Modeling and Tracing Stakeholders' Goals across Notations using RE-Tools

Sam Supakkul[1], Lawrence Chung[2], RJ Macasaet[3,4], Manuel Noguera[4], María Luisa Rodríguez[4], and José Luis Garrido[4]

[1] Sabre Inc., Southlake, Texas, USA
[2] The University of Texas at Dallas, Richardson, Texas, USA
[3] Pentathlon Systems Resources Inc., Manila, Philippines
[4] University of Granada, Granada, Spain
sam.supakkul@sabre.com, chung@utdallas.edu,
rjmacasaet@pentathlonsystems.com, {mnoguera,mlra,jgarrido}@ugr.es

Abstract. The ability to represent stakeholders' goals and their operationalizations, through tasks, resources, system requirements and specifications, helps better ensure that the projected system meets its intended goals. To offer such an ability, various notations have been developed, but somewhat independently of each other, each for its own concepts and with its own tool. As a result, it is difficult to establish end-to-end traceability among the various concepts. This paper presents RE-Tools - a toolkit towards an integrated modeling of such concepts, using *i**, the NFR framework, KAOS and Problem Frames. RE-Tools uses a meta-model approach to representing and integrating the notations, which includes shared, generalized meta-classes and cross-notational goal achievement contribution links. This approach is intended to allow for a uniform application of the Label Propagation Procedure in determining goal achievement across the supported notations, using either the open or closed world assumption.

1 General Information

Name	RE-Tools[1]
Version	2.0
Information	http://www.utdallas.edu/~supakkul/tools/RE-Tools
Frameworks	*i** [2], NFR Framework [3], KAOS [4], Problem Frames [5], UML
Purposes	Modeling and reasoning support, foundation for other tools [6]
Features	Integrated modeling environment, automated and integrated Label Propagation Procedure using open or closed world assumption, UML Profile-based meta-model, navigatable in-memory models, interchangeable XMI/XML file format, extension via plug-in and APIs, open-source, Windows OS supported.
Status	Publicly available since 2009, with around 1,300 downloads from around 65 countries to date, for use in teaching, research, and industrial practice, including Australia, Brazil, Canada, China, France, Italy, the Philippines, Spain, UK, and the US
Industry use	Internal use at Sabre and Pentathlon Systems Resources

2 Integrated Modeling and Tracing Goal Achievement

RE-Tools previously supported independent modeling of the supported notations and limited integration [1]. The tool has recently been enhanced to support a more comprehensive integration across multiple notations. Fig. 1 illustrates the key features of the enhancements, with the corresponding meta-classes shown in blue. Suppose the *Record ambulance location* Requirement in Problem Frames is determined to be satisficed (denoted by a check mark) by the system. Using the Label Propagation Procedure [7], the satisficing is propagated across the Make(++) contribution to derive that the *Ambulance location recorded* Goal in KAOS is also satisficed. The propagation is then repeatedly applied across the next contribution links in a cascading fashion upward the graph until the propagation cannot proceed any further because the needed information along the various paths is not available (e.g., the achievement of some sub-goals of an AND decomposition is undefined) or it has reached and evaluated the top-level Agents' goals (e.g., *Timeliness[Ambulance arrived at scene]* Softgoal).

To support the integrated modeling and reasoning, achievement contributions are represented as links between a parent Proposition and its offspring Proposition(s). Proposition is captured as a generalized meta-class in the meta-model, which is shared by the concepts in the supported notations (e.g., Hardgoal, Softgoal, BehavioralGoal, Task, Resource, Requirement). By applying the Label Propagation Procedure at the shared generalized Proposition meta-class level, RE-Tools performs reasoning about goal achievement generically and uniformly across the different notations.

3 Discussion and Future Work

The notion of (hard) goal achievement in *i** and KAOS are generally treated in an absolute sense (*satisfied* or (absolutely) *denied*), while the notion of Softgoal achievement in the NFR Framework is "good enough", reflecting the more subjective, interacting, relative nature (*satisficed*, *weakly satisficed*, *denied*, *weakly denied*). To support a more general and uniform label propagation, RE-Tools adopts and applies the weaker notion (satisficing) with an assumption that *satisficed* and (soft) *denied* could at least roughly be translated to the notions of *satisfied* and (absolutely) *denied* in some cases. However, additional research is required to better understand this approach, compare it with other integration approaches, as well as understand how to use multiple notations together.

Additionally, RE-Tools currently requires human intervention to trigger the automatic cascading label propagation process by first manually labelling the goal achievement of leaf-level nodes, which could be inaccurate or time-consuming. We are investigating ways to help alleviate these problems in some situations by integrating RE-Tools with other development notations and tools (e.g., BMPN [8] and URN [9]), and simulation tools [10] to automatically obtain the actual or simulated goal achievement of the leaf-level nodes. Furthermore, from the requirements modeling perspective, the tool may need to be extended to support other ontological concepts, such as quality constraints and domain assumptions.

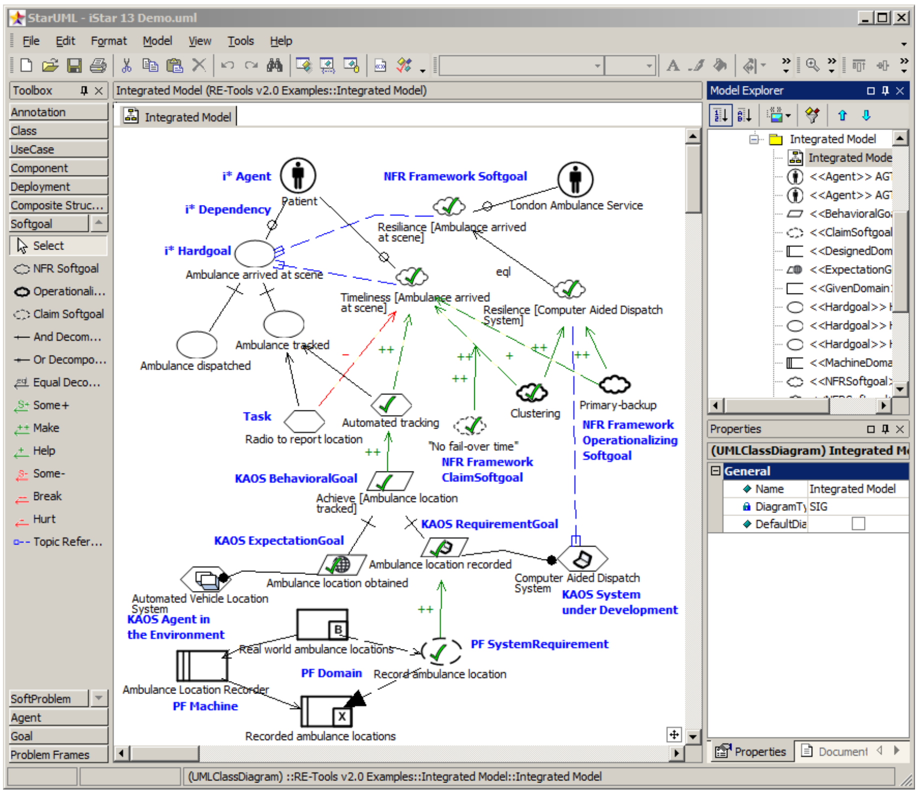

Fig. 1. A Sample Screenshot of an Integrated Model with Goal Achievement Reasoning

References

1. Supakkul, S., Chung, L.: The RE-Tools: A multi-notational requirements modeling toolkit. In: Requirements Engineering Conference (RE), 2012 20th IEEE Intl. (2012) 333 –334
2. Yu, E., Giorgini, P., Maiden, N., Mylopoulos, J.: Social Modeling for Requirements Engineering. The MIT Press (2011)
3. Chung, L., Leite, J.: On Non-Functional requirements in software engineering. In Borgida, A.T., Chaudhri, V.K., Giorgini, P., S.Yu, E., eds.: Conceptual Modeling: Foundations and Applications. (2009) 363–379
4. van Lamsweerde, A.: Requirements Engineering: From System Goals to UML Models to Software Specifications. Wiley (2009)
5. Jackson, M.: Problem Frames: Analyzing and Structuring Software Development Problems. Addison-Wesley (2000)
6. Supakkul, S., Hill, T., Chung, L., Tun, T., Leite, J.: An NFR pattern approach to dealing with NFRs. In: 18th IEEE Intl. Requirements Engineering Conf. (2010) 179–188
7. Chung, L., Nixon, B.A., Yu, E., Mylopoulos, J.: Non-Functional Requirements in Software Engineering. Kluwer Academic Publishers (2000)
8. Macasaet, R., Noguera, M., Rodrguez, M., Garrido, J., Supakkul, S., Chung, L.: A Requirements-Based Approach for Representing Micro-business Patterns. In: 7th IEEE Int'l Conference on Research Challenges in Information Science, IEEE (2013)
9. Amyot, D., Mussbacher, G.: URN: Towards a new standard for the visual description of requirements. In: Telecommunications and beyond: The BroaderApplicability of SDL and MSC. Springer (2003) 21–37
10. Hill, T., Supakkul, S., Chung, L.: Confirming and Reconfirming Architectural Decisions on Scalability: A Goal-Driven Simulation Approach. In: Proc. 8th Int'l. Workshop on System/Software Architectures, Springer (2009) 327–336