An Empirical Study on the Use of *i** by Non-Technical Stakeholders: The Case of Strategic Dependency Diagrams

Juan Pablo Carvallo
School of Systems and Telematics
Universidad del Azuay
Cuenca, Azuay, Ecuador
jpcarvallo@uazuay.edu.ec

Xavier Franch
Software and Service Engineering Group (GESSI)
Universitat Politècnica de Catalunya
Barcelona, Catalunya, Spain
franch@essi.upc.edu

Abstract— **Context:** Early phases of information systems engineering include the understanding of the enterprise's context and the construction of models at different levels of decomposition, required to design the system architecture. These time-consuming activities are usually conducted by relatively large teams, composed of groups of non-technical stakeholders playing mostly an informative role (i.e., not involved in documentation and even less in modeling), led by few experienced technical consultants performing most of the documenting and modelling effort. **Objective:** This paper evaluates the ability of non-technical stakeholders to create strategic dependency diagrams written with the *i** language in the design of the context model of a system architecture, and find out which difficulties they may encounter and what the quality of the models they build is. **Method:** A case study involving non-technical stakeholders from 11 organizational areas in an Ecuadorian university, held under the supervision and coordination of the two authors acting as consultants. **Results:** The non-technical stakeholders identified the majority of the dependencies that should appear in the case study's context model, although they experienced some difficulties in declaring the type of dependency, representing such dependencies graphically and applying the description guidelines provided in the training. Managers were observed to make more mistakes than other more operational roles. From the observations of these results, a set of methodological advices were compiled for their use in future, similar endeavors. **Conclusions:** It is concluded that non-technical stakeholders can take an active role in the construction of the context model. This conclusion is relevant for both researchers and practitioners involved in technology transfer actions with use of *i**.

Keywords- *enterprise architecture; system architecture; requirements engineering; i* framework; iStar; dependency; empirical study.*

1. INTRODUCTION

Modern enterprises largely rely on information systems specifically designed to manage the continuously increasing complexity of interactions with their context. *Enterprise Architecture* [1] is an increasingly adopted concept that encompasses several levels of architectural design starting from the strategic level, usually referred as *Business Architecture*, down to the *Data Architecture*, *System Architecture* and finally the *Technology Architecture* layers. Mapping the business architecture to the system architecture is a complex process, which requires deep understanding of the enterprise context and strategies. Because of this, early phases of the enterprise architecting process are usually oriented to model the enterprise context. Enterprise *context models* (CM) include environmental actors (i.e., actors in the context of an enterprise that interact with it) and the description of the relationships among them. Resulting models help understanding the purpose of enterprises on their environment, i.e. what is required from them, becoming a fundamental piece that helps enterprise decision-makers to design and refine their business strategies, and enterprise architects to understand what will be required from the resulting socio-technical system. However, far from easy, the construction of such models is usually a cumbersome task, mainly due to [2][3]:

- Communicational gaps among technical personnel (e.g. internal or external consultants) and their administrative counterparts. The former usually lack knowledge about business strategies, modeling, planning, and administration skills; whilst the latter have similar limitations in relation to methodological business processes and requirements elicitation, and systems modeling techniques.
- Limited knowledge of the enterprise structure, operations and strategy. This forces technical staff to spend important amounts of time studying and understanding the business. This effort needs to be reconciled with time constraints resulting from internal and external pressures and narrow windows of opportunity, which increases the risk of misunderstandings or misinterpretations.

Because of this, non-technical stakeholders do not participate actively in the early phases of the enterprise architecting process; instead, their role is mostly constrained to provide the information that technicians' request.

In order to deal with these problems, we proposed the DHARMA method [4], engineered to discover system architecture departing from the construction of CMs expressed in *i**. In the DHARMA approach, *i** CMs are interactively built with the

The final publication is available at Springer via http://dx.doi.org/10.1007/s00766-018-0300-7

participation of non-technical stakeholders, who often sketch drafts of the models without the participation of technical consultants. Although criticized by some authors, particularly for the difficulty to use its graphical notation [5] and the difficulty to manage the models when they grow [6], our experiences in the use of DHARMA supported the hypothesis of *i** being a framework that helped non-technical stakeholders in the early phases of enterprise architecting, Using this approach, non-technical stakeholders felt encouraged to help technical consultants achieve their requirements and architecting objectives by participating in the modelling activities [7].

The study reported in this paper has been designed with the goal of validating empirically the ability of non-technical stakeholders to learn and use the *i** notation in a real project, and find out which difficulties they may encounter and what the quality of the models they build is. Due to the purposes of the paper (see Section 3 for more details), we will focus on the first activity of the DHARMA method which makes use of one particular type of *i** models, the Strategic Dependency (SD) models and particularly the concept of dependency that gives the name to the model. The study has been designed in the form of a case study in which representatives of 11 organizational areas in an Ecuadorian university were involved. The results of the study (reported in Section 5) contribute to better understand the opportunities and limitations on the use of *i** beyond purely research studies, which is an area still lacking of contributions.

The rest of the paper is structured as follows: Section 2 presents the related work and Section 3 some background required to understand our work; Section 4 presents the case study; Section 5 presents the results of the work; Section 6 states the threats to validity found; Section 7 enumerates some methodological advices stemming from the work and Section 8 presents a final discussion and hints future work.

2. Related Work

There exists little work in relation to *i** usability analysis, particularly in cases involving non-technical stakeholders. Probably the most significant work in relation to this paper is the one presented in [5] which focuses on *i** visual syntaxes and semantic transparency. However, that work intends to improve *i** symbols in relation to several dimensions including semiotic clarity, perceptual discrimination, complexity management and visual expressiveness among other, more than a direct observation on the real usage. Other works [6][8] focus on guidelines to simplify, conduct and improve the process of drawing *i** diagrams.

Our intention in this case study is significantly different from these approaches; we are not interested in improving the *i** framework usability or end users ability to draw *i** diagrams, but to evaluate the ability of non-technical stakeholders to quickly grasp the main concepts included in the framework, to assimilate them and to produce models of enough quality, thus becoming more proactive in the earlier phases of the requirements engineering and enterprise architecture design processes.

The closest work in the field of Enterprise Architecture is the series of papers by Engelsman and Wieringa [9-11]. They have extended ArchiMate [12] with goal-oriented concepts and provided initial validation of the usability of this extension; it is worth to remark that his extension has been adopted in the Open Group standard for enterprise architecture modelling. Although not expressed in *i**, the goal of the validation is somehow related to our objectives. The authors defined understandability in terms of the correctness of use of the goal-oriented concepts and performed a study involving 19 language users (enterprise architects) which received some training on the language. In general, most of the evaluated concepts were misunderstood by most of the enterprise architects and thus the authors propose a simplification of the goal-oriented extension. The main reason behind the misunderstandings was conceptual similarity between concepts making it difficult to distinguish one from another; for instance, the notion of assessment was considered too close to goal, and similarly decomposition was confound with influence. Looking into the details, their analysis included two concepts closely related to our study: stakeholder (“individual, team, or organization (or classes thereof) with interests in, or concerns relative to, the outcome of the [enterprise] architecture”) and goal (“some end that a stakeholder wants to achieve”). While the notion of stakeholder (which is similar to the *i** concept of actor) was correctly understood by all the participants, goal had a more diverse comprehension, although still 17 out of 19 participants used the concept correctly more than 90% of its occurrences in the models. The results were thus encouraging, but the study didn’t cover the concept of strategic dependency that is so important in order to correctly delineate the relations of an organization with its context. It is worth to remark that other concepts like realization and decomposition were part of the analysis; although these concepts do not apply to our study, they are related to other kind of *i** models, the Strategic Rationale (SR) models, so they can become relevant in the future in this context.

Other works on the use of goal-oriented languages in enterprise architecture have not explored usability empirically [13][14]. Thus we argue that a study on the understandability of intentional dependencies for modeling the relationships between organizations and their context is still missing in the academic literature.

3. Background

In the early 90s, intentional modelling emerged as a way to capture the rationale in prescriptive models of requirements, information systems or software processes. They advocate to have an understanding of the “whys” that underlie the “whats”, embracing motivations intents and rationales [15]. In this concept emerged several approaches, being the *i** framework one of the most widely adopted by the software engineering community.

3.1 THE *i** FRAMEWORK

The *i** framework was proposed in the early 90s with the purpose of modelling and analyzing information systems from a goal-oriented perspective. Due to the importance of the concept of actor as an autonomous entity, the framework is also classified as agent-oriented. The first complete description of the framework appeared in Eric Yu's PhD thesis [16] (advised by John Mylopoulos) by 1995, comprising both the modelling constructs (which we refer to as the *i** language) and several proposed treatments. Since then, several releases, versions and dialects of the *i** language have been formulated basically in the areas of requirements engineering, information systems and business modelling. Recently, with the purpose of establishing a shared core language, the iStar 2.0 language was issued as a community effort [17]. Given its recent formulation, our study was carried out using the seminal language proposed by E. Yu in [16]. Although there are some differences in the concepts managed by both versions, they are not really fundamental and we do not think that they have an influence in the results of the study.

Models using the *i** language are of two different types. On the one hand, Strategic Dependency models (SD) show a networked structure of actors and the dependencies among them. On the other hand, Strategic Rationale models (SR) establish the rationale of the actors and how do they attain their needs. Given the purposes of the study (see Section 3.2) we focus here on SD models.

Figure 1 shows an example of SD model coming from E. Yu's thesis [16]: a model for a healthcare system. We can see several actors in the model that collaborate in order to achieve the overall objectives. This collaboration implies that each actor depends on others; therefore, the concept of dependency emerges. In a dependency, there are three parties involved: the *depender*, which is the actor that depends; the *dependee*, that is the actor upon which the depender depends; the *dependum*, that expresses the reason for the dependency. These dependencies are of four different types, following the type of the dependum:

- Goal dependencies. A depender depends upon a dependee in order to attain a goal. For instance, the Patient depends upon the Insurance Company in order to get Covered for his sickness.
- Soft-goal dependencies. A variation of goal dependencies in which the attainment of the goal has not clear cut criteria. For instance, the Physician depends upon the Claims Manager for getting a Fast processing of his claims, but the interpretation of Fast can be different for both involved actors.
- Task dependency. A depender depends upon a dependee in order to get a task done. For instance, the Insurance Company depends upon the Claims Manager in order to get its Claim processed.
- Resource dependency. A depender depends upon a dependee in order to get access to a physical or informational resource. For instance, the Claims Manager depends upon the Insurance Company in order to get the Patient's information.

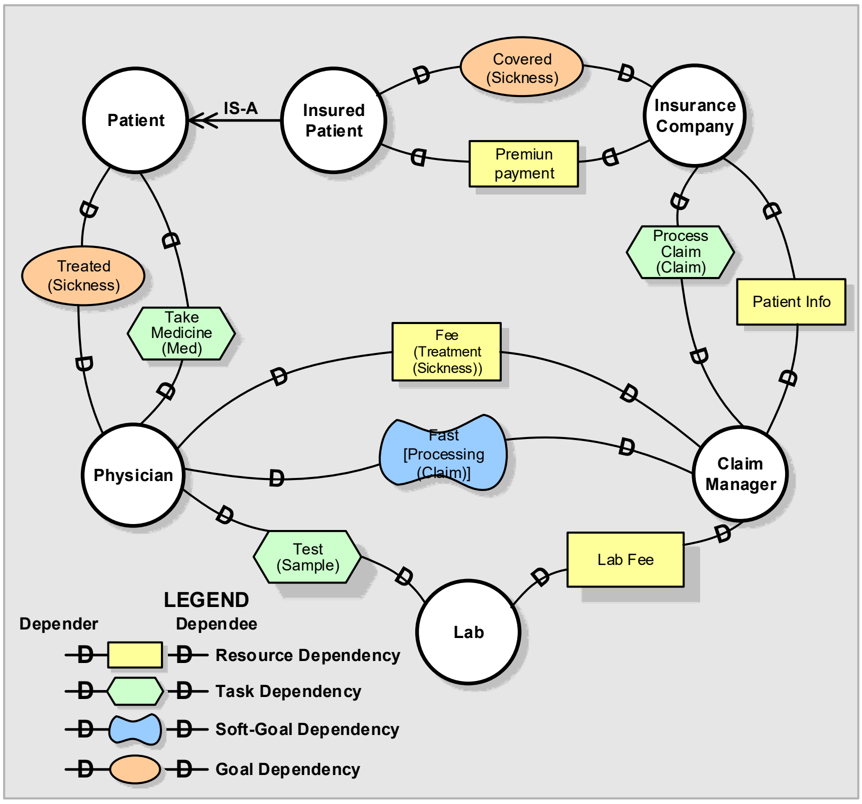

Figure 1. An example of *i** Strategic Dependency (SD) model (beautified model from [16])

3.2 The DHARMA Method

The DHARMA method (Discovering Hybrid ARchitectures by Modelling Actors) has been used in the context of this work. DHARMA aims at the definition of system architecture using the *i** framework [4]. We define a system architecture as a set of system actors (which represent atomic software domains that structure the system), services that must be covered by them and their relationships. The process resulting from the method is initiated by constructing a CM and ends with the identification of the system architecture (actors that structure the system, the services that must be covered by each of them and the relationships among them).

The method is structured into four basic activities that may be iterated or intertwined as needed (see Figure 2):

- **Activity 1: *Modelling the enterprise context.*** The organization and its strategy are analysed in detail, in order to identify the role that it plays inside its context. This analysis surfaces the different types of actors that exist in its contexts, and the strategic needs among them and the organization. *i** SD diagrams are used to elicit and represent the actors and relationships that form the departing CM.
- **Activity 2: *Modelling the environment of the system***. In this activity, a system-to-be is placed into the organization and the impact that it has over the context is analysed. The system may be a typical information system, or it may be a hybrid system including hardware components, maybe with some embedded software. The strategic dependencies identified in the former activity are analysed with the aim of determining which of them may be directly satisfied by the system, and which others are needed by the system providing its operational level. As a result, the dependencies are redirected inside the *i** SD diagram and also new dependencies arise. The model includes the organization itself as an actor in the system environment, in which its needs are modelled as strategic dependencies over the system.
- **Activity 3: *Decomposition of system goals***. In this activity, the system-to-be is analysed and decomposed into a hierarchy of goals that are needed to satisfy the strategic dependencies stated by the environment actors. The goals represent the services that the system must provide, to interact with the actors in the environment. An *i** SR diagram for the system is built, using means-end links of type goal-goal (representing then a decomposition of objectives into sub-objectives).
- **Activity 4: *Identification of the system actors***. The goals included in the SR model are analysed and systematically grouped into subactors that represent atomic domains. The objectives are grouped into services, according to an analysis of the strategic dependencies with the environment and an exploration of the existing off-the-shelf software marketplace. The relationships between the different actors that form the basic structure of the system are described according to the direction of the means-end links that exist among the objectives included inside them.

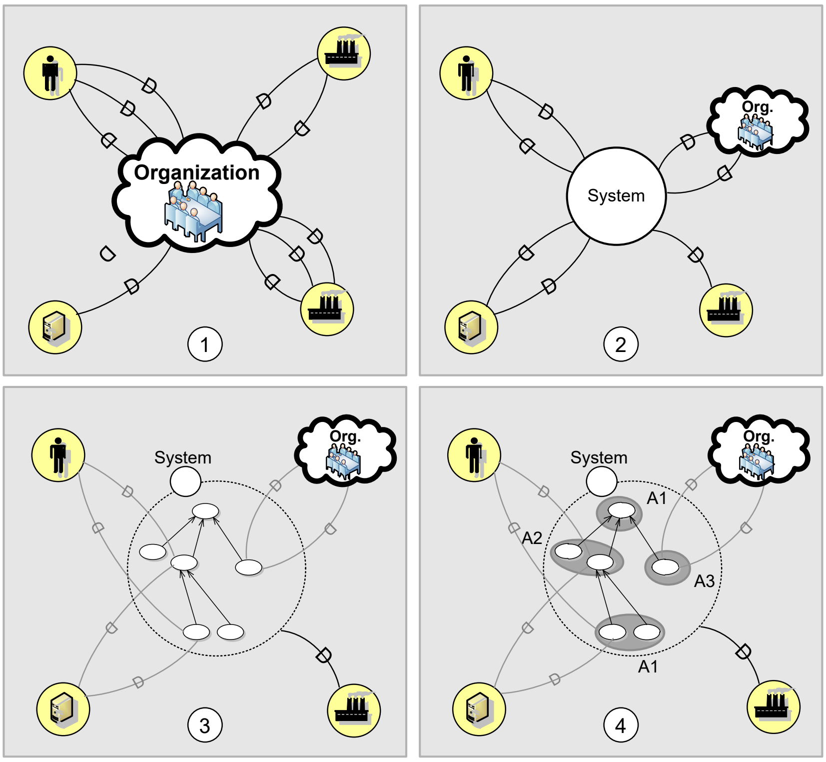

Figure 2. The DHARMA method.

As mentioned above, in this work we focus on the first activity of the DHARMA method, which makes intensively use of *i** SD models to describe the context of the organization. In our experience in real projects with DHARMA, non-technical stakeholders are well capable of contributing in activities 1 and 2, but not in activities 3 and 4; *i** SR decomposition conducted in these activities requires specialized knowledge in several technologies, thus they are more appropriated for requirements engineers or functional analysts.

From the academic and industrial point of view, we conducted and reported several cases in which we have used the DHARMA method. Particularly the cases of ETAPATELECOM, an Ecuadorian-based telecom company, and the Cuenca airport also in Ecuador, are worth to mention because of the size of the deliverables (e.g., context models encompassing more than 20 external context actors and hundreds of dependencies, which lead to the identification of system architectures comprising more than 20 subsystems in each case) and the organizational impact occurred as consequence of the project (e.g., restructuring of IT departments and the approval of founding for the execution of IT strategic plans encompassing portfolios with over 30 projects, required to implement the resulting architecture during 4-5 year periods, among others). We published in 2014 [7] a first compendium of lessons learned from the experiences run until that moment, which has been extended and enhanced in this paper, see Section 7 (Methodological Advices).

These experiences allowed us to construct a catalog of patterns of context elements, represented as *i** Strategic Dependency models, which include generic environmental actors and their associated strategic dependencies [18]. Patterns in the catalogue are used to improve CM construction, by reusing the stored intentional elements in new modes, instead of starting from scratch (see Figure 3 for an example that applies a pattern for CRM systems to a particular case). The catalogue, distinguish two levels of abstraction, a higher level applicable in general to any kind of enterprise and a lower level, which considers enterprise strategies which describe how a particular enterprise operates.

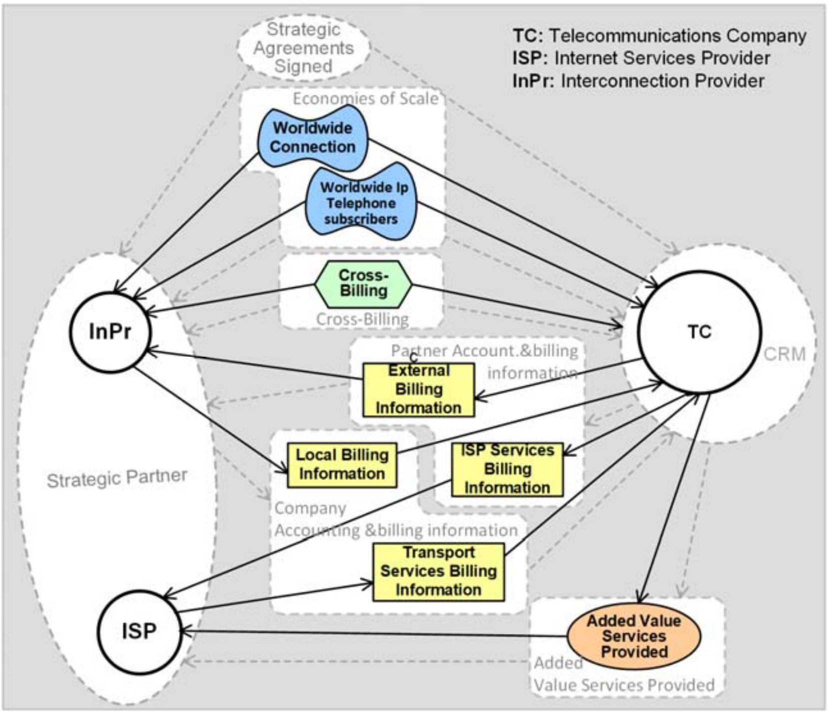

Figure 3. Example of instantiation of the CRM pattern in the domain of a Telecom (TC) company.

4. The Case Study

Next sections describe the case study and the activities performed in order to collect and analyze the data required to answer the research questions.

4.1 Overview

The industrial case study was conducted in the Azuay University (www.uazuay.edu.ec) a private medium size (10.000 students) Ecuadorian university. The University is based in the city of Cuenca, the third largest city in the country. Founded in 1968, the university offers several undergraduate programs though its 28 schools, which belong to 6 faculties (Business Administration, Juridical Sciences, Science and Technology, Design, Medicine, and Philosophy, Literature and Education). It also offers several graduated Master and specialization programs as well as Continual Education courses, open to the general public, which addresses specific needs of the region and the country.

Starting in 2011, change in Ecuadorian regulations has forced institutions to prioritize national and international accreditation, improve their research and community engagement activities, and in general, review their whole strategy. This context provides an ideal scenario for the enterprise architecture approach and thus the system architecture to be reengineered. In March 2012, we were hired as consultants to conduct a yearlong process, in which we used DHARMA to discover the hybrid architecture of the information systems required by the institution. One specific requirement of the administration was to actively involve stakeholders from different organizational areas, with different professional backgrounds and managerial levels. Participants were specifically selected by the Rector of the institution, in some cases involving more than one stakeholder per area e.g., the general secretary for each of the six faculties of the institution. The resulting architecture was then used as a basis to identify and prioritize the projects that structured the project portfolio of the university's IT strategic plan and to restructure its internal organization (subareas, managerial and technical profiles, processes etc.). This portfolio encompassed IT projects of different nature (software acquisition, software development, technology platform modernization and IT processes definition among other).

4.2 Goal and Research Questions

We state the goal of this study applying the Goal Definition Template from GQM [19]:

- **Analyze** *i** Strategic Dependency (SD) models
- **for the purpose of** evaluating *i** dependencies
- **with respect to** their understanding and correct application
- **from the viewpoint of** different types of non-technical stakeholders
- **in the context of** a case study in an Ecuadorian university project

To explore this goal, we focus on the concept of dependency, which makes *i** different from other goal- and actor-oriented approaches. In particular, dependencies lie at the core of the DHARMA approach for modeling enterprise contexts and system architectures. In order to design our study, we proposed four research questions to be addressed:

- **Q1: Do non-technical stakeholders understand and apply correctly the concept of *i** dependency?** We want to find out if the concept of social dependency is easily understandable by non-technical stakeholders; if not, one could argue that *i** is not an appropriate notation for our purposes. Therefore, we will investigate if they elicit the right ones, if they model them well, and if they miss some. The results of this research question frame the answers to the other three.
- **Q2: What are the most common mistakes made by non-technical stakeholders when representing *i** dependencies?** There exist several causes to ill-represent dependencies (e.g., wrong direction of the dependency, or incorrect description of the dependum), and this research question will collect the reasons and provide some arguments for the causes.
- **Q3: What are the types of *i** dependencies that are more difficult to understand and apply correctly for non-technical stakeholders?** In our experience, some types of dependencies are more prone to errors than others, and this research question will investigate this aspect.
- **Q4: Which are the groups of participants experiencing more difficulties on understanding and applying correctly the concept of dependency in *i**?** The concept of non-technical user is very broad and it could happen that not all of them experience the use of *i** the same way. Thus, this question considers the diversity of non-technical stakeholders with different professional backgrounds (e.g. lawyers, financial, human resources, etc.) and different levels of responsibility (e.g. managerial vs. operative roles) in order to find differences among these groups concerning the main difficulties that they experience when using the notation.

It is worth to mention that, since we offered to stakeholders the catalogue of actors coming from this previous work [18], we didn't evaluate the ability to apply such modeling concept, but instead we focused on the concept of dependency, which is key in order to build organizational models, both to represent dependencies within the context as dependencies inside different areas of the organization. We remark that the results from Engelsman and Wieringa (see Section 2) also support the fact that modeling actors (or stakeholders in their work) seems not to be a big barrier for practitioners.

4.3 Timeline and Subject Selection

The general timeline, including the main activities to be performed in the consultancy process, is presented in Table I.

After the needed initial preparation (Task 1 in Table I), the first activity conducted, as proposed in the DHARMA method, was the construction of the *i**-based CM (Task 2 in Table I). The CM construction was performed with the contribution of 13 *Organizational Areas* (OA), selected from the organization value chain and prioritized by the Rector of the University, who acted in the role of sponsor of the project. Once selected, we interviewed managers of each OA in order to define the specific stakeholders to be involved in the process. In some cases, managers decided to be directly involved whilst in others, mainly because of schedule constraints, they selected stakeholders with a more operational role to act on their behalf. The number of selected participants per area and their profiles is listed in Table II.

TABLE I. GENERAL TIMELINE FOR THE PROCESS.

Task	Description	Duration (work days)	ABC Activity
1	**Preparation**	**3 days**	
1.1	Review of resources with the Rector	2 days	Pre-ABC activities
1.2	Approval of Schedule	1 day	
2	**Construction of Context Model of the organization**	**35 days**	
2.1	Workshop/training seminar	2 days	Activity 1
2.2	Construction of first CMs by OAs	5 days	
2.3	Review of first CMs	3 days	
2.4	Correction of CMs by OAs	5 days	
2.5	Transcription of resulting CMs	7 days	
2.6	Refinement of resulting CMs	5 days	
2.7	CMs validation workshops	2 days	
2.8	Consolidation of validated CMs into organization's CM	3 days	
2.9	Elaboration and presentation of first report	3 days	
3	**Modelling of technological objectives and definition of components**	**35 days**	
3.1	Identification of automatable context dependencies	5 days	Activity 2
3.2	Identification of system objectives	10 days	Activity 3
3.3	Review of identified system objectives	10 days	
3.4	Grouping of system objectives into system actors	5 days	Activity 4
3.5	Review of identified actors	5 days	
3.6	Elaboration and presentation of second report	5 days	
4	**Building the project portfolio (charters and project plan)**	**60 days**	
4.1	Identification and definition IT projects	10 days	Post-ABC activities
4.2	Construction project charters	20 days	
4.3	Prioritization of IT projects	5 days	
4.4	Approval of projects by Board of Directors	5 days	
4.5	Construction of project plans for approved projects	30 days	
4.6	Elaboration and presentation of final report	5 days	

TABLE II. AREAS AND STAKEHOLDERS PARTICIPATING IN THE PROCESS

No.	Operational Area (OA)	Nb. of Participants	Job Position	Profession
1	Faculties	6	Mid-manager	Lawyers
2	Graduate school	1	Mid-manager	Business Administrator
3	Professional / Continuing Education	1	Manager	Business Administrator
4	Research Deanship	1	Operational	Systems Engineer
5	Financial Direction	1	Manager	Business Administrator
6	Administrative Coordination	1	Mid-manager	Economist
7	Treasury	1	Operational	Accountant
8	Human Resources	1	Manager	Business Administrator
9	Students Welfare	3	Manager / operational	Psychologists
10	Legal Department	2	Mid-manager	Lawyers
11	Communications Department	2	Manager / operational	Communicator
12	Libraries	1	Operational	Librarian
13	External Relations Department	1	Mid-manager	Lawyer

4.4 EXECUTION OF THE STUDY

To construct the CM, two types of activities were conducted: training (Task 2.1 of Table I) and modelling (tasks 2.2 to 2.8 of Table I).

4.4.1 TRAINING

Non-technical stakeholders had neither previous training on the graphical framework nor the notations to be used. Therefore, they were not aware of their utility and objectives. On the one hand, without basic training they wouldn't have been able to participate actively in the process. On the other hand, we did not want to train non-technical stakeholders to become experts; this could be a time-consuming and costly process, struggling against their particular interests and objectives, with no clear return on investment and thus, increasing the risk of the process.

We quickly surveyed the participants in order to identify their degree of interest in the process, their willingness to learn a modeling notation and the maximum time that they would be happy to spend on training. Although most of them showed a high interest on learning new techniques and in the process, because of its relevance for the organization and the opportunity to improve IT support for their areas, they also manifested their concern about the overhead that the process could bring to their already demanding agendas. As a result, we agreed an average of eight hours as maximum time to be spent on training. With this constraint in mind, we designed a four-session seminar, each session two-hour long. In the first two sessions, we socialized the project and its objectives and we taught participants about strategic planning in technology and hybrid systems architecture [4].

In Session 3, participants were introduced to the *i** basic concepts for modeling SD diagrams and the DHARMA method: actors, dependencies, dependency directions and dependency types (goal, soft-goal, task and resource) were discussed and several examples were provided; open discussion among participants was encouraged at all times. In the last training session, examples of application of the DHARMA method and construction of *i** SD models were provided. A complete relation of contents of the seminar is listed in Table III.

For most well-known graphical frameworks there exist several best practices documented, which can be easily transferred to non-technical participants. There are also guidelines emerging from consultants' own experience that can be provided. Starting from the first day of seminar, we introduced participants to such best practices for *i** (coming from [4][7][18]) and encouraged to use them at all times. We also made sure that participants were clear about their responsibility and the consequences of their lack of involvement (e.g. information systems with poor functional coverage for their areas).

In the sessions 3 and 4 of the training seminar, we issued the following recommendations:

- When modelling dependencies, we asked participants to consider only one actor in the environment of the OA at a time.
- Only dependencies among the OA and their environmental actors had to be considered; dependencies among environmental actors shall be discharged as irrelevant for the CM of the organization.
- To make identification of actors and dependencies easier, participants were encouraged to use the ones included in the Customer Relationship Management (CRM) pattern presented in [18] as checklist.
- Participants were given the choice to draw by hand diagrams using the standard *i** notation or to use the tabular representation that we have introduced in [20], which is shown in Table IV, with an excerpt of the dependencies identified among the Faculties OA and the Institutes context actor.
- Guidelines included in [4] were handed over the participants in order to drive CM construction in a more systematic way.
- Regarding descriptions of intentional elements, the guidelines proposed in [20] were also provided.

It is important to notice that we asked non-technical stakeholders to create their own models without providing training or biasing them towards the use of any particular tool. As result, they drew CMs all with different notations (see Figure 4 for some samples); however, intentional elements were easy to recognize and the time spent in training, and by participants on drawing activities, was significantly reduced. This fact led participants to focus in more relevant issues such as the identification of dependencies or their intentionality.

4.4.2 MODELLING

Regarding modelling, we provided non-technical stakeholders with a detailed schedule containing the work breakdown structure (WBS), including main activities, their tasks decomposition, time assigned for their fulfillment and deliverables to be produced.

Task decomposition was fine-grained; therefore, models were completed in several steps. For instance, for the first modelling activity (Activity 2.2 of Table I), in which we asked participants to identify actors in the environment of their specific OA and the dependencies among them and their OA, we identified the activities (subtasks) listed in Table V. Each activity was assigned 2 to 8 hours' effort, but the whole activity was scheduled to be completed in a week. Once completed, two-hour review meetings were scheduled with each OA in order for the consultant to provide feedback on the resulting CM (Activity 2.3 of Table I). An additional week was given to participants in order to refine their models considering this provided feedback, and then they delivered their final models (Activity 2.4 of Table I).

TABLE III. COMPLETE RELATION OF TOPICS ADDRESSED IN TRAINING

Session	Topics
1 (2 hours)	Introduction to Technology Strategic Planning
	Objectives of Technology Strategic Planning.
	Why is Technology Strategic Planning relevant
	General concepts of Porter's model of the five market forces and chain of value
2 (2 hours)	The nature of contemporary information systems: Hybrid Systems
3 (2 hours)	Introduction to i*
	The DHARMA method
	Modelling the enterprise context
4 (2 hours)	The organization, chain of values and technology systems
	Discovering organizations systems and technology architecture
	Defining project portafolio: the DHARMA-PET method

TABLE IV. TABULAR REPRESENTATION FOR I* SD MODELS

Actor	Dependency Type	Dependency	Direction
01.03 Institutes	Goal	Agreements signed	→
	Resource	Agreement Documents	←
	Goal	Technical formation provided	←
	Resource	Teachers	←
	Resource	Infrastructure	←
	Goal	Syllabus Prepared	←
	Resource	Curricula & Contents	←
	Goal	Curricula Approved	→
	Goal	Contents Approved	→
	Goal	Grades and assistance registered	→
	Resource	Grades and assistance records	→
	Resource	Grades and assistance formats	→
	Goal	Enrollment/Inscription recorded	→
	Goal	Admission exams taken	→
	Resource	Regulations and Certificates	→

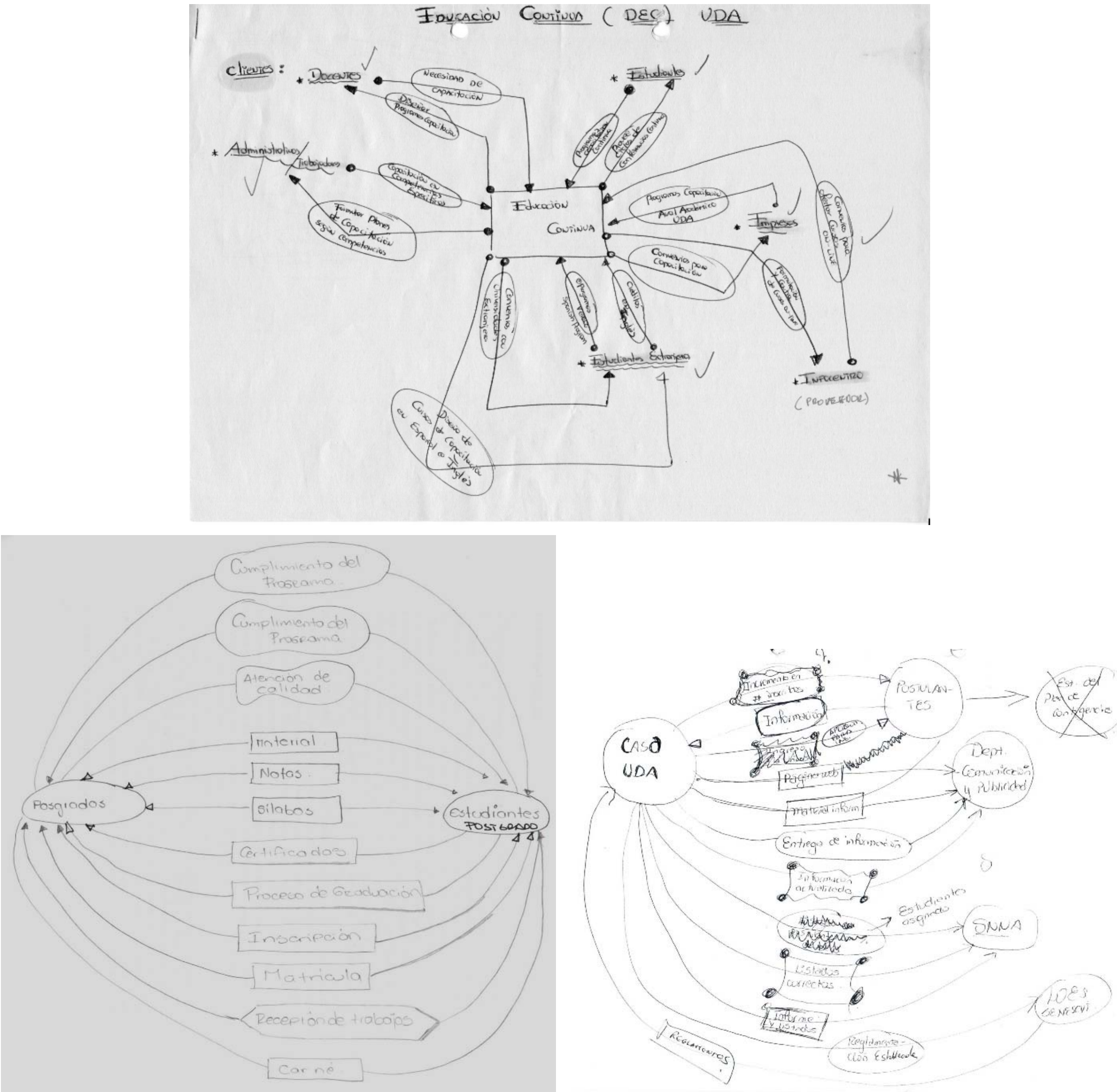

Figure 4. Sample CMs drawn by non-technical stakeholders.

TABLE V. WBS PROPOSED FOR FIRST MODELLING ACTIVITY OA.

Task	Description	Duration	Starts	Ends	Work Product
2.2	**Construct first CM of the OA**	**1w**	**4-jun**	**8-jun**	**1st. Version of CMs from OA view**
2.2.1	Environmental actors identification	4h	4-jun	5-jun	List of environmental actors
2.2.2	Goals identification	8h	5-jun	7-jun	CMs including only goals
2.2.3	Resources identification	2h	6-jun	7-jun	Enriched CMs including resources related to goals
2.2.4	Soft-goals and tasks identification	4h	6-jun	7-jun	Enriched CMs including soft-goals and tasks
2.2.5	Review and consolidation	4h	7-jun	8-jun	1st. Version of CMs from OA view

4.5 ANALYSIS

Analyzing *i** SD models is not an easy task. On the one hand, model construction does not follow any prescriptive method; they are greatly built based on modelers' point of view and perception. On the other hand, as they scale up, the number of graphical elements can become very large making the process extremely hard to manage. In cases like the one described in this paper, the problem is even more challenging: 13 models were built for the different OAs simultaneously, which included common elements which had to be identified and mapped.

To ease the process, we started by identifying what we wanted to validate, namely the elements of the notation and their attributes to be validated. We considered dependencies the central element for the analysis process. To derive the attributes, we further applied GQM to derive the metrics that will be used to answer the research questions, see Table VI. Explanations for each research question are:

- **Q1**. We define understandability as the ability of stakeholders to apply dependencies in a correct way when they build an SD model. When the stakeholders include a dependency, it may happen that it models a concept that exists in the real world (i.e., the organization context) or not (M1.3). In the first case, the representation of the dependency may be correct (M1.1) or not (M1.2). In addition, stakeholders may miss some dependencies that should have appeared in the model (M1.4). As a result, we obtain four metrics.
- **Q2**. In this question we focus in one particular set of dependencies, those identified in Q1 when measuring M1.2 (existing but ill-represented). We investigate the cause of such incorrectness which can be threefold: wrong type, wrong description or wrong direction, and we create a metric for each aspect. Please note that a dependency may suffer from more than one defect; in this case, the dependency is counted as many times as required by every metric.
- **Q3**. This research question goes deep into details related to the type of dependency. We have two sets of four metrics. In the first set, related to M2.1, each metric counts the number of dependencies written by stakeholders with a wrong type. In the second set, related to M2.3, each metric counts the number of dependencies written by stakeholders with a wrong description. In this second set, we made the decision of analysing the description with respect to the type that the stakeholders chose, even if incorrect, since they were supposed to write the description following the guidelines according to that type.
- **Q4**. Finally, in this question we didn't introduce any single metric but instead searched for statistical correlations of stakeholders' information with all the metrics introduced in the previous questions. More precisely, we considered information about job position and profession, and also considered the size of every OA participating in the study.

In order to simplify the analysis process while making it possible to evaluate all these metrics, we conducted the following activities for each of the resulting OA's CM:

- We, the authors, in our role of consultants, transcribed all CMs from the original handmade drawings (in a piece of paper) delivered by non-technical stakeholders, to their tabular representation (Activity 2.5 of Table I), using an Excel spreadsheet. For an excerpt of such tabular model, see Table IV again.
- Identified environmental actors were categorized as *External Context Actors* (ECA) (actors in the context of the organization) and *Internal Context Actors* (ICA) (actors that represent OAs, when they appear as part of the CM of other OA), and assigned an identification code.
- Additional columns (dependency type, direction and description) were added to tables, to be used by the consultants without modifying contents of the ones used by non-technical stakeholders. In these columns, the consultants stated, after their analysis, the correct type, direction and description, of each of the dependencies included by non-technical stakeholders in their CMs. For those cases in which dependencies had to be deleted because of being incorrect, cells in the added columns were left empty.
- Finally, the consultants added rows to include important dependencies missed by non-technical stakeholders in their CMs. In this case, new cells in relation to columns used by non-technical stakeholders were left empty which allowed keeping traceability for later analysis.

Once the review process was completed (Activity 2.6 of Table I, embracing the last three bullets above), workshops with each of the OAs were conducted. These workshops were designed to validate consultants' interpretations in relation to each of the dependencies stated by non-technical stakeholders on their models. The aim was to assure that semantics was preserved once dependencies had been transcribed to the tabular representation (Activity 2.7 of Table I). In this activity, incorrect dependencies were identified and marked.

TABLE VI. USING GQM: METRICS USED IN EVERY RESEARCH QUESTION

Goal: Analyze i* Strategic Dependency (SD) models for the purpose of evaluating dependencies with respect to their understanding and correct application from the viewpoint of different types of non-technical stakeholders in the context of a case study in an Ecuadorian university project	
Research Question	Metric
Q1: Do non-technical stakeholders understand and apply correctly the concept of *i** dependency?	M1.1: Number (and percentage) of existing and well-represented dependencies
	M1.2: Number (and percentage) of existing but ill-represented dependencies
	M1.3: Number (and percentage) of dependencies included that should not be there
	M1.4: Number (and percentage) of missing dependencies
Q2: What are the most common mistakes made by non-technical stakeholders when representing *i** dependencies?	M2.1: Number (and percentage) of incorrect dependencies with wrong type
	M2.2: Number (and percentage) of incorrect dependencies with wrong description
	M2.3: Number (and percentage) of incorrect dependencies with wrong direction
Q3: What are the types of *i** dependencies that are more difficult to understand and apply correctly for non-technical stakeholders?	M3.1-M3.4: Number (and percentage) of incorrect goal/soft goal/task/resource dependencies with wrong type
	M3.5-M3.8: Number (and percentage) of incorrect goal/soft goal/task/resource dependencies with wrong description
Q4: Which are the groups of participants experiencing more difficulties on understanding and applying the concept of dependency in *i**?	Statistical correlations among Mx.y from the previous questions and information about study participants and OA size

Validated CMs were merged into a final CM (Activity 2.8 of Table I). It is required for DHARMA's activity 2, 3 and 4, to depart from a single CM, including all actors and their intentional elements in relation to the organization. The aim of the process is to identify a system architecture for the organization as a whole, instead of a set of architectures for each OAs. The tabular representation helped to manage size and facilitate handling, particularly when models of different OA were put together. The following activities were conducted over the final organization CM.

- Rows were sorted by actor, dependency type, direction and dependency description, using Excel's built-in sorting capacities. This eased the identification and eventually removal of duplicated dependencies identified by more than one OA (including dependencies which were the same but had dissimilar description, type or direction). Columns for each OA were added to the table and used to keep track of duplicated dependencies that were deleted. In these cases, proper cells were marked with an "X" to signpost all the OAs that identified the same dependency. The final CM model after completing this process included: 69 ECA grouped in 6 categories (identified according to the CRM pattern we presented in [18]: *Strategic Partners*, *Control Agencies*, *Regulatory Agencies*, *Direct Customers*, *Suppliers* and *Sales Force*); 53 ICA grouped in 5 categories (*Internal Strategic Partners*, *Internal Controllers*, *Internal Regulators*, *Internal Customers* and *Internal Suppliers*); 1039 dependencies, 532 of them connecting the organization with some ECA and the remaining 507 connecting the organization with some ICA (as the answer to Q1 will show, some of these dependencies were removed).
- These 1039 dependencies needed to be split for further analysis: 800 were provided by non-technical stakeholders from 11 out of the 13 OAs and 239 in relation to these 11 OAs provided by the consultants. Please note that for the consultancy project, we the consultants had to provide the models for two OAs (Legal and External Liaisons OAs) since the non-technical stakeholders didn't manage to find the time to participate. However, for the purposes of this study, these two models (comprising a total of 85 dependencies) have not been considered.
- Mismatches among dependencies stated by non-technical stakeholders and the ones reformulated by consultants were identified in a systematic way; the contents of the dependency type, direction and description columns stated by stakeholders, were compared with the ones stated by the consultants, again using Excel built-in capabilities.

From the consultancy process point of view, the construction of CMs also turned out to be very valuable. Lots of data for cross-validation was made available. It helped to validate completeness of CMs constructed by related OA, but also to identify relevant dependencies on external environmental actors identified by more than one OA.

5. ANSWERS TO THE RESEARCH QUESTIONS

Table VII presents a summary of the numbers resulting from the process in the 11 OAs mentioned above. Table VIII summarizes the total numbers resulting from all areas. The detailed analysis of the results is presented next by answering the research questions stated in Section 4.2.

TABLE VII. EXCERPT OF THE DATA TABULATED FROM THE CONTEXT MODELS OF 11 OAS.

Area	Actors: External (ECA)	Actors: Internal (ICA)	Dependencies: Included in CMs by		Total	Goal	Soft-Goal	Task	Resources
Faculties	15	17	**Included in non-technical stakeholders CM**		**184**	**146**	**2**	**1**	**35**
			Incorrect	Type	44	40	0	0	4
			Incorrect	Description	132	120	0	0	12
			Incorrect	Direction	63	59	0	0	4
			Incorrect	Deleted	3	3	0	0	0
			Correct		36	14	2	1	19
			Added by consultant		**81**	**15**	**4**	**4**	**58**
Graduate school	8	10	**Included in non-technical stakeholders CM**		68	21	5	16	26
			Incorrect	Type	17	8	1	4	4
			Incorrect	Description	39	18	3	5	13
			Incorrect	Direction	17	8	2	5	2
			Incorrect	Deleted	4	1	0	3	0
			Correct		8	3	2	1	2
			Added by consultant		41	13	11	4	13
Professional / Continuing Education	10	4	**Included in non-technical stakeholders CM**		**54**	**54**	**0**	**0**	**0**
			Incorrect	Type	21	21	0	0	0
			Incorrect	Description	41	41	0	0	0
			Incorrect	Direction	12	12	0	0	0
			Incorrect	Deleted	1	1	0	0	0
			Correct		3	3	0	0	0
			Added by consultant		**30**	**10**	**8**	**1**	**11**
Financial Direction	7	9	**Included in non-technical stakeholders CM**		56	30	1	3	22
			Incorrect	Type	4	2	0	0	2
			Incorrect	Description	15	10	0	1	4
			Incorrect	Direction	2	1	0	0	1
			Incorrect	Deleted	1	0	0	0	1
			Correct		40	19	1	2	18
			Added by consultant		8	3	1	1	3
Treasure	3	7	**Included in non-technical stakeholders CM**		**26**	**12**	**6**	**0**	**8**
			Incorrect	Type	2	1	0	0	1
			Incorrect	Description	7	3	2	0	2
			Incorrect	Direction	12	4	3	0	5
			Incorrect	Deleted	0	0	0	0	0
			Correct		8	4	1	0	3
			Added by consultant		**11**	**4**	**3**	**1**	**3**
Human Resources			**Included in non-technical stakeholders CM**		52	16	18	0	18
			Incorrect	Type	1	1	0	0	0
			Incorrect	Description	15	13	1	0	1
			Incorrect	Direction	0	0	0	0	0
			Incorrect	Deleted	0	0	0	0	0
			Correct		36	2	17	0	17
			Added by consultant		0	0	0	0	0
Administrative Coordination	11	3	**Included in non-technical stakeholders CM**		**39**	**39**	**0**	**0**	**0**
			Incorrect	Type	12	12	0	0	0
			Incorrect	Description	33	33	0	0	0
			Incorrect	Direction	6	6	0	0	0
			Incorrect	Deleted	0	0	0	0	0
			Correct		1	1	0	0	0
			Added by consultant		**14**	**8**	**0**	**1**	**5**
Research Deanship	13	3	**Included in non-technical stakeholders CM**		**91**	**88**	**2**	**0**	**1**
			Incorrect	Type	40	39	1	0	0
			Incorrect	Description	68	67	0	0	1
			Incorrect	Direction	63	61	1	0	1
			Incorrect	Deleted	7	7	0	0	0
			Correct		6	3	3	0	0
			Added by consultant		**35**	**9**	**9**	**3**	**14**
Students Welfare	15	5	**Included in non-technical stakeholders CM**		84	39	8	2	35
			Incorrect	Type	3	0	0	0	3
			Incorrect	Description	24	10	1	0	13
			Incorrect	Direction	5	2	1	0	2
			Incorrect	Deleted	2	0	0	0	2
			Correct		58	27	6	2	23
			Added by consultant		14	2	8	0	4
Communica-tions Department	6	12	**Included in non-technical stakeholders CM**		94	91	0	0	3
			Incorrect	Type	41	39	0	0	2
			Incorrect	Description	57	57	0	0	0
			Incorrect	Direction	4	4	0	0	0
			Incorrect	Deleted	0	0	0	0	0
			Correct		14	13	0	0	1
			Added by consultant		1	0	0	0	1
Libraries	9	4	**Included in non-technical stakeholders CM**		52	14	19	0	19
			Incorrect	Type	10	5	2	0	3
			Incorrect	Description	31	11	13	0	7
			Incorrect	Direction	22	5	11	0	6
			Incorrect	Deleted	2	1	1	0	0
			Correct		14	3	3	0	8
			Added by consultant		4	1	1	0	2

TABLE VIII. TOTAL NUMBERS OBTAINED IN THE STUDY

Area	Actors		Dependencies						
	External (ECA)	Internal (ICA)	Included in CMs by		Total	Goal	Soft-Goal	Task	Resources
Total	69	53	Incuded in non-technical stakeholders CMs		800	550	61	22	167
			Incorrect	Type	179	155	3	6	15
			Incorrect	Description	448	372	19	8	49
			Incorrect	Direction	217	155	17	7	38
			Incorrect	Deleted	20	13	1	3	3
			Correct		221	90	35	6	90
			Added by consultant		239	65	45	15	114

Q1: Do non-technical stakeholders understand and apply correctly the concept of *i dependency?**

We found the results of the study very conclusive: non-technical stakeholders identified 800 dependencies with very little training, thus we think that numbers speak by themselves. Of course, it could have happened that these 800 dependencies were basically wrong, stating relationships that are not really dependencies; or the other way round, that in spite of this large number, still a significant number of other dependencies were missing. The numbers that we provide are again illustrative (see Table IX): from these 800 dependencies, the consultants considered that 97,5% were really needed (only 20 had to be removed either because they were redundant with dependencies identified by other areas or because they were covered by other dependencies in the model), whilst the consultants, in the consolidation process, found 239 missing dependencies (239 over 1039, i.e. 23%).

We analyzed the missing dependencies to understand why they were not identified. We found two main reasons: 1) non-technical stakeholders didn't pay enough attention to non-functional requirements that the consultants knew were important (e.g., security aspects), which can be seen in the percentage of soft goal dependencies added over the total, 65 out of 269 (27%); 2) some dependencies where not properly refined (e.g., goal dependencies were sometimes not elaborated into resource or tasks dependencies to make them more concrete). This second point is interesting, pointing out the asymmetry between Strategic Rationale models where different type of decompositions exists, and SD models, where the concept of dependency refinement cannot be represented.

Q2: What are the most common mistakes made by non-technical stakeholders when representing *i dependencies?**

As explained in Section 2.1, a dependency between two actors A and B is characterized by: the type of the dependency, which is determined by the type of its dependum (goal, soft goal, task or resource); the direction of the dependency (from A to B, or from B to A); the description of the dependum. The type and direction are a consequence of the graphical representation of the dependency in the model, while the description is textual. Precise guidelines for writing descriptions were given in the training sessions (see Section 4.4.1).

The number of errors are presented in Table X and detailed below.

Type. 179 dependencies out from 780 (i.e., 22,9%) were represented with a wrong type according to the intentions that was modelled by the dependum. Question Q3 provides the details for this part.

Direction. 72,2% of the dependencies identified by non-technical stakeholders (563 out of 780), were stated with a correct direction. The remaining 27,8% (217 dependencies) required to be redirected. The number is surprisingly high but when going into the details, the usual case was that various participants wrote all the dependencies in some of their models in the wrong direction, leading to this high percentage. When reflecting on this fact, it is worth to remind that some authors have already complained about the way to represent directions in *i** dependencies: a "D" to represent "depends on" which is not a very descriptive mnemonic [5][6]. The recent iStar 2.0 core standard [17] recognizes this graphical drawback as an important one that calls for a thorough study based on physics of notation [21].

Description. 448 dependencies out of 780 (57.4%) had an incorrect description according to the guidelines given. This is a high percentage but it must be said that the guidelines given were quite strict, with the goal of the description reinforcing the semantics of the dependency; therefore, violating some aspect the first time they used the notation was not considered extraordinary. Again, question Q3 provides more information in terms of the type of goal.

There are also dependencies which presented more than one type of error (see Figure 6). Out of 780 dependencies, 75 (9,6%) had wrong type and description, 125 (16%) had wrong description and direction, 19 (2,4%) had wrong direction and type, and finally 33 (4,2%) had wrong type, description and direction.

Q3: What are the types of *i dependencies that are more difficult to understand and apply correctly for non-technical stakeholders?**

Results point to the fact that the semantics of the four types of dependencies are easily understandable by non-technical stakeholders. Table XI shows the relationship among the numbers of dependencies of each type stated by non-technical stakeholders on their CMs, and the types that were reformulated by consultants after reviewing them. As mentioned in Section 4.4, workshops with non-technical stakeholders after their models were transcribed to tabular form, helped to clarify the semantics they had in mind. Still, several remarkable facts emerged:

TABLE IX. RESULTS OF THE EVALUATION OF METRICS IN RELATION TO RESEARCH QUESTION Q1

Metric	Number	% in relation to dependencies identified by non-technical stakeholders (800)	% in relation to total dependencies in final model (1039)
M1.1: Number (and percentage) of existing and well-represented dependencies	221	27,6%	21,3%
M1.2: Number (and percentage) of existing but ill-represented dependencies	559	69,9%	53,8%
M1.3: Number (and percentage) of dependencies included that should not be there	20	2,5%	1,9%
M1.4: Number (and percentage) of missing dependencies	239	0,0%	23,0%

TABLE X. TYPES OF ERRORS WHEN REPRESENTING DEPENDENCIES

Metric	Number	% in relation to dependencies identified by non-technical stakeholders (780)
M2.1: Number (and percentage) of incorrect dependencies with wrong type	179	22,9%
M2.2: Number (and percentage) of incorrect dependencies with wrong description	448	57,4%
M2.3: Number (and percentage) of incorrect dependencies with wrong direction	217	27,8%

TABLE XI. RELATION AMONG ORIGINAL AND REFORMULATED DEPENDENCIES IN THE FINAL MODEL

Metric	Number	Type stated by non-technical stakeholders	Type Stated by Consultans	Dependencies with wrong type	
				Number	%
M3.1-M3.4: Number (and percentage) of incorrect goal/soft goal/task/resource dependencies with wrong type	550	Goal	Soft Goal	38	7%
			Resource	92	17%
			Task	24	4%
			Deleted	13	2%
	61	Soft Goal	Goal	3	5%
			Resource	0	0%
			Task	0	0%
			Deleted	1	2%
	22	Task	Goal	0	0%
			Soft Goal	0	0%
			Resource	6	27%
			Deleted	3	14%
	167	Resource	Goal	7	4%
			Soft Goal	2	1%
			Task	5	3%
			Deleted	3	2%

- Goal dependencies are the ones with more mistakes identified: as much as 154 out of 550 (28%) goal dependencies require a change on their type. A total of 92 (17%) referred to the resource required to achieve a goal but such resources were drawn by non-technical stakeholders using the symbol of goal dependencies. On the other hand, 38 (7%) should have been stated as soft goal dependencies since there were not enough warranties to grant their achievement; e.g., the *Services Provided Timely* goal should be changed into a *Timely Services* soft goal. Finally, 24 (4%) were stated as tasks since there is only a particular way to achieve them; e.g., *Admission exams taken* which should be changed into *Admission exams taken at university facilities.*

  We think that the main reason for Goals to be confused with Tasks or Resource dependencies is that non-technical stakeholders tend to think on concrete daily-base activities used to achieve their goals (probably implemented on preexisting systems) or their concrete results, more than on the abstract goals that they are trying to attain. Regarding Goals that required to be reshaped as Soft Goals, we think that the main reason is that for most non-technical stakeholders, functional requirements (which may take the form of goals, tasks or resources) are more concrete and thus easier to identify. On the other hand, the identification of non-functional requirements requires a more advanced training to make them evident (although the guidelines discussed in the third bullet of this section helped to reduce this problem). This is true even in the case of technical personnel, since most of the structured or object-oriented analysis methods and artifacts, e.g. diagrams and notations, have been engineered to identify and document functional requirements.
- Regarding Task dependencies, 6 out of 22 (27%) were confused with resources. This mistake is easy to understand. Non-technical stakeholders referred to the action of obtaining particular reports instead of stating the report as the required resource (e.g., the task "*print invoice*" instead of the resource "*Invoice*"). Instead, just 5 out of 167 (3%) of resources had to be reformulated to tasks, and only a few more to goals and soft goals.
- Surprisingly, soft goals, which are considered by many the most semantically cumbersome type of *i** dependencies [6][7][10], are the ones with fewer number of mistakes in the study. Just 3 out of 61 (5%) had to be converted to Goal dependencies. We think that the reason for this is the way that soft goals were introduced in the training course.

Instead of focusing on the concept of non-functional requirement, which is probably closer to technicians than to the stakeholders involved in this study, soft goals were defined as goals that required an agreement among involved parties in order to decide their fulfilment. This agreement-based vision is closer to the language that non-technical stakeholders use and therefore it is more natural for them. On the downside, as mentioned above, some non-functional concepts were missing in their models.

In addition, we studied the errors related to the dependum description from the point of view of the type. This analysis revealed that there is not a single cause for the problem. Table XII provides details of the most common causes. It can be seen again that goals are considerably more error-prone than the rest of dependency types, while soft goals suffer from less errors. However, it is important to remark that the type of errors in both types of goals is quite different: while the main error with goals has to be with their statement, in the case of soft goals they have to be with the level of detail. In any case, it is worth to consider that the percentages given for soft goal apply to a low number, namely the 8 soft goals in which some error was found. In the case of resources, we find a combination of both types of errors (statement and excess of detail) while for tasks we have again very few erroneous instances so that the percentages need to be considered with caution. As a final number, only 46 out of 298 (15.4%) of the total number of errors remains as unclassified ("other", in the table).

TABLE XII. TYPICAL CORRECTIONS REQUIRED IN DESCRIPTIONS PROVIDED BY NON-TECHNICAL STAKEHOLDERS.

Metric	Total dependencies		Wrong description		Typical cases of corrections required in descriptions			
	Nb.	Type	Number	%	Problem	Number	%	Examples
M3.5-M3.8: Number (and percentage) of incorrect goal/soft goal/task/resource dependencies with wrong description	550	Goal	245	45%	Missing verb	66	27%	"Social security Payroll" replaced by "Social
					Incorrect verbal tense	110	45%	"Generate payment voucher" replaced by "Payment voucher generated"
					Wrong verb used	32	13%	"Agreements Realized" changed by
					Other	37	15%	Spelling, missing articles, etc.
	61	Soft Goal	8	13%	Overstatement of description	6	75%	"Process standardization both for classification and categorization" replaced by "Process standardization"
					Other	2	25%	Spelling, missing articles, etc.
	167	Resource	39	23%	Verb included	12	30%	"required books" replaced by "Books"
					Missing noun	4	10%	"Authorized" replaced by "Authorized snacks"
					Overstatement of description	18	46%	"Laboratory certificated, English sufficiency certificate, Computing sufficiency certificate" replaced with "Graduation Folder"
					Other	5	14%	Spelling, missing articles, etc.
	22	Task	6	27%	Wrong action	4	66%	"Web visualization" replaced with "Publish web
					Other	2	34%	Spelling, missing articles, etc.

Q4: Which are the groups of participants experiencing more difficulties on adopting and understanding the concept of dependency in *i**?**

In order to answer this question, we considered the three dimensions included in Table II (number of participants, job position and profession), and the results for each OA shown in Table VII (total number of dependencies and number of correct dependencies identified). We checked the total amount of dependencies as an indicator of potential difficulty on understanding the model (the more dependencies, the harder to understand).

We did not find any statistically significant correlation among the number of dependencies identified and the three dimensions included in Table II. However, results point to the fact that the number of dependencies has a direct relation to the number of actors that each OA interacts with (i.e., actors in their context), as can be seen in the tendency line depicted in Figure 5. This may turn interesting for future studies on valuing job positions or importance of OA in organizations.

However, we did find a statistical correlation among the quantity of correct dependencies and the job position in the organization. We used the analysis of variance (ANOVA) which is intended to determine if the means of at least three groups or more are different if compared to an overall mean. In our case, the null hypothesis (H0) considers the means of the four groups of job positions, Operational (O), Mid-Manager (MM), Manager/Operational (MO), and Manager (M) are equal, H0 : $\mu_O = \mu_{MM} = \mu_{MO} = \mu_M$, whilst the alternate hypothesis (HA) considers that at least one them is different. ANOVA uses F-tests to statistically test the equality of means in order to be able to reject H0. In order to do so, a small p-value is required, which requires a large F statistic (for more details on how to perform this test please refer to [28]). In our case, F-value resulted in of 7.5, and a p-value of 0.019 [1]. As shown in Table XIII, the number of correct dependencies increases as the job position of participants tends to be more operational (members of OA who perform and conduct business operations) and decreases as the job position tends to be more managerial (members of OA who lead operational stakeholders and perform strategic activities). It is important to notice that the Communications and the Students Welfare OA, included teams of participants with different job positions, operational and managerial, and these two OA performed better than the two OA

[1] Data set and R source code for the test may be downloaded from http://www.upc.edu/gessi/iStarNonTechnical/Correlation.zip.

that included only managers. As a matter of fact, the two areas omitted from Table VII, were omitted because at the end, managers involved in the process had no time to construct the models and a traditional approach had to be adopted, that is, they acted as informants whilst consultants had to construct the models based on information provided. Therefore, we may speculate that the main reason for the higher error rate in managerial stakeholders is more related to their lack of time for such operational activities rather than to their lack of knowledge or skills. Further empirical studies are needed to confirm this hypothesis.

Another important fact to consider is that the Human Resources department has been omitted from Table XIII, since its manager didn't participate in the same training program than the others. Instead, the CM of this OA was constructed after its manager got involved in a master's degree, in which the same contents were provided with increased number of hours of dedication. As it can be seen in Table VII, the number of correct dependencies identified for this area (correct type, direction and description) borders 70%. This is more than twice the percentage of correct dependencies identified by other areas (even if the job position is managerial instead of operational). This fact points to training as a very relevant issue for the process, which we will address in future studies.

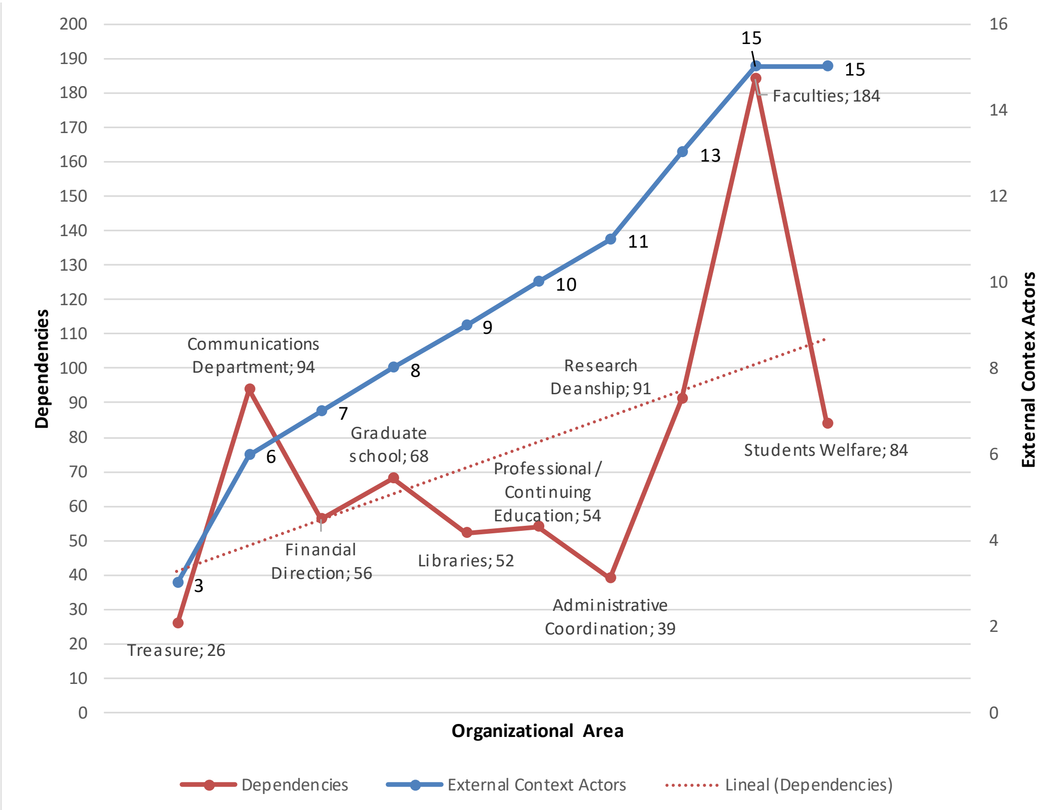

Figure 5. Tendency among number of external actors identified in relation to OAs and the number of dependencies identified.

TABLE XIII. RELATION AMONG CORRECT DEPENDENCIES AND JOB POSITIONS OF NON-TECHNICAL STAKEHOLDERS.

Dependencies		Operational Area (OA)	Nb. of Participants	Job Position	Profession
Nb.	Correct				
26	30%	Treasury	1	Operational	Accountant
91	28%	Research Deanship	1	Operational	Systems Engineer
52	27%	Libraries	1	Operational	Librarian
39	26%	Administrative Coordination	1	Mid-manager	Economist
184	20%	Faculties	6	Mid-manager	Lawyers
68	12%	Graduate school	1	Mid-manager	Business Administrator
94	15%	Communications Department	2	Manager / operational	Communicator
84	14%	Students Welfare	3	Manager / operational	Psychologists
56	13%	Financial Direction	1	Manager	Business Administrator
54	6%	Professional / Continuing Education	1	Manager	Business Administrator

6. Threats to Validity

As any other empirical study, ours is subject to threats that we discuss below.

Conclusion validity. Being *i** a notation with a high degree of freedom, especially with respect to the level of detail of the models, it may be argued that the role of "oracle" played by the consultants in classifying dependencies as right or wrong may have biased the results. At this respect, it is worth to mention that both authors (playing the role of consultants) have long experience on the use of *i**, therefore this risk is highly reduced. We want to remark that, given this freedom, we decided not to build a solution model in advance, but assess the models built by the stakeholders as they were produced.

Internal validity. Being a project "in the wild", we have been very careful in isolating unavoidable confounding factors that could endanger the conclusions that we were looking for. We have shown how 2 out of the 13 OAs have been left out of most of the analysis process because the stakeholders in these areas could not commit to deliver on time. Also, we have explicitly mentioned the fact that one of the stakeholders in the remaining 11 OAs had a different kind of training, more complete than the others. We couldn't avoid these situations but took mitigation actions as described along the paper. Concerning population, the selection of the OAs was external to our study, it came directly from the Rector; a risk is thus that some representative OA could have been left out. In any case, a look to the areas involved allows checking that there are the ones that one could reasonably expect will be usual participants in these types of projects. Selection of individuals was performed after interviews searching for the most appropriate individuals in terms of success of the project and not success of the study, mitigating then the risk of selecting the most convenient people for the vested interests of the researchers. We acknowledge the fact that some areas included more participants than others. Although this fact could have had some influence in the results, we did not observe a significant relation among the number of participants per area and the quality of the dependencies identified. On the other hand, we did not want to impose methodological issues that could have impacted negatively into the project schedule or the perception of usefulness by the involved stakeholders, other than those required to grant enough thoroughness in the experience.

We made sure of performing a rigorous planning of the study and establishing a solid protocol and templates for data collection and data analysis by following guidelines for software engineering [22]. Data collection has been intentionally loose due to the nature of the study, but instead data analysis has been carefully undertaken, as illustrated in the paper. Concerning the training material, it has been devised from long ago and it is even used in an MSc course in U. Cuenca (Ecuador), therefore it can be considered adequate.

External validity. We recognize that our results are tight to the context where we conducted the study and should be interpreted as such. In particular, we are aware that the number of participants in the study is not very high, as it happens often in qualitative studies that involve participants from industry, like this one. Therefore, the results obtained in the paper need to be considered in the context of the study. Furthermore, we acknowledge that several factors that were not explicitly requested in our study may influence the subject under research, such as organizational processes and policies, resources, professional background, education and cultural issues. Therefore, we do not try to make claims that cannot be sustained. Further studies need to be conducted to understand better the factors that may influence the results obtained in this particular study.

We also acknowledge the fact that previous studies on the use of the DHARMA method have been conducted in the same location; therefore, cultural and local factors could be a threat to the generalization of the findings.

7. Methodological advices

During this study, we have acquired valuable knowledge in relation to the comprehension and usage of *i** dependencies by non-technical stakeholders. Some of this knowledge comes from our observations, some other from the informal feedback and opinions given by the study participants along its execution. In order to make it actionable, we have structured the information in the form of methodological advices. Ten of these advices were already identified in a previous paper [7] (labelled as lessons learned) where they were explained in more detail; in this section, we summarize these 10 advices and add 5 new ones. Furthermore, we have grouped them in relation to four main process phases: introduction (4 advices), construction (6 advices), consolidation (3 advices) and project management (2 advices).

7.1 Introduction Phase

Methodological advice 1: Provide effective training on the *i framework**

Given that non-technical stakeholders will not usually know the *i** framework (nor the DHARMA method) in advance, training is mandatory. In our case study, without training, they would not have been able to participate actively in the process. However, it is important to constrain such training to information that will add value to the process and therefore keep non-technical stakeholders interested on its outcome. We consider that the ability on identifying dependencies demonstrated in this case study is a good indicator on the level of training required. Of course, margin for improvement always exist, but we observed that the training workload that we provided was near the limit the stakeholders were receptive to receive.

Methodological advice 2: Provide a roadmap to perform the work

Even after training, because of lack of experience, modelling activities and their objectives can be fuzzy to non-technical participants. They won't have the notion of time span or deliverables to be produced as result of each activity. In order to improve the process, participants should be provided with a detailed schedule containing the work breakdown structure (WBS), including main modelling activities, their task decomposition, time assigned for their fulfilment and deliverables to be produced. In task decomposition, it is important to be as atomic as possible in order to ease project follow up and management. An example of such decomposition can be seen in Table V (Section 4.4.2).

Methodological advice 3: Provide guidelines to improve quality

Non-technical participants tend to do their "best effort" and justify poor quality of results based on lack of experience and training. Best practices shall be introduced to improve quality of their work products. Fortunately for most graphical notations, *i** not being an exception, there exist several best practices documented, as well as tacit knowledge emerging from consultants' own experience, which can be transferred to non-technical participants. Best practices shall alert non-technical stakeholders on their responsibility and implications for the outcome of the process. As described in Section 4.4.1, we issued some of the guidelines included in [4][7][18], which proved useful for non-technical stakeholders in practice. However, they need to be introduced with caution for the reasons that will be explained in Advice 6 below.

Methodological advice 4: Provide clear naming conventions.

As reported in Table XII, one of the sources of errors by non-technical stakeholders is the incorrect naming of intentional elements. Incorrect naming hampers model understanding, since it may not be clear if the problem is a wrong name or a wrong type of dependency. Given the high percentage of such type of errors over the total, a clear advice is to provide even clearer naming conventions in the training material, following recommendation given by several authors [16][19].

7.2 CONSTRUCTION PHASE

Methodological advice 5: Help users to manage size

A common setback shared by most graphical modelling notations is the difficulty to manage drawing when models scale up. Advice shall be provided to non-technical participants to avoid over-scaling their models, and facilitate their handling. In fact, the decomposition of the big system model into smaller pieces corresponding to CMs for each OAs and the use of the guidelines provided in the two first bullets of Section 4.4.1, have demonstrated to be good tactics to handle such complexity. In fact, one of the lessons learned in this case study is that stakeholders are able to cope with the complexity of these models: none of them needed to use the tabular notation proposed in [11] (see Table IV), and only consultants found this representation necessary in order to consolidate the partial CMs into one single CM for the organization, whilst keeping adequate traceability of changes made.

Methodological advice 6: Avoid the use of specialized tools.

Mastering the use of specialized modelling software can be a time consuming and reward less activity, particularly in cases where users may never use the tools again, as it was the case of this study. In addition, this can unnecessarily increase project time span and costs. For this reason, we recommend to lead non-technical stakeholders creating their own hand-drawn models. Although this will produce several hand-drawn CMs with odd notations, e.g., the ones presented in Figure 4, this will help them to focus in more relevant issues, such as the identification of intentional elements. It is important to remark as supporting argument that these models are not supposed to be used afterwards (e.g., as starting point in a model-driven process, in which case some properties would be required [23]). It is also important to keep in mind, as mentioned in the introduction, that the final aim is not to train non-technical stakeholders in the use of specialized tools, but to empower them to more proactively participate in early phases of requirements modelling activities.

Methodological advice 7: Do not over-constrain user's imagination.

Modelling with non-technical stakeholders is a creative process, and particularly at the intentional level that *i** addresses, as reported by several authors who have proposed the use of *i** in a creativity context [24]. Overtraining non-technical users or excessive guidelines may constrain their thoughts, leading them to skip aspects that can be very relevant for the process. Giving value to free thinking is an overall recommendation, because in some cases this may lead to significant contributions, both for the process and the methods used to support it. For instance, when we asked non-technical stakeholder to identify environmental actors, we meant actors in the context of the organization (ECA). However, when creating the CMs from the perspective of their own OA, non-technical stakeholders also identified other OAs of the organization as actors on their context (ICA), as shown in Figure 4. This created the need of including dependencies between these actors. At the end, this indicates the need of improving the first activity of the DHARMA, dividing it into two activities: modelling the environment of the organization from the perspective of each organizational area and merging resulting models into a single organizational model. In this way not only contextual dependencies are identified (dependencies among OAs with ECA), but also dependencies among internal areas of the organization (dependencies among OAs with ICA). This is a very relevant issue, since it allows for designing processes of system architecture, required to support both external and internal needs, something that was missing from the original DHARMA method, which considered only external dependencies. Our current work is also going in this direction.

Methodological advice 8: Do not require excellence in the use of framework elements.

Since we led non-technical participants to draw freely, it is quite normal for their CMs, to contain several flaws. This can be seen through the results, tables and figures presented in Sections 4 and 5. However, at some point work products shall be reviewed by consultants anyway, and eventually be transferred to specialized tools. In spite of these mistakes, experience shows that corrections required are very simple to implement, as seen in this work: changing dependencies directions, their type or their wording, and by the like. See next two advices in relation to this.

Methodological advice 9: Apply patterns to develop the models.

Reuse of knowledge is an important methodological asset when modelling. Different domains may have their own concepts that appear over and over in subsequent projects. In the case of enterprise architecture, we have developed a set of patterns [18] that include a catalogue of actors that has been used in this case study (see third bullet of Section 4.4.1 and Section 3.2 for some details). As a consequence, the stakeholders involved in this case study have identified the right actors without much difficulty. Once this knowledge is augmented with dependencies, the construction of *i** SD models will be more effective and efficient.

Methodological advice 10: Review the models continuously and provide timely feedback.

Regardless of training, roadmaps and guidelines, we have observed in previous experiences that non-technical stakeholders require continuous (but decreasing) feedback in relation to their intermediate and final work products. Review meetings shall be scheduled in various milestones in order to analyse work products and provide advice required to refine their quality. In some cases, several iterations with different objectives are required. For instance, in sections 4.4.2 and 4.5 of this document, different types of meetings have been introduced, not only to response non-technical stakeholder's doubts and provide advice on their models, but also to gather feedback required by consultants to validate their interpretation in relation to elements included in their models.

7.3 CONSOLIDATION PHASE

Methodological advice 11: Plan for validation activities.

Analyzing *i** graphical models is not an easy task. On the one hand, models are not enforced by any prescriptive method; they are greatly built based on modellers' point of view and perception. On the other hand, as it was mentioned earlier, the number of graphical elements included can become very large making the process very hard to manage. Several activities shall be engineered in order to make analysis a manageable task (see Section 4.5 for some examples); these may include identification of relevant elements of the notation, construction of specialized supporting artefacts, transcription of original models to them and, identification and categorization of common instances of relevant elements incorporated in several models. In the usual case, validation workshops will be required in order to validate that the transcription of elements and interpretation of consultants aligns with original semantics.

Methodological advice 12: Use models constructed by related areas for cross-validation

As mentioned in Section 4.5, in addition to ECA, CM constructed by a given OA typically include ICA, representing other OAs acting as dependers or dependees in the model (see Table IV for an excerpt of dependencies identified in our case, among the Faculties OA and the Institutes ECA). This fact can be used as a validation mechanism: if OA2 appears as internal context actor in a dependency of a CM1 constructed by participants of OA1, then, that same dependency shall appear in the CM2 when constructed by the participants of the OA2, with OA1 acting as an internal context actor in that model. If a needed dependency is missing from CM2, it can mean either that participants of OA1 made a mistake when building he model or the other way round. This validation procedure has proved be good to spark discussion among participants in related OA and eventually improve quality of their CMs by including new dependencies or refining existing ones. In this way, dependencies included in one CM can be used as evaluation checklist for the dependencies included in CMs constructed by other OAs to/from which intentional elements exist in the model.

Methodological advice 13: Be aware of consolidation activities

Many empirical studies will require individual models to be constructed and later consolidated into a final model, encompassing all of the elements included in individual ones. Far from easy, this activity could become one of the most difficult tasks to achieve in the empirical evaluation process. Converting *i** models to a common tabular representation and merging them into an Excel sheet, using the activities and guidelines provided in Section 4.5, can greatly improve this process. Excel's built-in sorting capacities help identifying and eventually eliminating duplicated dependencies, incorporated in several models (including dependencies which were the same but had dissimilar description, type or direction).

7.4 Project management

Methodological advice 14: Keep up participant's motivation

Day to day events tend to turn participant's attention towards operational activities, closer to their positions in the organization. Some effort must be placed in order to keep them focused on modelling activities. The effort increases for managerial levels when compared to the one expend in operational ones; managers often lack of the pressure that higher supervision imposes over more operational positions (this may explain some of the results in relation to Q4 in Section 5). Some of the previous advices, e.g. 2, 3 and 4, may help to achieve this goal; however, in this study it has become clear that they may not be enough (given that the stakeholders of two areas didn't participate at the end). The introduction of gamification, e.g. internal competitions among participant's or achieving bonuses, could be a way to keep them focussed and committed to its objectives.

Methodological advice 15: Keep sponsors involved and informed

Well established and committed sponsorship is a key component in successful projects [25]. High-level managers define organizations strategy and priorities, without their continuous support projects may be considered of little relevance and thus undervalued by participants. Some responsibilities on the project have to be assigned to sponsors, e.g. the selection of areas and their participants in the project; assignment of permissions and time slots for their participation; review of progress reports; prioritization of interview activities and resulting projects portfolio and deliverables approval; among others (see sections 4.3 and 4.4 for details in our case). Particularly, activities 1.1, 1.2, 2.4, 3.6 and 4.6 of the schedule presented in Table I, were included to keep sponsors involved through the project. In addition to being review milestones, these meetings were used to remain participants of their responsibilities and motivate them in the achievements of the goals in the project. Tables presenting relations among participant areas and their progress in relation to main project milestones (e.g. training concluded, initial models constructed, models reviewed by consultants, models approved, etc.), can be used as support for executive presentations.

7.5 Reflections on methodological advices

With the exception of methodological advices 7, 8 and 13, which are specifically oriented to improve project execution (particularly to help consultants manage the project time span), the remaining methodological advices focus on the improvement of typical problems identified in this work (see Section 5 for details and Section 8, bullets Q1 to Q4 for a summary). In particular, significant number of discarded dependencies (related to metric M1.3 in Table VI), wording of dependencies (related to M2.2), incorrect type and direction of identified dependencies (related to M2.1 and M2.3 respectively), and reduced manager's involvement (related to the metric for correlation involving types of study participants). See Table XIV for the relationship among these problems and the proposed methodological advices.

We observed that these guidelines were essential for the achievement of non-technical stakeholders tasks in this project. However, we are aware that there exists some room for their improvement. Training is a particularly critical issue. On the one hand, as stated in the methodological advice 1, it is needed for non-technical stakeholders not only to gain basic knowledge in the notations and artifacts that they will be using to perform and support their work, but also to understand what is expected from them in terms of involvement and deliverables in the project. On the other hand, in industrial settings, availability for this kind of specialized training can be very limited, not only because it is highly constrained by participant's daily activities, but also because some of the knowledge gained may not be used again in their common tasks. Being aware of this reality, we believe that the training seminar has to be significantly improved in future editions, both in terms of coverage (e.g., to include additional topics as non-functional requirements and their analysis), but also in terms of the exercises to be performed in class (e.g., to help them better recognize when to use goals and soft goals, tasks or resources, and also to improve dependencies naming).

Other important aspect to be improved in future works is, in relation to the methodological advice 3. It is not only important to provide individual guidelines to improve quality, but also a method for non-technical participants to perform model construction. |The method shall formalize at least, the techniques required to identify ECA and ICA, the order for the identification of dependencies accordingly to their type, guidelines to identify soft goals and the cases where it is recommended to use them instead of goals, and finally some guidelines to validate resulting models.

8. Discussion and Future Work

In this paper, we have reported an empirical study about the perception and mastering of the *i** framework by non-technical stakeholders with focus on one particular construct, dependencies. Although the paper describes the experience in relation to the involvement of non-technical stakeholders in the first activity of the DHARMA method, the fact is that the four activities of the method were completed in this case study. The process concluded successfully with the definition of 28 projects of different nature, to be implemented by the university in a period of four years. Estimated budged for their implementation bordered 1'000.000,00 USD. As a remarkable observation, all the stakeholders involved in this study considered that the actions described in this paper were useful for this positive conclusion.

TABLE XIV. RELATION AMONG METODOLOGICAL ADVICES AND PROBLEMS THAT THEY HELP TO IMPROVE.

Phase	No	Guideline	Reduce number of discharded dependencies	Wordings in dependencies	Improve dependencies identification (Goals, Soft Goals)	Improve managers involvement	Improve project execution
1.1 Introduction Phase	1	Provide effective training on the i* framework	I	I	I		I
	2	Provide a roadmap to perform the work				I	I
	3	Provide guidelines to improve quality	I	I	I		I
	4	Provide clear naming conventions.		I			
1.2 Construction Phase	5	Help users to manage size	I			I	I
	6	Avoid the use of specialized tools.				I	I
	7	Do not over-constrain user's imagination.					I
	8	Do not require excellence in the use of framework elements..					I
	9	Apply patterns to develop the models.	I	I	I		I
	10	Review the models continuously and provide timely feedback.	I	I	I		I
1.3 Consolidation Phase	11	Plan for validation activities.				I	I
	12	Use models constructed by related areas for cross-validation		I	I		I
	13	Be aware of consolidation activities					I
1.4 Project management	14	Keep up participant's motivation				I	I
	15	Keep sponsors involved and informed				I	I

If we focus on the study itself, we have formulated four research questions for which we summarize the answers, based on the interpretation of the results summarized in Figure 6

- Q1: Do non-technical stakeholders understand and apply correctly the concept of *i** dependency? The stakeholders didn't experience significant difficulties in understanding the notion of dependency. Numbers show that 70% of the dependencies included in final consolidated CM were identified by non-technical stakeholders in the 11 models constructed by them. Just 3% of the dependencies identified by non-technical stakeholders on their models were discharged by consultants for being considered not suitable.
- Q2: What are the most common mistakes made by non-technical stakeholders when representing *i** dependencies? According to the results, the wording of the dependencies needs to be more carefully described in the training materials to lower the high percentage of errors in this respect (56%).
- Q3: What are the types of *i** dependencies that are more difficult to understand and apply correctly for non-technical stakeholders? Goal dependencies resulted in the most confusing type of dependency. Anyhow, the arguably low percentage of confusion on type classification (22,38%) allows us stating that the types are well understood in general, considering the fact that the stakeholders were novices to the notation.
- Q4: Which are the groups of participants experiencing more difficulties on adopting and understanding the concept of dependency in *i**? Not surprisingly, operational stakeholders performed better than managerial stakeholders, although we think that if the manager would have invested more time in the case study, this may have help to bridge this gap. We think that the problem has more to do with dedication than with training.

As a final question, we could ask: can *i** CMs constructed by non-technical stakeholders be used in practice? This is the most important observation of the study. In our opinion, the study shows that the answer is "yes": the consultants were able to understand most of the models and the problems reported in this paper didn't prevent their correct understanding. These models may serve then as communication means among non-technical stakeholders and IT departments. To add to this conclusion, the proposed process was not too invasive for the organization's dynamics. It took a senior and a junior trainee consultant, a total of 20 hours to complete the full analysis process: 1) transcribing models to tabular representation, and 2) correcting, type, direction and description of incorrectly stated dependencies. Additional 11 hours were spent on validating semantics of corrected dependencies with non-technical stakeholders. In a typical IT strategic planning process which makes use of the DHARMA method to define system architecture, this time is not really significant, i.e. around half-week in the context of a 6-month project which plans for 4 years of information system implementation in the organization. Therefore, we may argue that time is not a problem for the adoption of *i** in highly strategic processes with complex operationalization. In the context of the DHARMA method [4] and the patterns formulated from previous experiences [20], this observation adds to the identification of actors as instances from an existing catalogue (see Section 3), leading to a usable method in practice.

This study paves the road to further research. First of all, since we have collected a great amount of data in this study, we do not discard to look for more questions to answer, remarkably in relation to productivity (e.g., time required to build a model). Also, similar studies would help to increase the knowledge and support more conclusive observations. On the other hand, similar strategies could be applied to *i** Strategic Rationale (SR) models. In fact, we have some experiences also in this respect due to the use of SR models in later activities of our DHARMA method. Last but not least, we could think of similar studies focusing on other languages, where the aspect of adoption in practice is also subject of current research [26][27], although not adopting the co-creation perspective of our study.

As mentioned in the paper, the experiment was conducted using Excel to represent individual models in a tabular way. Although this tool proved useful for basic actions (e.g., recording actors, dependencies, their types or directions), it is clearly limited for more advanced actions (e.g., merging individual models into final organizational model, detecting redundancies, etc.). Because of this, we plan to use in future case studies a tool specifically designed to support the method. It is expected that the tool will contribute in the reduction of some common errors (e.g. directions of dependencies, naming the model elements, etc.).

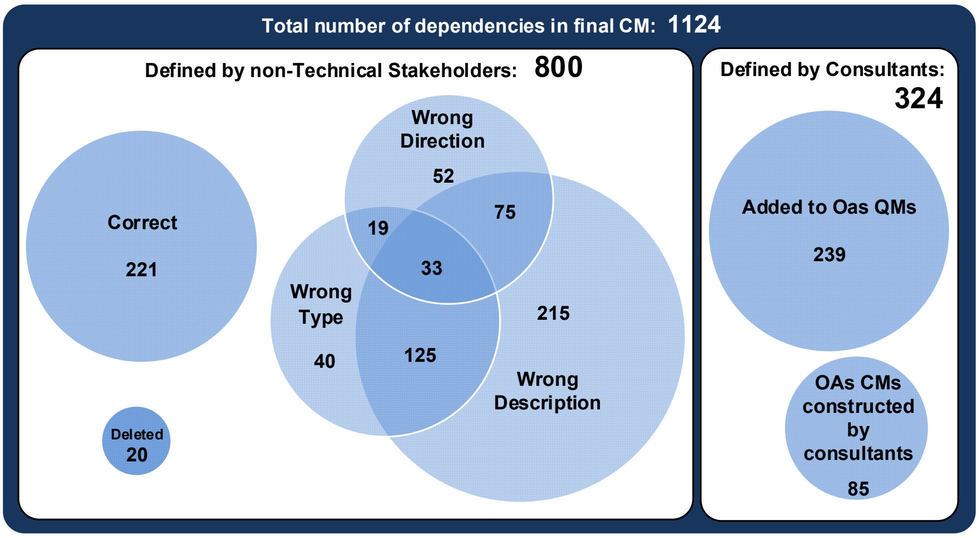

Figure 6. Dependencies in final CM.

References

[1] The Open Group. The Open Group Architecture Framework (TOGAF) version 9. The Open Group, 2009.

[2] Kilov, H. Using RM-ODP to bridge Communication Gaps between Stakeholders. WODPEC 2004.

[3] Chou, T.-H., Kanno, T., Furuta, K. Modeling and Bridging the Gap between Different Stakeholders. IESS 2012.

[4] Carvallo, J.P., Franch, X. On the Use of *i** for Architecting Hybrid Systems: A Method and an Evaluation Report. PoEM 2009.

[5] Moody, D.L., Heymans, P., Matulevicius, R. Improving the Effectiveness of Visual Representations in Requirements Engineering: An Evaluation of *i** Visual Syntax. RE 2009.

[6] Estrada, H. et al. An Empirical Evaluation of the *i** Framework in a Model-Based Software Generation Environment. CAiSE 2006.

[7] Carvallo, J.P., Franch, X. Lessons Learned on the use of *i** by Non-Technical Users. iStar 2014.

[8] Hadar, I., et al. Comparing the Comprehensibility of Requirements Models Expressed in Use Case and Tropos: Results from a Family of Experiments. IST 55(10), 2013.

[9] Engelsman, W., Wieringa, R. Goal-Oriented Requirements Engineering and Enterprise Architecture: Two Case Studies and Some Lessons Learned. REFSQ 2012.

[10] Engelsman, W., Wieringa, R. Understandability of Goal Concepts by Requirements Engineering Experts. MReBA 2014.

[11] Engelsman, W., Wieringa, R. Understandability of Goal-Oriented Requirements Engineering Concepts for Enterprise Architects. CAiSE 2014:

[12] The Open Group: ArchiMate 2.0 Specification. Van Haren Publishing (2012).

[13] Schultz, F., Meissner, J., Rosssak, W. Tracing the Interdependencies between Architecture and Organization in Goal-Oriented Extensible Models. ECBS-EERC 2013.

[14] Marosin, D., van Zee, M., Ghanavati, S. Formalizing and Modeling Enterprise Architecture (EA) Principles with Goal-Oriented Requirements Language (GRL). CAiSE 2016.

[15] Yu, E.S.K., Mylopoulos, J. Understanding "Why" in Software Process Modelling, Analysis and Design. ICSE 1994.

[16] Yu, E.S.K. *Modelling Strategic Relationships for Process Reengineering*. PhD Dissertation, University of Toronto, 1995.

[17] Dalpiaz, F., Franch, X., Horkoff, J.: iStar 2.0 Language Guide. CoRR abs/1605.07767, 2016.

[18] Carvallo, J.P., Franch, X. Building Strategic Enterprise Context Models with *i**: A Pattern-Based Approach. TEAR 2012.

[19] Basili, V.R. Applying the Goal/Question/Metric paradigm in the experience factory." Software Quality Assurance and Measurement: A Worldwide Perspective, 1993.

[20] Franch, X., et al. Systematic Construction of *i** Strategic Dependency Models for Socio-technical Systems. IJSEKE 17(1), 2007.

[21] Moody, D. The "Physics" of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. TSE 35(6), 2009.

[22] Runeson, P., Höst, M. Guidelines for conducting and reporting Case Study Research in Software Engineering. EmSE 14(2), 2009.

[23] Giachetti, G. et al. Verifying Goal-oriented Specifications used in Model-Driven Development Processes. Information Systems 64, 2017.
[24] Horkoff, J., Maiden, N.A.M., Lockerbie, J. Creativity and Goal Modeling for Software Requirements Engineering. Creativity & Cognition 2015
[25] PMI. A Guide to the Project Management Body of Knowledge (PMBOK Guides). Project Management Institute 2016.
[26] Torchiano, M. et al. Relevance, Benefits, and Problems of Software Modelling and Model Driven Techniques - A Survey in the Italian Industry. JSS 86(8), 2013.
[27] Ho-Quang, T. et al. Practices and Perceptions of UML Use in Open Source Projects. ICSE-SEIP 2017.
[28] https://docs.google.com/presentation/d/1esRTM6awLD0rPXatlGydWNL0YD-8h2a-cdpQe5QXcPo/edit#slide=id.g15b2e35842_0_1504