The Qualitative Systematic Approach to Requirement Analysis (QSARA)

Ban Al-Ani,

Dept. of Software Engineering, Faculty of IT, University of Technology Sydney

alani@it.uts.edu.au

Abstract

Stakeholders' understanding of what is expected of a system evolves with the continuous review and revision of the requirements document. Problems arise when no record is kept of their understanding over long periods of time and when an attempt is made to share that understanding. This paper presents the steps of a Qualitative Systematic Approach to Requirements Analysis (QSARA). The approach has been designed to assist in the analysis of unstructured requirements documents expressed in a combination of natural language text, tables and diagrams. Applying the proposed approach will produce a graphical representation of system features. QSARA's contributions include assisting stakeholders in capturing, modifying and sharing their understanding of documented requirements. It also facilitates the detection of errors and inter-dependency of features. The paper includes an example from a pilot study of the proposed approach that was conducted to validate its effectiveness.

Keywords: *requirements, requirements analysis, understandings, qualitative research, grounded theory, documentation of understanding, communicating understanding.*

1. Introduction

Documented system requirements do not arise naturally. They have to be engineered and are subject to continuous review and revision as they evolve and stakeholders gain a deeper understanding of what is expected of the system [16]. Requirements can be expressed through informal (e.g. natural language), semi-formal (e.g. data flow diagrams), and/or formal (e.g. Z language) notations.

Despite the problems attributed to the use of informal notation, its use is still common practice in view of the need for effective communication between the customers and developers [17]. The primary cause of problems arising during requirement development is often attributed to the lack of requirements understanding and the ability to communicate that understanding [11].

Algorithmic-like methods for understanding requirements at later stages of development have been proposed (e.g. 11) in addition to methods that convert requirements expressed in natural language notation to another. Such methods provide a means to analyse description viewpoints or re-use of requirements rather than provide a means systematically understand the early drafts of documented requirements [10, 4].

This paper reports on ongoing research of a Qualitative Systematic Approach to Requirement Analysis (QSARA) that assists stakeholders during the early stages of requirements development. The approach is based on qualitative grounded theory principles as defined by [14]. It adopts a hermeneutic mode of analysis, which is primarily concerned with the meaning of textual and non-textual information [2].

The QSARA is designed to assist analysts to gradually develop their understanding of an unstructured requirements document by identifying core system features and collecting requirements describing these feature. It can assist analysts to capture, modify and share their understanding of documented requirements. QSARA can also be used to facilitate the detection of errors and the inter-dependency of documented features. The outcome of the approach is a graphical representation of core feature characteristics, which can be understood by stakeholders of differing skills and backgrounds. The graphical representations can be used to index, structure and refine the early drafts of documented requirements.

The paper is made up of four main sections. The first describes the main principles of qualitative research relevant to the proposed approach. The second section defines QSARA steps. The steps are used to understand textual, diagrammatic and tabular requirements describing expected system features. Three of the features extracted from the pilot study will be constructed to illustrate the usage of QSARA. The third section presents an appraisal of the approach, which includes a review of the goals and benefits it achieves. The paper concludes with an outline of the lessons learnt during the pilot study and suggestions for ways in which this approach can be refined further.

2. Qualitative Research

Qualitative research typically includes information that is not arrived at by means of statistical procedures or other means of quantification. The credibility of this type of research relies heavily on the confidence in the analysts to be sensitive to the data and to make appropriate decisions in the field [14, 9].

Qualitative research has an emergent (as apposed to predetermined) design, and analysts focus on this emerging process as well as the outcome or product of the analysis. There are several types of qualitative research (e.g. grounded theory and conversational analysis), which have been used to analyse data gathered during research into education [9], ethnography [13] in addition to other sociological research [3, 1]. It has also been adopted to assess the quality of requirements documented in natural language [17].

The Qualitative Systematic Approach to Requirement Analysis (QSARA) is based on the grounded theory type of qualitative research that uses a systematic set of procedures to develop and inductively derive theory about phenomenon grounded in data.

Grounded theory is used in this research to analyse documented requirements because "it can be used to uncover and understand any phenomena about which little is yet known" [14]. This is an accurate description of the situation analysts find themselves in, in the early stage of developing a requirements document.

Traditionally this type of qualitative research is based on conducting the following [14]:

- Open Coding: the process of breaking down, examining, comparing, conceptualising and categorising data.
- Axial Coding: a set of procedures whereby data are put back together in new ways after open coding, by making connections between categories.
- Selective Coding: the process of selecting the core categories, systematically relating it to other categories, validating those relationships, and filling in categories that need further development.

While there is diversity in the practices of qualitative research, they are generally based on these three coding processes that will be explored further within the context of the main QSARA steps.

3. Definition of Proposed Qualitative Systematic Approach to Requirement Analysis (QSARA) Steps

The proposed QSARA can be applied to an unstructured draft of the requirements document compiled from the elicited user needs (gathered though interviews, questionnaires, etc). It was used during an experimental pilot study to understand a requirements document made up of fifty-two pages of textual and non-textual requirements. The document describes upgrades to an existing public transport system responsible for scheduling and signalling of public transport vehicles. Additional details of this document cannot be provided because of a confidentiality agreement.

Several software tools were used to support the approach during the experimental pilot study. QuestMap™ was the primary software tool used to document the graphical representation of constructed features. The tool is based on the Information Based Interchange System (IBIS) developed by [12]. Its main components are nodes and hyperlinks that can be generated between nodes. QuestMap™ can generate different types of nodes represented on screen by icons (each consisting of a *label* field and a *details* field) and different types of links are created when nodes are linked [5, 6]. Other secondary tools include Extractor™ [15], which is used to summarise text and generate a list of key words and key phrases. Tools developed specifically to support qualitative analysis do exist (i.e. Nudist and Atlas/ti are considered the main qualitative data analysis tools -3). However, the purpose of this research is not to develop an approach to suit a tool or based on a tool, but rather to develop a method and find a suitable tool to support it. It is essential that the QSARA not be constrained by existing tools limitations during its development. As a result the approach utilised some of the features available in existing generic software and adapted them to suit its purpose during the experimental study.

The main steps of this approach are listed below.

1. Conceptualising requirements (based on open coding).
2. Categorising concepts (based on axial coding and open coding).
3. Consolidating categories (based on selective coding).

An overview of these steps and their interaction is presented in figure 1.

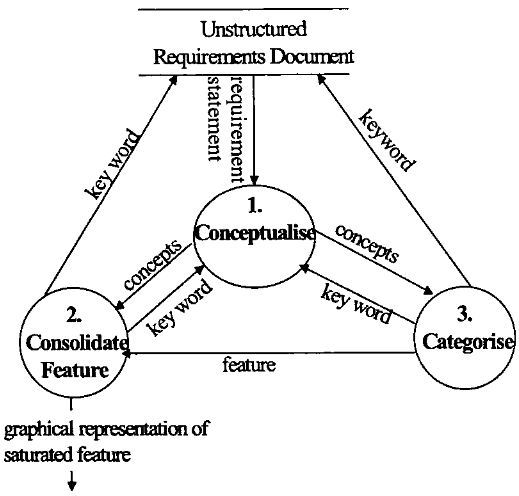


Figure 1. Data Flow Diagram of the QSARA demonstrating the main steps as processes and the data needed and generated by each step.

A detailed definition of the QSARA steps are described in the following sections together with examples from the pilot study that were documented using QuestMap™ [5, 6], Microsoft Word™ and Extractor™ [15].

3.1. Conceptualising

Strauss and Corbin [14] define conceptualisation, as being part of grounded theory's *open coding* process. The requirements document is conceptualised by decomposing requirements and highlighting key *concepts* (fragment of requirements). The purpose of this step is to identify which of the highlighted concept stands for a core system feature. The term *feature* is used to refer to "a bundle of behaviours that serves some useful and coherent purpose for a customer" [18].

Analysts can rely on stakeholders' theoretical sensitivity to identify key concepts. *Theoretical sensitivity* is the ability to recognise what is important in data and to give it meaning [8]. The analyst can also use software like Extractor™ to identify key words or phrases [15].

A sample of *concepts* that were extracted from different requirement statements in the pilot study document is listed below:

Off-load mode of operation, Major fault detected, On-load operation, System response to operation panel, Timetable load computer, ...etc.

When one such concept is identified the analyst can move on to the next step of QSARA, which is categorising.

3.2. Categorising

At this stage a core system feature is only represented by a concept. The purpose of this step is to develop a concept into a system feature by gathering requirement statements that refer to that concept. These requirement statements are the feature's properties or behaviours and can be scattered throughout the document.

While it is beneficial to identify all system features described in the requirement document, it can be infeasible to do so because of the amount of time required. The outcome can also be an unmanageable list of overlapping features. Consequently, it is more practical to limit early categorisation to core or high priority features identified by stakeholders, software tool and/or a methodology.

Executing the following can develop a feature's properties further:

1. Identify a requirements statement that describes a specific system feature. The analyst initiates a search to find explicit occurrences of the statement's key words/phrases. Searches are initiated based on the premise that requirement statements used to represent the same system feature can contain the same key phrases(s) or words. These key phrases can be used to search for other requirements that represent the same description. The analyst can read through the document and/or use an automated search service to conduct the search.

   In this example, the keywords were *off-load, load, mode* and *operation*. These words were used to search the document for related requirement statements using Microsoft Word™ search option. Several requirements were extracted as a result of this search, namely:

   Each computer shall receive and process all inputs. One pair shall operate in hot stand-by "off-load" mode while the other is on line in the "on-load" mode.

If the system is the "off-load" computer the system log...etc.

If the system is the "off-load" computer is not running...etc.

If the system is the "off-load" computer is running...etc.

As a result of these findings searches were also conducted for occurrences of the words *hot, stand-by* and *on-load.*

2. Extract the requirement statement from the document by electronically copying it from a soft-copy of the document and electronically pasting into a new document.

The requirements were electronically cut from the Microsoft Word™ document during the pilot study. A QSARA *concept* node was created (QuestMap™ *idea* node – [5]) and the *concept* placed in the *label* field of that node. The requirement statement was placed in the *detail* field of the same node. This node is represented on screen by a bulb icon. An example of the list of QSARA *concept* nodes and the contents of one is shown in figure 2.

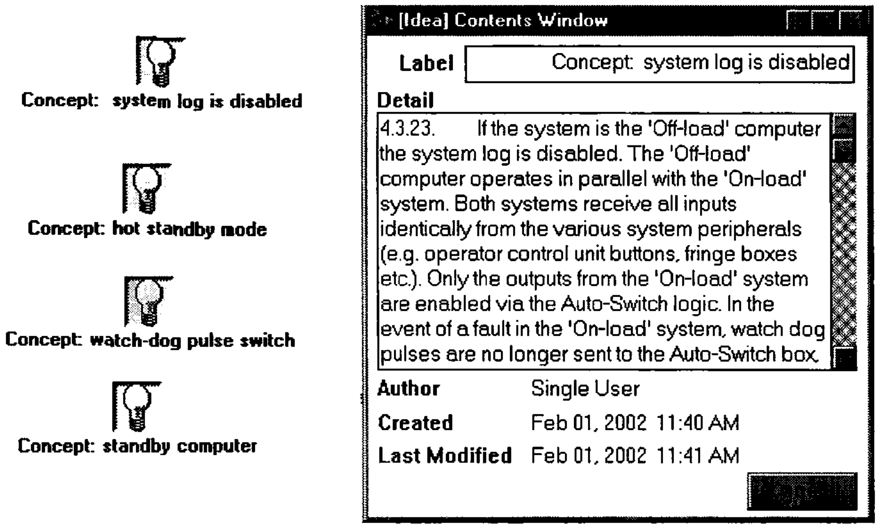


Figure 2. The *concept* nodes created during the *categorisation* step of QSARA's pilot study and the contents of one of those nodes. The bulb icons are displayed with the contents of the *label* field within the QuestMap™ environment. The label was chosen to represent core system features. The *details* field contains the requirement statement providing concept context.

The extracted statement led to further searches for requirements referring to *on-load mode,* which was identified as a potential key concept. The products of the searches were another set of *concept* nodes, which contain requirement statements describing the on-load system feature.

3. Link the extracted requirement to the corresponding concept by using QuestMap™ feature that allows the creation of links between *concept* icons. Gathering or *categorising* the concepts is described as *axial coding* in grounded theory [14].

The example used thus far produced two categories, namely: *off-load* and *on-load* mode of operation. This led to the conclusion that these were in fact *sub-categories* of *load mode operations*. Consequently, they were documented as subcategories as demonstrated in figure 3.

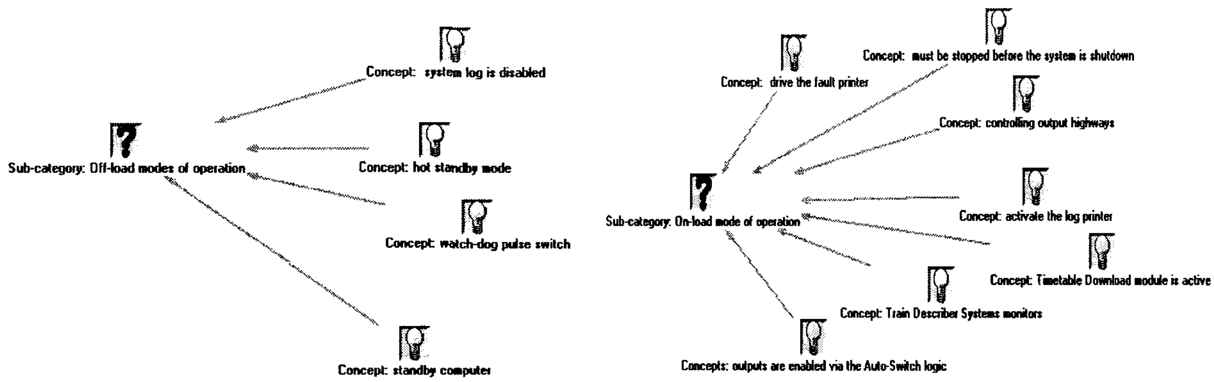


Figure 3. An example of features constructed by categorising the *concept* nodes.

4. The analysts can also make assumptions at this stage of requirement development. Assumptions should be documented clearly as an assumption and not a requirement. These were placed in a QuestMap™ *argument* node during the pilot study. The assumption is linked to the descriptions being constructed. No assumptions were made with regards to the *on-load* and *off-load* modes of operations features

Assuming that these are the only documented statements that describe these features, it is possible to proceed to consolidate categories.

3.3. Consolidating Categories

The categories identified in the previous stage should be re-examined to determine how they are linked, which is part of the *axial coding* process. The categories are compared and combined in new ways as the analyst begins to assemble the big picture. The purpose is not only to describe but also to *consolidate* categories by acquiring new and relatively accurate understanding of high priority system features of interest [9]. This consolidation is defined within the *selective coding* process of grounded theory. Categories can be consolidated by conducted the following:

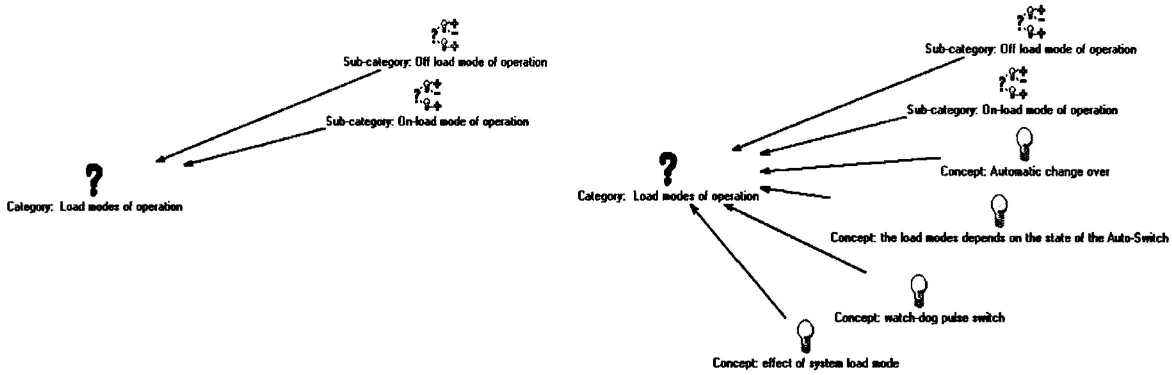
1. Question how the subcategories relate to one another and whether they are actually a part of another feature. System features that appear to be inter-related are grouped into the same category. Descriptions emerge from requirement statements that are pieced together to form a comprehensive picture of their purpose. The coherence of the description rests with the analysts who must rigorously study how different system features fit together in a meaningful way when linked [1]. *Consolidating* should be avoided early in QSARA process, because analysts are still in the early stages of understanding.

2. Search for explicit requirements that refute or support any assumptions made in the previous stage. If explicit requirement information is found to support the assumption within the requirements document, then a link is created between the requirement statements. A link is also created if a conclusion is reached that invalidates an assumption.

   If no evidence is found to support the assumption then the analyst can attempt to find other sources that will support or refute assumptions (e.g. field note or other stakeholders) and place them in a QuestMap™ *support* or *refute* node respectively. The *support/refute* node is then linked to the assumption.

The pilot study revealed that the system described in the requirements document had a *loads mode operation* feature. This feature consisted of two modes of operation thus far, namely: *off-load* and *on-load* load modes. These *subcategories* were discovered in the previous stage of QSARA and placed in a *map node* of *loads mode operation* feature [5].

Further analysis of identifiable concepts revealed that there exist other requirement statements, which describe different aspects of *load mode operations.* These requirement statements were extracted as described in the *conceptualisation* step of QSARA and placed in *concept* nodes as described in the *categorisation* step. They were linked to the category to further *consolidate* the description of this system feature, as illustrated in figure 4.



a. The load modes feature

b. The consolidated load modes feature.

Figure 4. An example of a constructed a feature (figure 3.a), which is later *consolidated* a feature (figure 3.b).

Once descriptions are consolidated it becomes feasible to *test* the analyst's understanding by presenting the graphical representations during brainstorming sessions or walkthroughs attended by stakeholders concerned with system development. Ideally the constructed understanding will closely approximate stakeholders' requirements.

The iteration between the QSARA stages is terminated when all core categories are consolidated and no additional requirements can be found within the document at this stage of requirement evolution. This is referred to as *theoretical saturation* [7].

4. An Appraisal of the Proposed QSARA

This paper presented a Qualitative Systematic Approach to Requirement Analysis (QSARA) that can be used to achieve the following goals:

1. **Assisting stakeholders in understanding documented requirements.**

   This is achieved by adopting the guidelines to construct descriptions of core system features. The analyst can use the approach to construct a description by linking the requirements that are part of the same description.

   It can also be used to document assumptions that can be used to expand on existing requirement statements once approved by key stakeholders.

2. **Achieving analysis results that can be more readily stored, retrieved, modified and expanded at later stages of requirement evolution.**

   Using automated support to document analysis finding can further simplify documentation of analysts' understanding. The graphically constructed descriptions can be modified and shared amongst stakeholders. The informal notation used means that all stakeholders can comprehend the analysts understanding of the proposed system features.

3. **Facilitating the detection of errors within the requirements document.**

   Errors can be detected when assumptions are documented and a search reveals that no explicit information is captured. A search could also result in contradicting requirement statements being found within the document. A comparison of understanding can also reveal ambiguity in documented requirements.

   It can also be used to demonstrate the *inter-dependency of features*. This can assist the analyst determine which requirements and features will be affected by changes to the requirements document.

The proposed approach has several benefits, namely:

1. Its informality means that its application can be adapted to suit individual needs. For example, the QSARA does not enforce restrictions with regards to a requirement statement's format or a concept's granularity.
2. The requirement document (captured in natural language) need not be translated to another notation to apply the approach. This is a significant advantage because at this early stage of development, requirements are rarely in a formal format. In addition, existing studies found natural language to be a popular notation used to document requirements.

Requirement information can be represented textual and non-textual notations. This is a significant advantage, as requirements do not necessarily consist of textual information alone, but could consist of diagrams, audio recording and/or visual recording which can also be analysed using the proposed QSARA.

5. Concluding Remarks and Future Work

Two important lessons were learnt when applying the approach during the pilot study, namely:

1. There must be a constant interplay between proposing concepts, categories and subcategories to achieve a higher category *concept* density. The analyst should not attempt to adopt on a waterfall or spiral model for this approach, as there is no set sequence of steps. The boundaries of the proposed QSARA steps overlapped in actual practice. It is only for explanatory purposes that QSARA was presented as a sequence of distinct steps.
2. There exists a need to validate analysts understanding against reality. It is only by comparing features that variations in understanding can be identified. These variations can lead to the detection of ambiguous requirement statements in addition to the detection of incompletely or inconsistently documented features.

While a substantial amount of work has been done to develop the approach, it possible to refine it further by:

1. Conducting an empirical study of QSARA application. The empirical study will involve two groups, which consist of several participants. Group A will analyse a case study document using analytical methods chosen by the individuals. Group B will analyse the same case study document using the QSARA. The purpose of this study will be to observe the time needed to adopt and apply the approach by individuals unfamiliar with QSARA. It will also provide a means to study any difficulties that individuals encounter when learning and/or applying the method. It would then become possible to improve the method based on the findings.
2. Explore the possibility of developing a requirements paradigm as suggested by [14]. No such paradigm was adopted at this stage of QSARA, to allow flexibility in development. However, a paradigm can improve a category's concept density and assist in the detection of errors by prompting analysts to look for excluded contradictory concepts, for example. A paradigm can also assist in linking sub-categories by prompting the analyst to think of the kind of relationship that exists between them (e.g. *causal*) and not just document that a relationship exists.

6. References

[1] Aronson, J., (1994). "A Pragmatic View of Thematic Analysis", The Qualitative Report, V. 2, N. 1, Spring http://www.nova.edu/ssss/QR/BackIssues/QR2/aronson.html.

[2] Avison, D., Myer, M., (1997), "Research in Practice: Qualitative Research in IS".

[3] Barry, C.A., 1998. "Choosing Qualitative Analysis Software: Atlas/ti and Nudist Compared", Sociological Research Online, V. 3, N. 3. http://www.socresonline.org.uk/socresonline/3/3/4.html.

[4] Cybulski, J., Reed, C., (1998). "Computer Assisted Analysis and Refinement of Informal Software Requirements Documents", Proceedings of 1998 Asian Pacific Conference on Software Engineering, pp. 128-135.

[5] GDSS, (1999a). "Organisational Memory and QuestMap™", http://www.gdss.com/OM.htm.

[6] GDSS, (1999b). "The IBIS Manual: A Short Course in IBIS Methodology", http://www.gdss.com/IBIS.htm.

[7] Glaser, B., Strauss, A., (1967). The Discovery of Grounded Theory, NY: Aldine.

[8] Glaser, B., (1978). Theoretical Sensitivity. Mill Valley, CA: Sociology Press.

[9] Hoepfl, M, (1997). "Choosing Qualitative Research: A Primer for Technology Education Researchers", Journal of Technology Education, V.9, N. 1.

[10] Kobayashi, Y., Enoki, H., Ohta, T., (1995). "Understanding Natural Language Requirements Descriptions for Telecommunication Services", 7th International Conference on Tools with Artificial Intelligence, pp. 295-302.

[11] Rickman, Dale M., (2001). " A New Process for Requirements Understanding", 20th Conference Digital Avionics Systems (DAC), Volume 1, pp. 736-741.

[12] Rittel, H., Hunz, W., (1970). "Issues as Elements of Information Systems Working Paper 131", Institut fuer Grundelagen der Planung, Stuggart.

[13] Seidel, J. V., (1998). "Qualitative Data Analysis", The Ethnograph V. 5, Qualis Research, Appendix E.

[14] Strauss, A., Corbin, J., (1990). Basics of Qualitative Research, Grounded Theory Procedures and Techniques. California: Sage publications.

[15] Turney, P.D., (1999). Learning to Extract Keyphrases from Text. NRC Technical Report ERB-1057, National Research Council Canada.

[16] van Lamsweerde, A., (2000). "Requirements Engineering in the Year 2000: A Research Perspective", Proceedings 22nd International Conference on Software Engineering; Limerick, June 2000, ACM Press.

[17] Wilson, W. M., Rosenberg, L. H., Hyatt, L. E., (1996). "Automated Quality Analysis of Language Requirement Specification", October, pp. http://satc.gsfc.nasa.gov/support/PNSQC_OCT96/pnq.html.

[18] Anton, A., Potts, C., (2000). "Requirements Engineering in the Long Term: Fifty Years of Telephony Evolution", International Workshop on Feedback and Evolution Business Processes (FEAST 2000), London, UK, 10-12 July.