So, You Think You Know Others' Goals?

A Repertory Grid Study

Nan Niu and Steve Easterbrook, *University of Toronto*

This novel way to apply the Repertory Grid Technique detects stakeholders' inconsistent use of terminology when stating goal-oriented requirements and helps spark further discussion.

People filter their observations of the world according to their interests, each person using his or her own conceptual framework. So, when we ask them to describe something, their descriptions differ: they might focus on different aspects and choose different terms. In requirements analysis, stakeholders can have varied expertise areas, distinct responsibilities, and divergent goals. They might disagree over how to interpret phenomena in the problem domain, what the requirements are, and how to meet those requirements. As a result, they often express themselves using ambiguous or conflicting terms. For example, what one person calls "responsiveness" might correspond to "performance" in another's description. For cell phone software, one stakeholder might interpret "usability" as "easy to learn," another as "mobility." Such mismatches in stakeholders' vocabulary can be very hard to detect. We call this *terminological interference*.

Analysis of terminological interference is possible only if we can discover relationships between stakeholders' mental models and the terms they use to describe them. George Kelly's Personal Construct Theory addresses this issue.[1] According to PCT, individuals develop their own set of mental constructs to help make sense of their environment. Researchers have used this theory to develop techniques for exploring personal constructs, most notably the Repertory Grid Technique.[2] The RGT elicits personal constructs by asking people to compare and contrast objects in the domain of interest.

Recent research in requirements engineering (RE) has generated a number of notations for modeling stakeholders' goals and the relationships between them (see the "Goal-Oriented Analysis and Softgoals" sidebar). However, the community has paid little attention to how stakeholders can develop consensus on the meaning of the goals in a goal model. In this article, we show how to use the RGT to compare stakeholders' terms when they describe their softgoals (goals whose satisfaction can't be established in a clear-cut sense). We conducted a pilot study for a nonprofit organization to demonstrate our approach. The study shows that the technique can readily identify agreements and mismatches in stakeholders'

0740-7459/07/$25.00 © 2007 IEEE

Goal-Oriented Analysis and Softgoals

Goals express, at various levels of abstraction, stakeholders' many objectives for the system under consideration. Goal-oriented requirements engineering uses goal models to elicit, elaborate, structure, specify, analyze, negotiate, document, and modify requirements. Axel van Lamsweerde provides a guided tour of this line of research.[1]

Goal modeling shifts the emphasis in requirements analysis to the actors in an organization, their goals, and the interdependencies between those goals, rather than focusing on processes and objects. This helps us understand why a new system is needed and lets us effectively link software solutions to business needs. Requirements research has produced two principal goal-modeling techniques—Knowledge Acquisition in Automated Specification (KAOS)[2] and distributed intentionality (*i**).[3] Both use goal models to provide criteria for determining whether requirements are relevant and complete.

Goal-modeling frameworks distinguish between *hard goals*—states that actors can attain—and *softgoals*, which can never be fully satisfied. System qualities such as reliability, efficiency, and portability are typically expressed as softgoals to suggest that the intended software is expected to satisfy them within acceptable limits, rather than absolutely. Softgoals tend to express abstract concepts because they're difficult to express in a measurable way. We can use an exploration of trade-offs between softgoals as the basis for requirements negotiation.[4]

References

1. A. van Lamsweerde, "Goal-Oriented Requirements Engineering: A Guided Tour," *Proc. 5th IEEE Int'l Symp. Requirements Eng.* (RE 01), IEEE CS Press, 2001, pp. 249–261.
2. A. Dardenne et al., "Goal-Directed Requirements Acquisition," *Science of Computer Programming*, vol. 20, nos. 1–2, 1993, pp. 3–50.
3. E. Yu, "Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering," *Proc. 3rd IEEE Int'l Symp. Requirements Eng.* (RE 97), IEEE CS Press, 1997, pp. 226–235.
4. L. Chung et al., *Non-Functional Requirements in Software Engineering*, Kluwer Academic Publishers, 2000.

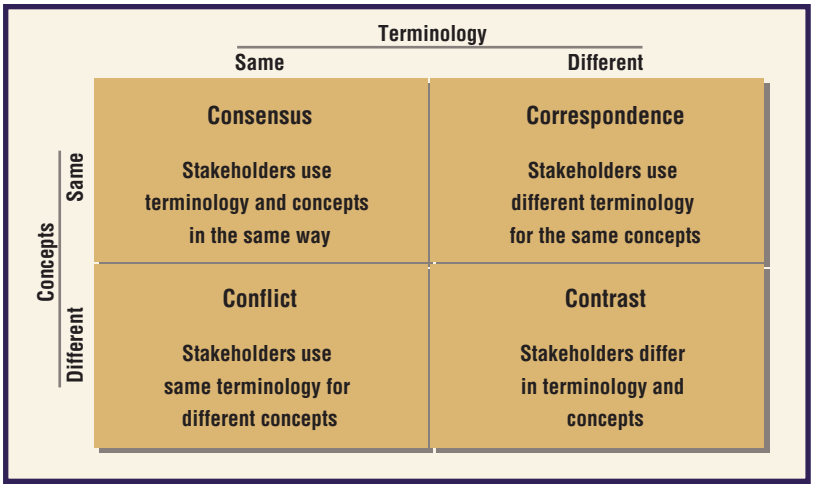

Figure 1. Relationships between stakeholders' terminology and concepts when they perceive a shared problem situation. (adapted from Mildred Shaw and Brian Gaines[3])

terminologies and can be performed without preliminary training or specific resources.

Terminological interference

When people observe a complex problem domain, their observations are inevitably incomplete. Personal values and experiences act as a filter, leading them to focus on aspects that are particularly salient to them. This gives rise to many partial conceptual structures. When asked to articulate these, individuals choose terms that are personally meaningful. Often, people find it necessary to adapt or invent new terms to describe situations that they haven't previously needed to articulate.

When stakeholders perceive a shared problem situation and attach terms to their concepts, four possible conditions exist for the relationship between their terminology and concepts (see figure 1). The challenge in knowledge elicitation is to discover which situation applies for a given set of stakeholder terms:

- *Consensus* is desirable; it gives stakeholders a basis for communication using shared concepts and terminologies.
- *Correspondence* lays the ground for mutual understanding of differing terms through the availability of common concepts.
- *Conflict* can cause significant communication problems during RE activities.
- *Contrast* doesn't involve interference, strictly speaking. But the lack of shared concepts could make communication and understanding among stakeholders very difficult.

We interpret each correspondence and conflict as an instance of terminological interference; each can cause communication problems, if not identified and managed. On the other hand, we believe that terminological interference is both inevitable and useful in RE. It's inevitable because stakeholders have complementary perspectives and are unlikely to have agreed on a well-defined, shared terminology for describing the problem situation. It's useful because it provides an opportunity to probe differences in the stakeholders' conceptual systems, challenge ill-defined terms, and identify new and productive distinctions for important concepts in the problem domain.

The Repertory Grid Technique

The PCT (see the related sidebar) assumes that the meaning we attach to events or objects defines our subjective reality and thereby the way we interact with our environment. Developed within the PCT, the Repertory Grid Technique acts as an instrument for capturing the dimensions and structure of personal meaning. The RGT provides a way for people to verbalize how they construe certain objects within an area of interest. These verbalizations are known as *constructs*, and the objects they refer to 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 construe certain information sources. In this example, the elements would be various information sources, such as TV, Radio, Newspaper, Newsgroup, and so forth. A simple way to elicit a person's constructs is to select a *triad* of elements and ask for a way in which 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 might say that A and B have many focuses, whereas C is singly focused. The construct ranging from "many focuses" to "single focus" can be considered a rating scale using, for instance, a scale from 1 to 5. The person can now assign each element a rating on that construct. The same triad might elicit more constructs. For instance, the person might also say that B and C are text-based, while A delivers multimedia services. As the person generates more and more constructs using different triads and rates the elements on them, a picture can be built up of an individual's ways of construing the domain.

Figure 2 presents a sample repertory grid with constructs elicited in our current example. Each column represents an element from the domain and each row represents an elicited construct. Constructs are bipolar, so we label them using the terms that participants gave to describe the two poles during the elicitation. Each entry in the grid indicates how the participants rated the element in that column according to the construct in that row, using a five-point scale. By convention, "1" means we can best describe the element using the pole to the grid's left, and "5" means it's best to use the pole to the grid's right, with the remaining values indicating intermediate points on the scale.

Personal Construct Theory

George Kelly's basic PCT postulate is that a person's thought processes are psychologically channelized by the ways in which he or she anticipates events.[1] A key idea in PCT is the image of the person as scientist. According to the theory, each person constructs a model of the world (much as a scientist constructs a theory), acts on the basis of that model (as the scientist creates an experiment to test the theory), and then alters the model in the light of feedback from the results of his or her actions (as the scientist uses data from the experiment to modify the theory). This view shares much of the spirit of the Inquiry Cycle,[2] in which requirements models are theories about the world, and designs are tests of those theories.

A key message of PCT is that individuals set the measure of their own freedom and their own bondage by the level at which they choose to establish their convictions. Constructs are ways of construing the world, enabling people to respond to their experiences in ways that are explicitly formulated or implicitly acted out.[1] For example, the way in which I interact with my desk is determined by the way I construe it—do I polish it carefully because I see it as something to be looked after, or do I put my feet up on it because I see it as a convenient resting point? Thus, in Kelly's theory, the notion of objectivity disappears, and the best we can do along these lines is intersubjectivity, thinking rather of a dimension representing degree of agreement between construers and degree of certainty of judgment.

PCT represents a coherent, comprehensive psychology of personality that has special relevance for psychotherapy. Researchers have developed PCT in conversational models of learning, using tools such as the Repertory Grid Technique.[3] In fact, the RGT has long been recognized as a domain-independent method for externalizing individuals' personal constructs, and researchers and practitioners in diverse fields such as psychology, education, business management, and so on have applied it in a wide variety of situations far removed from clinical psychology.

References

1. G.A. Kelly, *The Psychology of Personal Constructs*, Norton, 1955.
2. C. Potts, K. Takahashi, and A.I. Anton, "Inquiry-Based Requirements Analysis," *IEEE Software*, Mar./Apr. 1993, pp. 21–32.
3. F. Fransella, R. Bell, and D. Bannister, *A Manual for Repertory Grid Technique*, 2nd ed., John Wiley & Sons, 2004.

	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.

RGT Applications in Requirements Engineering

Mildred Shaw and Brian Gaines developed one of the initial approaches to repertory grid requirements elicitation. They based the approach on their knowledge engineering work and introduced George Kelly's Personal Construct Theory as a universal foundation for modeling methodologies.[1]

Neil Maiden and Gordon Rugg specify a situation where the RGT fits for requirements acquisition: package selection. Purchasing a software package often involves selection, and in such cases, requirements for the new system should act as selection criteria. If candidate packages are known, the RGT becomes an effective acquisition method because it explicitly encourages respondents to give criteria that discriminate between elements such as software packages.[2]

Marc Hassenzahl and Rainer Wessler explore the RGT's practical value in gathering design-relevant information about early artifact prototypes designed in parallel. Personal constructs (for example, boring-interesting, graspable-abstract) that people employ when confronted with design alternatives play an important role in narrowing the design process.[3]

Harry Delugach and Brian Lampkin adopt the RGT's triad method to elicit and classify requirements, distinguish and measure correlations between requirements, and reveal system qualities.[4] They use requirements as elements in their work so that the elicited constructs show how stakeholders construe these requirements.

However, the ability to compare stakeholders' constructs depends on the stakeholders having agreed on a well-defined set of elements first. Because we can never be sure that two stakeholders understand a particular requirement in the same way, it's unclear how useful it is to elicit personal constructs with respect to the requirements themselves. Recent work has taken a finer-grained look at requirements goal models by treating softgoals as personal constructs so as to discover early aspects[5] and to identify softgoal contributions.[6]

References

1. M. Shaw and B. Gaines, "Requirements Acquisition," *Software Eng. J.*, May 1996, pp. 149–165.
2. N. Maiden and G. Rugg, "ACRE: Selecting Methods for Requirements Acquisition," *Software Eng. J.*, May 1996, pp. 183–192.
3. M. Hassenzahl and R. Wessler, "Capturing Design Space from a User Perspective: The Repertory Grid Technique Revisited," *Int'l J. Human-Computer Interaction*, vol. 12, nos. 3–4, 2000, pp. 441–459.
4. H. Delugach and B. Lampkin, "Acquiring Software Requirements as Conceptual Graphs," *Proc. 5th IEEE Int'l Symp. Requirements Eng.* (RE 01), IEEE CS Press, 2001, pp. 296–297.
5. N. Niu and S. Easterbrook, "Discovering Aspects in Requirements with Repertory Grid," *Proc. 2006 Int'l Workshop Early Aspects at ICSE*, ACM Press, 2006, pp. 35–41.
6. B.G. Baixauli et al., "Eliciting Non-Functional Requirements Interactions Using the Personal Construct Theory," *Proc. 14th IEEE Int'l Requirements Eng. Conf.* (RE 06), IEEE CS Press, 2006, pp. 347–348.

Although the triad method is widely used in the RGT, there are many other ways to construct a repertory grid.[2] In our approach, we extracted both the elements and the constructs of a grid from existing goal models to give us a lightweight technique that would supplement goal-oriented requirements analysis methods. However, we believe the triad method can also be useful in eliciting softgoals directly from stakeholders and could thus enhance existing approaches to goal elicitation.

Collected repertory grids are amenable to clustering analysis and many other measurements. Although most repertory grids are descriptive rather than evaluative in nature, the RGT does provide evidence of explanatory and predictive potential.[2]

Terminological interference in goal models

RE researchers have used the RGT mainly as a knowledge acquisition tool (see the "RGT Applications in Requirements Engineering" sidebar). In our work, we apply the RGT instead as a means of addressing terminological problems in goal-oriented requirements models. Haruhiko Kaiya and his colleagues provide a technique to help identify requirements discordances, in which each stakeholder evaluates and compares goal preferences with each other and then uses the differences in evaluation results to detect the conflict of understanding the goals.[4] In addition to detecting conflicts, the RGT also helps establish correspondences between stakeholders' descriptions of goals.

In our approach, we treat goals as personal constructs, to examine whether different stakeholders use the same terminology when describing their goals. In particular, we've focused on softgoals—that is, goals whose satisfaction can't be established in a clear-cut sense. Softgoals are often hard to express in a measurable way, so ensuring that different stakeholders understand them in the same way is difficult.

To compare different stakeholders' constructs, we must have an agreed set of elements. Requirements goal models typically describe *tasks*, which contribute in various ways to the satisfaction (or otherwise) of goals. Because tasks are much more concrete than softgoals, it's more likely that stakeholders will agree on their meanings: empirical evidence suggests that people are better at comprehending concrete RE concepts than abstract ones.[5] In particular, we assume that people focusing on similar topics can readily agree on the definition of a common set of concrete tasks within the area of interest. We then compare

stakeholders' softgoals by how they relate to this shared set of concrete tasks rather than by any terms the stakeholders use to describe them. Our approach involves four highly iterative and interactive activities: extraction, exchange, comparison, and assessment.

Extraction

A key RGT assumption is that a finite set of elements defines the context. We must carefully choose elements within the area of interest of the constructs we wish to study.[2] For instance, it bends our minds to consider "antique" or "modern" numbers and "prime" or "nonprime" furniture.

When analyzing goal models, we begin with some core agent or key activities in the system; this generally provides a well-scoped area of interest. We carefully record each grid's context so that we can perform sensible exchange and comparison.

We treat each softgoal in the context as a construct, identified as a pair of polar extremes corresponding to "make the goal" and "break the goal." Then we select concrete entities, such as tasks related to the chosen constructs, as elements.

We then rate each element on each bipolar construct. Some ratings can be obtained from the goal models directly, some can be derived through label propagation algorithms,[6] and the remainder needs to be completed by the stakeholder. We define a five-point scale 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)

Exchange

Each grid expresses how a particular stakeholder views the domain and in what terms he or she makes sense of the underlying elements. In a shared context, each stakeholder's personal construct system overlaps to some degree with others, and this lets people exchange their grid data to share their individual perceptions of the domain.

We exchange only concrete entities (that is, tasks) between stakeholders because at this stage, abstract constructs have meaning only within each person's individual conceptual system. A construct is a discriminator, not a verbal label, so it's not transferable to another person without discussion and negotiation.[2]

On the other hand, the concrete entities are exchanged, because to make comparisons across individuals and investigate construct similarity requires that each construes the same set of elements. This structural exchange keeps us from making assumptions about the meanings of individuals' constructs.[7]

Comparison

We compare stakeholders' softgoals according to how they array the set of common tasks in a particular context. We can examine any two constructs' relationship by seeing to what extent one construct's ratings of all the elements tend to match, or differ from, the other's ratings.

If two softgoals relate to the tasks in the same or very similar way, we note them as a potential "correspondence," even though they might be labeled differently. If two softgoals contain the same term but relate to the tasks in a markedly dissimilar way, we note them as a potential "conflict."

Assessment

The most important reason to analyze stakeholders' goal models and compare their softgoals in a repertory grid isn't the grid itself but the discussion that follows. Our approach is of practical value if our findings help identify the sources of existing interferences and then generate follow-up questions to resolve them.

The most important reason to analyze stakeholders' goal models and compare their softgoals is the discussion that follows.

Pilot study

We carried out a pilot study to investigate the applicability of our RGT-based approach. The study context was a nonprofit organization, Kids Help Phone (KHP), which counsels kids and parents across Canada through the phone and Internet. Our pilot was part of a research project investigating the use of a goal-oriented framework for systematically analyzing the requirements for new Web-based counseling services.[8]

At the project's start, KHP project members conducted 14 stakeholder interviews, covering all major roles in the organization, and then used the interview transcripts to develop goal models. In the process of constructing these models, the team encountered

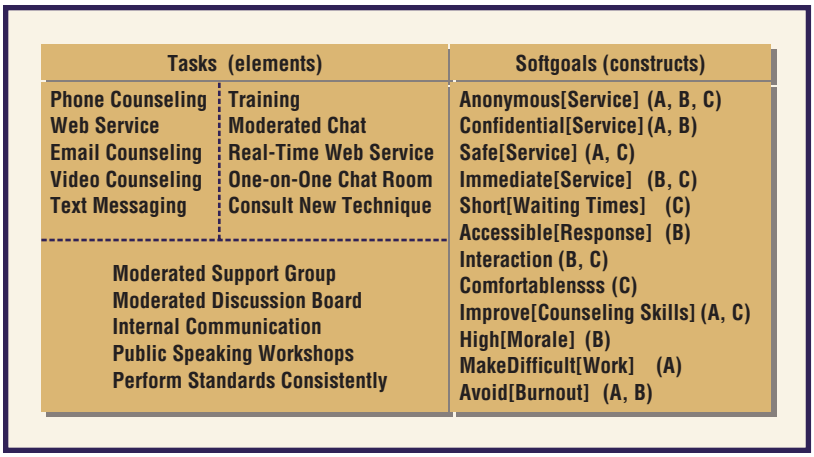

Figure 3. List of all Kids Help Phone counseling elements and constructs.

several problems emerging from diverse stakeholders with competing goals and differing vocabularies. For this reason, we considered the models appropriate candidates for applying the RGT.

Of particular interest for our study was a set of observations from the project's earlier stages, when the team had to deal with terminological issues. The observations were as follows:

- Using each stakeholder's own vocabulary, or a close paraphrase, helped to avoid modeling bias.
- Using more descriptive terms to name the model entities improved the model's readability.
- Modelers who recorded where the terms came from and why they were chosen were able to build traceability on the fly.
- Modelers sometimes had to fabricate terms to express stakeholders' requirements, and modelers differed in terminologies among themselves.

These observations led us to explore whether better approaches were available to handle terminological interference.

Extraction and exchange

We focused our study on the most important issue for KHP's service planning: the counseling role itself. The data set contained interviews with three counselors, and the team had developed a separate goal model from each of these transcripts. Three KHP project members, whom we call Ana, Bob, and Cem, offered us these models and helped us extract model elements, complete repertory grids, and assess comparison results. Figure 3 shows the extracted entities from these goal models—15 tasks and 12 softgoals.

In our approach, all the analysts share the set of tasks, which together determine the common ground. To circumvent bias, we asked a designated requirements engineer other than the original modelers to extract and consolidate the common set of tasks. The engineer extracted softgoals directly from each of the three models. The effort of extracting tasks and softgoals took about one hour.

Because we treat softgoals as personal constructs, we append the first letter of the stakeholder's name to the name of each softgoal, so that those with the same label but different owners are treated as distinct constructs. Strictly speaking then, figure 3 contains 20 softgoals, not 12. We then asked each analyst to rate all 15 tasks in his or her individual list of softgoals using the five-point scale defined earlier. They did so manually because of lack of tool support. In our study, each rating exercise lasted approximately 30 minutes.

Comparison and assessment

Using the FOCUS program,[2] we analyzed the three repertory grids collected from Ana, Bob, and Cem after the extraction and exchange procedures and generated one resultant grid. FOCUS performs a two-way hierarchical clustering analysis and reorders the grid so that similarly rated elements are adjacent and similarly used constructs are adjacent. The full version of the resultant grid has 15 elements and 20 constructs, as figure 3 indicates. Because of space constraints, figure 4 illustrates this idea using a projection with five tasks and six softgoals. FOCUS can also reverse constructs' poles to show correlated values, so we've annotated the poles with "+" to denote the high end (5) of the scale and "–" to denote the low end (1)—that is, "breaking" a softgoal.

FOCUS builds *dendrograms* (tree diagrams, often used to illustrate how the clusters produced by a clustering algorithm are arranged) that illustrate the strength of association between elements and between constructs. For example, in figure 4, the upper dendrogram demonstrates the relationships between softgoals. To highlight the grid's clusters, FOCUS gives dark shading to the ratings of 4 and 5, light shading to the ratings of 3, and no shading to the ratings of 1 and 2. This helps the users of FOCUS easily identify blocks in the grid.

Figure 4 shows that the terms used to express softgoals interfered greatly. For instance, from Bob's perspective, the softgoals Confidential and Anonymous were indistinguishable in terms of the tasks shown in the grid. If the stakeholders used them interchangeably, correspondence would be established; otherwise, we'd need to distinguish the constructs through further elicitation. For example, we might ask Bob to specify a task that makes one softgoal but not the other. Although both Bob and Cem used the term "Anonymous," they probably weren't referring to the same concept. These two constructs were associated at the 70 percent level, one of the lowest matching scores between any two softgoals shown in figure 4. We'd flag this terminological inconsistency and explore it further.

The projected repertory grid shown in figure 4 illustrates how we explore terminology problems. In our study, it would be inappropriate to deduce too much from the resultant grid. We'd generate plausible hypotheses to be tested with follow-up discussion, rather than generating any firm conclusions about terminological interference.

To evaluate the results of our investigation, we presented preliminary findings to the requirements analysts Ana, Bob, and Cem, who created the initial goal models on the basis of the interview transcripts and have been in constant communication with the KHP organization. The precision of our approach was satisfactory: the analysts confirmed all the terminological interferences that we detected—five correspondences and three conflicts—in the context of counseling. However, assessing whether the approach can detect *all* occurrences of terminological interference in the goal models is harder. In this sense, the pilot trial resulted in only true positives. To validate the completeness of our approach, we must explore whether any softgoals that have interferences fail to manifest in the repertory grid's clustering analysis. We're designing further studies to investigate this.

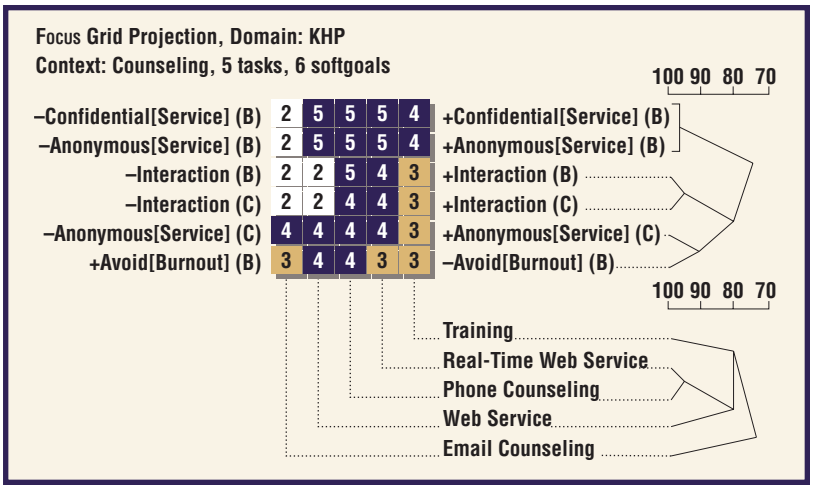

Figure 4. Part of the grid resulting from a two-way hierarchical clustering analysis using Focus.

Observations

We can make several observations on the basis of our findings and discussions with the analysts. First, in some cases, correspondence between terms is easy and trivial to establish, so there's no need to resort to any complicated procedure. For example, all requirements analysts agreed that high-level softgoals about counseling such as Good, Helpful, Proper, and High-Quality had the same meaning, even though different stakeholders had adopted them. It would have been a waste of time to rate tasks on these softgoals, so we excluded them from our grid design.

Second, although statistical evidence shows high similarity between two terms, subtle and important differences might exist and must be investigated. When comparing Bob's and Cem's Anonymous[Service] softgoal on the basis of all 15 tasks in our study, we reached the similarity level at 86.7 percent, which was the second-highest match for each construct. If we treated this as a consensus, we would have missed an important distinction. In Cem's opinion, the task Email Counseling contributed to Anonymity positively because people could protect their identities in the virtual space. But Bob thought Email Counseling could hurt Anonymity because most parents now install censorship software to protect their kids online, and this made the Anonymity of Email Counseling vulnerable. Exploring this difference yielded a more complete view of what Anonymous[Service] really meant to the counselors. This observation also indicates that setting numerical thresholds for establishing correspondence between constructs is unlikely to be useful in practice. In many cases in the repertory grid, it is the subtleties that matter. Although our data supports quantitative analysis, qualitative inspection provides richer insights.

Third, conflicting relationships exist between mutually agreeable terms and concepts. For instance, both Ana and Bob agreed on the meanings of the task Consult New Technique and the

Can we generalize our approach to cope with requirements artifacts other than goal models?

softgoal Avoid[Burnout]. However, Ana perceived a negative relationship because Consult New Technique would MakeDifficult[Work] and therefore hurt the softgoal Avoid[Burnout]. From Bob's standpoint, the task Consult New Technique could contribute positively to High[Morale], which helped to Avoid[Burnout]. This difference, which was captured in the grid, reflected conflict beyond the terminological level and sparked another discussion.

Limitations

Although the pilot study demonstrated our approach's usefulness, several limitations of the RGT became obvious. Most notably, we assume that stakeholders can mutually understand a common set of tasks in a given context, but people might not be able to accurately interpret other people's elements. Because the actual phrasing of elements might have a major impact on the proposed method, we've assigned a dedicated requirements engineer to consolidate the common ground for grid analysis. In the future, we plan to use a simple workshop to establish overlaps between stakeholders and combine the RGT and the nonfunctional requirements catalogue[6] to manage terminological interference.

The smooth establishment of an agreed set of tasks in our study could be the result of comparing models that were essentially developed for different stakeholders fulfilling the same organizational role. To further investigate our approach's scope of applicability, we plan to extend the interference analysis to heterogeneous stakeholder groups, such as people in different organizational roles or having different working experiences.

An earlier KHP case study found that serious scalability issues arose for goal modeling.[8] Our approach helped to address some scalability challenges by taking a prefixed element-construct view of models. The respondents from our pilot confirmed that extracting tasks and softgoals in the area of interest reduced the model's complexity. They also pointed out that stakeholders could perform the proposed interference detection method without preliminary training or specific resources. However, some respondents commented that the effort of manually completing the extracted repertory grid was considerable and that the grid-rating process could have been more efficient and scalable. We're developing semiautomated tool support for grid generation via label-propagation algorithms,[6] but it remains uncertain whether automatically filled partial grids can detect interferences. We'll clarify this issue through additional experiments.

Our pilot is a single study with one application rather than a controlled experiment on multiple representative subjects. We need more empirical evidence to strengthen the exploratory findings reported here. We must also address threats to validity—for example, does coincidence rating[2] lead to bias in the results? How many actual interferences does grid analysis fail to detect? To what extent can we generalize our approach to cope with requirements artifacts other than goal models?

Problems with stakeholder terminology are endemic in RE. As our study shows, stakeholders don't use their terminologies consistently when stating requirements. However, these terminological problems need not be a barrier to understanding stakeholders' goals, if approached carefully. One of the RGT's strengths is that it avoids the problem of imposing an unnatural terminology on stakeholders—it essentially treats a term's meaning as a relationship between signs and actions.

The results of terminological-interference analysis are informative but not judgmental. So, we don't address whether any particular use of the terminology is right or wrong, or even whether a particular definition of a term is better than another. Most RE activities take place without correct and consistent models, and so, as Martin Feather indicated,[9] hinge on getting right from wrong. We expect our proposed approach to act as a helpful initial step toward a more comprehensive framework for thoroughly understanding and adequately reflecting stakeholders' desires and needs.

Acknowledgments

We thank the members of the Kids Help Phone project at the University of Toronto for their generous support throughout our study. We also thank Erik Simmons for his insightful comments. Funding was provided by Bell University Laboratories and the Natural Sciences and Engineering Research Council of Canada.

References

1. G.A. Kelly, *The Psychology of Personal Constructs*, Norton, 1955.
2. F. Fransella, R. Bell, and D. Bannister, *A Manual for Repertory Grid Technique*, 2nd ed., John Wiley & Sons, 2004.
3. M.L.G. Shaw and B.R. Gaines, "Comparing Conceptual Structures: Consensus, Conflict, Correspondence, and Contrast," *Knowledge Acquisition,* vol. 1, no. 4, 1989, pp. 341–363.
4. H. Kaiya et al., "Improving the Detection of Requirements Discordances among Stakeholders," *Requirements Eng. J.*, vol. 10, no. 4, 2005, pp. 289–303.
5. Y. Pisan, "Extending Requirement Specifications Using Analogy," *Proc. 22nd Int'l Conf. Software Eng.* (ICSE 00), IEEE CS Press, 2000, pp. 70–76.
6. L. Chung et al., *Non-Functional Requirements in Software Engineering*, Kluwer Academic Publishers, 2000.
7. N. Niu and S. Easterbrook, "Managing Terminological Interference in Goal Models with Repertory Grid," *Proc. 14th IEEE Int'l Requirements Eng. Conf.* (RE 06), IEEE CS Press, 2006, pp. 296–299.
8. S. Easterbrook et al., "Do Viewpoints Lead to Better Conceptual Models? An Exploratory Case Study," *Proc. 13th IEEE Int'l Requirements Eng. Conf.* (RE 05), IEEE CS Press, 2005, pp. 199–208.
9. M.S. Feather, "Requirements Engineering—Getting Right From Wrong," *Proc. 3rd European Software Eng. Conf.* (ESEC 91), LNCS 550, Springer, 1991, pp. 485–488.

About the Authors

Nan Niu is a doctoral candidate in computer science at the University of Toronto. His research interests include managing crosscutting concerns during requirements engineering activities and manipulating requirements and design models in collaborative software engineering. He received his MSc in computing science from the University of Alberta. Contact him at the Univ. of Toronto, Dept. of Computer Science, 40 St. George St., Toronto, Ontario, Canada M5S 2E4; nn@cs.toronto.edu.

Steve Easterbrook is a computer science professor at the University of Toronto. His research interests include analysis of requirements for complex software-intensive systems, how the views of different stakeholders can be compared and merged, and the role of modeling for understanding and communicating key ideas about systems. He received his PhD from Imperial College, London. Contact him at the Univ. of Toronto, Dept. of Computer Science, 40 St. George St., Toronto, Ontario, Canada M5S 2E4; sme@cs.toronto.edu.