Information and Software Technology 53 (2011) 1084–1096

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

A method for assessing confidence in requirements analysis

Kenneth Boness [a], Anthony Finkelstein [b], Rachel Harrison [c,*]

[a] School of Systems Engineering, University of Reading, Berks, UK
[b] Dept. of Computer Science, UCL, Gower Street, London WC1E 6BT, UK
[c] Dept. of Computing & Electronics, Oxford Brookes University, Oxford OX33 1HX, UK

ARTICLE INFO

Article history:
Received 24 August 2010
Received in revised form 5 May 2011
Accepted 6 May 2011
Available online 14 May 2011

Keywords:
Requirements analysis
Confidence factors
Meta-level assessment

ABSTRACT

Context: During development managers, analysts and designers often need to know whether enough requirements analysis work has been done and whether or not it is safe to proceed to the design stage.
Objective: This paper describes a new, simple and practical method for assessing our confidence in a set of requirements.
Method: We identified four confidence factors and used a goal oriented framework with a simple ordinal scale to develop a method for assessing confidence. We illustrate the method and show how it has been applied to a real systems development project.
Results: We show how assessing confidence in the requirements could have revealed problems in this project earlier and so saved both time and money.
Conclusion: Our meta-level assessment of requirements provides a practical and pragmatic method that can prove useful to managers, analysts and designers who need to know when sufficient requirements analysis has been performed.

1. Introduction

Whenever we attempt to engineer or re-engineer a software system it is widely accepted that arriving at a set of requirements in which we have a lot of confidence is the key to success [20]. However there has been little work to date on ways to arrive at estimates of confidence or on methods that can be used to determine how much confidence different stakeholders attach to a set of requirements. A large system may have very many requirements, each with a different set of confidences associated with it. Estimates of these confidences will help managers to make decisions concerning the costs and benefits of a project.

Previously we have reported on a technique for assessing risks during requirements analysis [2]. During subsequent case studies on real-world systems we came to the realisation that any method intended for use in the real-world has to be as simple and practical as possible if it is to have any hope of being adopted by industry. The new method we describe in this paper uses a simplified form of goal responsibility modelling [27] and replaces the probabilistic risk metrics of [2] with confidence assessments performed by experts using an ordinal scale. This is an important improvement because the probabilistic risk metrics used previously implied a level of precision which could not be guaranteed. Our new method further extends the earlier technique by moderating the assessments using argumentation theory [24] and propagating them within a system using tabulation. Our method is compatible with most requirements representations that depend upon the notion of stepwise refinement. We pay particular attention to the assumptions of stakeholders [8,9,18], which are so often neglected to the detriment of the development.

Our method for assessing confidence during requirements analysis can be summarised as follows:

(1) Construct a goal decomposition graph.
(2) Annotate the graph with estimates of confidence.
(3) Determine the feasibility and adequacy of the requirements.
(4) Consider whether the threats predicted by the feasibility and adequacy assessments are acceptable.

The method is a very practical approach to assessing confidence in requirements which extends our previous technique and is applicable to real-world requirements engineering. Without it managers, analysts, designers and developers are forced to make decisions about whether to continue analysing requirements or start building systems with very little information. The method is particularly useful during requirements analysis and the early stages of systems' development such as the inception and elaboration phases of RUP, and could also be used for planning Scrum sprints. It can be used both with new systems built from scratch and when systems must be constructed using existing COTS

* Corresponding author. Tel.: +44 1865 484506; fax: +44 8700 526595.
E-mail addresses: k.d.boness@reading.ac.uk (K. Boness), a.finkelstein@cs.ucl.ac.uk (A. Finkelstein), rachel.harrison@brookes.ac.uk (R. Harrison).

0950-5849/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.infsof.2011.05.003

components or legacy systems. It can also be used during contract negotiations, or to facilitate responses to invitations to tender. The method has been used during consultancy with a number of our industrial collaborators, and has been found to be helpful.

To demonstrate our method, we present a small but typical problem involving the calculation of body mass index. The method was validated by retrospectively applying it to a case study which we had assessed previously [2]. The case study consisted of a medium-sized project provided by an Administrative Division of University College London. We also carried out semi-structured interviews with the project manager. The results were encouraging and do suggest that our method is useful and usable.

This paper begins with a presentation of our simplified technique for constructing a goal decomposition graph called *goal sketching*. We then go on to discuss the factors which we would like to assess. This is followed by details of our method for assessment and a small but typical exemplar problem. The validation is described next and in the final sections of the paper we discuss related work and our conclusions.

2. Goal sketching

Our assessment method requires the use of a goal graph such as the one shown in Fig. 1. Such a graph could be produced using the KAOS [26] method. However our research, performed with industrial collaborators over many years, convinced us that producing a complete goal graph quickly in a real-world project can be difficult. This led to the development of what we refer to as goal sketching [1,3].

2.1. The goal sketching technique

Goal sketching can be used as a precursor to some other requirements analysis method (such as KAOS modeling [26], use case methods [4], traditional hierarchical requirements modelling [7], etc.) or it can be used alone. It closely resembles KAOS but aims to be very practical. A goal sketch is in fact a goal graph, but the goal sketching technique emphasizes the presence of assumptions and distinguishes them from products (the various system elements to be constructed). This raises awareness of assumptions in the goal analysis [8,9,18,16].

In goal sketching we set down the objectives and show how each objective is to be satisfied (*or at least satisficed [19,23]*) in the resulting system. This is the *keep all objectives satisfied* maxim of van Lamsweerde et al. [26]. There may be many ways to satisfy the objectives, (see, for example, [25]) resulting in the development of a number of different models during analysis. These models allow us to show the project stakeholders' alternative systems for normal operation. Further models can be built for commissioning, decommissioning and maintenance if these are also appropriate.

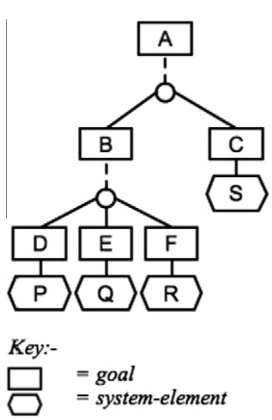

Fig. 1. A simple goal graph.

Fig. 1 illustrates goal-refinement and is referred to as a goal graph. Typically an analyst aiming to specify a system-to-be constructs a hierarchy of goals. The most abstract objectives are presented as root goals from which stem a system of sub-goals refined in steps until goals are reached that are sufficiently concrete that they can be assigned as responsibilities to elements of the intended system. For example, in Fig. 1, *A* is a root goal which is refined into *B* and *C*; both of these are more concrete than *A*. Goal *C* is sufficiently concrete that an element, *S*, of the system-to-be has been given responsibility to guarantee its satisfaction. *B* is not concrete and needs to be refined into *D, E* and *F* which are sufficiently concrete and so have been given to system elements *P*, *Q* and *R* for their satisfaction. This goal graph is said to be structurally complete as every leaf-goal (*C, D, E* and *F*) is guaranteed by an element of the system (*P, Q* and *R*) and consequently, (by refinement arguments joining the leaves to the root), every goal is satisfied.

This state of structural completeness shows how all objectives are understood as being satisfied. When there is uncertainty about an objective or how it might be satisfied it may be necessary for the analyst to approximate some of the goal-refinement in lieu of more complete information. This condition may be found at any point of a project, especially in the early stages such as RUP inception and elaboration. The value of applying the *keep all objectives satisfied* maxim through the device of structural completeness compels the analyst to reveal what is actually known about the requirements so that threats to the stakeholders' expectations can be exposed.

2.2. An example: calculating body mass index

By way of demonstration, we present in Fig. 2 a problem involving the calculation of body mass index that we used in our earlier paper [2].

From the requirements definition in Fig. 2 the goals for the system were determined by the analyst to be as follows:

(1) Normal operation of the walk-on scales must be maintained.
(2) The scales are for use in public places.
(3) WeighCom's good reputation must be maintained.
(4) The scales are to be constructed from prescribed components.

We assume that these goals have been agreed with the stakeholders. We will use the first goal (*normal operation of the walk-on scales must be maintained*) to illustrate our goal sketching technique. The goal sketch for this goal is shown in Fig. 3. This sketch extends the model produced using Objectiver [6] so that responsibility assignments are shown as hexagons. Obstacles are shown as parallelograms, and indicate anti-goals which can prevent a goal from being satisfied. The goals are numbered in the order in which they were analysed. Tool support is needed for both goal sketching and (as discussed later) the application of argumentation. This will ensure that our method is very practical, and is a topic of our current work. This goal sketch will also be used later to illustrate the application of our confidence factors.

3. The confidence factors

In our previous work [2] we identified four independent risk factors: (1) the environmental assumptions, (2) the achievability of the implementation of the requirements, (3) the integrity of the refinements and (4) the stakeholders' mandate. We have