Using *i** Meta Modeling for Verifying *i** Models

Antonio de Padua Albuquerque Oliveira [1, 2], Julio Cesar Sampaio do Prado Leite [2], Luiz Marcio Cysneiros[3]

[1] Universidade do Estado do Rio de Janeiro – UERJ
Rua São Francisco Xavier, 524 - 6 andar - Maracanã - Rio de Janeiro, Brazil
[2] Pontificia Universidade Catolica do Rio de Janeiro – PUC-Rio
Departamento de Informatica, Rua Marques de Sao Vicente 225 – Rio de Janeiro, Brazil
[3]York University – School of Information Technology
4700 Keele St. – Toronto, Canada
padua.uerj@gmail.com - www.inf.puc-rio.br/~julio - cysneiro@yorku.ca

Abstract. The i* Framework has been regarded as a suitable organizational modeling approach for representing early requirements of complex software systems. Intentionality in organizational context is the aim of i* Framework. We believe that a general lack of awareness about the i* language is the main reason for some authors mistakes including the lack of focus on intentionality. Aiming to help changing this scenario we made an exercise of modeling i* modeling using only i* concepts. Considering that building any diagram is more difficult than reading it we propose to use the i* meta model as basis for a series of check-list based questions. Based on the meta-model these questions work as a check-list for building an i* model, or if used after model creation as a basis for check-list reading as per Fagan's inspection. We believe our contribution relies on providing a systematic and well founded way of improving i* models quality.

Keywords: meta-modeling, Goal Oriented Requirements Engineering (GORE), early requirements, verification.

1 Introduction

The i* Framework uses two models: the Strategic Dependencies Model (the SD Model) and the Strategic Rationale Model (the SR Model). Furthermore several simple elements are used by these two models in order to represent social actors and dependency relationships among actors inside an organization. We have modeled i* in i* using the same perspective adopted by the i* Framework: "the intentionality perspective". Intentionality means to represent motivations and desires of actors [2].

In this way, first we considered the SD model as the organization and therefore actors (agents occupying positions and playing roles) are the elements (the actors and the four kinds of dependencies) that act in an SD model. Second, by the same token, we consider the SR model as the organization and i* elements were considered the actors (dependencies, all kinds of means-ends links and task-decomposition) that act in the SR model. Applying this abstraction exercise we believe that the intentionality of all elements and their relationships are exposed in a concise model.

This concise model is the basis for deriving the SD and SR check-lists. In this abstract, for space considerations we have shown just the SA diagrams, the SD and SR diagrams may be seen in a technical report [4]. However we have abstracted from these two meta-diagrams their key-points, as to better explain the check-list derivation.

2 Objectives of the research

Making check lists based on i star framework concepts

i* Modeling Framework's concepts and ideas are the basis for our meta-model which provides a clear statement: "*goals are states of affairs that an actor plans to achieve*" [2]; they are not activities or functions. Because there are some misuses of this definition we strongly recommend the adoption of the following standardization used by [3]: (i) **goal** → object + BE + verb in passive voice; (ii) **softgoal** → quality attribute + [object or task as topic]; (iii) **task** → verb in infinitive + object; and (iv) **resource** → name of the object.

In the next section diagrammatic results either by diagram or by key points are explored after enforcing this rule in representing the i* language.

3 Scientific contributions

Verifying an SD Model

The aim of the SD model is to represent strategic dependencies among actors. Using an abstraction we considered that all possible elements from i*, e.g. tasks, goals, positions, roles, and so on will be "actors" in our abstraction. Applying this abstraction we consider "actors" in SD model as being agents, which occupy positions and play roles.

Consequently, as we show in Figure 1, elements (links and nodes) are mapped as agents. Figure 1 is an SA Diagram, this diagram was proposed by Leite et al. [1] as way of structuring the i* concepts of actor, agent, role and position [2]. One agent, in the SD model, can occupy only two positions; either a position of an actor or a position of a dependency, because those "actors" can be classified in two kinds: "actors" which represent actors in the strict sense and "actors" which represent dependencies between actors, as per our abstraction. While occupying an actor's position an element can cover two kinds of roles: *dependee* or *depender*, roles are specializations of actor. On the other hand, while occupying a dependency position an element can cover four kinds of roles: *Goal Dependency*, *Resource Dependency*, T*ask Dependency*, or S*oftgoal Dependency.* So, each one of these four roles is a *dependum* as in [2].

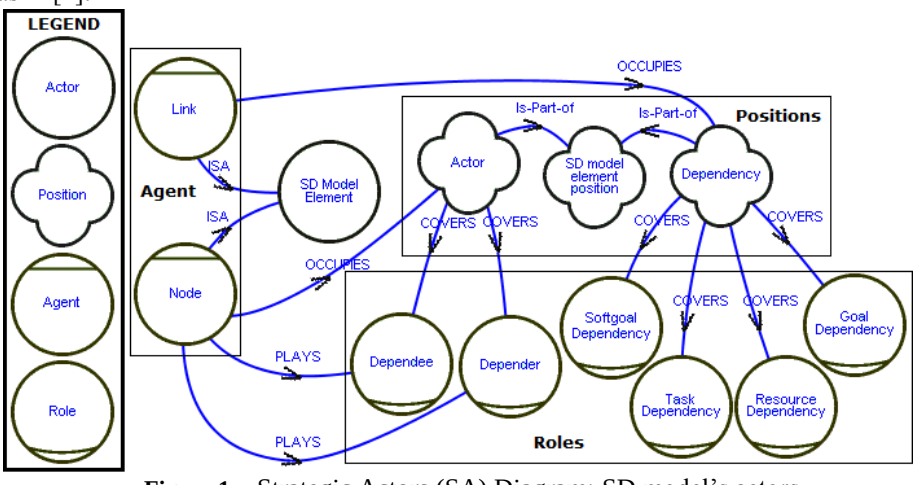

Figure 1 – Strategic Actors (SA) Diagram: SD model's actors

Building an SD Model for the SD Model

Continuing our abstraction exercise for the i* SD model, we created an SD model [4], from which we extracted three key points in order to represent the four types of dependency relationships (see the four roles of dependency (dependum) in Figure 1).

Key points: (1) Strategic dependency means that there is always a depender's goal to be achieved, (2) dependee has a commitment with a depender - Yu's thesis p. 12 [2], and (3) depender believes that dependee is able to carry out the commitment.

SD model Check List

- Is each element in the SD model either actor or dependency?
- For each dependency: Is one actor the depender and the other the dependee?
- For each goal dependency:
  - Does goal dependency obey the goal standardization?
  - Can the dependee achieve the goal the depender wants to?
  - Why the dependee is going to achieve the goal the depender wants to?
- For each softgoal dependency:
  - Does softgoal dependency obey the standardization?
  - Can the dependee achieve the softgoal the depender wants to?
  - Why the dependee is going to achieve the softgoal depender wants to?
- For each task dependency:
  - Does task dependency obey the goal standardization?
  - Can the dependee perform the task the depender wants to?
  - Why the dependee is going to perform the task the depender wants to?
- For each resource dependency:
  - Does resource dependency obey the standardization?
  - Can the dependee provide the resource the depender wants to?
  - Why the dependee is going to provide the resource depender wants to?

Verifying an SR Model

The aim of the SR model is to represent strategic rationale inside the actors' boundary. Applying our abstraction, the SR model has "actors" which can appear in the SD model and has other actors that are peculiar to SR model.

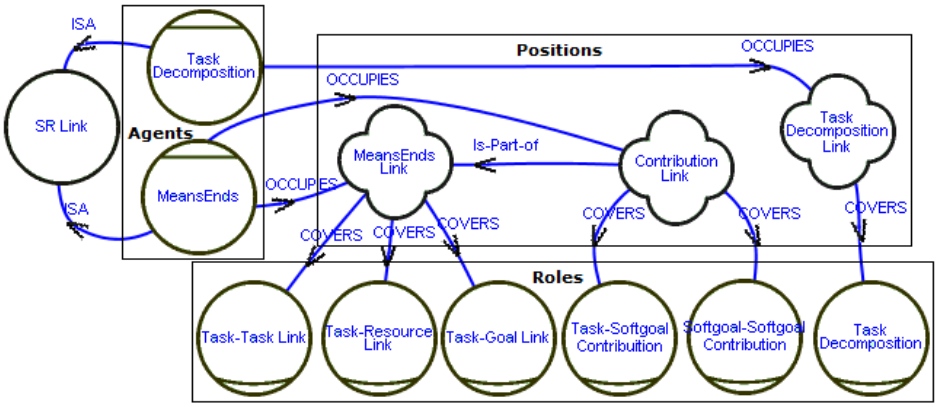

Figure 2 – Strategic Actors (SA) Diagram: SR model's actors

Figure 2 shows SR Links. SR models use two kinds of links, which were represented being agents: *Task Decomposition* and *MeansEnds*. We can observe that

the agent *Task Decomposition* can occupy only one single position in the SR model (Task Decomposition Link) which covers only one single role (Task Decomposition), but on the other hand the agent MeansEnds can occupy two positions: *MeansEnds Link* or *Contribution Link* which are considered part of *MeansEnds Link*. The position *MeansEnds Link* covers three roles: *Task-Task Link*, *Task-Resouce Link* and *Task-Goal Link*. The position *Contribution Link* covers two roles: *Softgoal-Softgoal Contribution* and *Task-Softgoal Contribution*.

Key points: (1) there are two constructs to represent the rationale inside actor's boundary: means-ends and task-decomposition, (2) there is only one way to represent task decomposition, (3) there are five kinds of means-ends link, and (d) means-ends links concerning a softgoal (always as an end) is named "contribution link".

Building an SD Model for the SR Model

Regardless of intentionality, an actor in the SR model should have in the highest level two ways to express desires and motivations: goals or softgoals. Accordingly, it was represented in [4] that the *Actor* depends on either the agent *EndSoftgoal* to have a softgoal satisfied or the agent *EndGoal* to have a goal be achieved. We called endSoftgoal and endGoal because Yu's thesis [2] placed softgoals, goals, resources, and tasks as end positions. There are also two possibilities: (a) a subTask may be an instance (INS) of a MeanTask and may be an instance (INS) of an EndTask and (b) a ResourceFor may be an instance (INS) of an EndResource.

Key points (derived from [4]): (1) intentionality is represented in an SR model by goals and softgoals, (2) there are two ways to represent the rationale inside actor's boundary: using a means-ends or a task-decomposition links, (2a) there is only one way to represent a task-decomposition link and there are five kinds of means-ends links (four have a task being a mean agent and one have a softgoal being a mean agent), (2b) a decomposed task can have four kinds of sub components: subTask, ResourceFor, subGoal and SoftgoalFor, (3) there are five situations of instances: (3a) a meanSoftgoal may be a an endSoftgoal, (3b) a meanGoal may be a subGoal, (3c) a resourceFor may be an endResource, (3d) a subTask may be a meanTask, and (3e) a meanTask may be a endTask.

Building an SR Models for the SR Model

As we noted before (Figure 2), an SR model is based on two "agents" for representing the rationales inside strategic actors, the links: Means-Ends and Task-Decomposition. In the meta-model we considered as organization "The SR Model" and consequently we represented the SR model actors for this organization [4].

Our experience in i* modeling suggested us a reduction mechanism: a proposal for simplifying the SR model. The means ends links *task-task* and *task-resource* should be eliminated. They are not necessary because they can be considered and modeled as a *task-goal* link, like "task Be performed" and "resource Be prepared".

SR model Check List

I - For each actor: Are goals and softgoals the roots in the highest level?

- Is each element a Contribution Link, a Task Decomposition Link or a MeansEnds Link?
- Is each MeansEnds Link, a Task-Goal Link, a Task-Task Link or a Task-Resource Link?

II - For each softgoal: Does the softgoal obey the standardization?

- In case of a single softgoal: Is there a NFR catalog for that softgoal?
- In case of a contribution: Is the contribution, a Task-Softgoal Contribution or a Softgoal-Softgoal Contribution?

III - For each goal: Does the goal obey the goal standardization?

- Can the actor achieve the goal by him(her)self? Why not?
- Is the task good enough to achieve the goal? Why?

IV - In case of Task Decomposition:

- For each softgoalFor: Has the softgoal answered the questions in II?
- For each subGoal: Has the goal answered the questions in III?
- For each subTask: Does subTask obey the standardization?
  - Can the actor perform the task? Why not?
  - Is the task necessary for the main task?
- For each resourceFor: Does it obey the standardization?
  - Is the resourceFor already prepared?
  - Is the resourceFor necessary for the main task?

V - For each goal dependency: Does dependency obey the standardization?

- Can dependee achieve the goal depender wants to?
- Why dependee is going to achieve the goal depender wants to?

VI - For each softgoal dependency: Does it obey the standardization?

- Can dependee achieve the softgoal depender wants to?
- Why dependee is going to achieve the softgoal depender wants to?

VII - For each task dependency: Does dependency obey the standardization?

- Can dependee perform the task depender wants to?
- Why dependee is going to perform the task depender wants to?

VIII - For each resource dependency: Does dependency obey the standardization?

- Can dependee provide the resource depender wants to?
- Why dependee is going to provide the resource depender wants to?

4 Conclusion

The main goal of this work is to improve the understanding of the i* framework so that requirements engineers can fully explore i* strengths. The work reminds the orthogonal role of each element, gives emphasis over what should be modeled and also shows the possibilities of i* modeling as a meta-modeling representation.

We have applied the i* Check Lists asking graduated students for verifying classmate's diagrams for simple modeling exercises. Our results are encouraging; however, we need to apply the strategy in different situations in order to get practical evidence of the effectiveness of our strategy. While carrying out these experiments we will also evaluate how well the approach scales to more complex models.

References

1. Leite, Julio; Werneck, Vera; Oliveira, A. Padua A.; Capelli, Claudia; Cerqueira, Ana Luiza; Cunha, Herbert; Baixauli, Bruno; "Understanding the Strategic Actor Diagram: An Exercise of Meta Modeling", The X Workshop on Requirements Engineering; Toronto, Canada - May/2007.
2. Yu, E. Modelling Strategic Relationships for Process Reengineering. PhD Thesis, Graduate Department of Computer Science, University of Toronto, Toronto, Canada - 1995.
3. Zheng You, Using meta-model-driven views to address scalability in i* models, Master of Science thesis, Graduate Department of Computer Science, University of Toronto, 2004, pp. 231.
4. Oliveira, A. Padua A.; Leite, J. C. S. P; Cysneiros, L. M.; "An exercise of Meta-Modeling: the Case of i*", In Monografias em Ciência da Computação, DI/PUC-Rio - 2010.