SOQUAREM: SOfware QUAlity Requirements Engineering Method

Rachida Djouab¹, Witold Suryn²

¹École de technologie supérieure
1100 Notre Dame St. West, Montréal, Québec, H3C 1K3 Canada
rdjouab@hotmail.com

²Department of Software and IT Engineering
École de technologie supérieure,
1100 Notre Dame St. West, Montréal, Québec, H3C 1K3 Canada
witold.suryn@etsmtl.ca

Abstract

IT industry needs reliable data about quality requirements to evaluate adequately systems and their architecture. The task is not easy due to nature of these requirements. In fact they are vague, difficult to define and often conflicting with other requirements. Today’s quality requirements management methods are dealing either with one quality requirement aspect (ex: FDAF framework for security or performance aspect) or are developed for a specific type of software (IESE NFR for embedded software). MOQARE is the only method which has been developed to deal with several types of requirements; however, it requires a MOQARE specialist to apply it in industry. The common key point related to all these methods is: “how to identify quality requirements from original source requirements of a system, user and business?” This step seems to be simplified or bypassed by simple interviews or questionnaires, but experience shows that this is a fundamental task required to ensure the correct, operational and properly detailed definition of quality attributes. In this paper, we present SOQUAREM (SOfware QUAlity Requirements Engineering Method), a method to identify and define quality requirements. The aim of SOQUAREM is to support systematic identification of quality requirements at the definition phase of software development lifecycle. It provides a general process model of quality requirements engineering and a business-based process for deriving them in a top down fashion.
1 Introduction

Quality requirements emerged in the two last decades with the evolving technology and the critical need to obtain software with high quality. Importance of quality requirements has been remarked when software developers were faced with returning software and unsatisfied users. Problems of maintenance and costs forced software engineering community to put emphasis on these requirements and develop quality requirements management methods. Tab.1 gives a classification of these methods according to used concepts and levels of their design. As illustrated in the table, business goal oriented methods (MOQARE (Misuse Oriented Quality Requirements Engineering) [1-2] and ATAM (Architecture Tradeoff Analysis Method)) [3] use business goals as main drivers in the software quality process. MOQARE is based on the MisUse concept and a checklist to derive quality requirements from vague requirements. It is applicable to quality requirements derived from business goals, but not proved to be comprehensible for non-technical stakeholders and novices. ATAM (developed by the SEI institute at the Carnegie Mellon University) supports evaluation of given architectural alternatives with respect to quality requirements attributes and identification of tradeoffs and sensitive risks early in the development process. Even if it provides details on elicitation of quality attributes scenarios, this method was only focused on eliciting architecture-centered quality attributes. ATAM used quality attributes at architectural level and presented a detailed description of quality scenarios concept, utility tree and architectural styles to evaluate software architectures. Goal-oriented methods (ASPIRE (Analysis of Software Product In Requirement Engineering) and Soft goal notation) use goal graph structure as driving force to elicit and refine NFRs. ASPIRE method is a structured process for eliciting complete and measurable NFRs (instances of quality attributes) and dedicated to embedded systems [4-6]. Except that the quality goals are not derived from business goals and the tailoring stage is not supported by context rich scenarios. The soft goal notation is applicable to all types of quality requirements [7-8], but focused on the documentation and negotiation of quality requirements, and not on their elicitation from business goals. Aspect oriented method (FDAF (Formal Design and Analysis Framework)) [9-12] is based on “Aspect Oriented Paradigm” concept to define quality attributes. FDAF framework has been designed for a specific quality attribute at architectural level. It has been developed to create architecture designs with NFRs aspects that cannot be described in the real time version of UML. The framework was not concerned with identification, representation or documentation problems at the requirements level. In addition, it is limited with analysis tools and modeling constructs supporting the description of component’s behavior and connections. SOQUAREM (SOftware QUALity Requirements Engineering Method) method has been proposed in this research study to deal with quality requirements in a structured and systematic manner. Dedicated to address all types of quality requirements, it is based on business context elements, scenario concept and the ISO/IEC 9126 [13] quality standard to infer the related quality attributes. Business goals are main drivers of SOQUAREM process.
They are provided by BMM (Business Motivation Model) concept and used as starting point in the derivation process to identify quality requirements and deliver detailed quality attributes. Traceability of these requirements to their original source is modeled in a utility tree.

The paper is structured as follows: section 2 presents SOQUAREM method where its specific features and key concepts are described. Further in the section, the meta-model and supporting techniques for SOQUAREM process are presented and the process model, quality attributes database and logic of SOQUAREM are described. Section 3 summarizes conclusions and continuations of the research work.

<table>
<thead>
<tr>
<th>QRs methods</th>
<th>Concepts</th>
<th>Paradigms and concepts</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Input Oriented Paradigm (IOP)</td>
</tr>
<tr>
<td>MOQARE</td>
<td>Requirement</td>
<td>²</td>
</tr>
<tr>
<td></td>
<td>Architectural</td>
<td></td>
</tr>
<tr>
<td>IESENK/ASPIRE</td>
<td>Requirement</td>
<td>²</td>
</tr>
<tr>
<td></td>
<td>Architectural</td>
<td>²</td>
</tr>
<tr>
<td>ATAM</td>
<td>Requirement</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Architectural</td>
<td>²</td>
</tr>
<tr>
<td>SOFT GOAL NOTATION</td>
<td>Requirement</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Architectural</td>
<td>²</td>
</tr>
<tr>
<td>FCAF</td>
<td>Requirement</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Architectural</td>
<td>²</td>
</tr>
</tbody>
</table>

Table 1: Classification of quality requirements methods
2 Presentation of SOQUAREM method

SOQUAREM method is business goals-centric; stakeholder’s-centered and scenarios-oriented. It is organized around 2 levels (Fig. 1): a) the business goals level which consists of identifying important business goals from the BMM model and BCT concepts; specific rules are used to refine business goals; consensus and free dialogue sessions are used to confirm the refined business goals with stakeholders and domain experts; b) the system quality attributes level where quality attributes are derived from the business goals according to the quality standard ISO/IEC 9126 and linkage rules. They are also operationalized by using scenarios template. Quality attributes scenarios are analyzed for possible conflicts and consolidated by using prioritizing techniques. Consensus session is used to confirm quality attributes scenarios with stakeholders. Utility tree describing traceability of quality attributes is produced. Finally quality attributes are linked to use case model by using mapping rules.

Figure 1: High conceptual levels of SOQUAREM
2.1 Specific features of SOQUAREM method

SOQUAREM solution is proposed to palliate some limitations of existing quality requirements management methods, which are: a) lack of processes and models describing identification, decomposition and representation of quality requirements and b) lack of structured methods or techniques to identify quality requirements. In fact, majority of the studied methods deal only partially or not at all with criteria related to identification, decomposition, representation, conflicts analysis, documentation, derivation of quality attributes from business goals, etc (Tab.2). For example:

- None of the methods fully supports the identification of quality requirements;
- Most of the methods are not based on ISO/IEC 9126 quality standard;
- IESE NFR doesn’t take into account the concept of business goals;
- MOQARE and ATAM do not take into account the concept of quality standard and the task of quality integration with FRS;
- MOQARE doesn’t take into account the task of conflict analysis;

Tab.2 summarizes assessment of studied methods according to established criteria. Tab.3 establishes comparison of studied methods with SOQUAREM according to cited criteria and through their artifacts.

<table>
<thead>
<tr>
<th>Quality requirements methods</th>
<th>Identification</th>
<th>Decomposition</th>
<th>Conflict analysis</th>
<th>Representation</th>
<th>Documentation</th>
<th>Consensus on quality definitions</th>
<th>Quality standard</th>
<th>Derivation from business goals</th>
<th>Integration with FRS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOQARE</td>
<td>Partially</td>
<td>Partially</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Partially</td>
<td>No</td>
</tr>
<tr>
<td>IESE NFR</td>
<td>Partially</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>ATAM</td>
<td>Partially</td>
<td>Partially</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Partially</td>
<td>No</td>
</tr>
<tr>
<td>SOFT GOAL NOTATION</td>
<td>Partially</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>SOQUAREM</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 2: Summary of methods and their characteristics assessment
Table 3: Comparison of SOQUAREM with quality methods through their artefacts

Innovative aspects of SOQUAREM are the following:

1. More interaction with stakeholders and domain experts during consensus and free dialogue sessions;
2. Use of intuitive modeling and motivation of business in the derivation process of quality attributes;
3. Structured derivation of quality goals from business goals by using Business Context Table (BCT) and Business Motivation Model (BMM). Derivation step of quality attributes from business goals is fully described in SOQUAREM;
4. Use of scenarios at the requirements level to resolve terminology problems and infer the correct quality attributes;
5. Use of verification rules like statement rules to define business goals, refinement rules to refine business goals, linkage rules to derive quality attributes from business goals and mapping rules to link quality attributes to the FRs model;
6. Use of globally recognised ISO/IEC 9126 as quality standard for SOQUAREM process;
7. Use of quality template to specify and document quality attributes;
8. Use of prioritizing methods (impact matrix and weighted method) to resolve conflicts among quality attributes.

2.2 Conceptual model of SOQUAREM

Fig.1 presents key concepts involved in the main activities of SOQUAREM process. The first activity related to identifying and refining business goals uses the concepts of BMM, BCT, free dialogue session, consensus session and statement and refinement rules. The next activity addressing derivation of quality attributes from the refined business goals applies BMM and BCT, quality scenarios description, quality standard ISO/IEC 9126, linkage rules, consensus session and prioritizing techniques. The two last activities dealing with documentation and representation of quality attributes use quality attributes template and utility tree.
Figure 1: Description of SOQUAREM concepts
The BCT questions are combined with BMM elements (see Fig.2) which should be developed with participation of concerned stakeholders during consensus sessions.

As shown in Fig.2, BCT elements are mapped with BMM artefacts to refine the business goals ($BG_k$), which are linked to quality attributes (according to quality standard ISO/IEC 9126, scenarios description and linkage rules) to obtain the final quality attributes list ($QAm$). QAs list is discussed with concerned stakeholders during consensus sessions.

![Figure 2: Required elements for identifying quality attributes](image)

Tab.4 shows definitions of key concepts of SOQUAREM process and Tab.5 defines the quality scenarios template used to describe quality attributes scenarios contributing to infer the correct quality attribute. Scenarios template is a key concept of SOQUAREM method. They provide the context for clarifying and operationalizing quality attributes. They are also used to map quality attributes to the functional requirements by using “Action” item.
<table>
<thead>
<tr>
<th>SOQUAREM concepts</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMM (Business Motivation Model)</td>
<td>Starting point of the SOQUAREM method. It is used to define motivation of the business context, state goals and sub goals of the business, related strategies and identifies relevant stakeholders with their corresponding expectations.</td>
</tr>
<tr>
<td>Business context Table (BCT)</td>
<td>Describes fundamental questions about elements of business context. It structures and details items of BMM business context according to the following keywords questions: How, What, Why and Who. BMM and BCT are used in the first three SOQUAREM steps to help refine business goals and derive quality attributes from business goals.</td>
</tr>
<tr>
<td>Free dialogue session</td>
<td>Used to identify and refine business goals from technological constraints, high level functional requirements and covering strategies.</td>
</tr>
<tr>
<td>Quality scenarios description (Tab. 5)</td>
<td>Where meanings of business goals are detailed according to specific items of the scenarios template. Quality scenarios provide a structured way to build quality attributes utility tree.</td>
</tr>
<tr>
<td>Utility tree</td>
<td>Developed for traceability of quality attributes. It shows how quality attributes are organized with the refined business goals and the associated quality attributes scenarios.</td>
</tr>
<tr>
<td>ISO/IEC 9126</td>
<td>ISO/IEC 9126 help stakeholders focus on the most recognized quality characteristics. It is used to infer the right quality attribute from the refined business goals.</td>
</tr>
<tr>
<td>Consensus session</td>
<td>Provides a way to communicate and consolidate quality attributes to stakeholders and obtain final list of prioritized quality attributes. Conflicts among quality attributes are also discussed with stakeholders. Consensus session allows discussing and confirming business goals and their corresponding quality attributes. Consensus session is used to: Confirm business goals with stakeholders; Discuss linkage of quality attributes to refined business goals with concerned stakeholders; Confirm consolidated quality attributes.</td>
</tr>
<tr>
<td>Quality attributes template</td>
<td>Documents quality attributes in the following items: context in which the quality attribute is applied, source of the quality attribute, representation of the quality attribute and impact of the quality attribute on the software process, etc.</td>
</tr>
<tr>
<td>Verification rules</td>
<td>Statement rules: to state and define business goals, refinement rules: to detail business goals, linkage rules: to link quality attributes to refined business goals and mapping rules: to map quality attributes to corresponding use case model.</td>
</tr>
<tr>
<td>Prioritizing methods</td>
<td>Used to find and resolve conflicts among quality attributes (such as impact matrix and weighted method).</td>
</tr>
</tbody>
</table>

Table 4: Key concepts of SOQUAREM
SOQUAREM represents an intuitive, scenarios-oriented approach to quality requirements engineering. Modeling elements in SOQUAREM include business goals; quality attributes scenarios and quality standard ISO/IEC 9126 (Fig. 3). Business goals, influencer and strategies are provided from the BMM model. They should be traceable to the following concepts: quality attributes actors and actions. Quality attributes are clarified and detailed by quality attributes scenarios. Scenarios give information about actions undertaken to achieve quality attributes and asset involved in achieving these quality attributes. Quality attributes are linked to functional requirements by means of actions and actors items of scenario template. They are also specified by using the ISO/IEC 9126 quality standard as supporting framework.

<table>
<thead>
<tr>
<th>Scenarios Items</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>Undertaken to achieve the quality attribute</td>
</tr>
<tr>
<td>Asset</td>
<td>Any part of the system (hardware, software, personnel, development process, data, etc.) involved in achieving the quality attribute</td>
</tr>
</tbody>
</table>

Table 5: Quality Scenarios Template

### 2.3 Meta-Model of SOQUAREM method

SOQUAREM represents an intuitive, scenarios-oriented approach to quality requirements engineering. Modeling elements in SOQUAREM include business goals; quality attributes scenarios and quality standard ISO/IEC 9126 (Fig. 3). Business goals, influencer and strategies are provided from the BMM model. They should be traceable to the following concepts: quality attributes actors and actions. Quality attributes are clarified and detailed by quality attributes scenarios. Scenarios give information about actions undertaken to achieve quality attributes and asset involved in achieving these quality attributes. Quality attributes are linked to functional requirements by means of actions and actors items of scenario template. They are also specified by using the ISO/IEC 9126 quality standard as supporting framework.
2.4 The SOQUAREM process structure

The SOQUAREM structure as illustrated in Fig.4 is organized around steps and uses various techniques and tools (like heuristics, mathematical and intuitive modeling and catalogs of NFRs methods inherited from the soft goal notation), quality standards and verification rules. Stakeholders and domain experts are involved during process
operation. Techniques used are either informal, heuristic or semi formal. The informal ones are consensus, free dialogue session, scenarios descriptions and template. Scenarios descriptions are used to detail meaning of quality attributes and make them operational. Heuristic techniques use descriptive methods to help clarify the business goals and identify quality attributes. Semi formal methods are using UML modeling to represent operational part of the quality attribute (actions undertaken to achieve it) and to link them to the functional requirements (represented in the use case model). Mathematical methods as utility tree, impact matrix and weighted methods are used to represent quality attributes and resolve conflicts among them. Verification rules are used during the whole process to regulate the operation process and are subdivided into: statement rules to define business goals, refinement rules to refine business goals into refined business goals, linkage rules to derive quality attributes from business goals and mapping rules to link quality attributes to the functional process by the use case model.

Figure 4: SOQUAREM process structure
2.5 SOQUAREM process model

The SOQUAREM process model is divided into six conceptual steps for defining and refining business goals, deriving, operationalizing, analyzing, documenting and representing quality attributes and finally for linking quality attributes to the functional process. These steps use various quality requirements elicitation techniques (Questionnaire, Consensus session, BMM, Scenarios, Prioritization, Utility tree and Template). Potential inputs of the process are BMM, BCT and domain experts. Main participants are quality requirements engineer, domain experts and selected stakeholders.

SOQUAREM process model is represented as (Fig. 5):

**Step 1:** State and identify the business goals: define relevant element of business context like business goals and business domain.

**Step 2:** Refine business goals: business goals are detailed according to additional business information such as organizational culture, regulations and guidelines, technological constraints and business strategies to achieve business goals.

**Step 3:** Link business goals to the corresponding quality attributes: detailed business goals are used to derive the quality attributes by using ISO/IEC 9126 quality standard and linkage rules.

**Step 4:** Build quality attributes scenarios by using the scenarios template and the consensus session techniques to infer the right quality attribute.

**Step 5:** Analyze conflicts between QAs and consolidate them by using prioritization methods.

**Step 6:** Link quality attributes to the functional requirements process by updating the initial use case model with additional information about QAs.
Figure 5: SOQUAREM process model
Fig. 6 illustrates the quality attributes database describing relevant information related to SOQUAREM process. Quality attributes are the core of the database. They are applied in a specific business domain. They are derived from refined business goals and specified according to ISO/IEC 9126 quality standard. They are detailed by quality scenarios and realized by “Action” item of the scenarios template. Quality attributes are concerned with stakeholders like business manager and developer and are required by development models like sequence diagrams and architectural model. They shall be verified or realized in all the software lifecycle phases. Finally, quality attributes are affecting other types of requirements like functional and cognitive requirements.
2.6 Logic of SOQUAREM

Fig.7 and 8 summarize linkage process and logic of SOQUAREM. Fig. 7 shows linkage process of SOQUAREM involving elements of business context (like business vision, business goals and strategies) to be refined and linked to system elements like quality attribute, actors and associated actions.

As illustrated by Fig.8, quality attributes are identified from business goals and integrated in the business process. SOQUAREM process is used at two levels: business level where elements of business context as BMM and BCT are used to help identify business goals and refine them into refined business goals. At the system level, quality attributes are linked to refined business goals by using scenarios and quality standard ISO/IEC 9126. They are also integrated to the functional process by means of mapping rules, use cases and business domain models. SOQUAREM provides traceability from business requirements to system specifications by mapping business context elements to quality attributes of system level in two ways:

1. Business goals are refined into sub goals and linked to quality attributes (blue, green and purple colors).
2. Influencer (external and internal) are mapped to:
   a. Actors responsible for achieving quality attribute (red color);
   b. Actions undertaken by actors to achieve quality attribute (brown color).

![Figure 7: linkage process of SOQUAREM process](image-url)
Figure 8: Logic of SOQUAREM process model
3 Conclusions and future work

The presented research study introduces and describes a software product quality requirements engineering process which deals with most of quality requirements aspects like identification, representation, prioritization and documentation. It is dedicated to all types of quality requirements at early stage of software and uses business goals as main drivers. Scenarios template, utility tree, documentation template and ISO/IEC quality standard are used as well to ensure completeness and traceability of quality attributes of the software product. It is also based on a detailed analysis of software requirements at business level to define the correct quality requirements.

The main objective of SOQUAREM process is to build bridge between business level elements and system level ones and allow more interaction between stakeholders, software people and domain experts during consensus and free dialogue sessions.

The possible continuation of this research should focus on:

1. Validating SOQUAREM process in large industry;
2. Developing supporting tool which will show relevant parts of SOQUAREM process model and help business and software people understand quality attributes management and make right decisions;
3. Integrating SOQUAREM process with software engineering processes/methods like RUP (Rational Unified Process), RAD (Role Activity Diagramming), Architecture centered design and OOAD (Object Oriented Analysis and Design);

4 References


