Managing Terminological Interference in Goal Models with Repertory Grid

Nan Niu Steve Easterbrook
Department of Computer Science, University of Toronto
Toronto, Ontario, Canada M5S 3G4
Email: {nn,sme}@cs.toronto.edu

Abstract

Terminological interference occurs in requirements engineering when stakeholders vary in the concepts they use to understand a problem domain, and the terms they use to describe those concepts. This paper investigates the use of Kelly's Repertory Grid Technique (RGT) to explore stakeholders' varying interpretations of the labels attached to softgoals in a goal model. We associate softgoals with stakeholders' personal constructs, and use the tasks that contribute to these goals as elements that stakeholders can rate using their constructs. By structurally exchanging grids data among stakeholders, we can compare their conceptual and terminological structures, and gain insights into relationships between problem domain concepts.

Concepts \ *Terminology*	Same	Different
Same	**Consensus** Stakeholders use terminology and concepts in the same way	**Correspondence** Stakeholders use different terminology for the same concepts
Different	**Conflict** Stakeholders use same terminology for different concepts	**Contrast** Stakeholders differ in terminology and concepts

Figure 1. Terminology and concepts

1 Introduction

Requirements Engineering (RE) aims to identify desires and needs of users, customers, and other constituencies affected by a software-intensive system. One problem of eliciting knowledge from multiple stakeholders is that they may only partially share their terminologies and conceptual systems. Stakeholders may disagree over how to interpret phenomena of the problem domain, how to describe their visions for the intended system, what the requirements are, and how to meet those requirements. Stakeholders may express their conceptual systems using incoherent, ambiguous, or conflicting terms. For example, they may use the same term to describe concepts that differ in meaning; or the same concept may be represented in different descriptions at different levels of abstraction, granularity, and formality. We call this *terminological interference*.

In a well-established scientific domain, experts develop a consensus over relevant distinctions and terms. Over time, they identify and test *objective knowledge* independent of individuals [13]. However, such objective knowledge is not yet available for most RE problem domains. If there is no pre-existing consensus over terminology, it is important to be able to compare the conceptual structures among multiple experts [4].

When stakeholders perceive a problem situation and attach terms to their concepts, there are four possible conditions for the relationship between their concepts and terms, as summarized in Figure 1 [16]. The challenge in knowledge elicitation is to discover which of the situations in Figure 1 apply for a given set of stakeholder terms:

- *Consensus* is a desirable situation, since stakeholders then have a basis for communication using shared concepts and terminologies.
- *Conflict* (also known as *terminological inconsistency* [6]) can cause significant communication problems throughout the requirements process.
- Discovering *correspondence* is important because it lays the grounds for mutual understanding of differing terms through the availability of common concepts.
- Strictly speaking, *contrast* does not involve terminological interference, but the lack of shared concepts could make communication and understanding among stakeholders very difficult.

We interpret both conflict and correspondence as instances of terminological interference. Both have the potential to cause communication problems, if they are not identified and managed. We believe that terminological interference is both inevitable and useful in RE. It is inevitable because stakeholders have complementary perspectives, and are unlikely to have evolved a well-defined terminology for describing the problem situation. It is useful because it provides an opportunity to probe differences in the stakeholders' conceptual systems, to challenge ill-defined terms, and to identify new and productive distinctions for important concepts in the problem domain. Explicit consideration of terminological interference also helps to keep stakeholders from reaching a false, and often too early, consensus [16].

Analysis of terminological interference is only possible if we are able to discover relationships between different stakeholders' mental models and the terms they use to describe them. Kelly's Personal Construct Theory [7] addresses this issue. It explains how an individual constructs a personal (i.e., idiosyncratic) view of his or her environment (e.g., artifacts, events). The theory has been used to develop techniques for exploring personal constructs, most notably the Repertory Grid Technique (RGT) [3]. RGT elicits personal constructs by asking people to compare and contrast objects in the domain of interest.

In this paper, we present a novel use of RGT as a means of exploring how stakeholders interpret the labels attached to softgoals in a goal model. We associate softgoals with stakeholders' personal constructs, and use the tasks that contribute to these goals as elements that stakeholders can rate using their constructs. Structurally exchanging grids data among stakeholders and systematically analyzing the results can help to identify agreements and mismatches in stakeholders' terminology, and offer interesting insights into relationships between problem domain concepts.

2 Repertory Grid Technique

Kelly's theory of personal constructs [7] assumes that the meaning we attach to events or objects defines our subjective reality, and thereby the way we interact with our environment. Repertory Grid Technique (RGT) was originally developed in the context of clinical psychology. However, it has long been recognized as a domain-independent method for externalizing individuals' personal constructs, and has seen applications in a wide variety of situations far removed from psychotherapy. The technique provides a way for people to verbalize how they construe certain objects within the area of interest. These verbalizations are known as *constructs*, and the objects are called *elements*. A construct is a bipolar dimension, where each pole represents the extreme of a particular view or observation.

As an example, the area of interest might be how people feel about certain information sources. In this example, the elements would be various information sources, such as *TV*, *Newspaper*, *Radio*, *Newsgroup*, and so forth. To elicit constructs, the person is asked to consider a "triad" of three

	TV	Newspaper	Radio	NewsGroup	. . .	
Many focuses	1	2	2	5	. . .	Single focus
Multimedia	1	4	2	5	. . .	Text
Entertaining	1	3	1	3	. . .	Not entertaining
Two−way	5	4	4	2	. . .	One−way
. . .	. . .	. . .	. . .	. . .	. . .	. . .

Figure 2. A sample repertory grid

elements and asked to say how two of them seem similar and how the third differs. For example, presented with the triad (A) TV, (B) Newspaper, and (C) Newsgroup, the person may say that A and B have many focuses, whereas C is singly focused. The labels "many focuses" and "single focus" can be treated as the two *poles* of a construct, to which we can then attach, say, a 5-point rating scale. Each element can now be assigned a rating on that construct. As more constructs are generated using different triads and the elements rated on them, a picture can be built up of an individual's ways of construing the domain.

Figure 2 shows a sample repertory grid of the above scenario. Rows represent constructs and columns represent elements. Note that constructs are described by labeling their poles, as these are the elicited terms. A five-point scale indicates where each element lies with respect to the poles of each construct. By convention, the poles to the left of the grid are the "1" end of the scale, and those to the right are the "5" end.

Collected repertory grids are amenable to cluster analysis and many other measurements. Most commonly used grid analysis methods and tools are discussed in [3]. Although most repertory grids are descriptive rather than evaluative in nature, RGT does provide evidence of explanatory and predictive potential [14].

3 Goal Models' Terminological Interference

RGT's "triad" method has been used in RE to gain individual perceptions of a set of requirements so as to measure correlations between requirements [9], reveal system qualities [2], and explore design space [5]. In all these projects, *requirements* were used as the elements, so that the elicited constructs show how stakeholders construe these requirements. However, the ability to compare stakeholders' constructs depends on having agreed a well-defined set of elements first. As we can never be sure that two stakeholders understand a particular requirement in the same way, it is not clear how useful it is to elicit personal constructs with respect to the requirements themselves.

In contrast to the triad-based approaches, we apply RGT as a means of addressing terminological problems in goal-oriented requirements models [8]. We treat goals as personal constructs, to examine whether stakeholders use the same terminology when describing these goals. In particular, we have focused on softgoals – i.e. goals whose satisfaction cannot be established in a clear-cut sense [1]. Softgoals are often difficult to express in a measurable way, and so it is hard to ensure that different stakeholders understand them in the same way. Our aim is to identify terminological interference in the labels attached to goals.

We apply RGT to existing goal models (e.g. i^* [17]). Using RGT, each construct (softgoal) is identified as a pair of polar extremes corresponding to 'make the softgoal' and 'break the softgoal'. Concrete entities in goal models, such as tasks that contribute to the extracted softgoals, are treated as RGT *elements*, since empirical evidence suggests that people are better at comprehending and making analogies between concrete concepts rather than abstractions in RE [12]. Each element is then rated on each bipolar construct. For each grid, some ratings can be obtained from the goal models directly, some can be derived through label propagation algorithms [1], and the remainder need to be completed by the stakeholder. A five-point scale is defined to make such measures both subtle and specific:

1 – break (strong negative)
2 – hurt (weak negative)
3 – neutral (unknown or don't care)
4 – help (weak positive)
5 – make (strong positive)

In a common context, each stakeholder's personal construct system overlaps to some degree with others, and this makes it possible for people to exchange their grids data to share their individual perceptions of the domain. Such exchange needs to be managed in a structured manner to gain sensible results that are amenable to interference analysis.

Figure 3 illustrates structurally exchanging two stakeholders' repertory grids. The scenario before the exchange is shown at the top of the figure. Both Jane and Sam build goal models to analyze the requirements for a computer game. Jane describes how a set of tasks $\{t_b, \ldots, t_j, t_m, \ldots, t_n\}$ and a set of softgoals $\{SG(J_1), \ldots, SG(J_q)\}$ are related. Sam describes the relationship between the tasks $\{t_m, \ldots, t_n, t_p, \ldots, t_s\}$ and the softgoals $\{SG(S_1), \ldots, SG(S_k)\}$.

Structural exchange allows the tasks in the goal model derived from one stakeholder to be assessed by another in order to determine whether the two stakeholders have consensus or conflict in their use of terminology and concepts. The bottom of Figure 3 shows how this exchange works in the game design example. Jane evaluates the tasks $\{t_p, \ldots, t_s\}$ that only appeared in Sam's model against her own softgoals $\{SG(J_1), \ldots, SG(J_q)\}$. And Sam analyzes how tasks in Jane's model contribute to his softgoals.

We only exchange concrete entities (in this case, tasks)

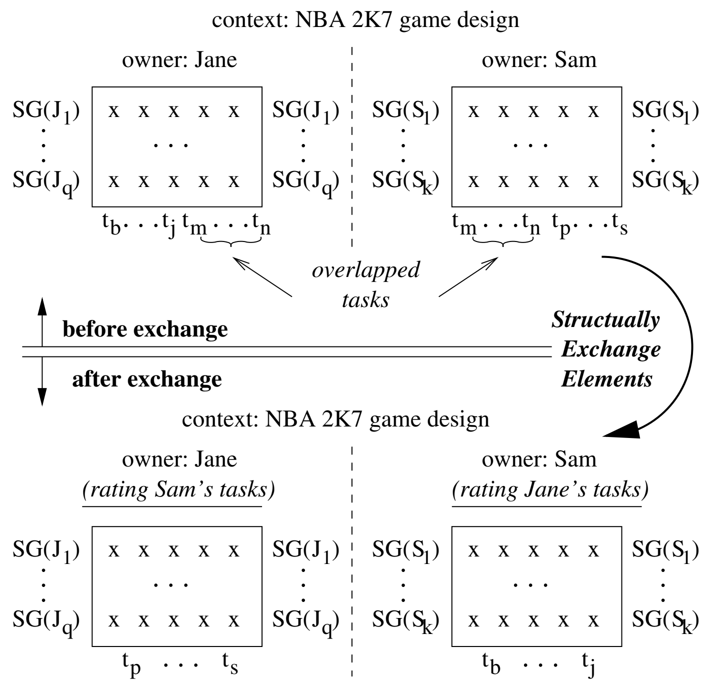

Figure 3. Exchange of two stakeholders' grids

between stakeholders, because at this stage, the abstract constructs only have meaning within each person's individual conceptual system. A construct is a discriminator, not a verbal label [3]. Construct names are merely convenient labels to remind the person of the thoughts and feelings that the construct provoked, and hence are not transferable to another person without discussion and negotiation [15]. The key idea is to compare the stakeholder's constructs by how they relate to a shared set of concrete entities, rather than by any terms the stakeholders use to describe them. In this way we avoid making any assumptions about the meanings of individuals' constructs.

On the other hand, the concrete entities *are* exchanged, because to make comparisons across individuals and investigate construct similarity requires that they each construe the same set of elements [10]. We assume that people focusing on similar topics would agree on the definition of a common set of concrete tasks within the area of interest, i.e., when presented with specific and relevant tasks that are devised by others, people are likely to grasp the essential meaning behind the notions.

We compare the stakeholders' constructs (softgoals) according to the extent to which they array the common set of tasks. Thus, if, for example, tasks rated to support 'interoperability' tend to be rated to break 'performance', then there is a negative association between these two softgoals.

To tackle the terminology problem on softgoals, our analysis needs to identify 'correspondence' and 'conflict' areas shown in Figure 1. If two softgoals orchestrate the tasks in the same or very similar way, 'correspondence' is established between these constructs even though they may be labeled differently. If two softgoals that have been labeled using the same terms align the tasks in a markedly dissimilar fashion, then 'conflict' is detected. These findings not only enable us to gain insights into stakeholders' use of terminologies and concepts, but also allow us to generate plausible hypotheses to be tested with subsequent efforts in eliciting and communicating requirements.

4 Conclusions

Managing terminological interference is fundamental and important in RE when multiple stakeholders and perspectives are involved. In this paper, we have designed a flexible method based on RGT for understanding stakeholders' terminological and conceptual structures. Our approach is appealing, since RGT avoids the problems of imposition of terminology, and the meaning of a term is essentially treated as a relationship between signs and actions. Our future work includes developing a well-categorized taxonomy to organize large amounts of artifacts in RE. According to our approach, such a taxonomy can be viewed as a catalogue of how constructs represented by linguistic symbols relate formally in a particular context.

Extracting both elements and constructs from existing models offers a novel, lightweight way of applying RGT to new situations, such as [11]. Abstract and concrete terms are treated as constructs and elements respectively, and are simultaneously presented to participants to externalize their individual views of these concepts. Structurally exchanging grids data offers a basis for comparing different lexicons in requirements models, and also generates a set of hypotheses to guide further RE activities. Since conceptual resolution is common and important in developing and evolving (large-scale) software systems, we believe that RGT can find its proper applications throughout the entire software lifecycle.

In future work, we plan to develop efficient methods for producing a core set of common elements that a group of participants can all meaningfully construe. This is critical to all RGT-based approaches, and can lead discussion to exploring the ongoing debate about "whether elements exist independently of constructs, or whether in fact elements are also constructs". We also plan to extend our proposed approach with RGT's triad method to develop a comprehensive RE framework for thoroughly understanding and adequately reflecting the stakeholders' desires and needs.

Acknowledgments. *We would like to thank the members of the Kids Help Phone project at the University of Toronto for their insightful comments on this work. We also thank the anonymous reviewers for their constructive suggestions. Financial support was provided by NSERC and by Bell Canada through the Bell University Labs.*

References

[1] L. Chung, B. Nixon, E. Yu, and J. Mylopoulos. *Non-Functional Requirements in Software Engineering*. Kluwer Academic Publishers, 2000.

[2] H. Delugach and B. Lampkin. Acquiring software requirements as conceptual graphs. In *Intl. Symposium on Requirements Engineering*, pages 296–297, 2001.

[3] F. Fransella, R. Bell, and D. Bannister. *A Manual for Repertory Grid Technique*. John Wiley & Sons, Ltd., 2nd edition, 2004.

[4] B. Gaines. An overview of knowledge acquisition and transfer. *Intl. Journal of Man-Machine Studies*, 26(4):453–472, 1987.

[5] M. Hassenzahl and R. Wessler. Capturing design space from a user perspective: The repertory grid technique revisited. *Intl. Journal of Human-Computer Interaction*, 12(3&4):441–459, 2000.

[6] J. Jutras. An automatic reviser: the transcheck system. In *Sixth Conference on Applied Natural Language Processing*, pages 127–134, 2000.

[7] G. Kelly. *The Psychology of Personal Constructs*. New York: Norton, 1955.

[8] A. Lamsweerde. Goal-oriented requirements engineering: A guided tour. In *Intl. Symposium on Requirements Engineering*, pages 249–262, 2001.

[9] N. Maiden and G. Rugg. Acre: selecting methods for requirements acquisition. *Software Engineering Journal*, 11(3):183–192, 1996.

[10] C. McKnight. The personal construction of information space. *Journal of the American Society for Information Science*, 51(8):730–733, 2000.

[11] N. Niu and S. Easterbrook. Discovering aspects in requirements with repertory grid. In *Early Aspects at ICSE: Workshop on Aspect-Oriented Requirements Engineering and Architecture Design*, 2006.

[12] Y. Pisan. Extending requirement specifications using analogy. In *Intl. Conference on Software Engineering*, pages 70–76, 2000.

[13] K. Popper. Epistemology without a knowing subject. In *Intl. Congress for Logic, Methodology and Philosophy of Science*, pages 333–373, 1968.

[14] A. Ryle. *Frames and Cages: The Repertory Grid Approach to Human Understanding*. Sussex University Press, 1975.

[15] M. Shaw. *On Becoming A Personal Scientist: Interactive Computer Elicitation of Personal Models of the World*. Academic Press, 1980.

[16] M. Shaw and B. Gaines. Comparing conceptual structures: Consensus, conflict, correspondence and contrast. *Knowledge Acquisition*, 1(4):341–363, 1989.

[17] E. Yu. *Modeling Strategic Actor Relationships for Process Reengineering*. PhD thesis, University of Toronto, 1994.