Goal based Tailoring of Quality Models for Quality Requirements

Arfan Mansoor
Software Architectures and
Product Line Group
Ilmenau University of Technology
Ilmenau, 98693, Germany
Email: arfan.mansoor@tu-ilmenau.de

Elena Rozova
Software Architectures and
Product Line Group
Ilmenau University of Technology
Ilmenau, 98693, Germany
Email: elena.rozova@tu-ilmenau.de

Detlef Streitferdt
Software Architectures and
Product Line Group
Ilmenau University of Technology
Ilmenau, 98693, Germany
Email: detlef.streitferdt@tu-ilmenau.de

Qaiser Abbas
Department of Computer Science
& Information Technology
University of Sargodha, Sargodha, 4100, Pakistan
Email: qaiser.abbas@uos.edu

***Abstract*—Context and motivation: Elicit mean to gather, acquire or to extract while requirements elicitation mean to gather or discover requirements. The activity is performed to determine the system requirements from stakeholders, system documents, domain knowledge and from other requirements sources. Question/problem: Requirements engineering is the most important part for a successful software system because here we come to a decision, 'what' is going to be built. Wrong decisions during this phase will have negative impact on the final product. Idea: The objective is to develop better understanding of requirements. In the start of requirements engineering process we have only few requirements (along with system vision statements) but at the end of this activity most of the requirements or in ideal scenarios all requirements need to be known at the appropriate level. The idea is to propose an integrative goal-quality model for requirements. The success of software product is highly dependent on Non-functional Requirements (NFR). In this paper an integrative goal model of influencing factors is presented. This helps to guide the tailoring of software quality model which is based on various project requirements, organizational needs, individual goals of developers and constraints of the environment. Contribution: The influencing factors help to integrate goal model to quality model and therefore helps in a systematic elicitation of project specific requirements.**

Keywords*—Requirements; goal models; quality models; meta model

I. Introduction

Eliciting clear, complete, and consistent requirements is a challenge and intricate task in requirements engineering due to number of reasons, for example, communication barriers that makes common understanding difficult. The requirements reside in scattered sources e.g., stakeholder, text documents, requirements models etc., and they are present in different forms e.g., as an idea, intentions or needs in the minds of stakeholders. For a successful requirements engineering process, all the relevant sources should be considered during requirements elicitation activity.

The first goal of requirements engineering (RE) process is defined as "all relevant requirements shall be explicitly known and expressed at the desired level of detail" [1]. This RE process is decomposed into four major activities. These are requirements elicitation, requirements analysis, requirements specification and requirements validation. Requirements elicitation is the earlier phase of requirements engineering process [1] where requirements are elicited from different sources e.g., from customer, user and other related documents. The elicited requirements at this stage are named as customer requirements or user requirements. Customer requirements are then analysed to discover problems especially problems related to inconsistent requirements (no requirements are contradictory), to identify the missing requirements (no needed services or constraints have been missed out) and to develop new and innovative requirements [2]. The feasibility of requirements in the context of budget and schedule is also carried out at analysis phase.

An important objective is to realize the relations among requirements and to find the requirements conflicts and overlaps. In case of conflicts the requirements are negotiated to find a compromise among the stakeholders. The research indicated the main origin of project failures is the lack of due diligence at the requirements engineering phase [3]. This study indicates 23.8% projects were canceled because of communication barriers between team members and end users, ambiguous requirements definition, and poor requirements management. Another study shows that 90% of system failures are tracked back to poor requirements elicitation [4]. Neglecting non-functional requirements is among top 6 risks in requirements engineering and it is often worse than forgetting a stakeholder [5]. Non-functional requirements are the heart and soul of why customers value your software and neglecting them often leads to re-architect which requires more time and cost.

There is no general quality model that fulfills all the needs and therefore quality models have to be tailored for specific

domain and for specific solutions. Most of the literature work has focused on the representation of requirements i.e., on requirements specification phase of RE process. The essential properties of software requirements specifications according to IEEE Guide [6] are following: unambiguous, complete, verifiable, consistent, modifiable, traceable, usable during operations and maintenance. In [1] the main goals of requirements engineering are characterized by three dimensions i.e., content dimension, agreement dimension, documentation dimension.

- Content dimension deals with understanding of requirements, all requirements should be known and understood in detail.
- Agreement dimension deals with agreements among relevant stakeholders about known requirements.
- Documentation dimension deals with documentation/specification of requirements in compliance with defined formats and rules.

Requirements elicitation process interleaves the first two dimensions and therefore a worthy requirements elicitation process must provide a solid base upon which the specification documents with desired attributes are produced.

In this paper a goal based integrated model for quality requirements is presented. A comprehensive list of quality attributes available in literature is also presented. In addition the need of a complete model of attributes in tailoring quality models is emphasised.

II. Requirements Engineering Challenges

The main challenges faced during the requirements engineering activities can be organized into these main groups: [7]

- Scope Concerns: They relate to determining the system boundary and the objectives of the target system. To little or too much information results in incomplete, ambiguous, unverifiable, and unnecessary requirements. These requirements are not accurate representation of actual user needs and therefore they are not implementable under system constraints.
- Understanding Concerns: They occur because of user's poor or incomplete understanding of his needs, technical capabilities and constraints. Analysts may lack in domain knowledge. Communication barrier exist between user and analyst, for example, both of them speak different languages. There are conflicting and unspoken or assumed requirements from different stakeholders. These problems results in requirements which are fuzzy in nature and hard to test.
- Volatility Concerns: The requirements are not discovered just at one time. They rather evolve over time in requirements engineering process. There are some requirements which are bound to change. The change may come because of user needs evolve or market trends are changed or there is a change in technology and for that a certain change is required.

Because of the above mentioned concerns, requirements engineering process need to be iterative in nature so that it can accommodate changes [8]. It is important to consider the context in which requirements are being elicited. Requirements elicitation process is performed in the following contexts [9]: Organization, Environment, Project, Constraints imposed by people provide contribution to the requirements elicitation phase.

- Organization context: Although requirements elicitation emphasis on the system's mission statements, but the overall organization context is often neglected. The requirements elicitation phase requires to understand the organization context where the system will work. The important factors of organizational context include: input submitters, output users, ideas and measures by which the system will impact to change the business process.
- Environment context: Environmental context is necessary because the developing system must interface with the other systems. Requirements elicitation phases is strongly influenced by the environmental constrains. For one type of applications there might be requirements of certain methods and tools but these may not be needed for other types of applications. Important environmental factors include: hardware and software constraints, domain knowledge of the target system, knowledge about the interfaces of target system with other systems, The role of target system in larger system perspective.
- Project context: The project context will also have influence on the requirements elicitation process. The factors of project context include: different stakeholder and their attributes. The stakeholders include end users, sponsors, developers, and requirements analysts. Some of the main attributes of these stakeholders are: style of management, hierarchies of management, knowledge and experience of domain, and technical experience.
- Constrains imposed by people: Requirements elicitation is a human intensive process and there are certain constraints that are imposed by the people involved in the process. Some of these constraints are managerial constraints which deals with the cost, time, and desired quality of the target system.

III. Requirements Elicitation using Goals

According to [10] requirements elicitation in terms of tasks should facilitate:

- Relevant requirements sources identification.
- Existing requirements elicitation from identified sources.
- Innovative and new requirements development.

Identification of relevant requirements sources starts by the analysis of already gathered documents. These documents contain information about the organization (i.e., enterprise goals) and may have system specific information (i.e., requirements). For identification of relevant requirements sources other approaches that complement goal based analysis are also used. In [10] a two step procedure is proposed: in first the

relevant requirements sources are identified and in second step requirements are elicited from those identified sources. The numbers of identified resources are restricted due to number of factors e.g., time, cost, availability of experts. In the first step techniques like interviews, workshops, or brainstorming sessions are used to identify relevant sources. The collected sources are added to the already identified sources. The process iterates until newly identified sources become sufficiently low or reaches to zero. For assessing the relevant sources a test named '100-dollar test' is proposed [11]. In this test each stakeholder is given 100 dollars to spend on the items and in the end the average amount of money spent on the items determines the relative weighting of that item. Now the requirements sources are prioritized depending on the amount spent on each item.

After the identification of relevant sources, the next step is to elicit and analyse requirements. Current system's exploratory analysis is important source for goal identification. In some situations high level goals are clearly described but in other situations they may be hidden and an explicit elicitation process is required to identify further goals. The intentional keywords are main clue for goals identification in provided requirements sources which includes interviews, transcripts, mission statements, policy statements, etc. Finding the deficiencies that can be formulated and by negating them is a naive method that produces an initial list of goals [12]. After the high level goals identification, they are refined to elicit further goals until system requirements are met. Scenarios, use cases and initial goal model are useful approaches for the system requirements elicitation.

Non-functional requirements are used as judgement criteria for the operation of a system, rather than describing the specific behaviours. Functional requirements state the functionality of the system, that is, 'what' the system must perform while non-functional requirements determine 'how well' the system must accomplish the 'what' (functional requirements). Normally the non-functional requirements emerge from functional requirements. They comprise of quality aspects and constraints: quality attributes are the system properties concerning stakeholders aspects and these quality attributes influence the system's satisfaction degree. Constraints on the other hand are mostly non-negotiable properties in comparison to quality attributes. Constraints are restrictions that play an important role in design bargain [13].

IV. Quality Models Comparison

Quality models are mostly used to describe and classify the non-functional requirements. Quality models available in literature are mainly composed of layers. The number of layers may be two or three. In two layers model, the layers represents characteristics and sub-characteristics of quality aspects while in three layers model third layer commonly represents the metrics of the quality aspects. Quality attributes identified by the analysis of quality model's characteristics presented in [14]–[19] are presented in Fig. 1. This gives a comprehensive list of quality attributes available in literature. The '*' represents the presence of attribute in the models mentioned in column.

V. Factors in Tailoring Quality Models

The main issue for tailoring of quality models is inadequacy of influential factors and attributes but goal models can provide a valuable information in that context. By using the goal models, the relation of decisions to the original goals are actualized. In the end there is also the requirement of an integrative model which can describe the stakeholder needs and their goals. This integrative model also need to identify the relations to the quality artefacts and the development environment. This will help in the early analyses of the possible influences of changing goals in the project development. Influence factors description by goal models consist of soft factors and is shown in Fig. 2 [20].

Our experience demonstrates that around 70% of the challenges in software development project are because of soft factors. Therefore, these factors need to be addressed for the success of a project. Fig. 3 shows a goal model refinement of these soft factors.
The **strategy** of a company is a critical soft factor for non-functional requirements selection. The refinement of strategy results in the following sub-goals:

- The intentional product **domain** or domains is one sub-goal of the strategy. Requirements engineering elicitation phase requires the knowledge of required domains. A detail knowledge of the domains will have a broad impact on the architecture of the product.
- Another sub-goal of strategy that will influence the soft factors like performance, memory, available development environment, available compilers of the system is **technology**. For example, the realization of variabilities with the C language has reduced capabilities compared to C++.
- **Stability** is another sub-goal of the strategy. Stability is specifically important for new companies. This goal might impact the feasibility of the development as strategy is exposed of risk for changes.
- The **roadmap** consist of the timing constraints for release planing. For specific release, features are identified. The roadmap will deal with periodic updates of the features.

Like the strategy, personal factors will also have their influence onto other elements. Personal goals of individuals in a software project are derived from the stakeholder model. There are number of stakeholders and each stakeholder may have his own priorities. The importance of their priorities depends on their role in the software project. Because of privacy concerns the personal information might be required to be kept private. This data is used in interaction with other models, for example, strategy and standards.

- Each stakeholder experience should have a role description as described in the elementary development processes (e.g., OpenUp, SCRUM). This helps in future further personal development. The personal experience level in each development step is related to the project roles.
- Each stakeholder might have personal preferences regarding the application domains and technological

Factors/Attributes/ Characteristics	Boehm's Model	McCall's Model	Romann Model	Sommerville Model	Dromey's Model	FURPS/ FURPS+	ISO9126 Model
Maintainability	*	*	*		*		*
Flexibility	*	*	*				
Testability	*	*		*			maintainability
Correctness	*	*	*				maintainability
Efficiency	*	*	*	*	*		*
Reliability	*	*	*	*	*	*	*
Integrity	*	*	*		*	*	*
Usability	*	*	*	*		*	
Portability	*	*	*	*	*		*
Reusability	*	*	*		*		
Interoperability		*	*	*		*	
Human Engineering	*		*	*		*	
Understandability	*						
Modifiability	*						maintainability
Functionality			*	*	*	*	*
Performance			*	*		*	
Supportability			*			*	
Clarity	*						
Documentation	*		*	*		*	
Resilience	*						
Validity	*						maintainability
Generality	*		*	*		*	
Economy	*		*				

Fig. 1. Comparison of quality models.

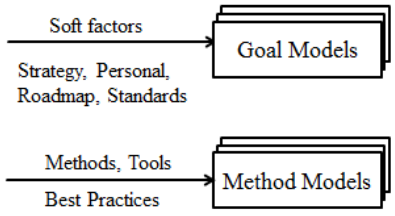

Fig. 2. Goal and method models influence factors.

choices. The personal preferences might also include the methods being used in the development process, specific templates that are used for certain deliverables.

- Stakeholders can have their own approach in addition to the company strategy for certain operations. The complete agreement among the approaches of all the stakeholders is an inconceivable task. For own roadmap development the goal awareness and their relations to own strategy is an important step that helps to integrate individual analysis of the strategies of different stakeholders.

In the end **standards** being followed have their influence on the technology goals. These are important for the strategic planning and these standards may recommend or require technologies and/or tools. As an example, IEC61508 safety standard advocates test case generation tools. The standards may depend upon a definitive development process architecture and distribute knowledge about development methods.

VI. Integrated Model

Mostly the quality models will not fit perfectly for the developing system [21] and therefore the adaptation of these quality models for specific project is required. Here the focus is on the integration of goal models and quality models that helps in the derivation of customized quality models. In last section influencing factors were identified. The adaptation of quality models is based on those specific factors. The general tailoring process consist of three steps:

1) Specifying the goal: The process begins with specifying the higher level goal which defines the needs of the project or organization.

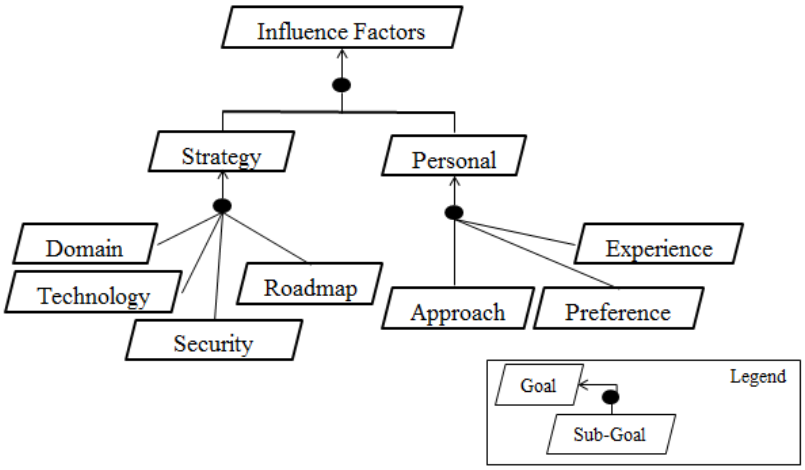

Fig. 3. Influence factors goal model.

2) Specifying quality aspect: The quality related aspects belonging to identified goal are specified. For that, quality models are used. The quality model used to identify the quality aspects is called the reference model. In Fig. 1 all widely used aspects are identified. So, instead of using one particular quality model which may lack quality aspects present in other models this is used as reference model.
3) Tailoring the model: Once all quality aspects are chosen, the ones that are not needed in the final analysis are discarded.

The defined meta-model in Fig. 4 is used to describe quality models in use, integrate the relevant attributes that are specific to stakeholder goals. The meta concepts GoalModel and QualityModel are central to overall meta model. A single goal have OR or AND refinements until the LeafGoal is achieved. LeafGoal can be the Task assignable to Agent or it may be a Quality Attribute(QA) derived from QualityGoal. QA influence other QA and it can also contribute to Task in positive or negative manner. Agents are of two types SoftwareAgent, EnvironmentAgent. Task is generalized form of UserTask and SystemTask having related User QA (UserQA) and System QA (SystemQA). For organizational specific QA, OrganizationalQA meta concept is defined. Each OR refinement may have variation points. Meta concept VariationPoint explicitly define the variability in goal model. VariationPoint represents the variation subject while Variant define concrete type of variation.

VII. Conclusions and Future Work

A classification of quality models from various authors is presented. In comparison of quality models, the quality factors from all these models are presented. In the last an integration of quality models and goal models is discussed and an integrated meta model is presented as an output of that integration. The model helps in the utilization of stakeholder specific goals in the development process. This model supports the soft factors in addition to concrete requirements. Elicitation of quality attributes that influence the development process for different domains is the future research aspect and this work will be a starting point for that future work.

References

[1] K. Pohl, "Fifth international conference on advanced information systems engineering (caise '93) the three dimensions of requirements engineering: A framework and its applications," *Information Systems*, vol. 19, no. 3, pp. 243 – 258, 1994. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0306437994900442

[2] M. Deraitus and A. Miller, "Customer requirements and user requirements: why the discrepancies?" in *Requirements Engineering Conference, 2003. Proceedings. 11th IEEE International*, Sept 2003, pp. 279–.

[3] [Online]. Available: http://www.bcs.org/content/conwebdoc/19584

[4] D. Apshvalka, D. Donina, and M. Kirikova, *Information Systems Development: Challenges in Practice, Theory, and Education Volume 1*. Boston, MA: Springer US, 2009, ch. Understanding the Problems of Requirements Elicitation Process: A Human Perspective, pp. 211–223. [Online]. Available: http://dx.doi.org/10.1007/978-0-387-68772-8_17

[5] L. Chung and J. C. Prado Leite, "Conceptual modeling: Foundations and applications," A. T. Borgida, V. K. Chaudhri, P. Giorgini, and E. S. Yu, Eds. Berlin, Heidelberg: Springer-Verlag, 2009, ch. On Non-Functional Requirements in Software Engineering, pp. 363–379. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-02463-4_19

[6] "Ieee guide for software requirements specifications," *IEEE Std 830-1984*, pp. 1–26, Feb 1984.

[7] M. Christel and K. Kang, "Issues in requirements elicitation," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU/SEI-92-TR-012, 1992. [Online]. Available: http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=12553

[8] L. A. Macaulay, *Requirements Engineering*. London, UK, UK: Springer-Verlag, 1996.

[9] G. Kotonya and I. Sommerville, *Requirements Engineering: Processes and Techniques*, 1st ed. Wiley Publishing, 1998.

[10] K. Pohl, *Requirements Engineering: Fundamentals, Principles, and Techniques*, 1st ed. Springer Publishing Company, Incorporated, 2010.

[11] D. Leffingwell and D. Widrig, *Managing Software Requirements: A Unified Approach*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2000.

[12] A. Van Lamsweerde, "Goal-oriented requirements engineering: A guided tour," in *Proceedings of the Fifth IEEE International Symposium on Requirements Engineering*, ser. RE '01. Washington, DC, USA: IEEE Computer Society, 2001, pp. 249–. [Online]. Available: http://dl.acm.org/citation.cfm?id=882477.883624

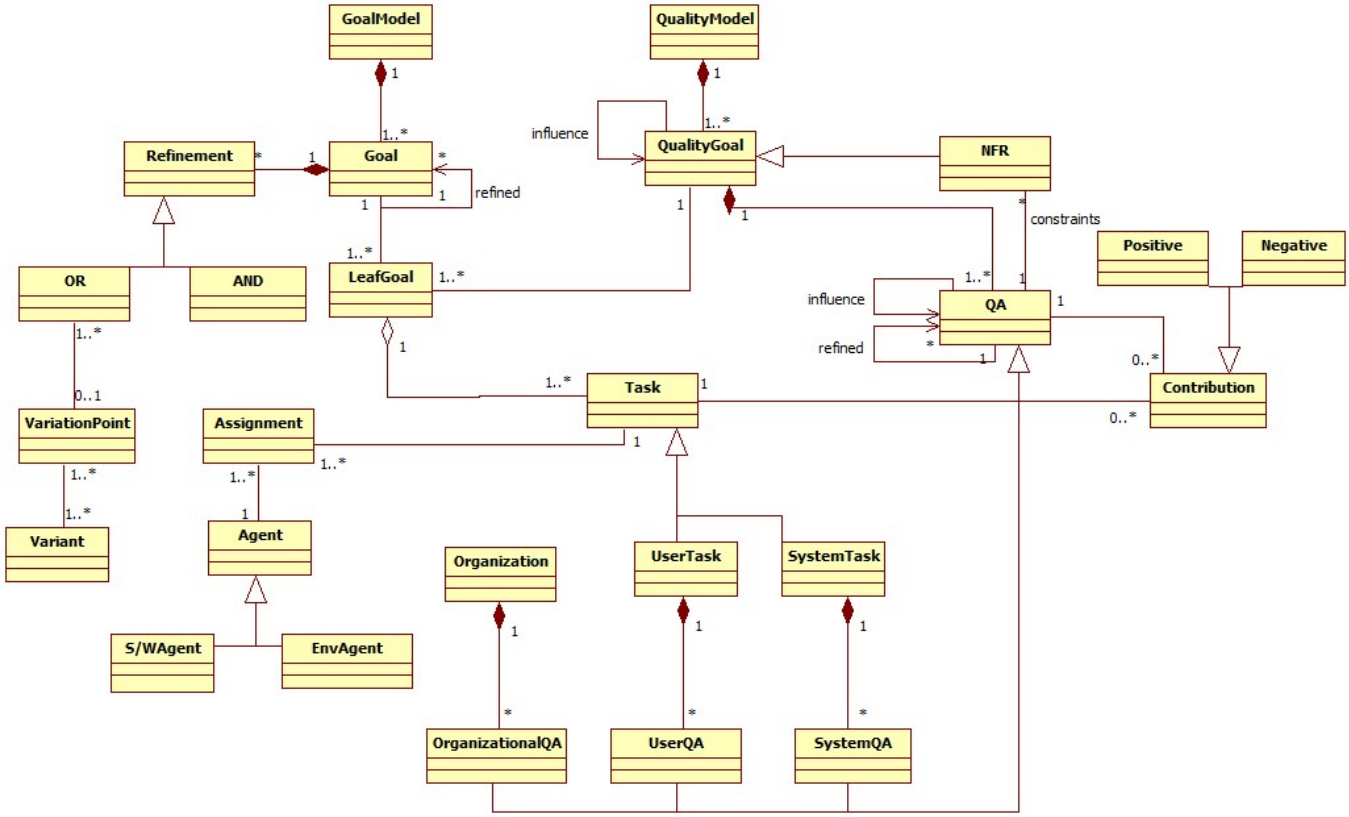

Fig. 4. Integrated meta model.

[13] M. Kassab, O. Ormandjieva, M. Daneva, and A. Abran, *Software Process and Product Measurement: International Conference, IWSM-Mensura 2007, Palma de Mallorca, Spain, November 5-8, 2007. Revised Papers*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, ch. Non-Functional Requirements Size Measurement Method (NFSM) with COSMIC-FFP, pp. 168–182. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-85553-8_14

[14] B. W. Boehm, "Software engineering," *IEEE Trans. Comput.*, vol. 25, no. 12, pp. 1226–1241, Dec. 1976. [Online]. Available: http://dx.doi.org/10.1109/TC.1976.1674590

[15] G. C. Roman, "A taxonomy of current issues in requirements engineering," *Computer*, vol. 18, no. 4, pp. 14–23, Apr. 1985. [Online]. Available: http://dx.doi.org/10.1109/MC.1985.1662861

[16] I. Sommerville, *Software Engineering (5th Ed.)*. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., 1995.

[17] R. G. Dromey, "A model for software product quality," *IEEE Trans. Softw. Eng.*, vol. 21, no. 2, pp. 146–162, Feb. 1995. [Online]. Available: http://dx.doi.org/10.1109/32.345830

[18] R. B. Grady, *Practical Software Metrics for Project Management and Process Improvement*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1992.

[19] I. O. for Standardization, *ISO/IEC TR 9126-4: Software engineering - product quality - Part 4 : Quality in use metrics*. ISO, 2004. [Online]. Available: http://books.google.de/books?id=LM5xkQEACAAJ

[20] A. Mansoor, D. Streitferdt, and M. K. Hanif, "Goal model integration for tailoring product line development processes," *International Journal of Advanced Computer Science and Applications: IJACSA*, vol. 7,2016, no. 7, pp. 618–623, Aug 2016. [Online]. Available: https://www.db-thueringen.de/receive/dbt_mods_00029929

[21] E. C. Groen, J. Doerr, and S. Adam, "Towards crowd-based requirements engineering A research preview," in *Requirements Engineering: Foundation for Software Quality - 21st International Working Conference, REFSQ 2015, Essen, Germany, March 23-26, 2015. Proceedings*, 2015, pp. 247–253. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-16101-3_16