In this section, we are interested in the MAA approach for continuous assessment. Analysing detailed process information against a common reference framework can result in better understanding of the influence of process capability on the object of study in a GQM measurement programme.

4.4.1 Background for continuous assessment

For continuous assessment purposes, the ISO 15504 is used as a reference framework for the software process capability as it is possible to find links between measurable objects and the ISO

15504 framework (cf. Figure 5). Specifically, the *assessment indicators* provide the adequate detail for connecting actual process information.

The indicators of *process performance* are used to determine whether a process exists in practice. For example, the software design process (cf. ENG.1.3 in ISO 15504 reference model) is considered as existing if it can be determined that there exist documents that specify

- an architectural design that describes the major software components that will implement the software requirements;
- internal and external interfaces of each software component;
- a detailed design that describes software units that can be built and tested;
- consistency between software requirements and software designs.

If a software design process is functioning in an organisation it should be straightforward to determine the existence of the documents that satisfy the goals listed above. This information could be contained, e.g. in a document management system that keeps track of the documents produced against a specified process. A report from this system would then help the assessor in determining whether the software design process is performed in the organisation.

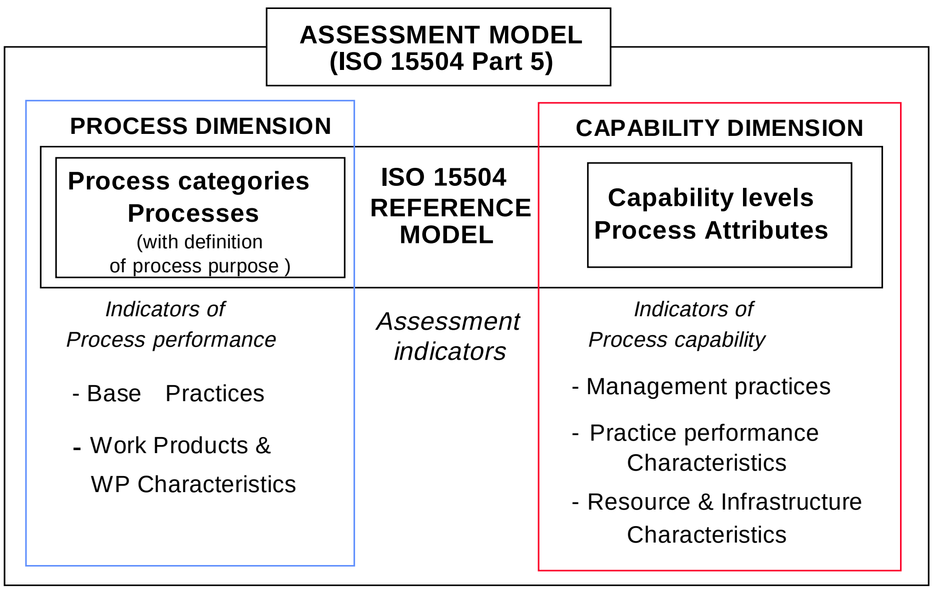


Figure 5: The ISO 15504 framework

Further, the ISO 15504 indicators of process capability are used to determine how capable an existing process is. Linking information from the measurement system to the management practices, characteristics of practice performance, resource and infrastructure can assist an assessor in determining how well the process is performed as intended by the definition of ISO 15504. For example, the performance management attribute 2.1 of SPICE level 2 can be considered as fulfilled if

- objectives for the performance of the process will be identified (for example, time-scale, cycle time and resource usage);
- the responsibility and authority for developing the work products of the process will be assigned;
- the performance of the process will be managed to produce work products that meet the defined objectives.

Using the ISO15504 indicators, a special branch of a GQM plan is created where the goal is usually to understand, monitor or improve process capability. This provides additional viewpoint to a measurement programme yielding supplementary information to better understand the object of the study. A special tool support to link a GQM plan with ISO15504 indicators is being created within the PROFES project.

4.4.2 Expected benefits and limitations

There are two main areas where continuous assessment is expected to bring benefits over the traditional approaches:

- Process visibility

- Assessment cost

With continuous assessment, the process implementation becomes more visible. It is possible to see in detail what is done in the software process against a solid process framework. For example, this enables close observation of improvement activities so it is more apparent whether new practices are adopted and successful long before the usual re-assessment. Continuous assessment also provides the means to detect process deviations earlier thus helping to manage process implementation in two ways: Firstly, giving early signals on practices that are not being adopted, indicating that people should be supported with the process adaptation. Secondly, suggesting potentials for process change.

The assessment costs are expected to be reduced with continuous assessment. The working hypothesis is that collecting information from the software process as it becomes available reduces the time needed for the interviews and document analysis during an assessment. Appropriate tooling (e.g. MetriFlame [16]) can greatly support this data collection. The key is to integrate the data collection into the work processes in such a way that it is a natural part of the work. This can be achieved in two ways: Either the data collection is essential for the work to be performed (e.g. writing an inspection report) or that the work automatically leaves marks in the tools and databases of the company.

5 The Role of GQM in validating the PROFES improvement methodology

Validation of the PROFES improvement methodology started at the beginning of the PROFES project in early 1997. Based on the first blueprint of the methodology, the validation study was planned with early involvement of the methodology users. The investigation is separated into two 15 months periods, during which the PROFES improvement methodology has been applied yet in multiple projects at the three industrial application projects of PROFES at Dräger MT-M, Ericsson Finland, and Tokheim. The projects were subject to detailed observation by the researchers who are responsible for the validation work. Hence, the basic design of the empirical work in PROFES is a twice repeated, three times replicated case study. Figure 6 depicts the overall structure and the main phases of the PROFES methodology validation study.

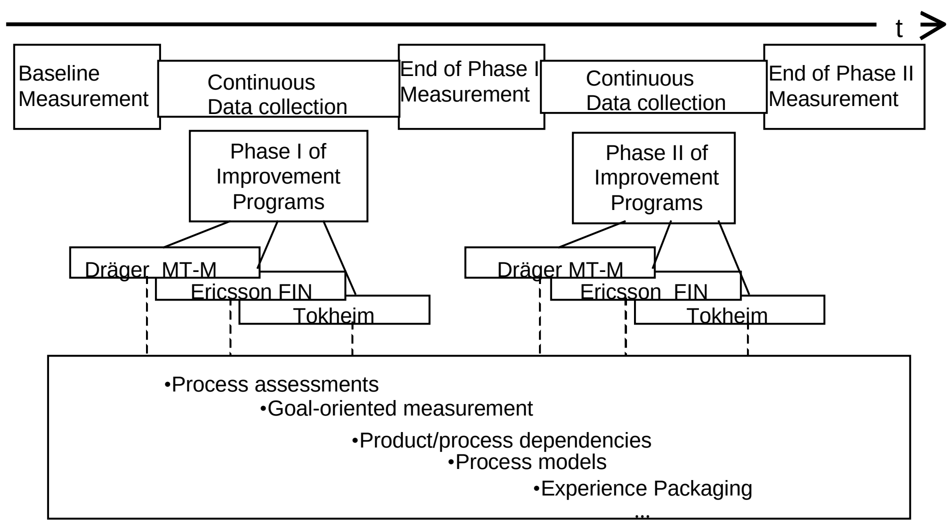


Figure 6: Design of the PROFES methodology validation study and its main phases

The PROFES methodology validation involves three basic types of validation criteria (i.e., multi-facetted validation):

- Achievement of product quality improvements through application of the PROFES improvement methodology (to be demonstrated by identifying causal links between methodology and product quality)
- Other aspects of benefit from applying the PROFES improvement methodology.
- Cost-effectiveness of the PROFES improvement methodology.

GQM has been used to identify and define validation criteria. Two overall GQM goals were defined that differ in their viewpoints:

Goal 1: Analyse the PROFES improvement methodology with respect to cost/benefit for the purpose of characterisation from the viewpoint of the *methodology user* in the context of PROFES.

Goal 2: Analyse the PROFES improvement methodology with respect to cost/benefit for the purpose of characterisation from the viewpoint of the *methodology provider* in the context of PROFES.

For each goal, questions and metrics have been gained by interviewing representatives of the PROFES application projects or methodology developers, respectively. The results are defined in the form of two GQM plans, which have been used to plan data collection and analysis. Figure 7 outlines their structure. It lists validation criteria and assumed impacting factors of the PROFES methodology validation.

Methodology User Viewpoint	***Methodology Provider Viewpoint***
Product Improvements Achievement of product quality goals **Process Improvements** Standardisation of work practices Focusing of process definition Improvement of work practices Improvement of efficiency of work practices Reduced risk of failure **Systematic Improvement** Reduced risk of failure Focused improvement actions Integrated business, product, and process issues Tailorability Efficient management involvement Compatibility with quality awards **Findings, Awareness, Understanding** Knowledge about software and system Awareness of software development capabilities Awareness of crucial software development issues Awareness of necessity of improvement New findings **Team Building & Organisational Culture** Contribution to group synergy Awareness of necessity of improvement **<u>Possible Impacting Factors</u>** Maturity of the software organisation Infrastructure of the software organisation Other ongoing improvement initiatives Project management's awareness of the improvement methodology Higher management's expectations on the improvement program	**Product Improvements** Product quality improvements **Process Improvements** Process definition Process consistence Process stability **Methodology characteristics** Domain-specific for embedded systems development Customer viewpoint Quality and improvement awareness **Methodology definition and support** Coverage of methodology (roles, phases, activities) Guidance of methodology (processes, guidelines) Documentation of methodology Tool support of methodology **<u>Possible Impacting Factors</u>** Size of measurement program Maturity of the software organisation Infrastructure of the software organisation Other ongoing improvement initiatives Organisational culture: Management commitment for the improvement program Organisational culture: Improvement attitude within the software project Degree at which quality improvement is integrated with regular software development activities

Figure 7: PROFES validation criteria and expected impacting factors

Non-product-related improvements at the PROFES applications involve for instance the fast process capability increase to level 3 on the process maturity scale, meeting the ISO 9001 certification criteria, better design and testing practices, as well as better integration of quality assurance with the project team. Example benefits that are specifically due to GQM measurement are: Enhanced definitions of software development processes, increased knowledge about software and system, and the effective fine-tuning of improvement actions using GQM measurement data. Similar kinds of benefits have been identified concerning ISO 15504 process assessments, process modelling, software engineering experience management, and other parts of the PROFES improvement methodology.

The third type of methodology validation criteria in PROFES is cost-effectiveness. The GQM interviews for planning the evaluation work have resulted in the following facets of cost-effectiveness: Overall effort for the improvement program, effort for the improvement program by key personnel (i.e., managers, software engineers, improvement team, and external consultants), and tailoring effort for the improvement methodology when setting up the improvement program. The related measurements have provided detailed effort data about the execution of BOOTSTRAP process assessments and GQM measurement programmes. It involves the number of hours spent by each participant of the improvement program for each activity of the respective method. Table 2 shows an example effort model for one variant of BOOTSTRAP assessments.

Activity	Role					Total
	Lead Assessor	Assessor	Manager	Engineer	Facilitator	
Preparation	18	20			2	40
Opening Briefing	0.5	0.5	2	1	0.5	4.5
Assessment SPU	7.5	7.5	2.5		1	18.5
Assessment Project	27	26	4.5	4	3	64.5
Evaluation	32	16				48
Review	10	10				20
Final Meeting	7	7	4	4	6	28
Report Preparation	44	4				48
Report Review	2	8				10
Total	148	99	13	9	12.5	281.5

(Effort in person hours)

Table 2: Example effort model of BOOTSTRAP process assessments

6 Conclusions and Future Work

The PROFES improvement methodology, which has been developed in a European research project, is novel in multiple ways: (1) It focuses on process improvement that is driven by explicit product quality goals. Therefore, explicit links are forged between aspects of the software process and their impact on the resulting software product quality. (2) The PROFES improvement approach integrates multiple improvement techniques that have in the past been applied in isolation; in particular these techniques are process assessments, goal-oriented measurement, process modelling, product assessment, and systematic experience reuse. (3) PROFES promotes a systematic and iterative approach of continuous improvement that is a-priori independent of any specific improvement technique.

GQM substantially contributes to realising these characteristics of the PROFES improvement methodology. GQM supports the characterisation and monitoring of product quality characteristics and process performance, and it helps to identify and evaluate of product-process dependencies. In addition, it can easily be integrated with other assessment, modelling, or improvement techniques, and it incorporates the principles of continuous improvement.

The contributions of GQM are mainly due to three facts: (1) GQM is a valid and effective measurement approach, i.e., it helps to assure that the right information is measured in the right way. (2) GQM is generic for a wide range of applications. (3) GQM can be customised to suit any specific application scenario (e.g., project monitoring, continuous process assessment, or evaluation of improvement program success).

GQM has been used to effectively link assessment and measurement. Continuous assessment provides a new perspective – process capability – into a measurement programme. The continuous assessment work within PROFES will be continued in the application projects to gain more experience on using the approach, e.g. to find out the most significant cost factors. Also, for a limited set of processes an initial set of indicators suitable for continuous assessment are gathered. These can be used as a starting point for planning continuous assessment in other companies. Finally, the tool support for automated data collection will be extended to provide more support for continuous assessment by mapping measurement data and definitions to BOOTSTRAP processes.

7 Acknowledgements

The authors would like to thank Andreas Birk, Markku Oivo, and Matias Vierimaa for sharing much of the work reported in this paper, and for providing substantial input to sections 4.1 and 5, section 1, and section 4.3, respectively.

Furthermore, we would like to thank all members of the PROFES consortium for many fruitful discussions and Seija Komi-Sirviö for reviewing an earlier version of this paper. Special thanks go to the application projects at Dräger MT-M, Ericsson Finland, and Tokheim.

The Commission of the European Community has supported this work through ESPRIT project no. 23239 PROFES (PROduct Focused improvement of Embedded Software processes).

8 References

[1] Basili, V.R., Caldiera G., Rombach, H.D., Goal Question Metric Paradigm, in John J. Marciniak, editor, *Encyclopaedia of Software Engineering*, Vol. 1, pp. 528–532, John Wiley & Sons, 1994.

[2] Basili, V.R., Caldiera G., Rombach, H.D., Experience Factory, in John J. Marciniak, editor, *Encyclopedia of Software Engineering*, Vol. 1, pp. 469–476, John Wiley & Sons, 1994.

[3] Basili, V.R., Weiss, D.M., A Methodology for Collecting Valid Software Engineering Data, *IEEE Transactions on Software Engineering*, Vol. SE-10, No. 6, pp. 728 – 738, 1984.

[4] Birk, A., Derks, P., Hamann, D., Hirvensalo, J., Oivo, M., Rodenbach, E., van Solingen, R., Taramaa, J., Applications of measurement in product-focused process improvement: A comparative industrial case study, *Proceedings of the 5th International Symposium on Software Metrics* (Metrics '98), Bethesda, MD, 20 - 21 Nov. 1998, IEEE Computer Society, 1998.

[5] Birk, A., Järvinen, J., Komi-Sirviö, S., Kuvaja, P., Oivo, M., Pfahl, D., PROFES - A product driven process improvement methodology, *Proceedings of the European Conference on Software Process Improvement* (SPI '98), 1 - 4 Dec. 1998, Monaco: John Herriot, 1998.

[6] Birk, A., Kempkens, R., Rombach, H.D., Ruhe, G., Systematic Improvement of Software Engineering Processes, *Proceedings of Fruehjahrstagung Wirtschaftsinformatik '98*, Braunschweig: Vieweg, pp. 265-280, 1998.

[7] Birk, A., van Solingen, R., Järvinen, J., Business Impact, Benefit, and Cost of Applying GQM in Industry. An In-Depth, Long-Term Investigation at Schlumberger RPS, *Proceedings of the 5th International Symposium on Software Metrics* (Metrics '98), Bethesda, MD, 20 - 21 Nov. 1998, IEEE Computer Society, 1998.

[8] Briand, L.C., Differding, C., Rombach, H.D., Practical guidelines for measurement-based process improvement, *Software Process Improvement and Practice*, 2(3): 253-280, 1996.

[9] Cugola, G., Fuggetta, A., Fusaro, P., Gresse von Wangenheim, C., Lavazza, L., Manca, S., Pagone, M.R., Ruhe, G., Soro, R., A Case Study of Evaluating Configuration Management Practices with Goal-Oriented Measurement, *Proceedings of the 4th International Software Metrics Symposium 1997* (Metrics'97), IEEE Computer Society Press, pp. 144-151, 1997.

[10] Gresse, C., Hoisl, B., Wüst, J., A process model for planning GQM-based Measurement, *Technical Report* STTI-95-04-E, Software Technology Transfer Initiative, University of Kaiserslautern, October 1995.

[11] Hamann, D., Järvinen, J., Oivo, M., Pfahl, D., Experience with explicit modelling of relationships between process and product quality, *Proceedings of the 4th European Conference on Software Process Improvement* (SPI '98), Monaco, 1 - 4 Dec. 1998.

[12] Hartkopf, S., Ruhe, G., Leippert, F., How to Make Sense of Empirical Software Engineering Data. An Integrated Approach, *Proceedings of the 10th International Conference on Software Engineering and Knowledge Engineering* (SEKE '98), pp.209-218, 1998.

[13] ISO/IEC TR 15504, Information Technology - Software Process Assessment - Parts 1-9. *Technical Report* Type 2, International Organisation for Standardisation (Ed.), Case Postale 56, CH-1211 Geneva, Switzerland, 1998.

[14] Kuvaja, P., Similä, J., Krzanik, L., Bicego, A., Saukkonen, S., Koch, G., *Software Process Assessment & Improvement – The BOOTSTRAP Approach*, Blackwell Publishers, 1994.

[15] van Latum, F., van Solingen, R., Oivo, M., Hoisl, B., Rombach, H.D., Ruhe, G., Adopting GQM-Based Measurement in an Industrial Environment, *IEEE Software*, 15(1):78–86, January 1998.

[16] Parviainen, P., Järvinen, J., Sandelin, T., Practical experiences of tool support in a GQM-based measurement programme, *Software Quality Journal*, 6(4): 283-294, 1997.

[17] Parviainen, P., Oivo, M., Väyrynen. K., From goal definition to experience packaging: industrial experiences of a GQM-based measurement program, to be published in *Proceedings of ESCOM-SCOPE 99*, East Sussex, England, 27-29 April, 1999.

[18] Vierimaa, M., Kaikkonen, T., Oivo, M., Moberg, M., Experiences of practical process improvement, *Embedded Systems Programming Europe*, Volume 2, No 13, pp. 10-20, 1998.

[19] van Solingen, R., Berghout, E., The Goal/Question/Metric method: A practical guide for quality improvement of software development, McGraw-Hill Publishers, 1999.

ABOUT THE AUTHORS

Dirk Hamann received the Diplom-Informatiker degree (M.Sc.) in computer science with a minor in economics from the University of Kaiserslautern, Germany, in 1996. Since 1996, he is working as a researcher and software engineering consultant at the Fraunhofer Institute for Experimental Software Engineering (IESE) in Kaiserslautern. He has been working together with companies like Bosch, Bosch Telecom, Daimler Chrysler, Deutsche Bank, and Dräger Medical Technology in transfer projects. His current research activities and industrial activities include software measurement, software process improvement, and software process assessment. He is a certified BOOTSTRAP 3.0 assessor and member of the IEEE Computer Society. Dirk Hamann can be reached at Fraunhofer IESE, Sauerwiesen 6, D-67661 Kaiserslautern, Germany, Email: hamann@iese.fhg.de.

Dietmar Pfahl studied applied mathematics, software engineering and economics at the University of Ulm, Germany, and the University of Southern California (USC), Los Angeles, USA. His education was complemented by studies of linguistics, literature and logic at the Ludwig Maximilians University in Munich, Germany, and the University of the Basque Country in Donostia (San Sebastian), Spain. In 1986, he received a M.Sc. degree in applied mathematics and economics (Diplom-Wirtschaftsmathematik) from the University of Ulm. Between 1987 and 1996 he was a junior scientist at the German Aerospace Research Establishment (DLR) in Oberpfaffenhofen, Germany, and a research staff member and software engineering consultant with Siemens Corporate Technology in Munich, Germany. Since 1996, he is with the Fraunhofer Institute for Experimental Software Engineering (IESE) in Kaiserslautern, Germany, where he works in different national and international research and transfer projects with software industry.

Janne Järvinen has received his Master of Science and Licentiate of Philosophy degrees from University of Oulu in 1993 and 1997, respectively. He has industrial experiences from companies like Ericsson, Siemens, Modera and CCC Software Professionals working as a software engineer and project manager. Janne Järvinen has been involved in the international SPICE-project from 1994 mainly as a core team member in the assessment instrument group. He was a member of the Esprit/Bootstrap project team and he is a certified Bootstrap 3.0 lead assessor. Currently he is on a leave from VTT Electronics where he has been working as a research scientist concentrating on software process assessment and improvement in the domain of embedded software. Janne Järvinen has written about 20 scientific and professional articles of software process assessment and improvement.

Rini van Solingen (M.Sc.) is working as a senior quality engineer in industry on software quality and process improvement, already for many years. Within this topic he is specialised on software measurement. He has given practical support to e.g. Dräger, Ericsson, KEMA, Philips, Proctor & Gamble, Schlumberger, TNO en Tokheim on the industrial application of software measurement. He participated in the industrial co-operation projects: ESSI/CEMP, SPIRITS, SPACE-UFO, and PROFES. He is author of the McGraw-Hill software measurement book: 'The GQM method: a practical guide for quality improvement of software development' (http://www.gqm.nl). Within Eindhoven University of Technology, Faculty Technology Management, he performs Ph.D. research on product quality focused software process improvement. He is a member of the IEEE Computer Society.