Categorization-based requirements engineering process with a common instrument for all roles within this procces.

Tatiana Chuprina
Model-based System Engineering
Fortiss GmbH
Munich, Germany
Email:chuprina@fortiss.org

***Abstract*—In this paper categorization-based concept named "Aspects" is presented, which have been implemented in Auto-FOCUS 3 development tool as an instrument supporting requirements engineering activities such as requirements structuring and their systematic review.**

I. Problems and Motivation

In software-intense systems development there are many types of different obstacles and challenges, which are still make an influence on quality and cost of designed systems as well as on a development process flow remaining unsolved or not proper considered.

According to information from A Review of Requirement Engineering Issues and Challenges in Various Software Development Methods, challenges regarding requirements quality, communication gap between roles included in development process such as requirements engineer, reviewer or software developer etc., lack of structured process of mapping requirements to design (Semi-Automatic Process of Generic Template Creation) bring about plenty impediments into software development process of various kinds and may result in great increase of resources from industry [1]. Moreover, Daniel Mndez Fernndez provides a statistics of failure projects causing by the mentioned problems among another, pointed the most critical issues such as Incomplete or hidden re-quirements, Communication flaws within the project team, Separation of requirements from solutions, Time boxing, Moving targets [16]. All these points show a demand of requirements engineer-ing as a complete field of study.

Importance of requirements engineering activities extremely rises for safety-critical systems. An example is an Avionics domain where risk of the error should be minimized from the initial stage of the development process, during requirements elicitation and analyzing. Otherwise, the flaw can affect a system design on different phases of development process and results in crucial consequences.

Therefore, motivation for this research is to elaborate and investigate a general concept (named The Aspects) which is in help for solving the mentioned problems regarding requirements quality by their structuring using the Aspects. Additionally, to study an impact of the Aspects on requirements development process.

II. Related Work

The main idea of the Aspects concept arises from analyses of general requirements categorizations provided by scientific overview in requirements engineering field and by collected experience of partners from industry with compliance to their standards.

The scientific part of the research based on such study work as doctoral thesis Categorizations of Product-related Requirements in Practice Observations and Improvements by Jonas Eckhardt [3]. There the author considered general requirements categorizations commonly applied in industry and recognized within scientific community (i.e. Pohl, 2010; Robertson and Robertson,2012; Sommerville and Kotonya, 1998; Van Lamsweerde, 2001)[2][2,4,5,6] and investigated requirements categorization based on a system model and its impact on requirements quality, relying on state-of-the-Art research results such as Broy 2016 [10], Mager 2015 [11] and others. The next paper, which made an impact to start this research was Does Goal-Oriented Requirements Engineering Achieve its Goal?[15], where the authors raise doubts about applying of goal-oriented requirements engineering (GORE) in practice due to missing sufficient documentation for industry and lack of comprehension between scientific society and industry. And as a consequences, the article Supporting Requirements-Engineering Research That Industry Needs. The NaPiRE Initiative by Daniel Mndez Fernndez[16] describes current problems in industry with concern to requirements engineering field and possible reasons for existing gap between researchers and industry.

The industry perspective of this study supports by practical knowledge assembled by industry partners of Fortiss with conformity to industry standards and regulations such as DO-178C [7], DO-331[8], ISO29148:2011[9] etc.

The Aspects incarnate a core of the mentioned categorizations and go far ahead including analysis of industry practic-es, thereby providing a possibility for different granularity within any category.

Currently, the Industry operates with Requirements in a form of text (use cases, specifications with detailed lines of description or bulleted list, etc.), which usually poor structured and error-prone, resulting in bed quality of requirements and the problems associated with this, described above.

For managing textual requirements and supporting traceability between requirements, industry usually uses such tool as DOORS, BTC and others.

Alternatively, in practice can be applied patterns to bring the requirements into formalized view. For example, using Requirement Quality Suite [12] or languages such as EARS [13] or UML [14] etc. However, this approach doesnt give flexibility in sense of requirements engineering and being lack of guidance for a user (requirements engineer, reviewer or developer).

In contrary to such technics, the Aspects method provides a noticeable degree of freedom for requirements engineering. It allows to work with unstructured or unorganized textual requirements and adhere incrementally a skeleton based on the Aspects, categorizing requirements. Furthermore, the con-cept supports a method for elaborating requirements or re-viewing them.

Moreover, the idea of the Aspects is general and allows a user to be independent from a specific tools. The Aspects can be applied within already well-known for a user chain of tools and thereby, save costs and time for a project realization in-stead of employee teaching a new extra tool.

Additionally, the mentioned above tool examples have on a hunch an assumption, that input for the designing phase should be brought to a completed structure before developers start their job. Such implication doesnt look realistic due to changeability of requirements during timeline of a project and could cause developers to be involved in requirements engineering performance. It can lead to risks rise, difficulties in standards compliance and reflect on costs of a project. The Aspects concept allows to avoid such problems and change requirements during a whole system life cycle.

III. Solution

An idea behind the concept of the Aspects arises from general requirements categorization and goes ahead with arrange of requirements in formal structure categorizing them by their characteristics and thereby, forming an attached combination of the categories (Aspects). This combination of the Aspects constitutes a structure of the requirements scope, a new Artifact. The Aspects compose different types with diverse granularity.

Commonly, a development process starts from the requirements elicitation mostly as text and analysis. AF3 supports requirements in textual format and provide the author to choose which Aspect is relevant to the considered requirement.

There are several types of Aspects (i.e. timing , functional, signal etc.) which represent diversity of requirements' objectives (inclinations) Figure 1.

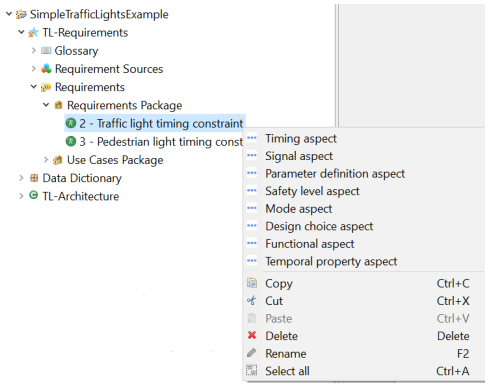

Fig. 1. Listing of the Aspects in AutoFOCUS 3

The Aspects concept is implemented in AutoFOCUS 3 model-based development tool and domain specific language for embedded systems design. There are eight main Aspects supported by AutoFOCUS 3: Timing aspect - the requirement refers to timing con-straints Signal aspect - the requirement defines a signal, its type and its range Parameter definition aspect - the requirement defines a parametric value. This allows to reused other re-quirements by changing the value of the parameter Safety level aspect - the requirement defines a safety level of the system Mode aspect - the requirement defines an operational mode of the system Design choice aspect - the requirement was derived from a design choice of the software developer Functional aspect - the requirement specifies a partic-ular behavior by relating inputs and outputs Temporal property aspect - the requirement defines a property expressed by temporal patterns

Depending on a characteristics of considered requirement the appropriate Aspect or set of Aspects can be applied to it. The number of applied Aspects can indicate about the requirements complexity exposing that too many responsibilities endow on it and it should be revised for further transformation (i.e. splitting into several less complex requirements). According to different Aspects may be taken in account diverse analysis methods and review activities during requirements examination. Furthermore, every Aspect provides a checklist specified according to a represented requirement category, which gives an ability to write requirements and review them in one tool, instead of using different tools for every activity. In AutoFOCUS 3 checklists implemented as automatized.

The Aspects incarnate a common instrument for all roles involved in requirements engineering process and narrow a gap between requirements engineers, reviewers and developers by formalizing requirements and holding up a guidance for these roles.

The following use cases will be discussed for the Aspects better explanation:

1. Use Case provides an example of requirements and shows how the Aspects concept can be applied for struc-turing requirements. This Use case can be validated by empirical experiments considered to be conducted further within this

research. 2. Use Case of avionics component development applying the Aspects. This use case shows how the Aspects concept supports a process of requirements review. The use case was approved within industrial project (ASSET) conducted by several companies from industry with Fortiss. 3. Use Case Beyond AF3 is an inner project of Fortiss, which validate a general nature of the Aspects concept. In other words, it demonstrates independency of the As-pects concept from a precise chain of tools.

A. Use Case1: Requirements categorization using the Aspects.

Stakeholders very often provides requirements in textual form, which have no structure and can consist of bulleted lists or long sentences of specification and etc. Example of such row requirement: The input for the component shall be Z number of latency limits, assigned to the ring buffers, with allowed range [Xms; Yms]. To translate this system specification (system requirement) into High Level Requirement (HLR), which further will substitute appropriate Low Level Requirements (LLR) for system design, this system requirement should be analyzed and structured accordingly. For instance, to structure this system requirement the fol-lowing Aspects can be applied to it: a signal aspect, a parame-ter aspect, timing aspect and temporal aspect as well. Therefore, the system requirement content is not changed, but main characteristics of this statement are defined. The number of matched aspects shows, that the requirement can be divided into several less complex requirements matching the aspects categorization. In this case, the high Level Requirements could be a list of following: The input for the component shall be Z number of la-tency limit signal and parameter aspects; latency limits are assigned to the ring buffers - pa-rameter aspect; latency limits has allowed range [Xms; Yms] pa-rameter and temporal aspects;

Now, these transformed requirements can be revised by separate reviewer manually (as it is required by some standards in industry i.e. aerospace and defense, DO-178C[7]). And after checking succeed and HLR have been also conformed by stakeholders, the procedure with the Aspects application repeats again in order to structure HLR into LLR and then a new review starts.

The process of requirements reviewing is described further in Use Case 2.

B. Use Case 2: Requirements engineering and reviewing with use of the Aspects.

There is a description of the work flow supported by Aspects during requirements analysis and review. This method was realized in AutoFOCUS 3 within ASSET project develop-ing a component for avionics system (Figure 2).

The process divided into parts by roles. The main roles here are a requirements engineer (REer) and a reviewer; the responsibilities of roles Stakeholders and SA design can be delegated to the main roles [17].

The Aspects can be applied to system requirements from initial stage of the development, from system requirements

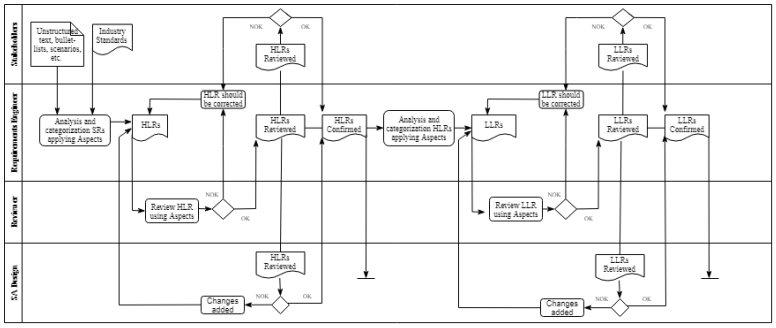

Fig. 2. Work flow supported by Aspects during requirements analysis and review.

(SRs) elicitation (see use case1). SRs analyzing and structuring requirements by the Aspects transforms into HLRs. Now HLRs can be provided to a reviewer for checking. The reviewer checks manually the HLRs, which are still in the text form and the Aspects attached to them. In context of aspects, a reviewer can examine HLRs for the following points: if the Aspects applied for every requirement and if the combination of Aspects matches to the HLRs.

Review process is maintained by check-lists inferred form the Aspects (every aspect has its own check-list). In AF3 these check-lists implemented as a semi-automatized feature to reinforce requirements review.

In positive case, the HLRs can be provided further to SA design team and/or to stakeholders for next consideration. HLRs is a textual artifact with attached aspects, which supports the HLRs structure. Therefore, stakeholders work again with a document of text requirements. In case the check of HLRs fails, a reviewer re-applies the Aspects to HLRs and restructures them and sends back to REer for repeated analysis and correction. REer now have a hints in a form of aspects structure attached to HLRs by a reviewer, so that he/she can clearly understand the nature of the correction in HLRs and restart the requirements analysis again.

For Developers use there exists a Design Choice aspect, so that they can indicate a new changes in requirements com-ing from their side and by that initiate a revision procedure again.

After this procedure, the HLRs confirmed and can be as an input to SA design process and to further LLRs. The proce-dure with LLRs starts with the analysis process and transform-ing the structure of the requirements applying the Aspects; and all procedure described above repeats for LLRs. The Aspect supports traceability between requirements on all stages of development. Furthermore, the Aspects appears here as a common instrument for all roles within requirements engineering and review processes guiding them through the work flow.

The Aspects concept implementation in AF3 reveals, that all described activities can be performed in the same tool. It gives an advantage with keeping traceability between all lev-els of requirements from SR to HLR and LLR as well as be-tween requirements and system design, which is critical for avionics.

C. Use Case3: General application of the Aspects.

This use case is in a research process and can be specified as a future work within this study.

The main idea of this use case is to apply the Aspects con-cept within other than AF3 tool i.e. GitLab, explore and vali-date if the concept holds up its main characteristics independently from AF 3 tool.

IV. Evaluation and Validation

Firstly, the Aspects concepts influence on requirements engineering process in a positive way has been proven by Industry Use case of avionics component development within ASSET project, in which an amount of industrial partners were involved. During the project, there were elaborated and validated a method of guiding a requirements review based on the Aspects. what impact the Aspects make on requirements reviewing process handle requirements complexity.

The second step in V&V process of the Aspects concept is validation of the general use of the Aspects within Beyond AF3 Fortiss project.

Next, for validation of the Aspects concept can be carried out a series of empirical study, which can be conducted within Fortiss research institute to explore, how user of the Aspects concept can handle a problem of requirements quality.

Though, according to Daniel Mndez Fernndez: Its not surprising that the biggest challenge in RE research is to provide proper empirical figures that demonstrate particular success factors. However, those factors are critical determinants of what works in practice and what doesnt.4 Consequences are that the state of empirical evidence in RE is particularly weak and that much of everyday industrial practices is gov-erned by conventional wisdom rather than empirical evi-dence. [16], which means, that in this study the question of an appropriate and adequate validation technics for the As-pects concept is going to be investigated.

V. ExpectedContributions

Conducting this research the following goals will be (being expected to) achieve: to bridge a gap between requirements engineers and reviewers providing to them a feasible method based on the Aspects, which guides both roles within requirements reviewing process and supports a communication flow between these two roles. That leads to ease of redundancy in the work-flow; to close a gap between requirements reviewer and system developers by providing a structured constitution of requirements which is in help to concentrate a precise overview on the requirements (as on a whole scope of them so as only on specific category of requirements or part of them, depends on categorization granularity). And as a result, a reviewer can provide a more specified and detailed response to developers on any changes in a system; moreover, the Aspects concept links a bridge between requirements engenderers and developers, so that a structure of Aspects itself includes a key hints for every requirements category, which help in seamless mapping requirements into design of a system; the Aspects is a general concept, which can be applied independently on a specific tool chain. They can be simply applied with a familiar development tools. It gives a flexibility in use and reduces a costs for new tools learning; semi-automatically check-lists, which are an extensions for every Aspect, useful for making a requirements review easier and consequently, less time- and resource-consuming;

All in all, the Aspects concept provides a common instrument for all mentioned above roles (requirements engineers, reviewers, developers). All these points can increase a quality of requirements and thereby, save a significant amount of resources in system development such as project time and expenses.

VI. CurrentStatus

Currently, the research considers a relevant and sufficient methods for proper evaluation of the Aspects concept. Simultaneously to this research task, further implementation of the Aspects concept in AF3 tool and its testing continues.

The following steps within the research appear as a checking of usability of the concept conducting a series of empirical experiments.

Moreover, the study is supposed to consider an impact of the Aspects application on other phase of development pro-cess, for example, such as testing, requirements reuse and etc.

Implemented within the MIRA framework[18] in AutoFOCUS 3 and reflects requirements engineering activities form the initial stage of a development process, the Aspects can represent a general concept supporting its unique method, which may be applied within other tools thereby, provides tool/platform- independance. The next pace in this direction of the research is a deeper investigation of this theory and its consideration on practice.

And the most challenging question about proper and adequate validation technics for the Aspects concept will be considered in future work in this study.

References

[1] A Review of Requirement Engineering Issues and Challenges in Various Software Development Methods
[2] Pohl, 2010;
[3] Jonas Eckhardt. Categorizations of Product-related Requirements in Practice Observations and Improvements
[4] Robertson and Robertson,2012;
[5] Sommerville and Kotonya, 1998;
[6] Van Lamsweerde, A. (2001). Goal-oriented requirements engineering: A guided tour. In Proceedings of the 5th International Symposium on Require-ments Engineering (RE),
[7] DO-178C.
[8] DO-331
[9] ISO29148:2011
[10] Broy, M. (2016). Rethinking Nonfunctional Software Requirements: A Novel Approach Categorizing System and Software Requirements. In Hinchey, M., editor, Software Technology: 10 Years of Innovation in IEEE Computer. John Wiley & Sons/IEEE Press.
[11] Mager, P. (2015). Towards a Profound Understanding of Non-Functional Requirements. Masters thesis, Tech-nische Universitt Mnchen.
[12] Requirement Quality Suite
[13] EARS
[14] UML
[15] Does Goal-Oriented Requirements Engineering Achieve its Goal?, Tuefl, Eckardt, Mund, Femmer

[16] Supporting Requirements-Engineering Research That Industry Needs. The NaPiRE Initiative by Daniel Mn-dez Fernndez
[17] Mini-Guideline to Requirements Engineering. Birgit Penzenstadler
[18] MIRA framework