Exploring the Effectiveness of Normative *i** Modelling: Results from a Case Study on Food Chain Traceability

Alberto Siena[1], Neil Maiden[2], James Lockerbie[2], Kristine Karlsen[2], Anna Perini[1], Angelo Susi[1]

[1]Fondazione Bruno Kessler - Irst, Trento, Italy
{siena, perini, susi}@fbk.eu

[2]Centre for HCI Design, City University, London
{N.A.M.Maiden@, J.Lockerbie@soi., Kristine.Karlsen@soi.}city.ac.uk

Abstract. This paper evaluates the effectiveness of an extension to *i** modelling – normative *i** modelling – during the requirements analysis for new socio-technical systems for food traceability. The *i** focus on modelling systems as networks of heterogeneous, inter-dependent actors provides limited support for modelling system-wide properties and norms, such as laws and regulations, that also influence the specification of socio-technical systems. In this paper we introduce an extension to *i** to model and analyse norms, then apply it to model laws and regulations applicable to European food traceability systems. We report an analysis of the relative strengths and weaknesses of this extended form of *i** with its traditional forms, and use results to answer two research questions about the usefulness and usability of the *i** modelling extension.

1. Introduction

Analysts are increasingly using *i**, the strategic goal modelling approach [21], to model and analyse requirements. *i** has been applied successfully to model requirements for air traffic management tools [8, 9] and decision support aids in agriculture [11] as well as to support individuals and groups in the work of charitable organisations [17]. Reported benefits to our projects have included automatic requirements generation from *i** models [9] and detection of omissions from UML requirements specifications [8]. However, the focus on modelling systems as networks of heterogeneous but inter-dependent actors provides limited capabilities for addressing the broader, system-wide properties and norms that also influence the specification and design of socio-technical systems. Examples of such norms include laws and regulations, which constrain and influence how actors in these systems shall operate. In this paper we report an extension to the *i** modelling approach to model norms in socio-technical systems, then investigate the effectiveness of this extension through its application to a large-scale case study – introducing new traceability technologies into two European food chains.

Whilst *i** has many strengths that have contributed to its increasing adoption, the representation of laws and regulations, as well as actors' adherence to these laws and regulations, is recognized as problematic because laws and regulations are difficult to represent using the standard actor-goals-dependencies metaphor found in the basic modelling approach – referred to as "basic" *i** in this paper. For example, whilst actors in socio-technical systems might seek to achieve compliance with reported regulations such as for food hygiene, regulation compliance on its own is often not a strategic goal or softgoal of actor. Furthermore, inclusion of new actors who generate and impose laws and regulations detract from the main analytic purpose of *i**. As a consequence, representing laws and regulations is often overlooked in early requirements work, with consequences for downstream analysis and design of socio-technical systems.

Previous work has introduced new *i** modelling concepts to represent and analyse norms to represent laws and regulations [15]. However, like all extensions to basic *i**, such as SecureTropos [5], the addition of new modelling semantics and syntax can increase the complexity and reduce the usability and adoption of the *i** modelling approach. Therefore, studies were needed to explore the coverage, effectiveness and usability of *i** modelling extensions prior to their widespread adoption.

In this paper, we report the extension of *i** modelling with norms to investigate whether such extensions deliver advantages such as the induction of new actor goals from the adoption of a norm, explanation of existing goals due to the imposition of norms, and the discovery of new roles/actors due to the imposition of a norm. We investigated TRACEBACK, a EU-funded Integrated Project seeking to introduce new technologies to improve traceability in European food chains. We used results to seek answers to two research questions:

Q1: Were analysts using the extended *i** semantics and notation with the norm concept able to infer new properties of the system being modelled?

Q2: Were analysts using the extended *i** modelling approach able to represent concepts related to norms, such as legislation, rules, etc. in an efficient manner (compared to basic *i**)?

The paper is structured as follows: section 2 reports the basic *i** modelling framework; section 3 introduces the normative *i** framework; section 4 reports the models obtained for the food traceability information system with both basic *i** and normative *i**; section 5 evaluates empirically the new framework by comparing the results of its application and the basic *i** models; section 6 discusses the results and tries to answer the raised questions; finally, section 7 concludes the paper.

2. *i** and REDEPEND

In TRACEBACK we applied the RESCUE requirements process with the *i** modelling approach and REDEPEND tool. RESCUE [6] supports a concurrent engineering process in which different modelling and analysis processes take place in parallel. Each stream has a unique and specific purpose in the specification of a socio-technical system:

1. Human activity modelling provides an understanding of how people work, in order to baseline possible changes to it [14]. In TRACEBACK we observed and documented the work activities and behaviour of actors in, for example, food

plants producing milk-based products;

2. System modelling enables the team to model the future system boundaries, actor dependencies and most important goals of actors in the dairy food chain using the *i** approach [21] and REDEPEND tool [7];
3. Use case modelling and scenario-driven walkthroughs enable the team to acquire complete, precise and testable requirements from stakeholders [18]. For example, we specified the behaviour of how food chain actors would work with new micro-devices and service-based information systems in improved traceability practices, then walked through scenarios to discover more complete requirements on these devices and information systems;
4. Managing requirements enables the team to handle the outcomes of the other 3 streams effectively as well as impose quality checks on all aspects of the requirements document [13].

In this paper we focus on the second stream using the *i** approach to food chains in terms of actor dependencies and goals.

*i** is an approach originally developed to model information systems composed of heterogeneous actors with different, often-competing goals that depend on each other to undertake their tasks and achieve these goals. *i** can be applied effectively to model food and food-related information chains, as we will demonstrate. Due to the physical characteristics of food production, actors and resources generated and consumed later in the food chain depend on actors and resources earlier in the food chain, which can be represented using dependency relationships central to the *i** approach.

Basic *i** modelling supports 2 basic types of model. The first *i** model produced was the Strategic Dependency (SD) model, which describes a network of dependency relationships among actors. The opportunities available to these actors can be explored by matching the depender who is the actor who "wants" and the dependee who has the "ability". Since the dependee's abilities can match the depender's requests, the system-wide strategic model is developed. For example, in TRACEBACK, the actor *Primary Dairy Producer* depends on a second actor *Farm* to attain the goal *hygiene standards met*, achieve the softgoal *quality milk received*, and obtain the resource *fresh milk*. More details of these models are reported in Section 4.

The second type of *i** model is the Strategic Rationale (SR) model, which provides an intentional description of how each actor achieves its goals and softgoals. An element is included in the SR model only if it is considered important enough to affect the achievement of some goal. The SR model includes the SD model, so it describes which actors may be able to accomplish something by themselves, or by depending on other actors. It specifies goals, tasks, resources and softgoals linked by dependency links from the SD model, task decomposition links, means-end links, and the contributes-to-softgoal links [21]. For example in TRACEBACK, the actor *Primary Dairy Producer* performs the task *undertake contamination recall procedures*, which achieves the softgoal *target recall undertaken successfully*, which in turn contributes positively to the actor *regulator* achieving the softgoal *contaminated products recalled efficiently*. Again more details of these models are reported in Section 4.

RESCUE is supported by REDEPEND [7], a tool based on Microsoft Visio and designed to provide systems engineers with *i** modelling and analysis functions. It provides drag-and-drop capabilities to visually develop *i** Strategic Dependency (SD) and Rationale (SR) models. REDEPEND also provides systems engineers with simple model verification functions for large-scale SD and SR models. In RESCUE we applied basic *i** to model essential actors, dependencies, goals and tasks in the dairy food chain. The next section reports methodological extensions to *i** to model normative contexts in socio-technical systems also applied to the dairy food chain.

3. Normative *i**

What distinguishes socio-technical organisations from simple groups of interacting people are norms [10]. Various types of norms exist in the real world, but, as pointed out in [16], the one that gains relevance at requirements time is the behavioural norm – essentially, behavioural norms impose actions to perform, goals to be achieved, resources to be used or principles to be respected.

Recent studies in requirements engineering address the problem of modelling regulations for requirements compliance. A survey on current approaches is given in [12]. Worthy of mention here is a proposal that relies on the analogy between regulations and requirements documents to model the objectives stated in the regulations [3]. However, the adopted goal-oriented framework – Kaos [2] – misses the capability of supporting agency in the models. In [1] the focus is on automatic extraction of obligations and rights from legal texts, so usefully supporting the analysts in parsing the law documents, but not in representing them. In [4], traceability links are used to map *i** models of the regulations into the *i** models for the stakeholders. We took these approaches into consideration before introducing a new modelling framework. However, our need attains principally to supporting the analyst in the *discovery* and *integration* of legal requirements, so none of the approaches were satisfactory for us in a domain like the food chain.

As with [3], we propose to use a goal-oriented approach, based on *i**, for modelling norms, but in contrast to the above mentioned work we focus on the interaction of norms, actors and goals during the requirements elicitation process. More specifically, as introduced in a previous work [15], our idea is to model contextually and homogeneously, but separately, the normative context of a domain and its stakeholders with their intentionality. We adopt the definition of "norm" as a means for communicating standards of behaviour [19], and which acts as an abstraction for any kind of deontic prescription (such as laws, regulations and so on). On the basis of this definition, in the present work we derive three properties of norms that are relevant for the requirements acquisition: i) the normative commitment relation; ii) the schema of the norm; iii) the compliance intentions.

The normative commitment relation. Intuitively, when we think of laws or regulations, we think of artefacts, i.e., text documents, that contain prescriptions. It is interesting to notice that, when a law commits something (the prescription) to someone, the commitment establishes a relation. The relation involves two subjects: the one who created the norm – the source of the norm; and the one who is addressed by the norm – the addressee [19]. So, as depicted in Figure 1, in *i** diagrams we represent laws in a ternary relation that links the source, the addressee and the legal artefact that contains the prescription. The double arrow represents the commitment direction, whilst the triangle represents the norm. The link between *EU*, *EC178/2002* and *Food industry operator* can be read as follows: the European Union has laid down the EC178/2002 law, which addresses all the operators that work in the food industry.

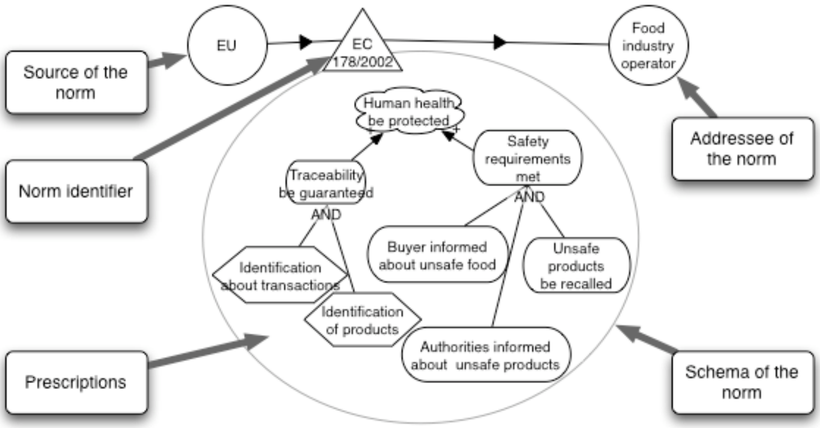

Figure 1. Normative *i: the normative commitment relation between two stakeholders and the schema of the norm.**

The schema of the norm. A norm artefact (e.g. a law's text) typically imposes some prescriptions. With the term *Schema of the norm* we refer to the behavioural pattern that the norm imposes to the addressee, that is, ways of acting, goals and principles to be adopted. In Figure 1 the schema of the norm is depicted as a balloon collecting a set of *i** intentions – goals, softgoals and tasks – and, possible relations among them – like decomposition or means-end. The depicted norm's schema can be read as follows: *Food industry operators* must ensure that the minimum requirements for food safety are met (hardgoal *Safety requirements met*), i.e., they must ensure that, in case of known, unsafe food, the products are recalled, and both the buyers and the authorities are informed (hardgoals *Buyer informed about unsafe food*, *Authorities informed about unsafe products*, and *Unsafe products be recalled*). They must label their products with an identification code (task *Identification of products*), and register any transaction (task *Identification about transactions*); but they must ensure the traceability of the food products they process, whatever other actions they do (hardgoal *Traceability be guaranteed*). While performing these tasks or fulfilling these goals, the leading principle that should inspire their conduct should always be the protection of the health of the consumers (softgoal *Human health be protected*); i.e., the accomplishment of the hardgoals has to be evaluated with regard to the root softgoal, and no other interpretations should be accepted.

The compliance intentions. We want to understand the actual impact of the law on the involved stakeholders, namely what do they put into action – if they do – to accomplish to the law imposition. In Figure 2, the interleaving between the actor's intentions and the law's schema shows how the *Food industry operator* intends to comply with the law. In the example, the law lays down for the actor the responsibility of recalling products if they are known to be unsafe. However, the lack of safety of products is not known *a priori* by the operator. So it will need to keep its products monitored, and so is how the goal *Products safety be monitored* is generated inside the actor's rationale. Such a goal is then further decomposed into two specific tasks (*Monitor unsafety of milk* and *monitor unsafety of dairy products*). In the intention of the food chain operator, the two tasks should ultimately contribute to the compliancy with the norm. So now we know that the monitoring activities have been undertaken by the operator for the specific need of complying with a prescription of the EC178/2002 law.

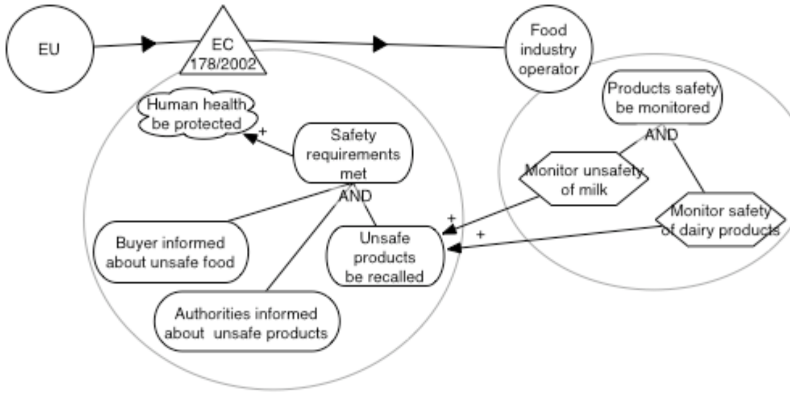

Figure 2. Schema of the norm EC178/2002 with the intentional entities inside the balloon representing the responsibilities established by the law.

When performing requirements elicitation, we interleave *i** modelling of domain stakeholders and normative modelling, as described below with the help of Figure 3. Figure 3(a) depicts a typical scenario that occurs while exploring a regulated domain. Let us suppose that we observe only Actor1, while Actor2, Actor3 and Actor4 are hidden. Here *hidden* means that the interviewed stakeholder(s) did not explicitly mention any norms, or if they did, they did this without highlighting their role. This is a typical problem of *tacit knowledge*. Returning to the example, we know that Actor1 is called to comply with two laws, Norm1 and Norm3, and so we proceed with the analysis of such laws (Figure 3(b), step 1). If the laws address other actors, they are added to the domain model (step 2). At this point Actor3 is still *hidden*. However, by analysing the source of Norm3 (step 3), we are able to find Norm2 (step 4), which in turn leads us to Actor3 (step 5). We store all this information in a norm diagram such as the one in Figure 3(b) for further analysis of the model. So we have discovered Actor2 and Actor3; but are those actors actually part of the domain? For sure we only want to model those actors that are relevant for the requirements specification. For this purpose, the analysis of the norm's schema allows us to discard those actors that are irrelevant for the problem under study. For instance, having discovered Norm1, we could observe that it lays down prescriptions attaining different topics, not in our interest. So, Actor4 will not enter in the description of the domain.

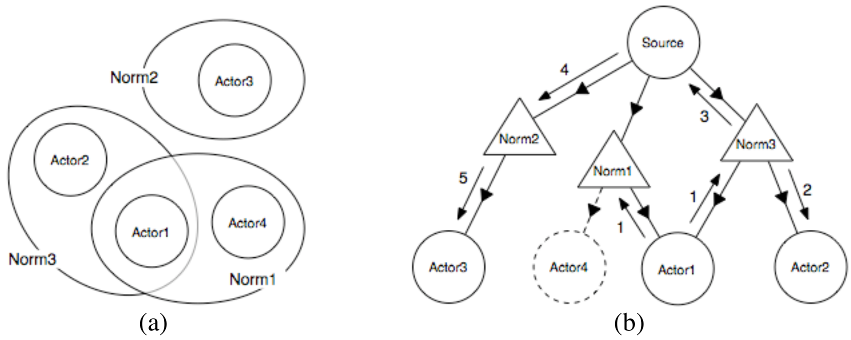

Figure 3. (a) the scope of three norms in a domain with four actors; (b) the process of norms/actors discovery and representation in the same domain.

4. The TRACEBACK Food Traceability Case Study

As Section 2 reports we applied the RESCUE process, *i** modelling approach and REDEPEND tool to the EU-funded TRACEBACK Integrated Project. Assuring the total traceability of food and feed along the whole chain from production to consumption is a cornerstone of EU policy on the quality and safety of food. This is a complex procedure involving identification, detection and processing of a vast amount of information. Profit margins of food producers and processors are already very tight, so they require a tracking mechanism that is not only reliable and easy to use, but does not entail a major cost burden. With a concerted effort and input from expert institutions, modern technology could provide such a system. TRACEBACK is developing innovative solutions based on micro-devices and innovative service-based architectures to provide innovative new information services to actors from primary food producers to consumers and health authorities. Solutions, which will include new micro-devices and a service-oriented reference architecture for traceability information systems (RATIS), are to be trialled on two major product chains – feed/dairy and tomatoes. In this paper we focus on models developed for one of the selected food chains – dairy products such as milk-based products.

During application of the RESCUE process a team of 3 analysts, produced *i** SD and SR models describing actors in the dairy food chain. The models were developed using information from descriptions of current processes and workflows in the dairy food chains in Europe, one-on-one interviews with stakeholders who fulfil modelled actor roles in these food chains, *i** modelling workshops at project partner sites, and electronic distribution of SD and SR models to stakeholders for comment and feedback. Overall the process lasted 6 months. Key results are reported in 4 basic *i** models – 1 SD and 1 SR model each for the 2 TRACEBACK-enhanced food chains in the European dairy and tomato food chains.

In addition to the RESCUE work, normative *i** models were developed by another analyst following the process sketched in Section 3. The analyst independently explored the domain with the purpose of both discovering the applicable norms and finding related stakeholders. Using documentation and information gathered from a one-on-one stakeholder interview, 7 models were developed with a drawing tool that can export to Visio/REDEPEND.

The SD and SR models for the dairy food chain and an excerpt from the normative *i** models are reported in the next section.

4.1 The Basic *i** SD and SR Models

The basic *i** SD model of actors in the dairy food chain is depicted in Figure 4, and the inset shows part of the model in a readable form. The model expresses 79 strategic dependencies between 13 actors from *feed suppliers* to *transportation* and even the *media* in a dairy food chain. The inset shows dependencies between the *Feed supplier* and *Farm* actors. For example, the *Farms* depend on the *Feed supplier* to achieve the softgoal *feed contamination detected early*.

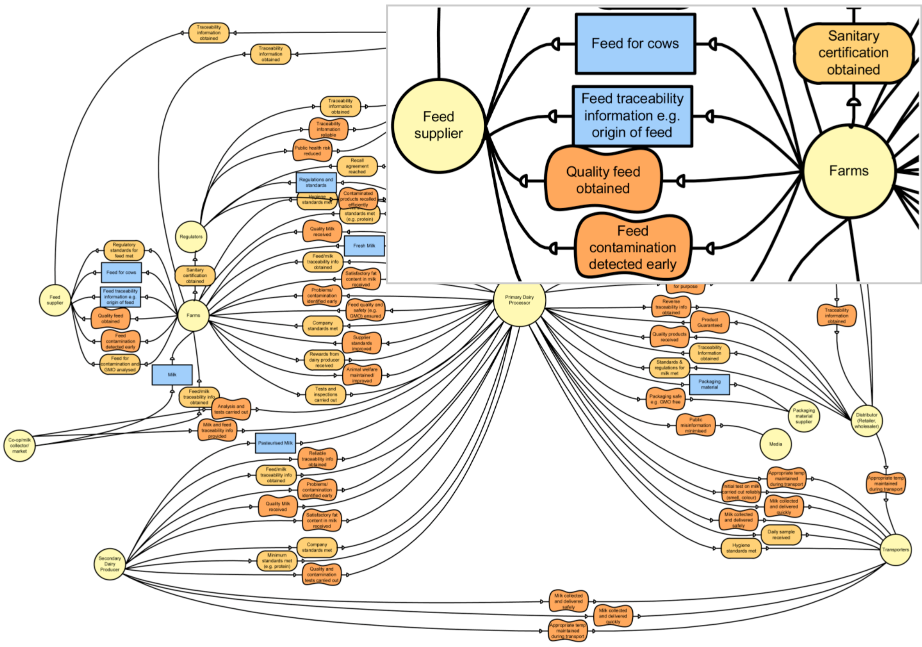

Figure 4. The basic *i SD model of actors in the dairy food chain, with an inset showing dependencies between the *feed supplier* and *farm* actors.**

The basic *i** SR Model for the same dairy food chain actors is depicted in Figure 5. The model specifies 251 different process elements and 257 different associations between these elements. The inset demonstrates part of the SR model, the *feed supplier* actor, in a readable form. The *feed supplier* undertakes the task *supply feed to farms*. To do this the *feed supplier* provides *feed traceability data* and uses the resource *feed for cows*, and seeks to achieve the softgoal *quality product stocked*.

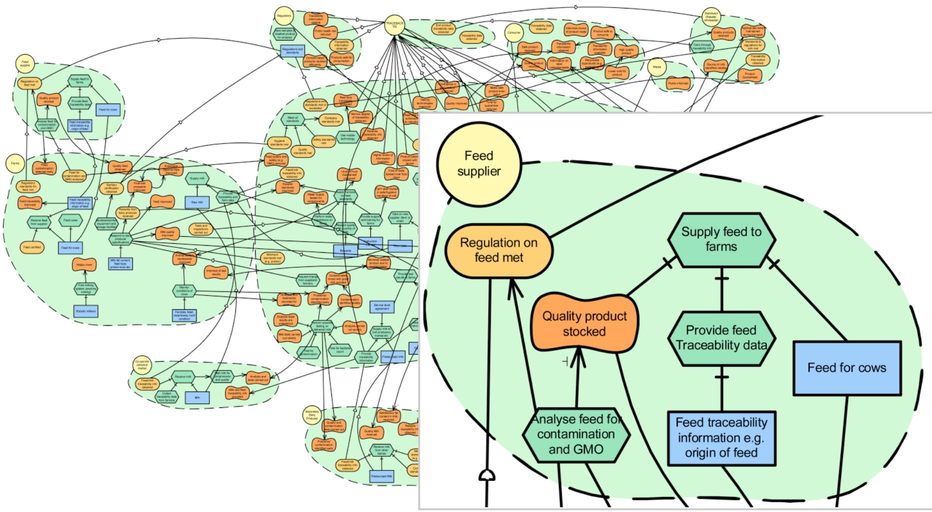

Figure 5. The basic *i SR model of actors in the dairy food chain, with an inset showing the expanded *food supplier* actor.**

4.2 The Normative *i** SD and SR Models

An excerpt of the normative *i** models is depicted in Figure 6. The actor *Food Safety Authority* has been instituted by the EC178/2002 for monitoring the entire food market, whilst the *Rapid Alert System*, which is comprised by the national governments and the EU bodies, is in charge of receiving and dispatching alerts on food-related events. In Figure 6 are also depicted the results of the norm's schema analysis, based on the same EU178/2002. In the following we discuss the four elements that are pointed out by the dashed arrows labelled 1, 2, 3 and 4:

1. The *Rapid Alert System* is devoted to the collection and forwarding of recalls across Europe, so the *Food industry operators* depend on it for dispatching alerts. At the same time, the *Rapid Alert System* depends on the food operators for having detailed traceability information to dispatch.
2. Some goals that had emerged as *Food industry operator* goals did actually come from EU laws. Recalling unsafe products or warning customer is not a free choice of producers, but are needed to comply with the law.
3. To minimise the impact that the recalling policy has on the budget, food industry operators try to discover potential unsafeties as early as possible (softgoal *unsafe products early detected*), so they monitor the quality of the raw materials and, when possible, the production processes of their suppliers. For example, in the picture we show how operators that work in the dairy production prescribe to the farmers a sort of non-legislative regulation, the Good Manufacturing Practices (GMP), to ensure the achievement of internal goals.
4. *The farmers*, in turn, put into action tasks and generate goals to be able to comply with the GMP.

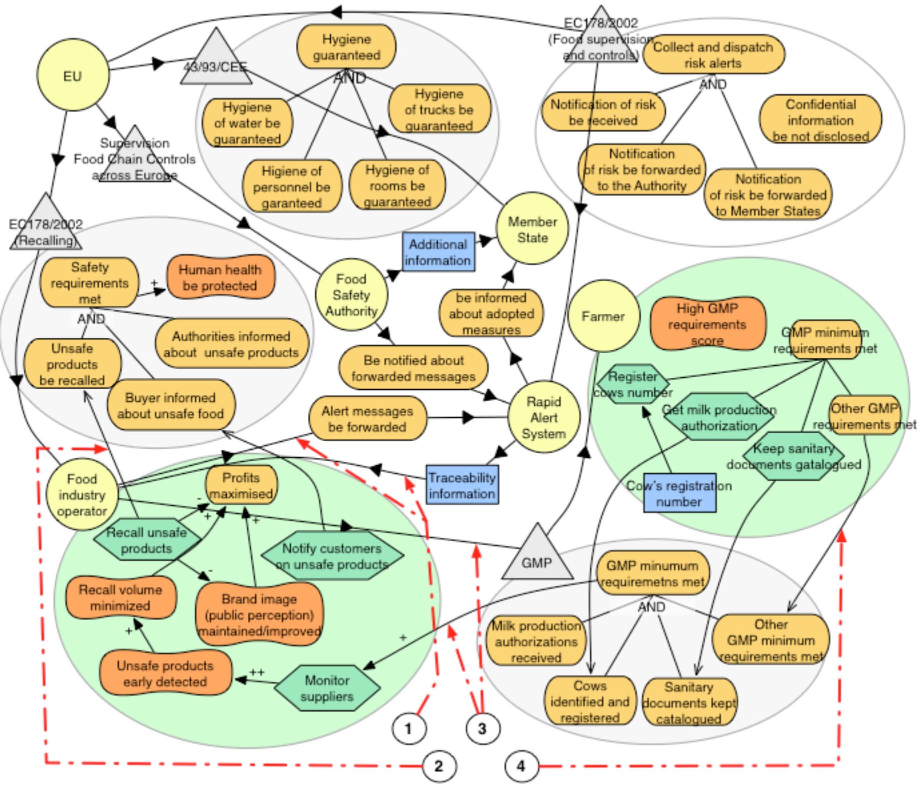

Figure 6. A snapshot on the domain, with regard to the regulation EC178/2002.

5. Empirical Analysis of the *i** Basic and Normative TRACEBACK Models

Whilst undertaking our RESCUE goal modelling process using basic *i**, we were aware of the existence and importance of laws and standards of behaviour but did not model these explicitly. Instead, we modelled these implicitly through the goals of the actors – for example, *Farms* seek to attain the goal *sanitary certification obtained*. However, a comparative analysis of the basic *i** and normative *i** models for the feed/dairy chain (Figure 5 and Figure 6) revealed that the identification and subsequent modelling of norms added some useful detail that was either overlooked or not clearly expressed in the basic *i** model. This analysis is described further below and summarised in Table 1.

Taking the GMP norm for *Farms* as an example, we can see that the norms approach leads to 2 new goals being introduced – *GMP minimum standards met* and *other GMP requirements met*. As mentioned before, we touched upon some of the GMP areas using the basic *i** approach such as *sanitary certification obtained* (goal), whereas undertaking the task *keep sanitary documents catalogued* is introduced in order to meet the high level requirement (goal) of the GMP. From this simple example we can see that the normative approach can add more precision.

Table 1. Summary of the comparative analysis undertaken between the feed/dairy basic *i SR model and its normative *i** equivalent**

Normative / *i** Actor	Matches to basic *i** model	Additions to basic *i** model	Amendments to basic *i** model
Food industry operator / Primary dairy producer	3 softgoals 2 tasks 1 goal	1 task	
Farmer/Farms		3 tasks 1 softgoal 1 resource	5 existing goals to be reconciled with 2 new goals
EC178/2002 / Primary dairy producer	1 goal	3 sub-goals	Must review the goal boundary between the primary dairy producer and the regulator
EC178/2002 / Regulator	1 softgoal		As above
43/93/CEE / Primary dairy producer	1 goal	4 sub-goals	
GMP / Primary dairy producer		3 goals	7 goals to be reconciled with 2 new goals and 2 norms
EC178/2002 (Food supervision and controls) / none specific		8 goals 3 actors	Primary diary producer and regulator actors to be reconciled with new goals and actors

Whilst we believed we had already modelled the strategic elements of the *Farms* actor, the introduction of the GMP norm resulted in: 1 new softgoal (10 already in basic), 2 new goals (6 already in basic), 3 tasks (8 already in basic), and 1 resource (6 already in basic). The resource, softgoal and 3 tasks constitute important additions to the model, whereas the 2 new goals encompass 5 of the original 6 goals related to standards, certification, analysis and inspections. Through further analysis of the GMP these goals could be aligned to the norm or modified accordingly.

Looking at the *Primary Dairy Processor*/*Food Industry Operator* actor boundary, we can see that the GMP norm contributed one additional task – *monitor suppliers* – whilst it is apparent that the other elements featured in the norms model were derived from the original *i** model. Under the basic approach, the goals contained within the GMP boundary should, in theory, feature within the actor boundary of the norm creator – in this instance the *Primary Dairy Processor*. A review of this actor boundary reveals 7 goals related to standards, regulations and requirements. However, none of these goals explicitly refers to the GMP goals of *milk production authorisations received*, *cows identified and registered* or *sanitary documents kept catalogued*, therefore we could argue for their inclusion within the basic *i** model.

Returning to the 7 goals relating to standards, regulations and requirements mentioned above, it is interesting to note that the high level goals *hygiene standards met* and *safety standards met* are elaborated upon in the norms model. The norm 43/93/CEE provides us with the additional detail of 4 hygiene-related sub-goals, whilst the EC178/2002 norm details 3 additional safety-related sub-goals. EC178/2002 also provides us with the softgoal *human health be protected* that is touched upon within the Regulator actor boundary in the basic *i** model by the softgoal *public health risk reduced*. This brings us back a limitation of basic *i** mentioned earlier, the issue of whose actor domain the goal belongs to – is it the goal of the regulator, the diary processor or both? Normative *i** provides us with the opportunity to treat the normative layer of the domain as a separate concern in domain modelling, hence removing this issue and supporting more effective analysis.

Another area completely overlooked by the basic *i** model was that of collecting and dispatching risk alerts, as addressed by the norm EC178/2002 (food supervision and controls). The normative model draws our attention to 3 new actors – *Food Safety Authority*, *Rapid Alert System* and *Member State* – which provide us with 8 additional goals. It is possible that overlooking these actors in the basic *i** approach may have had consequences further down the line for the analysis and design of the TRACEBACK socio-technical systems.

As mentioned earlier, we originally applied our standard RESCUE goal modelling process to TRACEBACK and did not explicitly model laws and regulations using basic *i**. Therefore, there is clearly an overhead associated with using the normative *i** approach that needs to be analysed with respect to the additional benefits it provides. We can divide our analysis into four main activities: interaction with stakeholders, inspection of documents, analysis of norm scope, and building the models. Such activities were mostly interleaved, but approximately we can estimate a 1-day interview with stakeholders; 3 days for deepening the knowledge on the norms; 7 days for exploring the norms scope and to identify the relevant ones; and finally 5 days to synthesize them and build the actual models. So, in total we can estimate that 16 person-days were spent applying the normative *i** approach to TRACEBACK.

6. Discussion

We used results reported in Section 5 to answer the 2 research questions about the *i** normative modelling extension. The answer to Q1, were analysts using the extended *i** semantics and notation, able to infer new properties of the system related to norms and legislation, is a tentative yes. In purely quantitative terms, 3 new actors and 24 new process elements, including 18 new goals and 1 new softgoal, were expressed and analysed in models developed for 3 separate pieces of legislature that impacted on two existing actors – the *primary dairy producer* and *farm* actors.

The comparative analysis we undertook showed that applying the normative approach generally added more detail to the standards-related goals already present in the basic *i** model – such added detail included cow registration and sanitary documentation cataloguing. In essence, we were able to disambiguate a number of high-level goals and derive more precise properties of the system being modelled. Furthermore, explicitly modelling the laws and standards adds richness to the models that can provide benefits later on in the software development process. As TRACEBACK is developing a service reference architecture that will provide multiple instantiations of traceability information systems, knowledge of each individual domain including GMP and EU laws is important. The normative *i** can be used as a reference model from which analysts explore the finer details to discover important system properties and final specifications.

Another point to note in support of the normative *i** approach is its usefulness and effectiveness where stakeholder access is limited. For example, we did not have the means to access the farms directly, so we obtained documentation from the dairy producers about the GMP and used normative *i** models to infer, from scratch, the missing knowledge. In this case the norms approach was a useful and effective way to better understand the domain and capture more detailed requirements.

Results from applying normative *i** to TRACEBACK also provided qualitative evidence to support our initial assertions. The basic *i** goal/actor metaphor cannot support a sufficiently complete representation and exploration of normative contexts in complex domains such as food traceability. Several problems identified and addressed subsequently in the project were a further exploration of important goal boundaries between the *primary dairy producer* and *regulator* actors or between the *primary dairy producer* and the *farmer*s. Evidence from TRACEBACK indicated that stakeholders often did not venture knowledge and model feedback beyond the boundaries of the actors representing them on the *i** models, and the modelling of norms helped us to overcome this limitation of the goal/actor metaphor.

The modelling process applied in TRACEBACK also provides an interesting insight with which to interpret our answer to Q1. Draft basic *i** models were already available when the normative modelling began. Clearly the basic *i** models did not explicitly model the norms. Instead, with hindsight, stakeholders' perceptions of norms can be inferred from the basic models. So for instance, the goal *Feed regulation met* in the SR model of the actor *Feed Supplier* depicted in Figure 6 represents the actor perception of the law EU178/2002.

In contrast to Q1, we were unable to answer Q2 conclusively and determine whether analysts using the extended *i** modelling approach were able to represent concepts related to norms, such as legislation, rules, etc. in an efficient manner (compared to basic *i**). We estimated the time in TRACEBACK to produce and analyse the normative models against the advantages reported previously. A crude quantitative analysis of the number of modelled elements per day revealed a productivity measure of 1.7 elements/day (27 new model elements divided by 16 person-days). Although this modelling rate is low we also need to take into account the qualitative benefits of the normative *i** approach. Also, further analysis of the data in Table 1 suggests little overlap between the modelled elements in the two models, with 9 matches to the basic *i** version compared with 27 additions. This result implies that normative *i** complements its basic equivalent giving us benefits that appear cost-effective.

Overall, our subjective opinion is that our application of normative *i** to TRACEBACK was cost-effective, but further research and a detailed cost benefit analysis would need to be undertaken to provide a more objective and definitive answer to this question.

Interestingly, the laws we considered were generally quite clear and readable. It was apparent that the well-organised structure and unambiguous nature of the legislature supported the cost-effectiveness of the normative *i** approach. In contrast, scope analysis resulted in being the most time-expensive activity, due to the large number of laws, several of them cross-referring each other and mostly out of scope. Building models of the legal documents is also quite time-consuming, but less than scope analysis, since norms are expressed in natural language, and to reduce ambiguity they tend to be extremely analytic. In order to get useful information from them to represent their intentional characteristics, we need to synthesize them.

7. Conclusions

In this paper we evaluated the effectiveness and efficiency of the normative *i** modelling, an extension to *i**, which aims at supporting requirements elicitation in domains articulated by norms. The analysis was performed on a case study based on a real project, TRACEBACK, devoted to the improvement of the traceability in European food chains. We used the normative *i** notation for modelling laws and regulations of the European food supply chain, and the resulting models have been compared with corresponding models, built previously with the basic *i** approach (basic *i**). Along with the comparison we addressed specific questions aimed at finding evidence of the effectiveness and the efficiency of normative *i**. Concerning effectiveness, from this experience it turned out that using normative *i** we were able to infer about the existence of several new goals and actors strictly related to the normative context, which were otherwise probably ignored. As for the efficiency of using normative *i**, we tried to characterize it in terms of extra time costs for this further analysis of the domain, resulting in about 5% of the overall time spent in modelling-related activities. An extra cost to be contrasted with the gain in modelling effectiveness. As a concluding remark, we consider our experience significant towards proving the effectiveness and efficiency of normative *i** modelling. Large-scale applicability could be evaluated through an empirical study, asking two groups of analysts to perform basic *i** and normative *i** modelling in parallel [20], but to be feasible, this type of analysis will require a lab-size case-study.

From this experience we derived some interesting work directions for the future.

- The normative *i** framework needs to be supported by a formal semantics. A conceptual meta-model will complete the framework and make it comparable to other approaches. Work is currently ongoing in this direction.
- The normative and basic *i** could be integrated into one single interleaved methodology, also in order to minimize the possible model reconciliation effort.
- As pointed out in [12], a major problem in se is the traceability of normative prescriptions. Being able to separate normative from strategic requirements is the first step towards supporting traceability along the different phases of the software development.

Acknowledgements. This work was funded by the EU-funded FP6 TRACEBACK project FP6-2005-FOOD-036300.

References

1. Breaux, T. D., Vail, M. W. and Anton, A. I. Towards Regulatory Compliance: Extracting Rights and Obligations to Align Requirements with Regulations. Proceedings of the 14th IEEE International Requirements Engineering Conference (RE'06), IEEE Society Press, 2006, 49-58.
2. Dardenne, A.; van Lamsweerde, A. & Fickas, S. Goal-Directed Requirements Acquisition. Science of Computer Programming, 1993, 20, 3-50
3. Darimont, R. & Lemoine, M. Goal-oriented analysis of regulations. International Workshop on Regulations Modelling and their Verification & Validation, 2006.

4. Ghanavati, S.; Amyot, D. & Peyton, L. A Requirements Management Framework for Privacy Compliance. In the 10th Workshop on Requirements Engineering (WER'07), 2007, 149-159.
5. Giorgini, P., Massacci, F., Mylopoulos, J. and Zannone, N. Requirements Engineering meets Trust Management: Model, Methodology, and Reasoning. Proc. of the 2nd International Conference on Trust Management (iTrust'04), 2004.
6. Jones S.V. & Maiden N.A.M., RESCUE: An Integrated Method for Specifying Requirements for Complex Socio-Technical Systems, in Requirements Engineering for Socio-Technical Systems, ed. J.L. Mate & A. Silva, Ideas Group, 2005, 245-265.
7. Lockerbie, J.A. & Maiden, N.A.M., 2006, REDEPEND: Extending *i** Modelling into Requirements Processes, Proceedings 14th IEEE International Conference on Requirements Engineering, IEEE Computer Society Press, 361-362.
8. Maiden N.A.M., Jones S.V., Manning S., Greenwood J. & Renou L., Model-Driven Requirements Engineering: Synchronising Models in an Air Traffic Management Case Study, Proceedings CaiSE'2004, Springer-Verlag LNCS 3084, 2004, 368-383.
9. Maiden N.A.M., Manning S., Jones S. & Greenwood J., Generating Requirements from Systems Models using Patterns: A Case Study, Requirements Engineering Journal 10(4) , 2005, 276-288.
10. North, D. C. Institutions, Institutional Change, and Economic Performance, Cambridge University Press, 1990
11. A. Perini and A. Susi. Designing a Decision Support System or Integrated Production in Agriculture. An Agent-Oriented approach. Environmental Modelling and Software Journal, 19(9), September 2004.
12. Otto, P. N. and Antón, A. I. Addressing Legal Requirements in Requirements Engineering. 15th IEEE Inter. Requirements Engineering Conference, 2007, 5-13.
13. Robertson S. & Robertson J., Mastering the Requirements Process, Addison-Wesley, 1999.
14. Vicente, K., Cognitive work analysis, Lawrence Erlbaum Associates, 1999.
15. Siena, A. Engineering Normative Requirements. 1st International Conference on Research Challenges in Information Science (RCIS'07), 2007.
16. Stamper, R.; Liu, K.; Hafkamp, M. & Ades, Y. Understanding the Role of Signs and Norms in Organisations - a semiotic approach to information systems design. Journal of Behaviour and Information Technology, 2000.
17. Sutcliffe A.G., Analysing the Effectiveness of Socio-technical Systems with *i**, in Requirements Projects: Some Experiences and Lessons, in Social Modeling for Requirements Engineering, MIT Press, ed. Giorgini, Maiden, Mylopoulos, Yu, 2007.
18. Sutcliffe A.G., Maiden N.A.M., Minocha S. & Manuel D., 1998, Supporting Scenario-Based Requirements Engineering, IEEE Transactions on Software Engineering, 24(12), 1072-1088.
19. Van Kralingen, R. A Conceptual Frame-based Ontology for the Law. First International Workshop on Legal Ontologies, 1997.
20. C. Wohlin, P. Runeson, M. Hoest, M. Ohlsson, B. Regnell, and A. Wesseln. Experimentation in Software Engineering - An Introduction. Kluwer Academic Publishers, 2000.
21. Yu E. & Mylopoulos J.M., Understanding "Why" in Software Process Modelling, Analysis and Design, Proceedings, 16th International Conference on Software Engineering, IEEE Computer Society Press, 1994, 159-168.