

A GROUNDED THEORETICAL AND LINGUISTIC ANALYSIS APPROACH FOR NON-FUNCTIONAL REQUIREMENTS ANALYSIS

Research-in-Progress

Suranjan Chakraborty
Towson University, USA
8000 York Road, Towson, MD 21252
schakraborty@towson.edu

Christoph Rosenkranz
Goethe University, Germany
Grüneburgplatz 1, D-60323 Frankfurt
rosenkranz@wiwi.uni-frankfurt.de

Josh Dehlinger
Towson University, USA
8000 York Road, Towson, MD 21252
jdehlinger@towson.edu

Abstract

An important aspect of the requirements engineering process is the specification of traceable, unambiguous and operationalizable non-functional requirements. This remains a non-trivial task due to the lack of well-documented, systematic procedures that facilitate a structured analysis of the qualitative data that is typically the input to this activity. This research investigates the development of a procedural approach that can potentially fill this gap by incorporating procedural perspectives from Grounded Theory Method, Linguistic Analysis and the Non-Functional Requirement Framework, without significantly deviating from existing practice. This paper describes a preliminary version of this procedural approach along with empirical illustrations using data from a redesign initiative of a library website of a public university in the United States. The paper concludes with a preliminary assessment of the approach and a discussion of the contributions of the research.

Keywords: Requirements Engineering, Grounded Theory Method, NFR, Linguistic Analysis

Introduction

An important aspect of requirements engineering (RE) is the analysis of non-functional requirements (NFR). NFRs of an information system describe the software quality attributes (e.g., usability, security, etc.) (Roberston et al. 2006; Sommerville 2011). A significant challenge in software engineering is in addressing NFRs. Specifically, an analyst needs to ensure that NFRs are documented and implemented in a way to make them *traceable* (i.e., the ability to identify the decomposition, analysis, and implementation throughout a systematic process), *unambiguous* (i.e., the ability to express an NFR in a complete, understandable description), and *operationalizable* (i.e., the ability to decompose an expressed “fuzzy” software quality attribute into a concise, quantifiable, implementable requirement) (Sommerville 2011).

These challenges typically stem from the socio-cognitive context of RE (Galliers et al. 2000; Hansen and Lyytinen 2010). RE is the activity concerned with the identification and documentation of the purpose of an information system (IS) along with the context in which it will be used. RE is typically undertaken to develop an understanding of complex, socio-technical systems, and typically uses natural language-based qualitative data to develop semi-formal descriptions of the proposed IS (Sommerville 2011). The qualitative source data poses some fundamental problems. First, natural languages have inherent ambiguities, which can lead to multiple interpretations of the same text excerpts (Hansen and Lyytinen, 2010). Second, stakeholders are frequently unable to provide complete and accurate descriptions of their IS needs (Urquhart 2001). Despite such known problems, there is a lack of systematic, well-documented procedures within RE that enables the bridging between the qualitative data and the final system description. This shortcoming becomes exacerbated when one considers NFRs, as they typically represent intangible concepts open to subjective interpretations.

Given the above challenges, we propose the use of qualitative research methods to augment the quality of NFR analysis. Specifically, this research investigates the development of a procedural approach that can potentially fill this gap by incorporating perspectives from Grounded Theory Method (GTM), Linguistic Analysis, and the NFR Framework. We have followed a Design Science approach (Gregg et al. 2001; Hevner et al. 2004). The purpose of this paper is to describe an initial design and its rationales, a “systematic specification of design knowledge” (Gregor and Jones 2007, p. 314), to develop an approach that assists analysts and developers in eliciting, identifying, analyzing, and specifying NFRs. Our approach aims to meet specific *meta-requirements* (Walls et al. 1992) for the RE context. Specifically, we provide a set of *systematic procedures* that facilitates the determination of NFRs from qualitative text sources. In addition, these NFRs need to fulfill the requirements of being *unambiguous* and *traceable* to the qualitative source; finally, the approach needs to have a *seamless fit* with existing practice (Sommerville 2011). To meet these meta-requirements, we primarily make use of GTM and Linguistic Analysis as our kernel theories to derive the main design rationales underlying our approach (Simon 1996; Walls et al. 1992). Based on this justificatory knowledge, we derive a set of *design requirements*. GTM provides us with a systematic procedure for conducting the qualitative analysis, whereas Linguistic Analysis allows us to assert that the use of our approach will arrive at NFRs with sufficiently high levels of unambiguity and quality. In addition, we adopt: a typology of NFR from the well-known Volere framework (Roberston and Robertson. 2006); and the NFR Framework (Chung et al. 2000). From all four, we derive specific rationales concerning the objective of our approach.

Design Rationale and Frameworks

Grounded Theory Method and Linguistic Analysis

First, we suggest basing the inference steps and procedures of our approach loosely on GTM. GTM is described as “a qualitative research method that uses a systematic set of procedures to develop an inductively derived grounded theory about a phenomenon” (Strauss and Corbin 1990). Data analysis is conducted using three different coding procedures: open coding, axial coding, and selective coding (Strauss and Corbin 1990). *Open coding* involves “breaking down, examining, comparing, conceptualizing, and categorizing data” (Strauss and Corbin 1990). The initial output of GTM is open

codes, which are then aggregated into higher-order concepts called *categories* and their attributes (*properties*). *Axial coding* formalizes a set of procedures that allows for the reformulation of the data obtained from open coding, by developing explicit hierarchical relationships between categories and sub-categories. Axial coding is performed until all categories identified during open coding have been included in some category-sub-category relationship. *Selective coding* constructs relationships among the higher-order categories that were identified during the axial coding phase. This coding procedure selects the core categories and systematically relates them to other categories. Correspondingly, our first design requirement is to follow the steps of GTM in deriving NFRs from qualitative text data (e.g., interviews with stakeholders).

Second, we propose to leverage linguistic theory. Several studies have shown the central role of natural language in RE (e.g., Alvarez and Urla 2002; Cohn et al. 2009; Corvera Charaf et al. 2012; Urquhart, 2001). Thus, the RE process can be characterized as a “language development and formalization process” (Lyytinen 1985), in which the use of *natural and verbal language* poses major challenges (Urquhart 2001). A disadvantage of natural language is its inherent ambiguity, which provides space for different interpretations of the same requirements. This necessitates the emergence of a *semantic alignment* or the development of a shared language by all stakeholders during RE (Garrod and Pickering 2009). The efficiency of a shared language requires that representations (i.e., utterances, words, symbols) are effective, that is, always anchored to the unique and particular situation of their use (Hirschheim et al. 1995). *Language quality* - the degree to which a *symbol* obtains and retains a relationship or mapping to a *concept description* - then relates to how well a representation captures and retains a meaning assigned to it (Corvera Charaf et al. 2010). “Effectiveness” with regard to building shared language in RE then refers to the degree to which all stakeholders manage to achieve high language quality. Building on ontological foundations (Wand and Wang 1996), a minimal and consistent set of four dimensions allows assessing language quality during RE communication (Corvera Charaf et al. 2010): a) *Completeness*: a representing symbol is given for every concept description; b) *Meaningfulness*: every representing symbol is linked with one corresponding concept description; c) *Non-redundancy*: every single concept descriptions are linked to one concrete symbol; d) *Unambiguity*: no two concept descriptions map into the same symbol.

These four dimensions of language quality allow analyzing and assessing the mappings between symbols and concept descriptions. Consequently, our second design requirement is that our methodology needs to support the building of a shared language with high language quality during RE.

Volere Classification of NFRs and the NFR Framework

Volere offers a taxonomy of NFRs to aid an analyst in identifying, analyzing and categorizing NFRs during the documentation process (Roberston et al. 2006). The Volere taxonomy defines the following eight, top-level NFR categories to aid software engineers in eliciting, analyzing and documenting requirements: look and feel requirements; usability and humanity requirements; performance requirements; operational requirements; maintainability and support requirements; security requirements; cultural and political requirements; and legal requirements. This taxonomy is used here to aid in the discovery of the NFRs and provides our third design requirement.

The NFR Framework offers a structured notation for “representing and recording the design and reasoning process” relating to the discovery of NFRs (Chung et al. 2000). NFRs are treated as “soft goals” that need to be clarified, disambiguated, prioritized, and elaborated upon, if the proposed system has to incorporate the appropriate quality attributes. The NFR Framework achieves this by developing a *softgoal interdependency graph* (SIG) representing the main NFRs and then systematically decomposing the softgoals into sub-softgoals. This decomposition is continued until sufficient refinement is achieved to allow the analyst to comprehend potential development techniques that can be adopted to meet the non-functional goals. Graphical notations used to develop a SIG are as follows: ‘clouds’ are used to represent a softgoal and a ‘straight line’ is used to depict the interdependencies between softgoals. If several softgoals are collectively needed to meet a higher softgoal, it is depicted as an AND contribution. Additionally, softgoals can potentially have positive (i.e., a ‘+’ symbol) or negative (i.e., a ‘-’ symbol) contributions to other softgoals. Support of the NFR Framework provides the fourth and last of our design requirements.

The GTM-Linguistic Analysis Based Procedural Approach

The proposed procedural approach for identifying and specifying NFR from qualitative source data is described next. We provide an initial ‘proof-of-concept’ demonstration of the procedure using data available from a real-life RE endeavor. The requirements collection effort was related to the website redesign of the library of a public university in the United States. As a part of this, a number of requirements gathering activities were performed, including surveys (where users were provided closed and open ended questions about their likes and dislikes of the existing site and also to provide recommendations), a focus group and card sorting exercise involving a total of 94 students, 58 faculty and staff participants. There were over a 100 surveys available for analysis along with qualitative text from the focus group discussions. We would like to mention here that the RE activity for the website redesign was independent of this research and none of the authors were involved in it. However the data was made available to the first and third authors and has been used solely to provide empirical illustrations for our proposed procedure.

Boundary Conditions and Assumptions

The underlying premise of the procedural approach is the application of GTM in analyzing qualitative text input to yield NFRs. However, there are key distinctions between the RE context and the traditional context of theory development. First, in theory development the structure of the component and interrelationships are unknown and need to be discovered; the emphasis, therefore, is on a true emergence of theoretical ideas from data. In this particular RE context, the key components (i.e., categories of NFRs) is well established, therefore the emphasis is on the emergence of rich descriptions of the components and their interrelationships. Second, in theory development the conceptual elements vary in terms of salience and pertinence for a given phenomenon as well as the context of its study. In RE, the NFR types remain invariant. There is, however, contextual variation of their implications and salience. Finally, in theory development the emphasis is on complete immersion and not replication. In the RE context, the emphasis is on immersion as well as replication across context.

Therefore, our application of GTM has been adapted to facilitate the emergence of a rich, traceable, unambiguous description of the NFRs. This adaptation takes two key forms. The first concerns our approach to *theoretical sensitivity*. Theoretical sensitivity is considered important because it allows researchers to draw on the body of existing knowledge and is considered crucial for the emergence of contextually important theoretical categories from the raw data (Kelle 2007; Strauss and Corbin 1990). For the context of our application we had to concede a certain amount of forcing of pre-existing theoretical elements. Specifically, during the open coding process, we propose an explicit sensitization to specific NFR types from the Volere typology. This facilitates the development of procedural structure and promotes replication. However, the emergence of the conceptual elements that provided a rich description to these NFR types occurs in the true spirit of the GTM coding procedure. Our second adaptation concerns the GTM procedure. First, we augmented the open coding procedure by incorporating procedural elements of Linguistic Analysis to enhance language quality. This procedural deviation from GTM is motivated by known ambiguity and lack of clarity related to the use of natural language based qualitative text that serves as an input to RE (Hansen and Lyytinen 2010). The second deviation concerns the axial coding phase. During this phase, we have eschewed the use of the paradigm model proposed in (Strauss and Corbin 1990) to develop our relational model of NFRs.

First Phase: Open Coding

The input to the open coding phase is the qualitative text from documents obtained during a typical RE elicitation activity. In our context, this is represented by the qualitative text obtained from the responses of the faculty focus group and also from the open-ended questions within the faculty and student paper survey. An additional input is a definitional list of the NFR types from the Volere process model.

The first step of the open coding phase involves the fracturing of the qualitative text to identify open codes that represent the analytical building blocks. The open codes are any text excerpt that is identified by the

analyst to be descriptive of an NFR. In our procedure, applying Linguistic Analysis facilitates this. The procedural steps for the same build on Corvera Charaf et al. (2010):

1. *Step – Symbol-concept description coding (LC coding)*: This involves the scrutiny of all collected data for single concept descriptions that are relevant for NFRs by looking for utterances where a relationship between symbols and meanings are built. All sequences corresponding to a concept meaning are collected under a same *object of definition* (OD). The coder gives the OD name. Utterances are either classified as SY (symbols) or CO (concept descriptions) depending on the corresponding OD. Figure 1 gives an example.
2. *Step – Language quality coding (LQ coding)*: This involves the classification of the relation between SY and CO and identification of deficiencies in the dimensions of language quality. From a procedural standpoint, language quality codes are assigned to the symbols and concept descriptions identified during LC coding. Specifically, INC (incompleteness) is assigned if for a given CO1, no SY1 could be identified; MLN (meaninglessness) if no CO1 was introduced for SY1; RDC (redundancy) if another symbol (SY2) also refers to CO1; and AMB (ambiguity) if another concept (CO2) also refers to SY1.

Figure 1 shows the output from the above LC and LQ coding steps. The first column shows an exemplar text extract and the second column shows the concept descriptions and symbols identified from the text. The linguistic assessment of the text excerpt indicates a discussion about the customization of the website and its features. This, therefore, represents the OD. The symbols and concept descriptions identified from the text excerpt relate to and describe this OD. Column 2 also presents the language quality codes from the LQ step (represented as colored boxes). An inspection of these codes indicates some of the typical quality-related issues pertaining to utterances by a typical user. For example there emerged the issue of *incompleteness*, as illustrated by the concept description “are useful”, which alludes to some user expectation that it would be useful in terms of “regular usage”, “ease of use”, or “effectiveness” of the website, but the exact nature remains implicit and therefore elusive to the analyst. We propose that the LC-LQ coding steps is particularly useful in explicitly identifying areas in the source text that have lack of clarity and provides them with an opportunity to gain greater clarity at this early stage instead of propagating misspecification of NFRs.

3. *Step – Identifying the open codes*: This step involves the inspection of the conceptual ideas represented within the SY and CO that emerged from the previous step and developing contextually meaningful labels from these. These labels represent the open codes. It should be noted that these open codes can emerge from and be identical to the SY and OD identified in this phase as well as from ideas embedded within the concept descriptions. In addition, multiple open codes could emerge from the artifacts of the LC-LQ steps.

Some exemplar open codes that emerged from the SY, CO, and OD identified from the LC-LQ phase are presented in the third column of Figure 1.

4. *Step – Aggregating Open Codes to categories*: the input for this list is the comprehensive list of open codes that emerge from the previous step. Inspecting and ascribing higher order meaning to the text labels of the open codes facilitate this process. The identification of categories facilitates a coherent classification of the open codes and the emergence of a structure. For example, an inspection of the open codes that emerged in Figure 1 leads to the identification of four higher-level categories – *Useful Features*, *Website Objects*, *Usability Enablers*, and *Stakeholders*. An additional objective of this process is also to develop a rich description of the categories by identifying the properties that help comprehend the exact nature and implications of a category. Typically, the set of open codes that allude to the categories can be identified to form the property set of the particular category. For example, the open codes, tables, resources, and features are members of the property set of the category Website Objects (Figure 2). The process of identification of categories and their properties proceeds until the analyst is convinced that a level of saturation is reached in terms of the categories identified from the set of existing open codes. An aspect of this process is to identify this hierarchy through iterative inspection until a classification schema is reached that identifies all available open codes as categories, sub-categories or properties. Figure 2 presents the classification schema that

| Text Excerpt | LC/LQ Coding Example | Open Codes |
|---|--|--|
| <p>"Ideally the web site would be customizable by individual or department, with a "my favorites" feature that would allow the user to choose which tables / features / resources are useful, and those would be prominently displayed. What is important to one department / individual may not be to another."</p> | <p>Object of Definition (OD): „Website Customization and Features“</p> | <p>Customization My favorite feature Subjective feature importance Prominent display Tables Resources Individuals Departments User</p> |
| <p>Legend:</p> <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px; font-size: 8px;">Concept Description</div> <div style="border: 1px solid black; border-radius: 10px; padding: 2px; font-size: 8px;">Symbol</div> <div style="background-color: #c8e6c9; padding: 2px; font-size: 8px;">Language Quality Code</div> </div> | | |

Figure 1. LC-LQ/Open Code Table

Second Phase: Axial Coding

The input to the axial coding stage is the category cluster diagram (see Figure 2). This alludes to a certain hierarchy within the categories and sub-categories. The objective of axial coding is to refine the interdependencies into a formal representation of the relationships between the different categories, subcategories and their properties. To facilitate this we adopt the NFR Framework, a theoretical lens that has the context of NFR explicitly embedded within it. Specifically, the procedural coding within this phase is adapted to represent the interdependencies of the categories and the sub categories as soft goal interdependency graphs (SIGs).

This process takes a top down approach using as its starting point the highest-level categories identified during the open coding phase (e.g., usability, security etc.). In NFR Framework parlance, these represent the highest-level softgoal. A process of inspection and comparison of the conceptual implications of the various sub-categories and their properties may systematically develop the SIG for the particular NFR type. This process of refinement needs to continue until each branch of the SIG terminates in a leaf that can be construed as an explicit operationalization. The left side of Figure 3 shows the SIG that was constructed for the 'usability' NFR type.

While the SIG may be demonstrably derived from the nested category clusters, the process should not be considered mechanistic or trivial. The process of identifying the hierarchies of the sub-softgoals requires a certain amount of interpretation and sensitization of the context. For example, it may not always follow that sub-categories identified as in the same hierarchy in nested category cluster would map to soft goals in the same level in the SIG. In our example *Usability Enablers* and *User Friendly* emerged as sub categories in the same hierarchy with the usability category, but were placed at softgoals at different levels

within the SIG. In extension, the same can apply to properties of a particular sub-category (e.g., *Customization* and *Easier to Manage*) and their hierarchical position within the SIG. This aptly demonstrates the procedural strength of the axial coding in developing a clearer conceptualization about the NFRs and their interdependencies. Our intention was not to design a process that reduces the importance of the individual's analysis; therefore it is appropriate that the process of identifying the exact hierarchy of the sub-categories would require thoughtful interpretation of the context by the analyst.

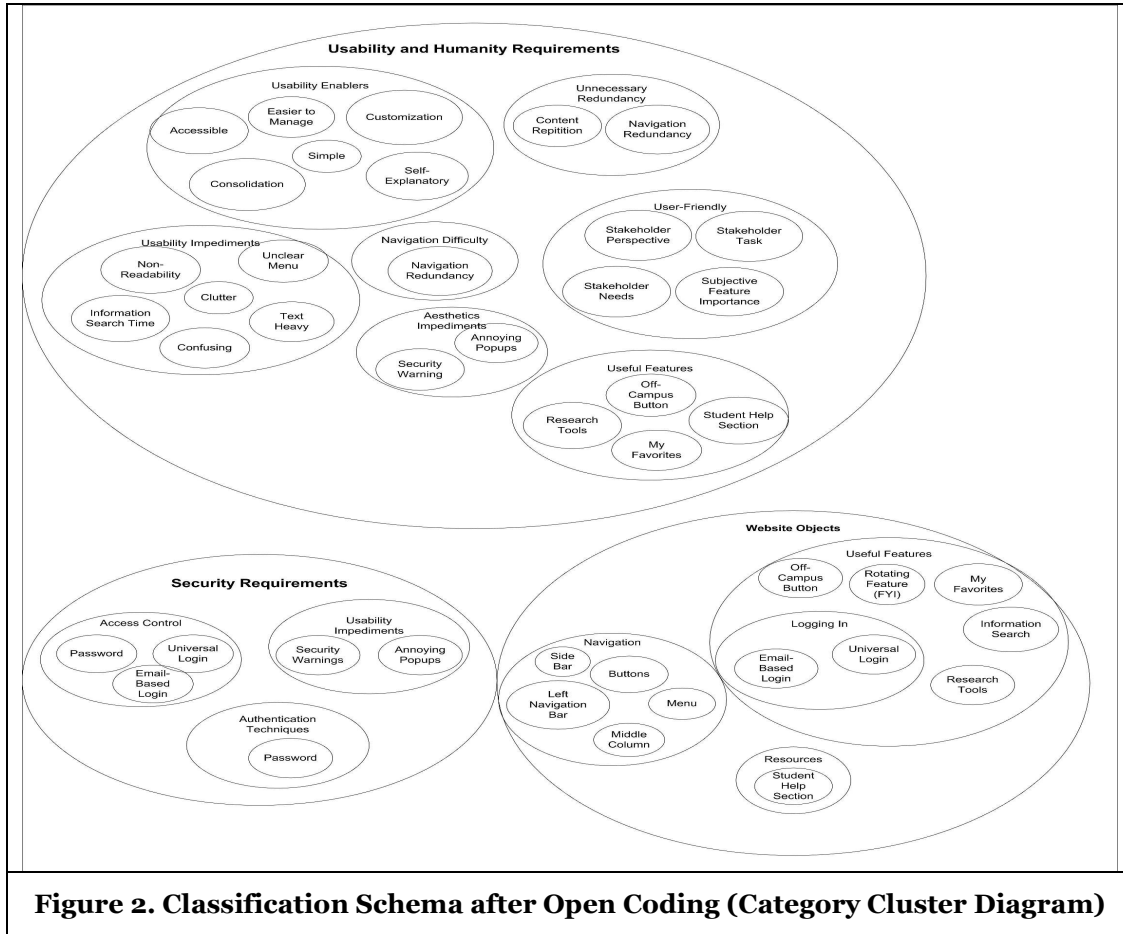
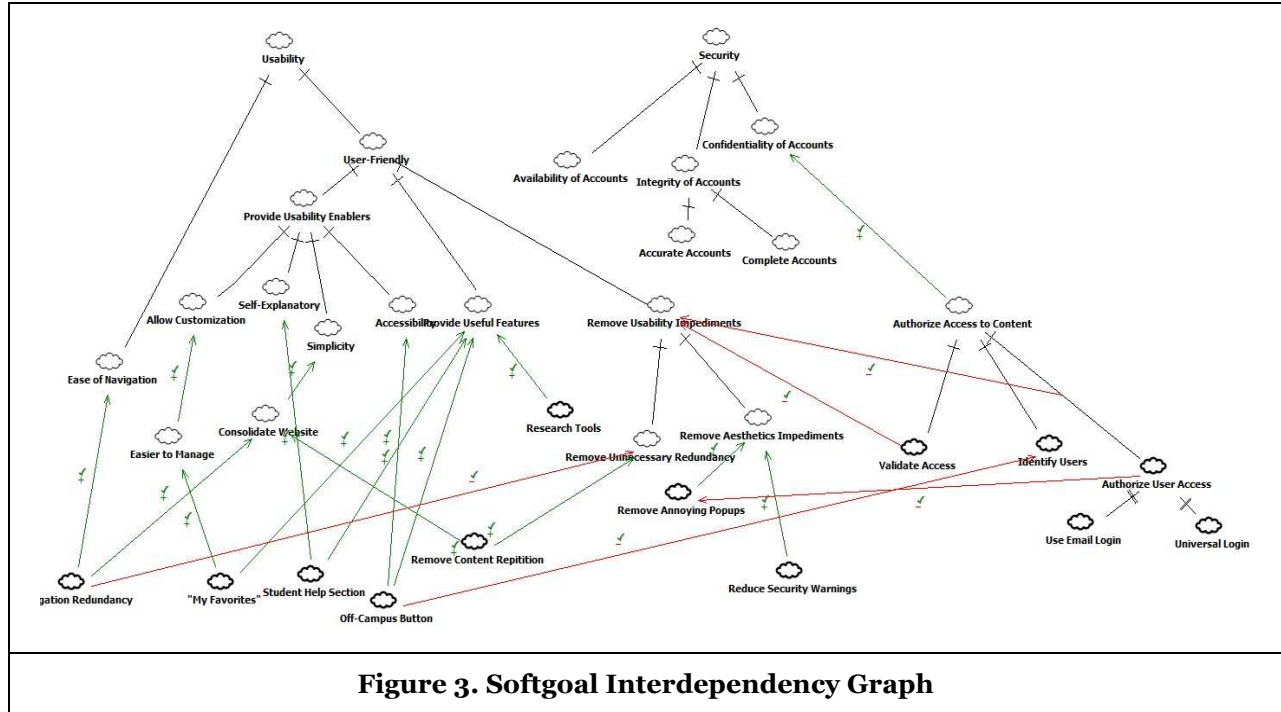


Figure 2. Classification Schema after Open Coding (Category Cluster Diagram)

In certain cases, in order to retain the spirit of the NFR Framework, it may be necessary to reframe the labels of certain sub-categories (and properties) to make them more appropriate representation of goals. In our example such reframing was necessary with certain categories. Moreover, not all properties of a sub-category may find its place in the SIG. This will occur in situations where such properties do not represent goals related to a NFR. The leaves of the SIG represent operationalizations of their parent softgoal and are deemed to contribute to achievement of such a goal. However, we would like to caution that the identification of such leaves for a branch might not necessarily indicate a complete SIG. Once such leaves are identified, the analyst should examine other soft goals (from different branches) to see if such leaves also contribute to other soft goals. Not performing such due diligence could not only result in incomplete development of the tree but the also the missing of possible conflicts. A case in point is the leaf *navigational redundancy* in Figure 3. While initially it was found to positively contribute to ease of navigation, a conflicting negative contribution was also identified for *remove unnecessary redundancy*. This represents an important dialectic in the specifications that would need to be resolved by the analyst if the overall quality of a system is to be ensured.



Third Phase: Selective Coding

The objective of the selective coding phase is to develop a more complete description of the NFRs specifications by identifying the interdependencies (conflicts and synergies) between the NFR types. The NFR Framework continues to guide this phase and the targeted NFR Framework artifact from this phase is the *full softgoal interdependency graph* that describes the complex relationships among non-functional requirements and select operationalizations. Procedurally, this is enabled by identifying the core NFR for the particular context and then systematically identifying the interdependencies. This is achieved by examining the sub-soft goals and operationalizations of each SIG and identifying possible positive and negative contributions. The excerpt of the full interdependency tree showing the dependencies between usability and security are shown in Figure 3.

Discussion and Outlook

Our procedural approach meets the meta-requirement of *systematic procedure* by prescribing an ordered analysis of qualitative data using the coding procedures of GTM. This is comprised of coding steps with well-defined sets of input, precisely described procedures and output. The coding steps are sequential in nature and utilize output from the previous step as a starting point. In addition, each coding step is self-contained with distinct objectives, making the methodology inherently modular, and therefore conducive to be carried out by multiple personnel sequentially. The methodology also allows explicit *traceability* through the development of intermediate artifacts at each step and also within steps. Therefore the elements of each phase can be uniquely traced within, as well as across the sequential coding steps. For example, a particular soft goal identified in the *full softgoal interdependency graphs* in the selective coding phase can be easily traced to the source qualitative text via the SIG for the NFR type, the Category Cluster Diagrams, and the LC/LQ Open Code Tables. An important objective of the methodology was to enable *unambiguous* understanding of a particular NFR. This is achieved in the methodology in two ways. First, this is realized through the use of the Linguistic Analysis steps during the open coding phase. These steps represent an explicit check to reduce ambiguities stemming from the use of natural languages as well as enabling the analysts to assess the quality of the qualitative data, identify nature and types of informational deficiencies, and subsequently seek clarification from the original source of the data. Second, a careful adherence to traceability in the design of the methodology provides the analyst with the

opportunity to trace any output element to a qualitative data source to better understand the context of its definition. Finally, we enable a *seamless fit* with existing practice in two ways. First, we have adapted the open coding phase of GTM to explicitly use the Volere NFR typology thereby constraining the methodology within the definitional boundaries of known NFR types. Second, we have adapted the axial and selective coding phases to develop design artifacts that conform to notational representation of the NFR Framework.

As a preparation for evaluation, Gregor and Jones (2007) argue for testable hypotheses about the artifact to be constructed. As to the dependent variables, empirical studies on RE suggest measuring the cost effectiveness of the RE process as an indicator for successful RE (El Eman and Madhavji 1995; El Eman and Madhavij 1996). Additionally, it has been asserted that in order to assess performance, one should also measure whether a treatment leads to improved quality of the requirements generated. Consequently, we expect that using our methodology, which supports both a GTM-based procedure for analyzing NFR and the building of a shared language, improving the language quality of the output, will result in higher RE performance in terms of cost effectiveness and quality of the requirements than using methodologies that do not support this. That is, the independent variable is the methodology support of the RE process. Correspondingly, we hypothesize that:

H1a: *The use of a procedural approach that supports both GTM-based procedures and steps to improve language quality to determine NFRs will result in a less costly RE process than the use of approaches that do not support this.*

H1b: *The use of a procedural approach that supports GTM-based procedures and steps to improve language quality will result in higher quality and purposeful (i.e., meaningful) NFRs being produced for a given RE task than the use of approaches that do not support this.*

H2: *The use of procedural approach that support GTM-based procedures in conjunction with the Volere taxonomy will result in a more traceable process for a given RE task than the use of procedures that do not support this.*

H3: *The use of a procedural approach that supports Linguistic Analysis-based procedures in conjunction with the NFR framework will result in a more unambiguous identification of NFRs for a given RE task than the use of approaches that do not support this.*

H4: *The use of a procedural approach that supports GTM-based and Linguistic Analysis procedures in conjunction with the NFR framework will result in more operationalizable definitions of NFR for a given RE task than the use of approaches that do not support this.*

The next step in this research is to empirically evaluate the above assertions. We propose to do this by conducting controlled laboratory experiments involving graduate students undertaking a course in Requirements Engineering in an American University, a large proportion who are working professionals in the IT field. The experiments will be designed to compare the performance of individuals in an RE task to identify NFRs using conventional RE approaches and the procedural approach described in this paper.

While this research is at its preliminary stage, it makes some important contributions to the field. First, by proposing the use of qualitative analytical techniques as a procedural bridge this work provides a potential resolution of some well-documented challenges in the specification of NFRs (e.g., traceability, ambiguity, etc.). Second, by formulating a methodology that uses analytic procedures of GTM and Linguistic Analysis and representational notation of the NFR Framework, this work explicitly demonstrate how qualitative analytical techniques can be incorporated into existing NFR specification processes. Third, through identification of, and adherence to, meta-requirements derived from the literature, this work has developed a methodology that has the potential of improving the quality of NFR specifications and the quality of information system design. Finally, the procedural approach is inherently self-contained and we suggest that it can be easily adapted to other RE techniques and methodological approaches with minimal adaptation and disruption. Additionally, the stages of the procedure, while sequential, are also sufficiently self-contained to allow them to be carried out by different people. Consequently, we believe the procedure can be adopted without significant additional costs to existing RE processes within an organization.

References

- Alvarez, R., and Urla, J. 2002. "Tell Me a Good Story: Using Narrative Analysis to Examine Information Requirements Interviews During an Erp Implementation," *The DATA BASE for Advances in Information Systems* (33), pp. 38-52.
- Chung, L., Nixon, B.A., and Mylopoulos, J. 2000. *Non-Functional Requirements in Software Engineering* Kluwer Academic Publishers.
- Cohn, M.L., Sim, S.E., and Lee, C.P. 2009. "What Counts as Software Process? Negotiating the Boundary of Software Work through Artifacts and Conversation," *Computer Supported Cooperative Work* (18), pp. 401-443.
- Corvera Charaf, M., Rosenkranz, C., and Holten, R. 2012. "The Emergence of Shared Language: Applying Functional Pragmatics to Study the Requirements Development Process," *Information Systems Journal* (Special Issue on New trends in Information System Development).
- Corvera Charaf, M., Rosenkranz, C., and Holten, R. 2010. "Assessing Language Quality in the Information Systems Development Process: A Theoretical Approach and Its Application," *31st International Conference on Information Systems (ICIS 2010)*, St. Louis, Missouri, USA.
- Emam, K.E., and Madhavji, N.H. 1995. "Measuring the Success of Requirements Engineering Processes," *Proceedings of the 2nd IEEE International Symposium on Requirements Engineering: IEEE Computer Society*, p. 204.
- Emam, K.E., and Madhavji, N.H. 1996. "An Instrument for Measuring the Success of the Requirements Engineering Process in Information Systems Development," *Empirical Software Engineering* (1:3), pp. 201-240.
- Galliers, R.D., and Swan, J.A. 2000. "There's More to Information Systems Development Than Structured Approaches: Information Requirements Analysis as a Socially Mediated Process," *Requirements Engineering* (5:2), pp. 74-82.
- Garrod, S., and Pickering, M.J. 2009. "Joint Action, Interactive Alignment, and Dialog," *Topics in Cognitive Science* (1:2), pp. 292-304.
- Gregg, D.G., Kulkarni, U.R., and Vinzé, A.S. 2001. "Understanding the Philosophical Underpinnings of Software Engineering Research in Information Systems," *Information Systems Frontiers* (3:2), pp. 169-183.
- Gregor, S., and Jones, D. 2007. "The Anatomy of a Design Theory," *Journal of the Association for Information Systems* (8:5), pp. 313-335.
- Hansen, S., and Lyytinen, K. 2010. "Challenges in Contemporary Requirements Practice," *Proceedings of the [43th] Annual Hawaii International Conference on System Sciences* Computer Society Press (CD-ROM), p. (12 pages).
- Hevner, A., March, S., Park, J., and Ram, S. 2004. "Design Science in Information Systems Research," *MIS Quarterly* (28:1), pp. 75-105.
- Hirschheim, R., Klein, H., and Lyytinen, K. 1995. *Information Systems Development and Data Modeling. Conceptual and Philosophical Foundations*. Cambridge, UK et al.: Cambridge University Press.
- Kelle, U. 2007. "The Development of Categories: Different Approaches in Grounded Theory," in *The Sage Handbook of Grounded Theory*, A. Bryant and K. Charmaz (eds.). London: SAGE, pp. 191 - 213.
- Lyytinen, K.J. 1985. "Implications of Theories of Language for Information Systems," *MIS Quarterly* (9:1), pp. 61-76.
- Robertson, S., and Robertson, J. 2006. *Mastering the Requirements Process*. Upper Saddle, NJ: Addison-Wesley.
- Simon, H.A. 1996. *The Sciences of the Artificial*, (3 ed.). Cambridge, MA, USA et al.: MIT Press.
- Sommerville, I. 2011. *Software Engineering*, (9 ed.). Boston: Addition Wesley.
- Strauss, A.L., and Corbin, J.M. 1990. *Basics of Qualitative Research* Newbury Park, CA: Sage.
- Urquhart, C. 2001. "Analysts and Clients in Organisational Contexts: A Conversational Perspective," *The Journal of Strategic Information Systems* (10:3), pp. 243-262.
- Walls, J., Widmeyer, G., and El Sawy, O. 1992. "Building an Information System Design Theory for Vigilant Eis," *Information Systems Research* (3:1), pp. 36-59.
- Wand, Y., and Wang, R. 1996. "Anchoring Data Quality Dimensions in Ontological Foundations," *Communications of the ACM* (39:11), pp. 86-95.