Applying a Foundational Ontology to Analyze Means-End Links in the *i** Framework

Renata Guizzardi

Ontology and Conceptual Modeling Research Group (NEMO)
Universidade Federal do Espírito Santo
Vitória, Brazil
rguizzardi@inf.ufes.br

Xavier Franch

Software Engineering for Information Systems Research Group (GESSI)
Universitat Politècnica de Catalunya
Barcelona, Spain
franch@essi.upc.edu

Giancarlo Guizzardi

Ontology and Conceptual Modeling Research Group (NEMO)
Universidade Federal do Espírito Santo
Vitória, Brazil
gguizzardi@inf.ufes.br

Abstract— **The *i** community has raised several main dialects and dozens of variations in the definition of the *i** language. Differences may be found related not just to the representation of new concepts but to the very core of the *i** language. If on the one hand this is caused by large adoption of the framework in the academic setting, on the other hand, it also poses some threats. For example, novices have trouble learning how to use the language, and besides these inconsistencies prevent *i** from being largely adopted in business settings. Based on positive results from previous work related to conceptual modeling languages, we believe that foundational ontologies may present a promising solution for this problem. Foundational ontologies may help clarifying the semantics of core *i** constructs and provide practical guidelines for their use. Last, they may serve as the basis to propose a normative definition of the framework. In this paper, we develop this idea, first by justifying the use of foundational ontologies and, in particular, the UFO ontology. Then, we raise some problems found in the *i** literature. And then, we show the outcomes of adopting an ontological approach for the specific case of the *i** framework. We focus here on one of the most characteristic *i** construct, namely the means-end link.**

Keywords-*i**; iStar; foundational ontology; UFO; means-end.

I. INTRODUCTION

Throughout the years, different research groups have proposed variations to the modeling language proposed in the *i** framework (from now on "the *i** language") [1][2][3]. Some come from paradigm shifts, others propose some particular type of new construct, and still others issue slight modifications related to the core constructs of the *i** language. This third type of variations mainly appear because the definition of the *i** language is neither fully detailed nor formal, and researchers may have interpreted the same constructs in different ways. The absence of a universally agreed metamodel has accentuated this effect [2].

Two possible positions can be adopted at this respect. One may argue that due to the social intention behind *i** modeling, a certain degree of freedom is convenient and then these slight changes should be acceptable. On the contrary, a more strict position is to consider necessary the existence of a shared body of knowledge on *i** with a well-defined meaning. In this paper,

we align with the second option: our position is that those concepts that form the core of the *i** language shall be well-defined and agreed by the community. This is important to allow a uniform and consistent use of the language, in a way that the community members are able to understand and communicate well through their models. Moreover, if it is a community objective that *i** gains industrial acceptance, then it is necessary to provide one interpretation of its core concepts.

There may be some discussion about where the boundaries of the *i** language core are, but some constructs like actor, goal and dependency, to name a few, seem out of discussion. In this paper, we are interested in the analysis of means-end links. Therefore, we search for a proposal that may serve as basis for a community agreement about what a means-end link exactly is. Agreement shall be first at the syntactic level by referencing to some *i** metamodel (e.g., which type of intentional element may be involved in a particular context). But syntax is not enough; still different modelers may interpret syntactic-equivalent models in different ways. We need a deeper understanding of the meaning of means-end link. In this paper, we propose the use of the UFO foundational ontology [4].

UFO has been previously applied to evaluate and (re)design other conceptual modeling languages. For instance, we may cite the work presented at [4], which formulated a new UML stereotype compliant with the findings of this foundational ontology. UFO has also been successfully used to design a language underlying a Knowledge Management Systems methodology [5] and to align Goals and Business Processes [6]. Finally, UFO has been employed with success in the analysis of alternative enterprise modeling standards and languages such as RM-ODP [7] and Archimate [8], ARIS [9], BPMN [10], REA [11], among others, In all these cases, the ontological approach has proven itself useful in clarifying the semantics, and consequently, the possible uses of the language's construct. This makes it much easier for the designer (novice or not) to apply the language in practice, once all ontological commitments have been made explicit.

Our work proposes the use of UFO to provide an ontological foundation to the *i** framework. In this paper, we take one particular construct, means-end links, as subject of research. The goal therefore is twofold. On the one hand, to

provide a well-defined semantics to means-end links so that the *i** community may use this construct with a non-ambiguous meaning. But on the other hand, we are interested in gaining more insights on the use of foundational ontologies in general, and UFO in particular, in the analysis of conceptual modeling languages.

The paper is organized as follows. We start by advocating for the adoption of foundational ontologies in general, and UFO in particular (Section II). We present next a few examples extracted from the *i** literature which illustrate some representative inconsistencies (Section III). In Section IV, we present an UFO-grounded ontological view of the means-end link. Finally, Section V concludes this paper.

II. ON THE USE OF FOUNDATIONAL ONTOLOGIES

A. Ontologies in Conceptual Modeling

Ontologies are recognized as important conceptual tools in Computer Science since the end of the 60s, especially in the areas of conceptual modeling and artificial intelligence [4]. In the past decade, we observed an explosion of works related to ontologies in several scientific communities. This is motivated by the potential of ontologies to solve semantic interoperability problems (e.g., application and database integration).

An important point to notice is the difference in meaning of the term "ontology" when used, on the one hand, by the conceptual modeling community and, on the other hand, by the artificial intelligence, software engineering and semantic web communities. In conceptual modeling, the term is used in accordance with its original definition in philosophy, i.e. as a formally and philosophically well-founded model of categories that can be used to articulate conceptualizations in specific engineering models and knowledge domains [4]. Conversely, in the other areas mentioned above, the term ontology has been used to describe: (i) a concrete engineering artifact, designed to serve a specific function, without (or with minimum) concern to theoretical foundational aspects; or (ii) domain models (e.g. biology, finance, logistics, etc.) expressed in a knowledge representation language (e.g., RDF, OWL, F-Logic or conceptual modeling language (e.g., UML, EER, ORM) [4].

B. Advantages of using Ontologies

With respect to the analysis and (re)design of conceptual modeling languages (i.e. the focus of this particular paper), we must understand ontology as in conceptual modeling, i.e. as a theoretical body of knowledge or foundation (hence the name foundational ontology). Using this foundational ontology as a reference model (see Figure 1) enables the evaluation, comparison, and identification of correspondences between different modeling languages. In other words, a foundational ontology (e.g. UFO [4], DOLCE [12], etc.) can be employed here as a well-founded basis for: (1) making explicit the ontological commitments of each modeling language; (2) defining (ontological) real-world semantics for their underlying concepts; and (3) providing guidelines for the correct use of these concepts.
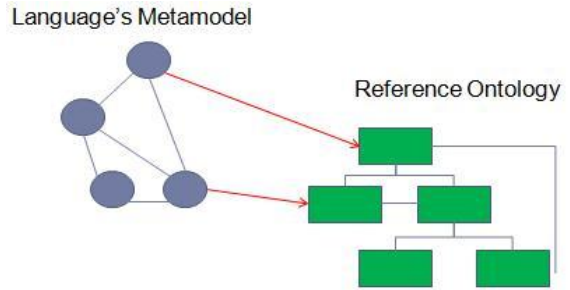

Figure 1. Language Metamodel grounded in a Reference Ontology

With respect to (1), it is our belief that when a modeling language does not make explicit the semantics of the constructs it applies, it leaves room for dubious interpretation and mixed used of such constructs, which may seriously undermine the understanding of the models designed using such language, also damaging communication between designers of different models, and between designers and stakeholders. In other words, making explicit the ontological commitments means making clear the intended meaning for each of the applied construct, articulating this intended meaning with the help of general ontological categories. This, in turn, allows for free discussions about such meaning, besides applying the constructs in a more consistent and coherent manner.

Regarding (2), it is important to emphasize the nature of foundational ontologies. Foundational Ontologies are typically founded in studies from areas such as philosophy, linguistics, logics and cognitive science. Moreover, being domain independent well-grounded formal theories, they can serve as a foundation for analyzing general modeling language´s primitives and domain specific concepts. For instance, when presenting a concept such as "part", the ontology makes explicit the very essence of what parthood is, i.e., regardless with we are talking about of parts of cars, organizations or human bodies. In this example, a theory such as mereology is deeply investigated to provide support for this definition. The ontology then helps one to answer: what is a part and how is it different than the whole? What are essential parts? Are there parts which may be shared among different things or this is a theoretical impossibility? Giving real-world semantics to modeling constructs enable one a much clearer understanding of how to use the modeling language, as well as helps us to realize if such language is able to model all the phenomena of the world which deserves to be modeled in particular domains. A good example is given by UFO-A, which offers is a rich theory of parthood relations [4]. Using this theory one can make explicit the distinction between, for instance, Essential Parthood (Person-Brain) and Mandatory Parthood (Person-Heart). This formal characterization underlying these distinctions allow: a) being explicit about the formal semantics of modeling primitives in Conceptual Modeling; and b) more importantly, being explicit about which ontological relations in reality are represented by these modeling primitives.

As for (3), usage guidelines come as a consequence of understanding well the semantics behind constructs. With this, designers are likely to better grasp in which situation one construct is preferable or which design patterns may appear. Also, [5] and [6] show how two distinct languages can

interoperate, having UFO as a reference model, which also guides designers on the practical application of both languages.

One could argue that there are several foundational ontologies, so grounding a modeling language with respect to one of them does not solve the problem, only transferring the discussion to another layer of abstraction. It is true that a few foundational ontologies are now prominent in the field of computer science and it is also certain that they present disagreements w.r.t. some philosophical choices. However, despite the plurality of existing foundational ontologies, one can rely on pre-theoretical criteria for their selection [13]. For instance, for the purpose of providing a foundation for conceptual modeling, one can certainly defend the primacy of an ontology that takes linguistic competence and cognition seriously (as it is the case of UFO) over a revisionary natural-science oriented one. At minimum, choosing one Foundational Ontology and using it as a reference model for a modeling language offers the following advantage: it enables people to debate their distinct views on the language with basis on a solid theoretical framework. In other words, it helps people understand where they agree and where they disagree, making the debate easier to flow. This is exactly the point which makes this proposal attractive in the case of the *i** framework.

C. Foundational Ontologies for the i* Framework

As already stated in the introduction, throughout the years, different research groups have proposed variations to the modeling language underlying the *i** framework [1][14]. The result of such efforts led to a large adoption of *i** at least in the academic environment, but also in a few industrial settings. While this is quite rewarding, the popularity of the language also introduces some challenges: (a) people have different understanding of the *i** core concepts as well as the extensions, which makes them to apply the language constructs in various distinct ways, inconsistent between each other; (b) novices have great difficulties in learning the language, due to conflicting examples found in literature; (c) new industrial partners resist adopting a framework, given that the academics do not provide a clear and uniform standard; and (d) *i** tools developed for, sometimes, complementary purposes, hardly interoperate.

The problems highlighted in (a), (b), (c) and (d) are typical interoperability problems. Going back to the proposal of grounding the modeling language on a foundational ontology, such approach could enable the distinctions among the different *i** dialects to be properly formalized. This can assist the debate towards defining an industrial standard, while also allowing distinct dialects to co-exist, providing that semantic interoperability is guaranteed by using the same formalization model.

D. The UFO Foundational Ontology

It has been aforementioned that several foundational ontologies exist today. A few examples of some prominent work are GFO [15], DOLCE [12], OntoClean [16] and BWW [17]. UFO started as a unification of the GFO (Generalized Formalized Ontology) and the Top-Level ontology of universals underlying OntoClean. However, as shown in [4], there are a number of problematic issues related the specific objective of developing ontological foundations for general conceptual modeling languages (e.g., EER, UML, ORM) which were not covered in a satisfactory manner by existing foundational ontologies such as GFO, DOLCE or OntoClean. To cite few examples, in contrast with BWW, it elaborates on social concepts, and on a cognitive and linguistic view of events [18]; in contrast with BWW and GFO, it elaborates on multitude of finer-grained categories of classifiers which are fundamental for conceptual modeling [4]; in contrast with OntoClean, it targets other types of non-taxonomic relations and in contrast with DOLCE, it develops a theory of material relations which proved to be of great significance in addressing a number of phenomena in conceptual modeling. Thus, with the purpose of developing a foundation for addressing the specific phenomena of concern in conceptual modeling, UFO has been developed into a full-blown reference ontology based on a number of theories from Formal Ontology, Philosophical Logics, Philosophy of Language, Linguistics and Cognitive Psychology. This ontology is presented in depth and formally characterized in [4], for the case of UFO-A, and discussed in more details in [18], for the case of UFO-B and C.

For the aforementioned reasons, we have chosen UFO here as our foundational ontology for the analysis of the *i** framework.

E. Analyzing a Modeling Language by applying a Foundational Ontology

Guizzardi [4] provides a framework for evaluating modeling languages. This framework verifies how clear and expressive a language is, by focusing on its notation, but also evaluates how well this language is able to represent the state of affairs for which it is proposed (also referred in this work as *domain appropriateness* and *comprehensibility appropriateness*).

"The domain appropriateness of a language is a measure of the suitability of a language to model phenomena in a given domain, or in other words, of its truthfulness of a language to a given domain reality. (...) Comprehensibility appropriateness refers to how easy is for a user a given language to recognize what that language's constructs mean in terms of domain concepts and, how easy is to understand, communicate and reason with the specifications produced in that language." ([4], pg. 28).

The proposed framework is based on the construction of an ontology to describe the conceptual domain of discourse. This ontology is then used as a type of 'mirror' for the modeling language, i.e. for verifying how well this modeling language is able to represent the concepts and relations represented in the ontology. This verification results is a measure of the quality of the domain appropriateness of the given language. The ideal situation is that in which the modeling language metamodel is isomorphic to the reference ontology, i.e. there is a one-to-one mapping between the concepts underlying the language and the concepts defined in the ontology. For an in depth discussion on this topic, one should refer to [4].

Figure 2 exemplifies how the method enables the identification of four undesirable properties in the modeling

language, namely: construct overload, construct excess, construct redundancy, and language incompleteness.
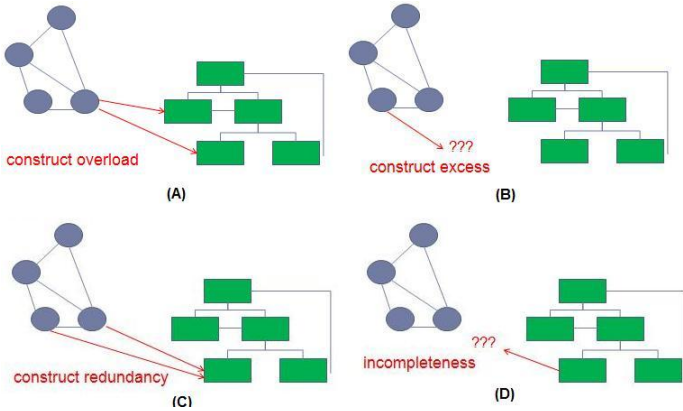

Figure 2. Inconsistencies in a language may be found when grounding its metamodel with respect to a reference ontology

Construct overload generally means having a single language construct representing two or more ontological constructs. As stated by Guizzardi ([4], pg. 31) citing [17], *"Construct overload is considered an undesirable property of a modeling language since it causes ambiguity and, hence, undermines clarity. When a construct overload exists, users have to bring additional knowledge not contained in the specification to understand the phenomena which are being represented."*

To be sound, every construct of the language should have an interpretation in terms of the domain conceptualization. As discussed in [4], if the language includes constructs devoid of a natural ontological interpretation, readers will assign their own individual interpretation to it, over which the language designers have no control. This, in turns, opens the possibility for non-shared interpretations among people communicating with the language and, hence, to semantic interoperability problems. Having one construct that does not map to any ontological concept is known as *construct excess*. The presence of this extra construct undermines the understanding of the specification. In other words, a specification is clear if the reader is able to link the language constructs to the entities of the domain of discourse. Consequently, only the entities of this domain (represented in the domain ontology) should be modeled with the use of language constructs.

A language should possess only one construct to represent each phenomenon in the domain or discourse (i.e. each entity in the domain ontology), thus avoiding *construct redundancy*. The same conceptual entity being represented by two or more constructs in a specification, consequently adds confusion to the meaning of the model. A reader may ask himself, for example, if the two constructs are actually the same or if there is any semantic distinction between them. Besides turning more difficult the understanding of specifications, this adds unnecessary complexity to the modeling language. In general, when facing redundancy, designers tend to ascribe slightly different meanings to the redundant constructs, which may not be fully understood by the model readers.

A modeling language is said to be complete if every concept in a domain conceptualization is covered by at least one modeling construct of the language. This is directly linked to the expressivity of the given language. In other words, if a language is incomplete, it fails to represent all phenomena in the given domain of discourse. The result of this *incompleteness* is either an incomplete specifications or construct overload, which are both undesirable for deteriorating the clarity of the specifications produced with the given language.

III.   THE *I** CASE: EXAMPLES FROM LITERATURE

In this section we analyze in detail the current landscape with respect to language variations proposed by the *i** community. In previous work we have undertaken several reviews that we use as our baseline for his analysis. In [19] we presented a thorough comparative analysis of the three main streamlines of the *i** language, namely the seminal *i** [20] and its evolution in the *i** wiki, GRL [21] and the Tropos' language [22]. In three recent papers [1][14][23] we located and analyzed papers on *i** published in 10 selected venues in the last 5 years, including moreover the recent *i** book [24]. We found 146 papers that were analyzed under different perspectives in each of those works: model interoperability [14], framework diversity [1] and language variations [23]. From this last perspective, that is the one of interest in this paper, we filtered the papers removing those that tackle different issues other than language (e.g., analysis techniques, visualization issues, etc.) and we selected 63 papers that were categorized in terms of addition, removal or modification of existing *i** constructs (see Table I).

TABLE I.   VARIATIONS PROPOSED BY THE *I** COMMUNITY IN THE LAST 5 YEARS (SELECTED VENUES ONLY). EACH PAPER INCREMENTS EACH COLUMN AT MOST IN 1.

	Actors	Actor links	Depen-dencies	Intentional elements (IE)	IE links
New	4	24	10	21	21
Removed	8	5	2	1	0
Changed	3	1	1	36	43

In this section, we present a few examples extracted from this literature review, which illustrate some of the inconsistencies we encountered. We particularly focus on the use of the means-end link. As this use is frequently mixed up with other relation types, examples also include cases of contributions and decompositions.

A.   *The use of means-end links*

Different *i** variants [23], whilst respecting the main idea of the means-end link (ME-link for short) as "a means to attain an end", state their own (if any) restrictions (see Table II). In his seminal proposal, Yu stated about ME-links: "the end can be a goal, task, resource or softgoal, whereas the means is *usually* a task". The term "usually" was dropped in the evolution of this variant, the *i** wiki version, where not just the means were completely restricted to tasks (i.e., not *usually* but *always*) but also the end was restricted to goals. The guidelines in the wiki state that other types of links are available for different

combinations of intentional element's types. Whilst being clear in terms of what types are permitted, this definition is very restrictive.

Another issue to remark is the relationship to the concept of "OR-decomposition". The concept of ME-link is close to that of OR-decomposition, in the sense that the source elements of the link (either "means" or "sub-elements") are interpreted in a kind of logical OR relationship with respect to the target. In fact, several proposals seem to not distinguish these two concepts, e.g. GRL has eliminated means-end links and just OR-decomposition (in addition to AND- and XOR-decomposition) is offered. On the contrary, some Tropos definitions are using both constructs in its language, like in the following quotations: "in an OR decomposition the subgoals represent alternative ways to achieve the root goal", and "the means/end relationship specifies a means (in terms of a goal, a plan or a resource) to satisfy the goal"; whilst others Tropos' papers are closer to the classical Yu's proposal or even whilst providing both constructs, it is not clear which is the real difference.

It is also worth to remark that existing works do not seem to address much the relationship between ME-links and actor links. In our ongoing work analyzing the meaning of the is-a actor link, we have mainly investigated if new means can be added to an end that is inherited in a subactor. To do so, a question needs to be answered: is a means-end relation always complete, is it always incomplete, or if none of the former, is it possible to distinguish an incomplete relation from a complete one?

Last, a final issue is whether the means for an end are to be considered exclusive or not (XOR vs. OR). For instance, Yu's thesis does not explicitly state this interpretation, but from the examples the means seem to be always exclusive. Just one analyzed proposal (GRL) allows explicitly declaring the type of logical operator, but we remind that GRL is not distinguishing among ME-links and OR-/XOR-decompositions.

TABLE II.   SUMMARY OF THE STATE OF THE ART.

Approach	Link	Intentional types (means->end)	OR/XOR/ not stated
Yu's thesis	Means-end link	(usually) T -> G | SG | T | R	Not stated
i* wiki	Means-end link	T -> G	Not stated
Tropos	Means-end link	G | T | R -> G	Not stated
	OR-decomposition	G -> G	Not stated
GRL	OR-decomposition	G | SG | T | R -> G | SG | T | R	Explicitly declared

B. Examples of uses of means-end links

This subsection is dedicated to raise some issues present in literature, regarding the practical use of ME-links. We will see that two situations arise: discrepancies in the interpretation of ME-links, and modeling errors.

Logical combination of means. As already stated, an analysis of the existing literature reveals that the use of the ME-links is neither uniform, nor consistent. In general, we note that distinct researchers infer their own particular meaning for the i* constructs, according to the needs of their current work. Even sometimes, the same author applies ME-links with different semantics in distinct contexts or publications. For instance, Figures 3 and 4[1] present two models in which the ME-link is applied. By the description found in [20], in figure 3, the tasks lease, buy and borrow are three alternative ways to achieve the goal has (item). Thus, the means are related through an XOR-type relation. This is clear by the statement in the paper claiming that *"A means-end relationship links an end to one (of possibly several) means for accomplishing it"* (p. 104). However, according to [25], the three plans of Figure 4 (get new exclusion set, get new preference set and get new location preferences) should all be executed to accomplish the replan dynamically goal. This makes it clear that the relation between the means, in this case, is an AND-type relation. This might seem a slight distinction but the model designer is hardly the only reader of the model. In fact, this is a clear illustration of *construct overload* (refer to section 2.E). Consequently, two readers might infer distinct realities from the same model. One could even argue that this meaning is clear by the textual description. However, written words are dubious and it is not possible to guarantee that such a description will always accompany the model or that it will be transparent and unambiguous.

Types of means and ends. Figure 4 adds another concern to our list. In the iStar wiki, it is stated that only tasks and resources may be means to goals; and resources may be means to tasks. Such statement only corroborates with the main i* reference [20]. But other researchers and practitioners have a broader view of the elements that may be considered means to tasks than the view advocated in the wiki. In the example of the figure, for instance, goals are means to goals: the plan the meeting goal is a means to the replan dynamically goal.

ME-links and OR-decompositions. One may consider that the relation means to end must only work from the concrete to the abstract. Following this reasoning, one could then affirm that it does not make sense to have a goal as means to another goal. And in this case, to relate goals, the designer may only apply decompositions. However, one may also infer something different: suppose that means-end and decomposition between goals are only logically differentiated, the former being an implication while the latter is a disjunction (in case of OR-composition) or a conjunction (in case of AND-decomposition). In case both means-end and decomposition are possible between elements of the same kind (e.g. goals), then we must distinguish the semantics behind these relations. Otherwise, this represents a case of *construct redundancy*. So, while both are equally possible interpretations, it is important to clearly define which one is the choice of the i* framework, otherwise the understanding of the models may be seriously undermined, since the reader will interpret the model according

---

[1] All figures have been designed using the TAOM4E 5.0. This tool models Tropos so the syntax of the figures may differ somewhat from the original ones. It is also important to say that we only included the parts of the models which were relevant to the present discussion.
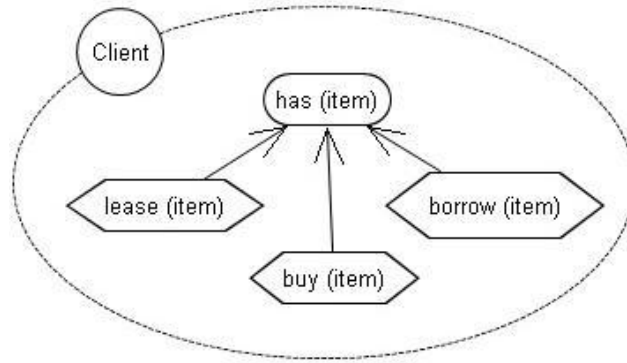
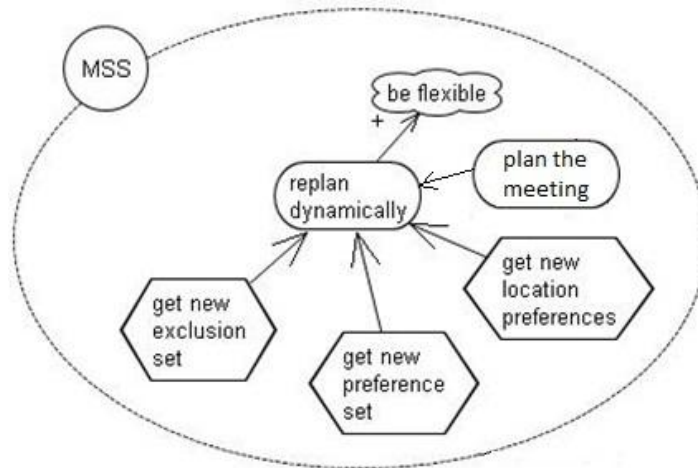Figure 3.   Means-end link as an XOR-type relation (adapted from [20])

Figure 4.   Means-end links as an AND relation [25]

to his own choice, which may differ from the designer's intentions.

ME-links and contributions. In Figure 5, one may wonder about the differences between ME-links and the contribution relation. The build a requirements Tropos model goal is a means to the root goal and at the same time contributes to two other goals (namely the organization described and the understanding customer's requirements goals). What is the difference between means-end and contribution? Does contribution infer partial satisfaction? This could be an interesting interpretation that in fact, differentiates both relations. However, we find in literature that contributions may have different values, e.g. make and help [2]. So one may wonder if the ME-link equals the make contribution. If so, again we have a case of *construct redundancy* and, unavoidably, one will try to ascribe distinct meanings to these constructs once they are applied in the same context, which could be misleading.

Application of more than one link. Figure 6 presents a case in which the same goal (namely the plan the meeting goal) is decomposed and, at the same time, has other goals as means. In such case, how should we interpret the distinctions between the relations? Also, this presents difficulties even in reading the model. Shall we start by considering the decomposition and then the ME-links (or the other way around, or the order does not matter at all)? It is difficult to understand why one should consider the relation between plan a meeting (goal) and [plan a meeting] automatically (subgoal) and [plan a meeting] manually (subgoal) a decomposition while the collect requirements and manage conflicts are considered means to this same goal. It seems to us that they refer to the same type of relation after all. Not having a suggestion on how to differentiate such cases, we could consider this yet another case of *construct redundancy* i.e. the existence of two constructs in the language to model exactly the same phenomenon.

---

[2] The definitions of the values of a contribution link are available in the istar wiki at http://istar.rwth-aachen.de/
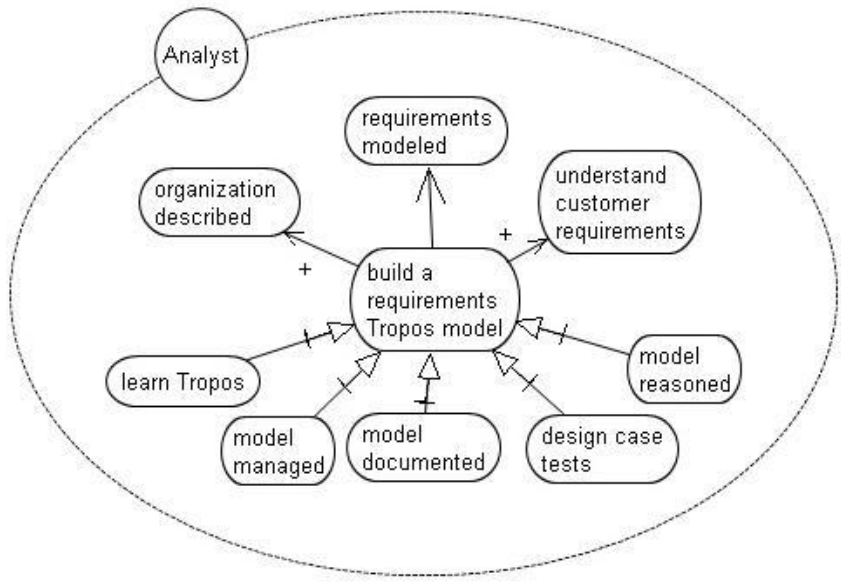
Figure 5.   Having both means-end and contribution links in the same context (adapted from [26])
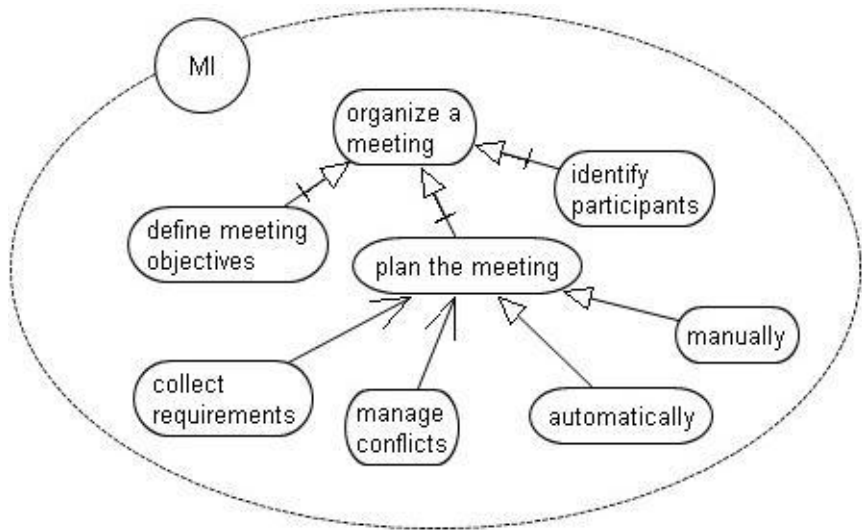
Figure 6.   Subgoals and means to the same goal [25]

A.   Means-end Links from an Ontological Point of View

Before analyzing *i** ME-links, it is important to provide an ontological view of the language intentional elements. Reading this section requires some basic knowledge about UFO [4][18].

We interpret the *i** *Agent* and *Role* as the concepts of Agent and Social Role in UFO (respectively). *Position* is also interpreted as a complex Social Role, since this *i** concept is defined solely with the purpose of aggregating different roles. The abstract *Actor* concept only captures general relations between *Agent*, *Roles*, *Positions* and other modeling elements and, thus, it has no specific interpretation in itself.

We interpret *i** *Goals* as Goals in UFO. Goals in UFO are related to sets of intended states of affairs of an agent. The relationship between an *Agent* in *i** and a *Goal* is interpreted indirectly by making use of the concept of Intention (or Internal Commitment) in UFO, which is a Mental Moment of an Agent. As previously discussed, UFO contemplates a relation between Situations and Goals such that a Situation (or possibly a number of Situations) may satisfy a Goal. In other words, since a Goal is a proposition (the propositional content of an Intention), we have that a particular set of state of affairs can be the truthmaker of that proposition. This set is named the goal satisfiability set.

The concept of *Softgoal* does not have a uniform treatment in the *i** community. Sometimes, softgoals are taken to represent non-functional requirements. In other places, a softgoal is considered as a fuzzy proposition, i.e., one which can be partially satisfied (or satisfied to a certain degree, or yet, *satisficed*) by Situations. We here take a different stance, reflecting an alternative, but commonly held view that a *Softgoal* is one "subjective to interpretation" and "context-specific". Hence, for softgoals, it seems to be impossible to eliminate a judging agent (collective or individual) from the loop. Thus, instead of considering in the ontology a new *satisfices* relation between Situation and Goal which perhaps should contemplate a fuzzy threshold of satisfaction, we take a different approach. We consider the relation of satisfaction as a ternary relation that can hold between an Agent, a Goal and goal satisfiability set. An instance of this relation is derived from the belief of an agent that a particular set of situations satisfies the goal at hand. Now, in this view, different agents can have different beliefs about which sets of situations satisfy a given goal. In fact, it is exactly this criterion which seems to capture the aforementioned notion of softgoals and its differentiae w.r.t. hardgoals: (i) a goal G is said to be a hardgoal iff the set of situations that satisfy that goal is necessarily shared by all rational agents; (ii) a goal G is said to be a softgoal iff it is possible that two rational agents X and Y differ in their beliefs to which situations satisfy that goal.

Seeing the distinction between these subcategories of goals under this light, allows us to talk about different levels of "softness" between different formulations of a goal. In one end of the spectrum, each individual agent would have a different belief about which situations satisfy a goal. In the opposite end, we have a hardgoal. In between, we can have communities of agents (or collective agents) of different sizes which share a common belief regarding this set of situations. In the last case, this collective agreement can be captured by a Normative Description [6].

A valid modeling strategy in *i** is the association of goals to roles. As we have discussed in [5], Social Roles in UFO are special types of Social Objects characterized by a number of Social Moment Universals (Social Commitments and Social Claims). They are typically defined through a Normative Description which is accepted in the context of a certain Social Agent (e.g., an Organization, a Social Community or a Society). Thus, when accepting to play a Social Role, one is making a meta-commitment of commiting to perform all tasks and adopt the goals associated with the role [27]. To cite one example, take the role of President, defined by a certain Constitution. By accepting to play this role, an agent commits to adopt all goals defined for this position. Moreover, in this situation, it is possible that some of the agent's prior goals become conflicting with those adopted via the commitment to the social role. For instance, while being President, one cannot work simultaneously for a private company and one cannot appear in public places without proper security measures. As pointed out by [28], an agent's autonomy within social context is restricted by the set of meta-commitments and claims she has, as a result of playing a specific role.

The mapping of the *Task* concept from *i** to some UFO concept is established in a direct manner. *Task* in *i** is a specific way of doing something to satisfy some *Goal* (or satisficing some *Softgoal*). From the UFO ontology, we have that an Action (instance of an Action Universal) is an intentional event performed by agents with the purpose of achieving goals. Consequently, the *i** *Task* construct can be interpreted as an Action Universal. It is important to emphasize that an Action may be either atomic or complex, mapping respectively to a Task as a single action or as a set of actions.

The concept of *Resource* has been interpreted as a Resource in UFO, i.e., as a Non-agentive Substantial (or Object) which participates in a Complex Action. A Complex Action is a composition of at least two basic Actions or Participations. Participations can themselves be intentional (i.e., Actions) or non-intentional Events.

B.   Ontological Analysis of the i Means-end Relation*

We first analyze *ME-links* between a *Task* and a *Goal*. In *i**, the *ME-link* is a ternary relation indexed to an *Agent*'s (subjective) point of view. However, as stated above, we consider in UFO that all agents have a consensual opinion regarding the satisfaction of a hardgoal. In this sense, we can exclude the agent's point of view from the definition of this *ME-link*, simply interpreting it as: a Task T is a means to a Goal G (G being the end) iff one or more executions of T (i.e., action instances of type T) produce a post-situation which satisfies G.

Similarly, the *ME-link* can also be defined between a *Resource* type and a *Goal and between a Resource* type and a *Task*. The former mode of this relation can be interpreted as follows: a Resource type is a means to a Task (end) iff every Action instance of that Task has as part a Participation of a

Resource of that type. In contrast, the *ME-link* between a *Resource* and a *Goal* should be interpreted as: a Resource type R is a means to a Goal G (G being end) iff every Action which satisfies that Goal has as part a Participation of a Resource of that type. By this definition, it is clear that, even if not explicit in the model, there is always a Task T (Action in UFO) which has as part a Participation of Resource R. In other words, Resource R is a means to the hidden Task T, which is a means to the Goal G.

In the case of a *Softgoal*, we should in fact consider the perspective (i.e., belief) of a particular agent, since its satisfaction set is not a consensus. Thus, we define the ME-link between a *Task* and a *Softgoal* as follows: a Task T is a means to a Softgoal S (S being the end) in the point of view of Agent A iff one or more executions of T produce a post-situation which A believes to satisfy S, or in other words, a post-situation included in the goal satisfiability set of A.

C. *Resolving the Dispute Regarding Means-end and OR-decomposition*

As already mentioned in section 3.1, the *i** dialects do not agree on the use of the ME-link and the OR-decomposition. Having two distinct relations that model the same phenomenon in the world is usually not a good practice, because, in general, the modeler will attempt to ascribe different meanings to each of them, or else randomly choose one or another when modeling such phenomenon. Thus, we believe it is in the interest of the *i** community to reach an agreement, having one uniform view in this regard.

In [29], we present two possibilities: 1) maintaining only the ME-link and excluding the OR-decomposition from the *i** core; and 2) having both ME-link and OR-decomposition, but having a clear semantic distinction between them. In this paper, we prefer to take a position, showing our preference to the second option.

Although leading to a more complex language, we argue that this choice favors expressivity. In this case, the aforementioned ontological definitions are applied if we consider means and ends of **distinct types**. And Means-end relation is seen as an implication, in which if the means is satisfied, then the end is also satisfied. Among intentional elements of the **same type** we consider OR-decomposition only, e.g. an OR-decomposition of goal G_0 into subgoals $G_1 \ldots G_n$ should be interpreted as: $(G_0 \leftrightarrow (G_1 \lor G_2 \lor \ldots \lor G_n))$. Thus, for the case of goals, OR-decomposition relations reflect logical relations between propositions. These distinctions are shown in Table III.

The distinctions between Means-End and Or-decomposition may help us interpret the model of Figure 6. In that case, both collect requirements and manage conflict should be modeled as tasks, as Means-end is no longer allowed between constructs of the same type. In this case, the satisfaction of the tasks implies the accomplishment of the goal. On the other hand, automatically and manually are alternative ways to accomplish the plan the meeting goal, which is correctly captured by the interpretation of Or-decomposition between goals as a goals disjunction.

TABLE III.   CONSIDERING BOTH MEANS-END LINKS AND OR-DECOMPOSITION

		End			
		T	R	G	SG
Means	T	Or-D	---	ME	ME
	R	ME	Or-D	ME	ME
	G	---	---	Or-D	---
	SG	---	---	---	Or-D

D. *Distinguishing between two types of Means-end relations*

There is still one pending issue, which relates to Figures 3 and 4 of the previous section. At this point, viewing a Means-end as an implication relation is not enough. There is still a conflict overload, since nothing is said if all means should be satisfied in order to satisfy the end (AND relation between means) or if the satisfaction of only one of the means is enough to satisfy the end (XOR relation between means). In order to correct this overload, the possibilities are: a) creating distinct arrows to denote each possibility or b) annotating the arrow with an AND or an XOR to make this difference explicit. The decision on one solution or another requires further investigation and here we just leave the issue open.

V.   CONCLUSIONS

The *i** framework is recognized as a distinguishing modeling framework for requirements analysis and organizational modeling. Some of the benefits of the use of *i** are [20]: (i) allowing a deep understanding of the organizational member's intentions and motivation; (ii) providing traceability between system's functionalities and requirements; and (iii) allowing the analysis of alternative ways a requirement may be accomplished. However, in order to fulfill its promises, the constructs of the *i** language must be well understood so as to allow consistent use. Moreover, it is desirable that the learning curve to apprehend the semantics and syntax of these modeling elements is kept reasonable. Otherwise, the effort for understanding the model or for learning how to design one will not justify its practical use.

We have seen that, in industrial practice, designers tend to apply a subset of the framework's constructs despite the effort of many researchers in extending the language to capture more complex phenomenon [14]. This trend also justifies the endeavor we have started in this work, by providing a clearer understanding of the *i** core constructs, grounding the semantics of these constructs in a formal theoretical body.

In this paper, we started by looking at the means-end relation and related concepts. We presented the results of a survey, showing a few distinctions found in among the *i** dialects in respect to this construct. We also illustrated some of the different existing interpretations by showing some examples collected from literature. And we finally provided an ontological analysis of this construct by the use of the UFO ontology as a reference model.

Our work offers several opportunities to other research lines. For instance, Moody et al. [30] have largely discussed about the adequacy of the current form that the *i** concrete syntax (i.e. its visual notation), identifying some situations that pose problems that somehow relate to the ones mentioned here (e.g., construct overload). The existence of an ontological foundation is a necessary input to design a concrete syntax, solving this kind of problems as recognized by Moody et al. themselves but also as discussed in [4].

Our subsequent steps in this research include debating the present definitions with the community of researchers and users of *i**, thus brushing them up. Moreover, we intend to visit the remaining constructs of the language, following the same approach to ground them in the UFO foundational ontology.

REFERENCES

[1] X. Franch, A. Maté, J.C. Trujillo, and C. Cares, "On the joint use of *i** with other Modelling Frameworks: a Vision Paper", in the 30th International Conference on Conceptual Modelling (RE 2011), Trento, Italy, 2011.

[2] X. Franch, "Fostering the Adoption of *i** by Practitioners: Some Challenges and Research Directions", in Intentional Perspectives on Information Systems Engineering, 2010, pp. 177--194.

[3] J. Horkoff, G. Elahi, S. Abdulhadi, and E. Yu, "Reflective Analysis of the Syntax and Semantics of the *i** Framework", in the 2nd International Workshop on Requirements, Intentions and Goals in Conceptual Modeling (RIGiM 2008), Barcelona, Spain, 2011.

[4] G. Guizzardi, "Ontological Foundations for Structural Conceptual Models". Ph.D. Thesis, University of Twente, The Netherlands, 2005.

[5] R. Guizzardi, and G. Guizzardi, "Ontology-based Transformation Framework from Tropos to AORML", in Social Modeling for Requirements Engineering, Cambridge: MIT Press, 2011, pp.547-570.

[6] E. Cardoso, P. Santos Jr., J.P. Almeida, R. Guizzardi, and G. Guizzardi, "Semantic Integration of Goal and Business Process Modeling", in the International Conference on Research and Practical Issues of Enterprise Information Systems, Natal, Brazil, 2010.

[7] J. P. Almeida, E. Cardoso, and G. Guizzardi, "On the Goal Domain in the RM-ODP Enterprise Language: An Initial Appraisal based on a Foundational Ontology", in the IEEE 6th International Workshop on ODP for Enterprise Computing, Vitória, Brazil, 2010.

[8] C. Azevedo, J.P. Almeida, M. van Sinderen, D. Quartel and G. Guizzardi, "An Ontology-Based Semantics for the Motivation Extension to ArchiMate", in the 15th IEEE International Conference on Enterprise Computing, Helsinki, Finland, 2011.

[9] P. Santos Jr., J.P. Almeida, and G. Guizzardi, "An Ontology-Based Semantic Foundation for ARIS EPCs", in the 25th ACM Symposium on Applied Computing (SAC 2010), Sierre, Switerland, 2010.

[10] G. Guizzardi, and G. Wagner, "Can BPMN be used for Simulation Models?", in the 7th International Workshop on Enterprise & Organizational Modeling and Simulation, London, UK, 2011.

[11] F. Gailly, G. Geerts, and G. Poels, Ontological Reengineering of the REA-EO Using UFO, International Workshop on Ontology-Driven Software Engineering, Reno, USA, 2009.

[12] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari. "Ontology Library", WonderWeb Deliverable D18, 2003.

[13] N. Guarino, and G. Guizzardi, "In the Defense of Ontological Foundations for Conceptual Modeling", Scandinavian Journal of Information Systems,vol.18, n. 1, 2006.

[14] C. Cares, and X. Franch. "A Metamodelling Approach for *i** Model Translations", in the 23rd International Conference on Advanced Information Systems Engineering (CAiSE 2011), London, UK, 2011.

[15] B. Heller, and H. Herre, "Ontological Categories in GOL". Axiomathes 14: 71-90, Kluwer Academic Publishers, 2004.

[16] N. Guarino, and C. Welty,C., "Evaluating Ontological Decisions with OntoClean". Comunications of the ACM, vol. 45, n. 2, 2002.

[17] R. Weber. "Ontological Foundations of Information Systems", Melbourne: Coopers & Lybrand, 1997.

[18] G. Guizzardi, R. Falbo, and R. Guizzardi, "Grounding Software Domain Ontologies in the Unified Foundational Ontology (UFO): The case of the ODE Software Process Ontology", in the XI Iberoamerican Workshop on Requirements Engineering and Software Environments, Recife, Brazil, 2008.

[19] C. Cares, X. Franch, E. Mayol, and C. Quer. "A Reference Model for *i**", in Social Modeling for Requirements Engineering, Cambridge: MIT Press, 2011.

[20] E. Yu, "Modeling Strategic Relationships for Process Reengineering, in Social Modeling for Requirements Engineering. Cambridge: MIT Press, 2011, pp. 11-152.

[21] D. Amyot, J. Horkoff, D. Gross, and G. Mussbacher, "A Lightweight GRL Profile for *i** Modelling", in the Workshop on Requirements, Intentions and Goals in Conceptual Modeling, LNCS, vol. 5833, 2009, pp. 254--264.

[22] P. Bresciani, P. Giorgini, and F. Giunchiglia, J. Mylopoulos, A. Perini. "Tropos: an agent-oriented software development methodology" in International Journal of Autonomous Agents and Multi Agent Systems, vol. 8, n. 3, pp. 203–236, May 2004.

[23] L. López, X. Franch, and J. Marco, "Making Explicit some Implicit Language Decisions in the i* Framework", in the 30th International Conference on Conceptual Modelling (ER 2011), Brussels, Belgium, 2011.

[24] E. Yu, P. Giorgini, N. Maiden, and J. Mylopoulos "Social Modeling for Requirements Engineering". Cambridge: The MIT Press, 2011.

[25] F. Giunchiglia, A. Perini, and F. Sannicolo. "Knowledge Level Software Engineering" in Intelligent Agents VIII: Agent theories, Architectures, and Languages. J.J. Meyer, and M. Tambe, Eds., Berlin: Springer, 2002, pp. 6-20.

[26] J. Horkoff, G. Elahi, S. Abdulhadi, and E. Yu, "Reflective Analysis of the Syntax and Semantics of the *i** Framework", in the 2nd International Workshop on Requirements, Intentions and Goals in Conceptual Modeling (RIGiM 2008), Barcelona, Spain, 2011.

[27] Castelfranchi, C. Commitments: From Individual Intentions to Groups and Organizations. In Proceedings of the First International Conference on Multi-Agent Systems. AAAI-Press and MIT Press, 1995, pp. 41-48.

[28] Bottazzi, E., & Ferrario, R. A Path to an Ontology of Organizations. Proceedings of the 1st International Workshop on Vocabularies, Ontologies, and Rules for the Enterprise (VORTE 2005). Enschede: Centre for Telematics and Information Technology, 2005, pp. 9-16.

[29] X.Franch, R. Guizzardi, G. Guizzardi, and L. López. "Ontological Analysis of Means-End Links", in the 5th International i* Workshop, Trento, Italy, 2011, pp. 37-42.

[30] D.L. Moody, P. Heymans, and R. Matulevicius, "Visual syntax does matter: improving the cognitive effectiveness of the i* visual notation". Requir. Eng. 15(2): 141-175, 2010.