Clarifying Goal Models

Ivan J. Jureta and Stéphane Faulkner

Information Management Research Unit, University of Namur, Belgium
iju@info.fundp.ac.be, sfaulkne@fundp.ac.be

Abstract. Representation and reasoning about information system (IS) requirements is facilitated with the use of goal models to describe the desired and undesired IS behaviors. One difficulty in goal modeling is arriving at a shared understanding of a goal model instance, mainly due to different backgrounds of the system stakeholders who participate in modeling, and the subsequent disparate use of terminology. Lack of shared understanding, or, in other words, the presence of multiple interpretations entails no guarantee that stakeholders' expectations expressed in the model instance will be appropriately understood during the subsequent steps of system development. Among the many potential causes of multiple interpretations, this paper focuses on a critical set of such causes, namely: ambiguity, overgenerality, synonymy, and vagueness of information represented in instances of goal modeling primitives. The "Goal Clarification Method" is suggested to guide the identification of unclear information and the subsequent clarification thereof.

1 Introduction

Requirements engineering (RE) is a structured approach to the assessment of the role that a future information system (IS) is to have within a relatively well-delimited human and/or automated environment. It involves the identification of goals to be achieved by the IS, their operationalization into implementable IS services and constraints, the identification of resources required to perform those services and the assignment of responsibilities for the resulting requirements to agents, such as humans, devices, and software. A usual starting point in RE is the elicitation of goals that the future IS will need to achieve once developed and deployed [20]. Goal modeling can be defined as the activity of representing and reasoning about IS goals using models, in which goals are related through relationships with other goals and/or other model elements, such as, e.g., actions that system agents are expected to execute, resources that they can use, or roles that they can occupy. With a number of currently established RE methods relying on goal models in the early stages of requirements analysis (e.g., [4, 6–8, 22]; see, [20, 17] for overviews), there seems to be a consensus that goal models are useful in RE.

If few system stakeholders (including, e.g., requirements engineers, users, and developers, among other) participate in the modeling of the future system's goals, arriving at a shared understanding of an instance of the goal model (henceforth, "goal diagram") is unlikely to involve significant difficulty. However, as the system under scrutiny gains in complexity—which tends to occur for most but toy systems, and is certainly true for IS employed in automating, e.g., government and health care services, air-traffic management, industrial production processes—the participation of many stakeholders, the inherent inability of individual human stakeholders to grasp the full extent of interactions and interdependencies between system components, to predict with detail and/or certainty the future conditions in which the system is expected to operate and the influence of such conditions on its functioning, makes the construction of the goal diagram an intricate task, critical for the success of the subsequent IS development activities.

As more stakeholders become involved in the modeling activity, differences in their individual backgrounds and the subsequent disparate use of terminology during goal modeling entail difficulty in arriving at a shared understanding of the goal diagram. Lack of shared understanding, or, in other words, the presence of multiple interpretations of the goal diagram, leaves open the possibility for misinterpretation thereof during the subsequent steps of system development. Since this entails no guarantee that the resulting system will satisfy the initial expectations, stakeholders cannot accept as appropriate a goal diagram that admits multiple interpretations. In this paper, such a goal diagram is referred to as an *unclear* goal diagram.

One possible approach to resolving problems of clarity in a goal diagram is to *propose techniques for identifying information that may lead to lack of clarity in the diagram and subsequently clarify it.* To facilitate the detection and clarification tasks, this paper proposes the "Goal Clarification Method (GCM)". The method focuses on four critical causes of unclear goal diagrams, namely, ambiguity, overgenerality, synonymy, and vagueness of information appearing in instances of goal modeling primitives. These four causes are commonly encountered in practice, so that addressing them is of apparent importance. By drawing on research in philosophy [11, 12, 14, 23, 29, 31, 32], linguistics [1, 18, 28], and artificial intelligence [2, 13], GCM proposes a set of practical techniques for identifying ambiguous, overgeneral, synonymous, and vague information in goal diagrams, and subsequently clarifying it to facilitate moving toward a shared understanding of the goal diagram among the stakeholders. In this paper, GCM is applied to the goal model from the Tropos RE methodology [4], mainly because it uses modeling concepts common in RE.

The problem of clarity of a goal diagram is first illustrated via an example (§2). The same example subsequently illustrates the features and use GCM (§3). Related work is discussed (§4), conclusions are drawn, and directions for future work are identified (§5).

2 Clarity Problem Illustrated

Integrated health and social care IS are being increasingly considered as a means for providing more effective health care to older people. In England, the electronic

Single Assessment Process (eSAP) system [24] is one such effort. Overall, the system aims at automating care processes, including, e.g., assessment procedures, collection and management of patient information, management of care plans, and the scheduling of appointments between the health professionals and the patients. Mouratidis and colleagues have studied security aspects of this system [25]. As the primary aim herein is not to discuss the complexity of this system, but to exemplify GCM, a fragment of eSAP requirements is considered.

As noted above, one of the goals of eSAP is to facilitate the scheduling of appointments between health care professionals and patients. The resulting IS ought to contain a component dealing with this issue. The goal model instance for the said component is considered herein. The meeting scheduling problem is selected for it is a classical and commonly discussed problem in goal modeling (e.g., [19, 7, 33]) and is easy to understand. This ensures that the remainder is readable and that the main features of the proposed method are salient.

In eSAP, the automated appointment scheduler should try to select a convenient date and location, such that most potential participants participate effectively. Each appointment participant should provide acceptable and unacceptable appointment dates based on his/her agenda. The scheduler will suggest a appointment date that falls in as many sets of acceptable dates as possible, and is not in unacceptable date sets. The potential participants will agree on a appointment date once an acceptable date is suggested by the scheduler. The appointment can be initiated by a health care professional or a patient.

A goal diagram for such a system component would be represented in Tropos as an instance of the $i*$ Strategic Rationale (SR) model. The $i*$ framework comprises, in addition to the SR model, the so-called "Strategic Dependency (SD)" model, which features a subset of the modeling primitives of the SR—the latter is therefore taken as the reference Tropos model in the remainder. An example SR diagram for the scheduler, taken as-is from [33], is reprinted in Fig.1. It shows roles such as *Automated Meeting Scheduler* and *Meeting Participant*, their interdependencies in the achievement of goals, the execution of tasks, and the use of resources, and their internal rationale when participating in the given IS. For example, the *Meeting Be Scheduled* goal of the *Meeting Initiator* can be achieved (represented via a means-ends link) by scheduling appointments in a certain way, consisting of (represented via task-decomposition links): obtaining availability dates from participants, finding a suitable date (and time) slot, proposing a appointment date, and obtaining agreement from the participants. Cloud-shaped elements designate softgoals which differ from goals in that there are no crisp criteria for their satisfaction. Softgoals are commonly used to represent nonfunctional requirements in a goal diagram. Actors are assigned to roles they can play: health care professionals and patients can both initiate and participate in appointments.

When confronted with the goal diagram in Fig.1, various stakeholders will ask different questions; for example:

- Do the 7 days allowed for scheduling a date include weekends or not?
- Can any date range be entered in the scheduler?

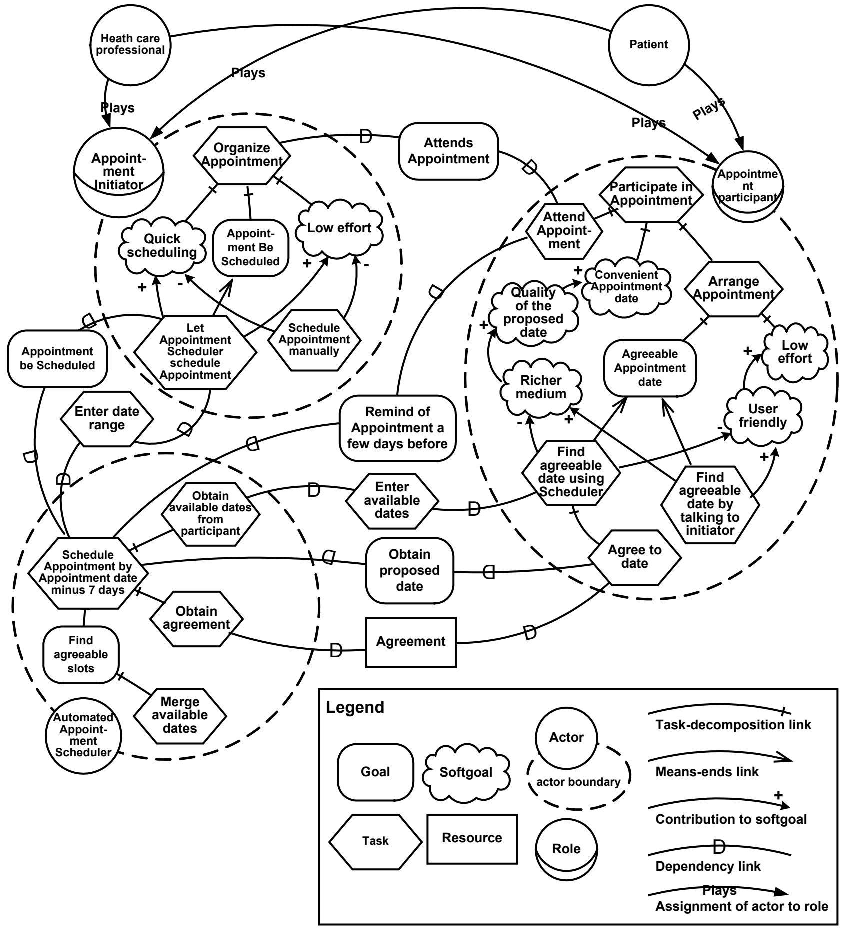

Fig. 1. An *i** Strategic Rationale diagram for the appointment scheduling problem in eSAP.

– Is the appointment participant role the same as participant role?
– How many is a few days before the appointment date?

The first question points to ambiguity of the task *Schedule appointment by appointment date minus 7 days*, as a stakeholder might understand these as either including or excluding weekends. Overgenerality is the problem behind the second question, which refers to the tasks *Enter date range* and *Enter available dates*—it remains unknown from the given model whether weekends are included in the referred days or not. The third question points to problematic use of terms "appointment participant" and "participant", that is, to their potential synonymy. Finally, the last question indicates vagueness, arising from the gradeable adjective "few"in the goal *Remind of appointment a few days before*.

If such questions are not answered within the diagram itself, additional information ought to be associated to clarify the diagram. Otherwise, there may be unnecessary reviews of the diagram, changes, or additional explaining. These activities require time and resources that could be used in other, more productive tasks. As goal modeling is an iterative activity, providing the clarifying information with the goal diagram results in better informed future iterations.

Although a stakeholder may perceive some information in a goal diagram as unclear, to act in order to clarify it, she ought to know how to detect a lack of clarity and to identify directions for the enhancement of unclear information. Moreover, as ambiguity differs from vagueness, synonymy differs from each of the latter, and so on, there can be various distinct techniques for detecting lack of clarity and subsequent clarification: i.e., clarity is a multi-faceted construct. In practical terms, in addition to perceiving a piece of information as unclear, any clarification method must enable the engineer to determine along which facets it is unclear, and clarify accordingly.

From there on, an active clarification process is conceptualized as a successive application of a set of basic *clarification techniques*, each being a transformation of information considered unclear into that perceived as clear by the stakeholder(s). The aim of the requirements engineer is to move on each dimension toward a direction assumed desirable: e.g., moving from "more" to "less" ambiguity, from more to less vagueness, etc. In addition to clarification techniques, clarity checking techniques are required to detect if some information is unclear along a particular facet. It follows that a way of helping modeling stakeholders in an active approach is to provide a rich catalog of clarity facets, to define each facet for easier identification, and to suggest clarity checking and clarification techniques to be applied when a facet is identified.

The catalog of four facets—ambiguity (§3.1), overgenerality (§3.2), synonymy (§3.3), and vagueness (§3.4)—introduced herein is incomplete and its extension is encouraged: it is impossible, knowing the extent of the literature on linguistic phenomena such as vagueness, to provide a full account herein. Practicality has therefore been the focus, with discussion and careful reuse of established results in linguistics, philosophy, and artificial intelligence. The proposed facet classification, along with the clarity checking and clarification techniques make no attempt at settling debates on the essence of concepts such as vagueness or ambiguity. Instead, the proposal draws on various literatures, taking as given some of the existing results while introducing techniques specialized for the problem at hand. The reader is reminded that this paper focuses only on ambiguity, overgenerality, synonymy, and vagueness of information represented in instances of goal modeling primitives. It thus does not address multiple interpretations that might result from, e.g., ambiguous syntax and/or semantics of a goal model. The available literature is followed in assuming the value of the chosen Tropos goal model for RE activities (e.g., [4, 10, 33]).

3 Goal Clarification Method

GCM integrates two components. The first is a clarification process, which organizes the identification of unclear information and subsequent clarification tasks. The second component is a set of detection and clarification techniques (see, §3.1–§3.4) employed within the clarification process. When constructing both the clarification process and the detection and clarification techniques, the aim was to remain lightweight and unintrusive with regards to the goal modeling framework. In this respect, GCM complements available RE methodologies without requiring any adaptation of the goal models these methodologies use. The clarification process involves the detection of unclear information in the goal diagram, the labeling of the unclear information, its clarification through in the thesaurus, and finally the change of the information in the goal diagram to avoid the identified clarity problem. The *thesaurus* complements the goal diagram. It is updated at each detection of unclear information. The unclear word or expression is carried over to the thesaurus, and information is provided therein to explain how the given information is clarified. The clarification decisions listed in the thesaurus are enforced throughout the system development project.

Technique 1. GCM *clarification process* proceeds in the following steps:

1. Choose a word or expression appearing in an instance of a modeling primitive (e.g., a particular goal, task, actor, or other) in the goal diagram and test it for lack of clarity along the four clarity facets. It is usually not necessary to apply detection techniques for all four facets—choice of the relevant facets for which to proceed to detection is not arbitrary, but guided by stakeholders' or engineer's questions about what a word or expression is intended to mean.
2. If the given word or expression proves unclear along one or more clarity facet (see, §3.1–3.4), label all the instance of the modeling primitives in which the unclear word or expression appears with C.Xn. X is either A (for ambiguity), G (for overgenerality), S (for synonymy), or V (for vagueness); n is a number used as a unique identifier for the given clarity issue.
3. In each of the labeled elements in the previous step, place brackets around the unclear word or expression, and label it accordingly (below, assume that the fragment of interest in an expression of the form: "... word(s) ..."):
   (a) If ambiguous, then "... $^{\mathsf{C.An}}$[word(s)]$^{\mathsf{C.An}}$ ...".
   (b) If overgeneral, then "... $^{\mathsf{C.Gn}}$[word(s)]$^{\mathsf{C.Gn}}$ ...".
   (c) If synonymous, then "... $^{\mathsf{C.Sn}}$[word(s)]$^{\mathsf{C.Sn}}$ ..." with the same label (i.e., C.Sn) applied to all words synonymous with "word(s)" within the diagram.
   (d) If vague, then "... $^{\mathsf{C.Vn}}$[word(s)]$^{\mathsf{C.Vn}}$ ...".
4. Carry the word(s) in brackets to the thesaurus, and clarify using a clarification technique for the corresponding clarity facet.
5. Enforce the result of clarification over the goal diagram and other artifacts (e.g., diagrams built in the steps following goal modeling) so that the agreed meaning is maintained across the development project. The goal diagram element is relabeled: the brackets produced in Step 3 above in the relevant modeling primitive instances are eliminated and the label produced in Step 2

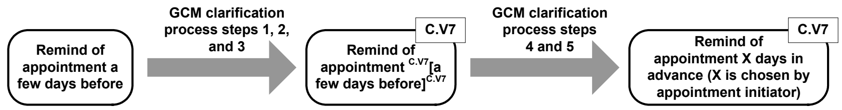

Fig. 2. Labeling a goal instance from the diagram in Fig.1 using Technique 1.

is maintained to relate (for traceability reasons) the content of the thesaurus produced in Step 4 with the relevant modeling primitive instance.

The above steps are repeated for any unclear information detected in the goal diagram. Consequently, a goal modeling primitive can carry more than one label (if the information it contains, e.g., an ambiguous expression which itself contains synonyms). In practice, Technique 1 is selectively applied when a participant in modeling consider a word or expression in the goal diagram can have multiple interpretations. Since it remains unrealistic to aim perfectly shared understanding, the amount of clarification to perform will be determined by the time and resource constraints placed on the modeling activity.

Example 1. To illustrate the labeling technique and the use of the thesaurus, consider the goal *Remind of appointment a few days before* from the goal diagram in Fig.1. Assume that a stakeholder identified the expression “a few days before” in this goal as vague. The result of the steps 1–3 of Technique 1 is shown on the left-hand side of Fig.2: the goal is labeled according to the indications given in Steps 2 and 3. Assume then that clarification resulted in avoiding vagueness by replacing the given expression with an indication of the number of days (selected by the appointment initiator) for reminding of the appointment in advance. Step 4 in Technique 1 resulted in adding the following fragment to the thesaurus:

(Vagueness, C.V7) **A few days before (the appointment date):** The appointment initiator should have a choice of automatically informing the participants 1, 2, ..., 7 days before the appointment date.

Step 5 of the clarification process consists of replacing the unclear with the clear interpretation in all modeling primitive instances in the goal diagram, in which the relevant unclear information appears.

The remainder introduces, discusses, and illustrates the techniques for detecting lack of clarity (used in Step 2 of Technique 1) and subsequently clarifying (Step 4 of Technique 1). Examples are given only in terms of thesaurus entries (without the graphical labeling as in Fig.2) for space considerations.

3.1 Ambiguity

While people seem capable in many cases to intuitively detect the occurrence of ambiguity, a useful criterion for doing so seems elusive (e.g., [29, 11]). The aim at present is to suggest a practical criterion for ambiguity that is acceptable in many cases. In addition, it is required that ambiguity is distinguished from the other three facets. For instance, separating ambiguity from overgenerality can be difficult if a word designating a class is considered ambiguous if the designated class is divisible onto subclasses. If this is accepted as a necessary condition for ambiguity, and knowing that a subclass contains instances of the subdivided class (see, §3.2), all ambiguous words would be considered as general. However, this is not appropriate: Hospers [14] has argued that a word is ambiguous neither because (i) the class of objects it refers to can be broken down into smaller classes or subclasses, nor (ii) when it may have many instances of use. Moreover, it is now accepted that ambiguity does not require generality [11].

Ambiguity is multiplicity of meaning of an expression, regardless of whether it originates from polysemy [28] of individual words or from multiplicity of structural analyses [11]. Polysemy occurs when a word, taken out of context, admits multiple meanings [28]—e.g., "run" as a verb has 29 distinct meanings and 125 sub-meanings according to the Webster's dictionary. Context of use tends to resolve problems of polysemy in communication [28], in that a word which is taken with other words in an expression loses most of its alternative senses. It is, however, not necessary to have a polysemous word in an expression for it to be ambiguous, as the expression "in hospitals, the police cannot shoot suspects with guns" illustrates. In this latter case, it is not polysemy that generates ambiguity, but the fact that the proposition admits different structural analyses [11]—each structural analysis leads roughly to one alternative reading of this proposition. The reader should note that matter is more elaborate than the above indicates: for instance, negation with "not" in English is often ambiguous for it leaves open whether it is the truth or the assertability of a proposition that is negated.

It is therefore difficult to propose a unique and general clarity check for ambiguity. For instance, it is useful to check if there is a state of affairs in which the ambiguous expression can both be affirmed and denied. For instance, in a state in which "armed police cannot shoot unarmed suspects on hospital grounds" and "armed police can shoot armed suspects on hospital grounds" both hold, the expression "in airports, the police cannot shoot suspects with guns" can both be affirmed true and false: in the last expression, the reader cannot know who carries guns (and is thus armed)—the police or the suspects, or both. A true reading of the given expression is, e.g., "in airports, the armed police cannot shoot unarmed suspects", while a false one is "in airports, the police cannot shoot armed suspects". Tautologies and contradictions cannot, however, be addressed by the given clarity check. Moreover, this check does not point to the source of ambiguity—finding it requires additional knowledge about the language being used, and thus varies across languages (e.g., [11]).

Although imperfect, structural analysis and word polisemy can be first used to detect potential for ambiguity. Once detected, the above clarity check is applied to verify if there is a state in which some alternative readings can be affirmed and others denied.

Definition 1. *An expression e is* ***ambiguous*** *if there are at least two of its alternative readings e_j and e_k such that one can be affirmed and the other denied* *within the domain knowledge (denoted $\mathcal{A}$, and assumed to contain anything the stakeholders can suggest during modeling) in which the expression is employed: ambiguous$(e) \not\models \bot$ iff:*

1. *There is a non-empty set E which contains alternative readings of e.*
2. *There are two readings e_j and e_k in E such that one gives rise to inconsistency given $\mathcal{A}$, while the other does not: $\exists e_j, e_k \in E$ s.t. $(\mathcal{A}, e_i) \models \bot$ and $(\mathcal{A}, e_j) \not\models \bot$.*

Technique 2. (Brute force ambiguity check) Identify as many alternative readings as feasible, and search for information relevant to the given domain knowledge which when combined with the readings is consistent with some but inconsistent with other. While it may seem cumbersome, this form of clarity checking is often feasible, for many alternative readings can be intuitively eliminated, whereas the remaining few can be subjected to scrutiny to relevant stakeholders through informal discussion.

Technique 3. (Ambiguity resolution in the thesaurus) The ambiguous element is labeled and introduced in the thesaurus with a list of alternative readings elicited by the engineer. Resolving ambiguity amounts to choosing one of the available readings and enforcing it throughout other fragments of the requirements specification through a thesaurus $\mathcal{D}$, where the the ambiguous word is carried over and accompanied with its chosen reading.

Example 2. Consider the the task *Schedule appointment by appointment date minus* ${}^{C.Aj}[$*7* ${}^{C.Ai}[$*days*$]^{C.Ai}]^{C.Aj}$ in Fig.1. One problem here (C.Ai) is that speaking of duration in terms of days engenders ambiguity by polysemy, as different and contradictory interpretations are available for "day"—namely, 24h or working day. Because this seems to be a case of ambiguity from polysemy, the Techn.2 above applies. Furthermore, 7 days can be interpreted to include only working days, or to include also weekends (C.Aj). The identified alternative readings are written down in the thesaurus:

(Ambiguity, C.Ai) **day:**
1. A day is to be understood as 24h; •
2. A day is to be understood as a working day, from 8am to 6pm;

(Ambiguity, C.Aj) **7 days:**
1. 7 days includes a weekend;
2. 7 days are counted by excluding sunday; •

The ambiguous words and expressions are labeled according to Technique 1. For each ambiguous element, alternative readings are given in the thesaurus, and the chosen reading is decorated with •. Choosing randomly one of the interpretations is inappropriate, for the probable impact an inadequate choice would have—e.g., scheduling problems in a hospital. Choice is made by the stakeholder clarifying the information.

Technique 4. (Structural analysis ambiguity check) In English, knowing that a "noun phrase can have complementary propositional phrases" and a "verb phrase can contain just a verb and propositional phrase" [11] allows showing that "in airports, the police cannot shoot suspects with guns" admits two structural analyses: 1) in airports, the police [[cannot shoot]$^{\text{verb}}$ [suspects [with guns]$^{\text{prop. phrase}}$]$^{\text{noun phrase}}$]$^{\text{verb phrase}}$; and 2) in airports, the police [cannot shoot [suspects [with guns]$^{\text{prop. phrase}}$]$^{\text{noun phrase}}$]$^{\text{verb phrase}}$.

Example 3. Assume that electronic agendas are available and that they can communicate with the automated appointment scheduler. Assume further that there is a goal "$^{\text{C.Ak}}$[Provide their availability information]$^{\text{C.Ak}}$" assigned to the electronic agenda software. This expression is ambiguous as the following two readings can be identified using structural analysis:

(Ambiguity, C.Ak) **Provide their availability information**

1. Users' agendas provide the availability of the users; •
2. Agendas that belong to the users provide information on whether they (the agendas) are available for fulfilling a request;

3.2 Overgenerality

Overgenerality is introduced to characterize a relationship present between expressions such as "allow all patients to initiate urgent appointments" and "allow patients with health status below Y to initiate urgent appointments", where the first is general with regards to the second since the the second involves a subclass of the first.

Definition 2. *A word or expression g is* ***overgeneral*** *if domain knowledge $\mathcal{A}$ contains an expression about a particular p of g (e.g., above, g can stand for the "all patients" and p for "patients with health status below Y"), whereby the expression with p contradicts an expression with g.*[1] *That is: overgeneral$(g) \not\models \perp$ iff:*

1. *$F(a) \in \mathcal{A}$, that is, there is an affirmable expression $F(a)$ in domain knowledge, where a occurs once in $F(a)$ and can be replaced with the particular p or the general g. By convention, $F(a)/g$ denotes the affirmable expression obtained by replacing a with g.*
2. *The expressions $F(a)/g$ and $F(a)/p$ are inconsistent given the domain knowledge in $\mathcal{A}$: $\mathcal{A}, (F(a)/g), (F(a)/p) \models \perp$.*
3. *general(p, g) holds, i.e., it must be verified that g is a general of p. g is general with respect to p iff for an affirmable expression $F(a)$, the expression $\neg(F(a)/g) \wedge (F(a)/p)$ is a contradiction.*[2]*;*

Technique 5. (Criteria-based resolution of overgenerality) Define criteria for distinguishing entities that fall in, from those that do not fall in the set designated by the expression $F(a)$ in which substitution for g and p gives inconsistency. The entry in the thesaurus explicates the criteria used to discriminate what particulars can be used as substitutes for a in $F(a)$.

[1] As a convention, $F(a) \in \mathcal{A}$ in the definition indicates that an expression $F(a)$ containing a word or expression of interest a is in domain knowledge $\mathcal{A}$.

[2] The given clarity check for generality was suggested in [23] and defended against alternative checks in [11]—it is taken as appropriate for the purpose herein.

Example 4. Consider the task *Enter* $^{C.Gi}$[*available dates*]$^{C.Gi}$ in Fig.1, "Available dates" might include sundays, which might be problematic. Using Technique 5 on the expression "A appointment participant can enter any available date" (this expression is evident by reading the diagram in Fig.1) and knowing from domain knowledge that a appointment cannot be scheduled on sundays, results in the following entry in the thesaurus:

(Overgenerality, C.Gi) **Available dates:** Any date excluding sundays.

3.3 Synonymy

Synonymy treats the use of common terminology in an inconsistent manner—it highlights the use of syntactically different terms, which are given the same semantics.

Definition 3. *Words w_1 and w_2 are synonymous within $\mathcal{A}$ if they are can be used interchangeably in $\mathcal{A}$: synonymous(w_1, w_2) holds iff:*

1. *Both words appear in $\mathcal{A}$: $w_1, w_2 \in \mathcal{A}$;*
2. *w_1 appears in expression $F_1(w_1)$ and w_2 appears in expression $F_2(w_2)$, and interchanging the two words within the expressions maintains the meaning of the new expressions equivalent with original ones. That is, $F_1(w_1)/w_2$ has the same meaning as $F_1(w_1)$, and $F_2(w_2)/w_1$ has the same meaning as $F_2(w_2)$.*

Technique 6. (Synonymy by interchangeability check) Intuitively, words used within similar expressions are candidates for synonymy. Clarity checking for synonymy proceeds by replacing a word in an expression with another word within one or more expressions in which the first appears. For instance, if $F(a)$ and $G(b)$ are two expressions appearing in domain knowledge, and the requirements engineer believes that both expressions refer to the same properties or behaviors of the IS, then if the engineer understands $F(a)/b$ (i.e., the expression $F(a)$ in which each occurrence of the word a is replaced with the word b) in the same way as $F(a)$, and $G(b)/a$ in the same way as $G(b)$, then a and b are synonymous for the given expressions.

Example 5. One potential synonymy in Fig.1 is for "$^{C.Si}$[appointment participant]$^{C.Si}$" and "$^{C.Si}$[participant]$^{C.Si}$". Applying Technique 6 on expressions obtained by reading the goal diagram (e.g., "Obtain available dates from participants") results in identifying the synonymy and resolving it with the following:

(Synonymy, C.Si) **Meeting participant = participant:** A participant is any the person who has confirmed attendance and attends the appointment, except the appointment initiator.

3.4 Vagueness

The locus of vagueness in many vague expressions, just as in “quick scheduling” in Fig.1, is the presence of a predicate headed by a *gradeable adjective*, (above: “quick”). Such predicate designates a property of having a degree of speed that is at least as great as some standard of comparison of speed, that itself is not part of the meaning of “effective” but is determined by the context in which the said adjective is used. From there on, truth assignment can change as the standard changes (and as context changes). This matter is, however, more intricate, as setting a standard of comparison seems to eliminate borderline cases altogether (and subsequently the Sorites paradox). While attractive, this seems removed from reality: assuming e.g., that 1 day is a mean speed of scheduling (and, say, the standard for comparison), then some stakeholders may still refuse to accept that the given speed of 1 day is quick, for they have witnessed situations in which speed was significantly lower (i.e., the variance in the sample from which the mean was computed can be considered high). A solution to this is suggested in [12], where for a borderline case to be described truthfully with the given vague predicate, it is necessary for it not to exceed the standard without exceeding it by a *significant* amount—in practice, two very similar cases along the scale of measurement associated to the vague predicate will be taken same (i.e., will carry the same truth valuation) if the cost of discriminating between them outweighs the benefits of doing so. They will count as the same for the given purposes [12]. Unfortunately, it seems that how much significant it is, is itself vague—the only realistic solution then remains seeking stakeholders’ agreement on the standard and its enforcement throughout a chosen context.

These established positions on gradable adjectives (e.g., [18, 12][3]) already provide relevant practical indications for the problem at hand.

Definition 4. *Adjective e is gradeable and assumed giving rise to* ***vagueness*** *within the expression in which it appears, if the following conditions are met:*

1. *1st assumption on the gradeable adjective: The adjective maps its arguments onto abstract representations of measurement, or degrees.*
2. *2nd assumption on the gradeable adjective: The set of degrees totally ordered with respect to some dimension (e.g., cost, size, etc.) constitute a scale. The 1st and 2nd assumption together give an ontology for gradeable adjectives which provides indications on what adjectives to consider when clarity checking for vagueness.*
3. *The adjective itself does not entail a standard for comparison, so that such a standard varies with context.*
4. *Presence of borderline cases: It should be possible to identify borderline cases of application of the given adjective.*

[3] The reader is reminded that the present work is not one focused on linguistics, so that no specific references will be given beyond overview and extensive discussions from the aforementioned field. For instance, no minority positions are mentioned herein. For details, the reader will refer to the works cited within the given references.

5. *The truth of the expression in which the adjective appears should vary with the change of standard which is accepted to distinguish when the adjective applies from when it does not.*
6. *The predicate generated by the adjective can be used in lines of reasoning that follow the one taken in the Sorites paradox (see above).*

Technique 7. (Vagueness check) Use the items in Def.4 as a checklist to identify gradeable adjectives that may give rise to vagueness in a given expression.

Technique 8. (Resolving vagueness from gradeable adjectives) For gradeable adjectives that generate vagueness, the desired solution is to specify the standard of comparison, and to treat borderline cases individually.

Example 6. Applying the Technique 7 to the goal *Remind of appointment a few days before* in Fig.1 indicates that "few" is an adjective to which all of the conditions enumerated in Def.4 apply. Consequently, "few" generates vagueness in the given goal. Solution is obtained by applying Technique 8, and is shown in Example 2: a standard of comparison has been established by defining the precise meaning of "few" in terms of particular values it can take.

4 Related Work

Chantree and colleagues [5] focus on ambiguities arising from the presence of coordination conjunctions *and*, *or*, and *and/or* in natural language requirements. Text appearing in instances of modeling primitives in goal models normally does not contain coordination conjunctions since these are avoided with goal modeling constructs such as AND/OR refinement or decomposition (see, e.g., [7, 33, 4]). Their technique can be combined with GCM: while GCM focuses on goal diagram content, their approach can be applied before goal modeling, on information represented in textual documentation from which the engineer is expected to derive goal diagram elements. Kamsties and colleagues [16] focus on ambiguity natural requirements and suggest techniques for detecting ambiguity without discussing clarification. Also, their approach does not consider ambiguity detectable by structural analysis. For instance, Berry and Kamsties [3] concentrate on ambiguity that may arise from the use of plural in natural language requirements. Fuzzy logic has been suggested for dealing with vague requirements (e.g., [21] among other—a more elaborate discussion is given in [15]). Somewhat close to our effort here is [27], where three dimensions along which any RE project or framework can be described, are identified. It is suggested there that the overall aim in any RE project is to move from informal to formal representations of requirements (along the *representation* dimension), from individual to shared views (the *agreement* dimension), and from an opaque understanding of the system to its complete specification (the *specification* dimension). In such a framework, the specification dimension can be enriched by subdividing it onto, among other, the four clarity facets identified herein. Since clarification is aimed at arriving at a shared understanding of requirements, it is indirectly related to research on viewpoints in RE ([26] and later—e.g., [30]).

Compared to related efforts, GCM focuses on lack of clarity in goal models, proposes an integrated approach to four causes of unclear goal diagrams, and gives general techniques both for identifying and resolving these. GCM can be combined with the cited proposals in that some of these fit into GCM as additional more specific techniques (e.g., Chantree and colleagues' heuristics for detecting ambiguity from coordination conjunctions [5] is a particular case of ambiguity—see, §3.1; same applies for Berry and Kamsties' treatment of ambiguity from plural) applicable not directly on goal diagrams, but on information used to build goal diagrams. GCM can be seen as complementing viewpoint-based approaches applied to goal models—for instance, in Sabetzadeh and Easterbrook's view merging framework [30], GCM can be used to assist modelers in resolving disagreement in view merges, provided disagreement arises from lack of clarity (and not from, e.g., difference of opinion).

5 Conclusions and Future Work

This paper introduces and illustrates the Goal Clarification Method for avoiding ambiguity, overgenerality, synonymy, and vagueness of information appearing in instances of modeling primitives in goal diagrams for RE. GCM complements available approaches for moving toward shared understanding of information represented in RE goal diagrams.

GCM requires the writing and updating of the thesaurus in order to enforce clarification choices throughout the goal diagram. A tool is needed to facilitate these tasks. When GCM was applied, a cross-referenced document was created and maintained with a word processor. A more advanced tool would automatically verify internal thesaurus consistency and automate the enforcement of clarification decisions. In practice, clarification techniques proved particularly useful in raising awareness of clarity issues. This lead modeling participants to rephrase pieces of information in the goal diagram and the thesaurus so as to check whether they understand them appropriately. When rephrased information appeared different than the original, clarification techniques were applied to check what clarity issue is present and subsequently resolve it. Clarification is time consuming: one potential direction for improvement lies in linking the thesaurus to lexicons such as, e.g., WordNet [9] in order to automatically generate lists of potential synonyms and facilitate the writing of definitions, which is useful in dealing with overgenerality. Also, since clarification assumes the participation of several modelers, managing the modeling process is another difficulty, which remains beyond the scope of GCM.

Current work focuses on the identification of additional clarity facets and the exploration of ways to at least partially automate GCM techniques.

References

1. Barker C (2006) Vagueness. In Brown K (ed.). *Encyclopedia of Language and Linguistics.* 2nd Ed., Elsevier.

2. Bennett B (1998) Modal Semantics for Knowledge Dealing with Vague Concepts. *Proc. Int. Conf. on Principles of Knowl. Represent. and Reasoning.*
3. Berry D, Kamsties E (2005) The syntactically dangerous all and plural in specifications. *IEEE Softw.* 22 1:55-57.
4. Castro J, Kolp M, Mylopoulos J (2002) Towards requirements-driven information systems engineering: the Tropos project. *Info. Sys.* 27 6:365–389.
5. Chantree F, Nuseibeh B, de Roeck A, Willis A (2006) Identifying Nocuous Ambiguities in Natural Language Requirements. *Proc. Int. Conf. Req. Eng.*
6. Chung L, Nixon B A, Yu E, Mylopoulos J (2000) *Non-Functional Requirements in Software Engineering.* Kluwer.
7. Dardenne A, van Lamsweerde A, Fickas S (1993) Goal-directed requirements acquisition. *Sci. Comp. Programming* 20:3–50.
8. Donzelli P (2004) A goal-driven and agent-based requirements engineering framework. *Req. Eng.* 9 1:16–39.
9. Fellbaum C (1998) *WordNet: A Lexical Reference System and Its Application.* Cambridge, Mass., MIT Press.
10. Fuxman A, Liu L, Mylopoulos J, Pistore M, Roveri M, Traverso P (2004) Specifying and Analyzing Early Requirements in Tropos. *Req. Eng.* 9 2:132–150.
11. Gillon B S (1990) Ambiguity, Generality, and Indeterminacy: Tests and Definitions. *Synthese* 85:391–416.
12. Graff D (2000) Shifting Sands: An Interest-Relative Theory of Vagueness. *Philosophical Topics* 20:45–81.
13. Halpern J Y (2004) Intransitivity and Vagueness. *Proc. Int. Conf. on Princip. of Knowl. Represent. and Reasoning.*
14. Hospers J (1953) *An Introduction to Philosophical Analysis.* Prentice-Hall.
15. Jureta I J, Faulkner S, Schobbens P-Y (2006) A More Expressive Softgoal Conceptualization for Quality Requirements Analysis. *Proc. Int. Conf. on Conceptual Modeling.*
16. Kamsties E, Berry D, Peach B (2001) Detecting ambiguities in requirements documents using inspections. *Proc. Worksh. Inspection in Softw. Eng.*
17. Kavakli P, Loucopoulos P (2005) Goal Modeling in Requirements Engineering: Analysis and Critique of Current Methods. In Krogstie J, Halpin T, Siau K (eds.) *Information Modeling Methods and Methodologies (Adv. Topics Database Res.).* IDEA.
18. Kennedy C (2006) Vagueness and Grammar: The Semantics of Relative and Absolute Gradable Predicates. *Linguistics and Philosophy.* In press.
19. van Lamsweerde A, Darimont R, Massonet Ph (1992) *The Meeting Scheduler Problem: Preliminary Definition.* Université catholique de Louvain.
20. van Lamsweerde A (2001) Goal-Oriented Requirements Engineering: A Guided Tour. *Proc. Int. Conf. Req. Eng. (RE'01).*
21. Liu X F, Yen J (1996) An Analytic Framework for Specifying and Analyzing Imprecise Requirements. *Proc. Int. Conf. Softw. Eng.*
22. Liu L, Yu E (2004) Designing information systems in social context: a goal and scenario modeling approach. *Info. Syst.* 29:187–203.
23. Margalit A (1983) A Review of Scheffler (1979). *J. Philosophy* 80:129–137.
24. Mouratidis H, Philp I, Manson G (2002) Analysis and design of eSAP: an integrated health and social care information system. *Proc. Int. Symp. on Health Information Managements Research.*
25. Mouratidis H, Giorgini P, Manson G (2005) When security meets software engineering: a case of modeling secure information systems. *Info. Syst.* ???
26. Nuseibeh B, Kramer J, Finkelstein A (1993) Expressing the Relationships Between Multiple Views in Requirements Specification. *Proc. Int. Conf. Softw. Eng.*

27. Pohl K (1993) The Three Dimensions of Requirements Engineering. *Proc. Int. Conf. on Adv. Info. Syst. Eng.*
28. Ravin Y, Leacock C (eds.) (2000) *Polysemy: Theoretical and Computational Approaches*. Oxford Univ. Press.
29. Richman R J (1959) Ambiguity and Intuition. *Mind, New Series* 68 269:87–92.
30. Sabetzadeh M, Easterbrook S (2006) View Merging in the Presence of Incompleteness and Inconsistency. *Req. Eng. J* 11 3.
31. Sorensen R (2003) Vagueness. In Zalta E N (ed.). *The Stanford Encyclopedia of Philosophy.* Stanford University.
32. Williamson T (1994) *Vagueness*. Routledge.
33. Yu E (1997) Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. *Proc. Int. Symp. Req. Eng.*