Establishing Information System Compliance: An Argumentation-based Framework

Giampaolo Armellin, Annamaria Chiasera
GPI S.p.A.
Trento, Italy
{garmellin,achiasera}@gpi.it

Ivan Jureta
University of Namur,
Namur, Belgium
ivan.jureta@fundp.ac.be

Alberto Siena
Free Univ. of Bolzano/Bozen
Bolzano, Italy
asiena@unibz.it

Angelo Susi
FBK-Irst
Trento, Italy
susi@fbk.eu

***Abstract*—This paper introduces a mixed modeling and argumentation framework applied to assess the compliance of requirements with legal norms, and reports the results of its application in an industrial project in healthcare. Domain experts applied a goal-oriented modeling framework for the representation of requirements and norms, then used argumentation techniques to assess the compliance of requirements with norms, and revise requirements model to ensure compliance.**

***Index Terms*—Requirements Engineering, Norms compliance, Goal-Oriented, Argumentation**

I. Introduction

With the increase in the use of complex socio-technical systems by organizations and firms to deliver services, came a growth in the interest of regulatory bodies in the compliance of these systems with local, national, and international norms.

The impact of this situation has been immense on Software Engineering as much as on business practices. It has been estimated that in the Healthcare domain, organizations have spent $17.6 billion over a number of years to align their systems and procedures with a single law, the Health Insurance Portability and Accountability Act (HIPAA), introduced in 1996 [1]. In the Business domain, it was estimated that organizations spent $5.8 billion in 2005 alone to ensure compliance of their reporting and risk management procedures with the Sarbanes-Oxley Act [2]. In Italy, the D.Lgs. n. 196/2003 national law lays down personal data protection rights, and the measures to be set up by any subject controlling sensible data.

In this setting, engineers are faced with new challenges in eliciting requirements that at the same time fulfill the needs of stakeholders and comply with relevant norms. The difficulty of this task stays in the nature of law: despite the assumptions of some past works, which basically tried to treat it as a formal system, law is a vague and ambiguous artifact, often incomplete in its prescriptions and sometimes containing contradictions. Assessing compliance for information systems requires therefore cross-disciplinary skills: computer science skills and legal knowledge to deal with the norms properties. Consequently, assessing compliance, even with a small law fragment, might require the acquisition and elaboration of a large body of information. As long as such information is kept by law experts or individual analysts, every further modification in the system (or in the law) causes the need for re-acquiring information again, thus generating the increase of costs.

Assuming that with appropriate engineering tools we can identify the measures to be implemented in order to comply with law prescriptions, it's unclear how to support such evidence in a real environment. After the requirements for the system-to-be have been elicited, the system has to be designed, developed and deployed. In each of these phases, wrong decisions may alter the compliance solutions set up in the requirements phase. Additional efforts are necessary to maintain the system aligned with law prescriptions, thus causing costs to grow. Unsurprisingly, legal compliance is gaining the attention of the software engineering community, which increasingly faces that problem, and approaches are being developed, to deal with it in a systematic way.

In this paper, we report on the application of a mixed modeling/argumentation framework. The framework has been applied on an industrial health care project, which had the purpose of developing a distributed system for the realization of an Electronic Patient Record for storing social and health information to be used in health care. The *Nòmos* modeling language was used by analysts to reason about laws and strategies and to offer model-based evidence that a set of given requirements indeed is compliant with a particular law. Subsequently, an argumentation framework has been applied on the generated models to build the knowledge base needed to support the analysis of the quality of the models.

The paper is structured as follows: section II recalls the scenario and the modeling tools adopted to model it; section III introduces the research problem arising from the depicted scenario; section IV describes the modeling/argumentation process undertaken; section V presents the findings of the approach; section VI surveys the related works; finally, section VII concludes.

II. Baseline

A. Industrial context

A health care case study. The present work moves from the outcomes of a research and development project in the health care domain. The project was intended to define the architecture for an integrated service-based system, aiming at increasing the possibilities of self-supporting life for elder

or disabled people in their own home. The system has been conceived as a network of interconnected components. Nodes of the network are mainly hospitals with their information systems. Such systems run their own databases, and provide some basic services — e.g., data search and retrieval to other members of the network. The project focused primarily on architectural and technological aspects of such system, but it also required the definition of Electronic Patient Record, as the building block for the information shared among services. The Electronic Patient Record came out of the critical need to store social and health information for use in health care. Information about patients contain sensitive data, thus requiring special care in designing it, and in defining how the data is accessed and used. This led to defining the new concept of Electronic Health Record (EHR) as the central element, around which components of the network operate.

EHR information is stored and accessed independently by the subjects that operate in the health care system: social workers, doctors, social cooperatives, relatives. The EHR, accessed via web, allows for a collaboration among the subjects, for improving health care and having social and health information, as well as economic and managerial data. When patients access a hospital, the reception operators collect information from patients, about their health status and their medical history. Through the system, it is possible to access EHR data, which collects every useful information available for the patient wherever in the network; alternatively, it should be possible to create the EHR from scratch, and broadcast it through the network. The network-wide collection of patient data forms the EHR for the patient.

B. Requirements modeling

Figure 1 represents how data are created or retrieved through the EHR system. The picture uses *i** [17], which is a goal-oriented modeling language, offering primitives to model a domain along two perspectives: the strategic rationale of the actors - i.e., a description of the intentional behavior of domain stakeholders in terms of their goals, tasks, and quality aspects (represented as soft-goals); and the strategic dependencies among actors - i.e., the system-wide strategic model based on the relationship between the depender, which is the actor, in a given organizational setting, who "wants" something and the dependee, that is the actor who has the ability to do something that contributes to the achievement of the dependers original goals. The system-to-be is modeled as a new actor of the organizational setting that enables the achievement of domain stakeholders goals, so expressing the requirements of the system. Goals express the "why" of choices, and are decomposed into sub-goals and operationalized by means of plans. Plans, in turn, may need resources to be executed. The picture represents an excerpt of the rationale behind the EHR system, limitedly to the [Healthcare Receptionist] actor. When a patient (actor [Patient]) accesses a healthcare center, at the check-in the EHR of the patient has to be retrieved from the system. The system is queried to have the data concerning that specific patient. If the data is not found locally, the EHR system queries the network and, in case of success, contacts another EHR system (owned by a different hospital), which in turn executes a local search. If the searched data do not exist in the network, the EHR system creates a new record. In this case, after the data insertion, the system broadcasts the data to the whole network. When the broadcast notification is received, each EHR system updates its local database.

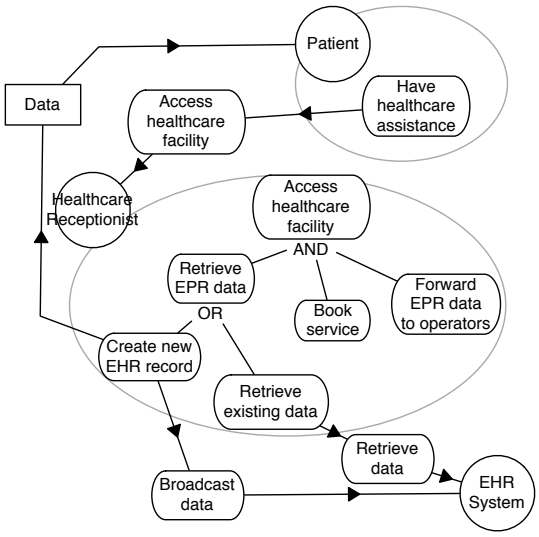

Figure 1. A schematic goal model for the EHR scenario.

Such transmission of sensitive data has raised legal issues. Privacy laws lay down prescriptions concerning the processing of personal data (in particular, sensitive data) of patients. For example, the Italian Personal Data Protection Code D.Lgs. n. 196/2003 requires the owner's confirmation for the data being processed. Before building the system, it is necessary to provide some kind of evidence that the described scenario does not violate the law.

C. Law modeling

Legal prescriptions have been modelled by means of the *Nòmos* modeling language. *Nòmos* [16] is a goal-oriented, law-driven framework intended to generate requirements through which a given information system can comply to a given law. Such requirements are referred to in the sequel as compliance requirements. *Nòmos* extends *i** by adding the capability to model law prescriptions and the link between intentional elements and legal elements.

Nòmos is based on the Hohfeldian ontology of legal concepts [9]. Such ontology contains intuitive concepts such as duty, privilege and claim, as well as more technical ones, such as power, no-claim, liability, immunity and disability. Legal concepts form normative propositions, which are the most atomic propositions carrying a normative semantics. Normative propositions contain information concerning: the subject, who is addressed by the normative proposition itself; the legal modality (i.e., the Hohfeldian concept); and the description of the object of such modality (i.e., what is actually the duty or privilege). Complex legal prescriptions are specified

in law documents composing normative propositions through conditions, exceptions, and other conditional statements. Such elements are captured in *Nòmos* by introducing priorities between normative propositions. For example, a data processor may be allowed (i.e., it has a privilege) to process the data of a subject; but the right of the subject to keep his/her data closed w.r.t. third parties has a higher priority on the privilege, thus constraining the way data is used by the processor.

Figure 2 exemplifies the *Nòmos* language used to create models of laws. The language is an extension of the *i** modelling language, so it inherits its notation. For example, actors of the domain are represented as circles, and they have an associated rationale, which contains the goals, tasks and resources of the actors. In the *Nòmos* language, the actor's rationale also contains the normative propositions addressing that actor, partially ordered through dominance relations. Finally, the holder and counter-party actors of a right are linked by a legal relation. The diagram in the figure shows an excerpt of the *Nòmos* models that represent some fragments of the Italian Personal Data Protection Code[1]. The figure shows the subject of right - the [User] - and its claim, toward the data processor, to have [Protection of the personal data] (as of Art. 7.1). However, the data processor is free to [Process performance data] (Art. 7.1). The general duty to protect the personal data is then overcame by a number of more specific duties, such as [Be informed of the source of the personal data], [Obtain updating, rectification or integration of the data], [Have owner authorization to process patient's data], [Be informed of the source of the purposes] (all from Art. 7.2), and so on. However, the data processor has the claim, towards the data subject, to refuse requests of information in (unlikely) case, for example, of the data that are processed for reasons of justice by judicial authorities.

III. Research Issues

By applying this framework to the EHR case, we produced significant results; an excerpt is schematized in Figure 3. Basically, as long as normative propositions (such as [Have owner authorization to process patient's data] in the picture) are added to the model, new goals are realized to be missing. This starts a modeling task, which *ends when no more missing goals can be identified*. However, we acknowledge that this condition of "no goals missing" is vague and arbitrary, thus causing the models to be attackable and their quality debatable.

This is particularly dangerous in an industrial setting scenario: stakeholder-generated requirements can be easily validated by stakeholders themselves; on the contrary, stakeholders not necessarily have the knowledge to evaluate law-induced requirements (i.e., requirements generated for law compliance purposes); or, if they have it, they not necessarily have the possibility to assist analysts while specifying requirements.

This problem has emerged in the EHR case. Compliance-related issues were raised by project partners *after* a large part of the requirements specification was already done, and

[1]An English translation of the law can be found at http://www.privacy.it/

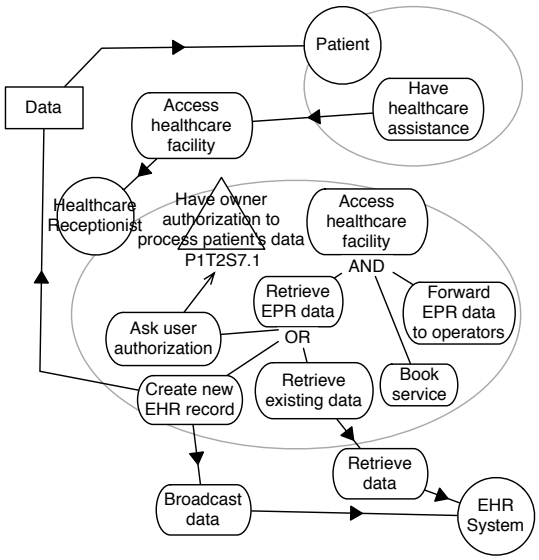

Figure 3. A goal model for the demo scenario of the EHR project.

the system development was already started. On the other hand, writing again the requirements from scratch was not an acceptable option, so an engineering effort was required to capture the key properties of system requirements, and double-check them with respect to the need of ensuring compliance.

A. Iterative Requirements Gathering Process

Although the system development has already started, requirements gathering is still an ongoing process. The complexity of the system is such that the requirements are not elicited in a once-and-for-all solution. Rather, the system *evolves* in an iterated cycle – at each iteration, needs and constraints given by the partner have to be incrementally acquired and translated into new requirements. With 'partner' we mean administrative bodies, healthcare facilities and aid agencies (both public and private), which will be managing sensitive data of patients through the EHR system and are therefore concerned by strong legal responsibilities. Companies are typically so worried of the legal consequences of breaches in the satisfaction of legal regulations that they often refuse to adopt new solutions unless it is given a proof it implements all the requirements necessary to be compliant.

The main problem becomes therefore the interaction with partners to manage a smooth transition to the phase of system adoption. The proposed solution adopts an *iterated requirements gathering phase*. As in Figure 4, at each iteration it is necessary to:

- 1: *Communicate* with partners about the status of the development and the adopted solutions.
- 2: *Bargain* with partners about the correct meaning of the adopted design decisions, on the one hand, and about the actual needs and constraints of partners, on the other.
- 3: If bargaining concludes successfully, there is an *Agreement*, which also concludes the iteration. The agreement

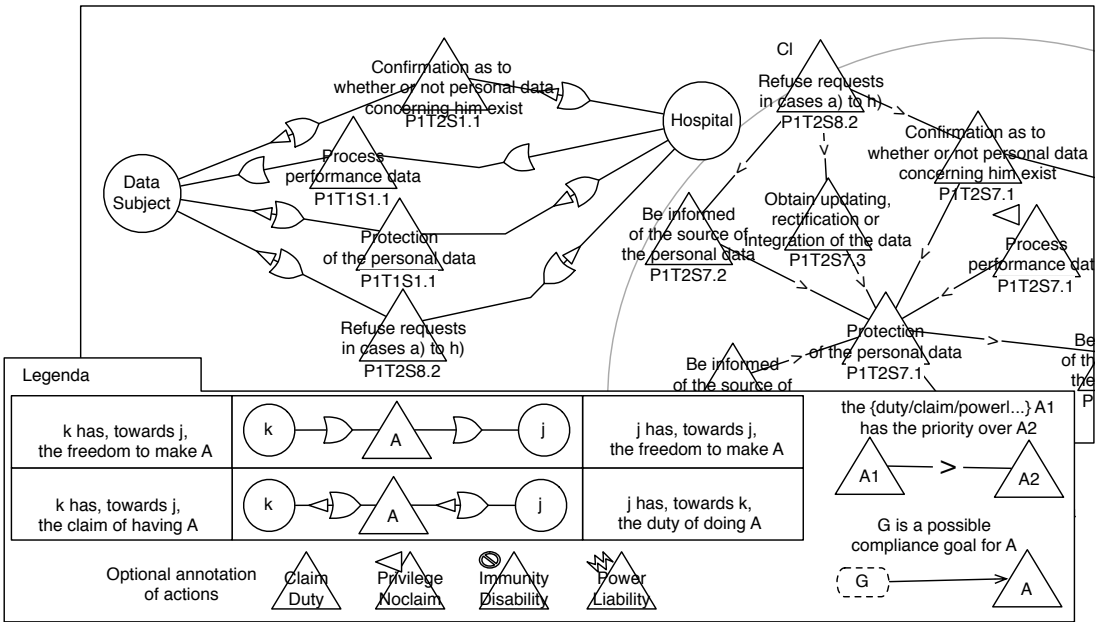

Figure 2. The *Nòmos* modelling languages: visual representation of the Italian Personal Data Protection Code.

generates the specifications for the subsequent design and implementation phase.

The process ends when an iteration produces no additional information; i.e., when the new specifications equal the existing specifications. On the other hand, **the process fails if the bargaining process fails to generate an acceptable solution**, also in the case a solution actually exists: incomplete information and misunderstandings can bring to wrong decisions, thus making the bargaining phase the critical one.

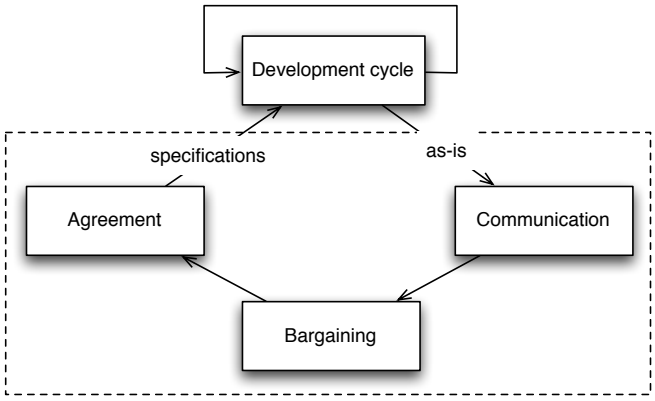

Figure 4. The develop-and-bargain continuous requirements gathering cycle.

The arising problem is therefore, how to support analysts and partners in the accomplishment of an iteration, to align the system with the needs of partners. Currently, the iteration has a free schema, guided by the initiative of analysts and with artifacts that are easily understandable even by domain experts that are not technician (e.g. social workers, nurses, accountants) like narrative documents and simplified UML activity diagrams.

IV. Argumentation-based Process

We hypothesize that an iteration may be viewed as an *evolutionary* transition of the requirements model from a state to another. Each state of the model is described by a finite number of *arguments* that justify it. The transition happens by adding new arguments, which in turn derive from the bargaining activity. In other words, during the bargaining, involved actors expose their doubts, issues, needs, desires and so on, forming a complex argumentation structure.

A. Recording Bargaining

Goal models are typically built through a process, which encompasses discussion and agreement about the proper modeling choices – for example, whether a goal should be decomposed into sub-goals or delegated to another actor. It is demonstrated that the justification process behind the construction of a goal model can be represented in terms of argumentation theory [10].

The justification of an element (a goal, task, or otherwise) in a *Nòmos* model proceeds in a recursive manner by defining and labeling a *dialectical tree* for that element, by following the steps below:

1) Choose an element (X hereafter) of the *Nòmos* model, and set it as the root of the dialectical tree.
2) Suppose that there are n arguments $A_1, \ldots, A_n$ (which here take the form of natural language sentences or paragraphs) against the compliance of X. Draw a dashed line from each A_i, for $1 \leq i \leq n$, to X, and read each line as saying that the corresponding argument *attacks* X.
3) For each A_i, for $1 \leq i \leq n$, set A_i as the root of a dialectical tree, and do step 2 above for A_i. Stop and move to step four below when no participants in the

justification/bargaining process have more arguments to give.

4) Label as "undefeated" the leaves of the dialectical tree for X. For any inner node of the dialectical tree for X, label it undefeated if and only if every child of that node is labeled "defeated". A node is defeated if and only if it has at least one undefeated node as a child. Move to step five below when the entire dialectical tree for X has been labeled.
5) The element X is justified, and thereby acceptable and compliant in the Nomos model, if X is labeled "undefeated".

The justification procedure starts by looking for arguments against X, then for arguments against these arguments against X, and so on until no participant in the modeling and compliance checking process have more arguments to give. The procedure then proceeds to label the dialectical tree of X, and the acceptability of X in the *Nòmos* model, and thereby its compliance, depends on the label that it receives.

B. Approach

We applied the argumentation framework at the end of these phases, trying to make explicit any implicit assumption in the mind of the IT experts in eliciting the requirements of the system and to motivate such requirements w.r.t. the privacy regulations the system should comply with. This discovery process allowed first to identify holes in the requirements and secondly to propose changes to the specification to better comply with the regulations. This ex-post analysis resembles the work of a detective as typically analysts and designers do not keep track in a systematic way of the reasons of their design choices as in general they do not need to prove anything to anybody. This project is different as this proof of concept is needed and with this experiment we want to make this implicit knowledge explicit.

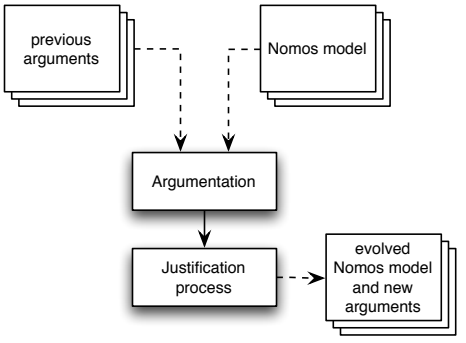

Figure 5. High level view of the activities and artifacts involved in the argumentation process: the *Nòmos* model and previous argumentations are the input to the new argumentation and justification process whose output is an evolved *Nòmos* model and a set of new arguments.

In particular, in our experience we performed the following activities (see also Figure 5); we:

1) started from the initial *Nòmos* model specification of the requirements of the EHR system
2) specified the arguments attacking:
   - model entities and relationships
   - previous attacks to entities and relationships

   so creating the dialectical trees associated to the model entities
3) detected the admissibility of the set of attacks via the justification procedure introduced above
4) if possible, transformed admissible attacking arguments into modelling actions devoted to the adding or deletion of model entities or relationships, so producing an evolved version of the model

The knowledge emerging during the argumentation step (step 2) is the main source for the model evolution at step 4 that has the objective to increase the model from the point of view of the law compliance. In particular, there are two points that have to be considered: a first point is related to the complexity of the structure of the set of arguments impacting on a single entity of the model; the evaluation of such complexity could give indications about the granularity of the model entities; in fact it can suggest possible *and/or decompositions* of the entities in the domain, in order to obtain smaller argumentation networks impacting on the single decomposing entities. A second point is related to the semantics of the arguments; in fact, analyzing the arguments, it is possible to make it emerge new model elements (both entities and relationships) to be added to the model in order to make it evolve towards a compliant model.

V. Applying the framework

Given a model of system requirements, as soon as it is shared with the stakeholders, such model can be attacked and undermined in its validity. Going back to the case exposed in section II, this happened when the rationale of the [EHR system] (not shown in the picture) is presented to the partners.

A. Description of the initial Nòmos model

In Figure 7 is depicted the compliance *Nòmos* solution that was identified during the initial requirements analysis.

The model contains two actors: the EHR representing the system-to-be and the data controller in charge of producing and managing patient's data (e.g. administrative bodies, healthcare facilities and aid agencies).

The top goal of the model is [Comply to privacy regulations], $G0$ for short, that is further decomposed into $G1$, [Support different authorization profiles], and $G2$, [Avoid duplication of medical information]. These two goals correspond to the laws $L1$ and $L3$ in [4] that are summarized in Table I. In practice, an EHR is privacy safe if it supports different authorization profiles and if its architecture avoids duplicating information.

Goal $G1$ is operationalized by the two tasks $T1$, [Define detailed privacy policies], and $T2$, [Enforce privacy policies], for defining detailed privacy constraints and for enforcing them. $G2$ is achieved by delegating the storage of sensitive information to the Data Controller via the task delegation $T3$ [Store sensitive information].

TABLE I
LAWS FROM GUIDELINESFSE AND PRIVACYPROTECTIONCODE

Name	Description
L1	criteria should be laid down to encrypt and/or keep separate the data suitable for disclosing health and sex life from any other personal data; [..] As for EHRs, secure communication protocols should be deployed by implementing encryption standards for electronic data communications between the various data controllers.
L3	The Electronic Health Record should be set up by prioritizing solutions that do not entail duplication of the medical information created by the health care professionals/bodies that have treated the given data subject.
Dlgs 196/2003 n.26(1)	Sensitive data may only be processed with the data subject's written consent and the Garante's prior authorisation, by complying with the prerequisites and limitations set out in this Code as well as in laws and regulations.
Dlgs 196/2003 n.26(2)	The Garante shall communicate its decision concerning the request for authorisation within forty- five days; failing a communication at the expiry of said term, the request shall be regarded as dismissed. Along with the authorisation or thereafter, based also on verification, the Garante may provide for measures and precautions in order to safeguard the data subject, which the data controller shall be bound to apply.

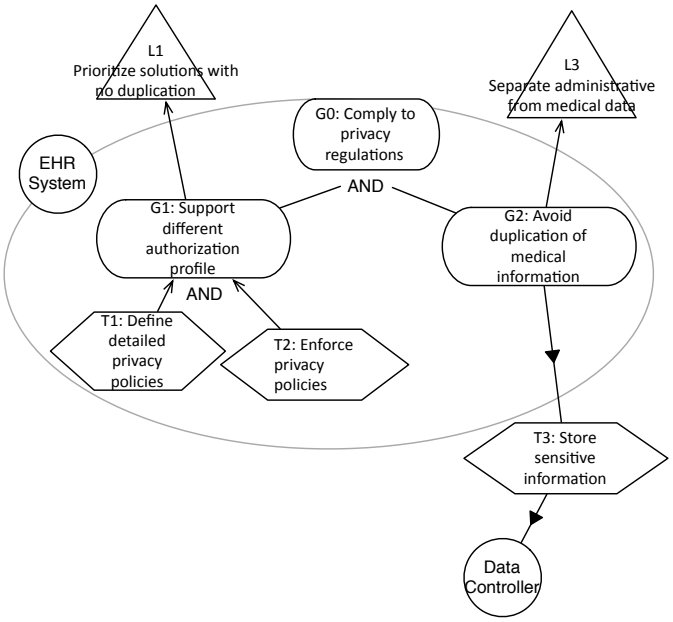

Figure 6. An excerpt of the compliance for the EHR actor, expressed by means of a *Nòmos* model.

B. Argumentation phase

Figure 7 depicts the requirements model revised in accordance with the list of attacks elicited from a series of interviews with the system designers. The arguments related to the model are depicted with rectangles containing the text of the argument and possibly a label starting with the letter "A", while the attacks of one argument to another or to model entities are depicted as dashed arrows. The arguments are summarized in Table II.

Arguments $A1.1$ and $A2.1$ attacking tasks $T1$ and $T2$ show that the implicit assumption behind the definition of these tasks on which the satisfaction of $L1$ is based are false because the EHR is not said to be a trusted party and it does not know which part of the data should be considered private and which instead are public. The solution is to delegate the realization of these two tasks to the data controller itself, the only trustable actor.

The argumentation sequence for goal $G2$ and task $T3$ is more complex and deserve more attention. Delegating task $T3$ is not enough to consider $G2$ satisfied as it does not prove that alternative solutions have been evaluated giving more priority to the ones avoiding duplication as required by $L3$, as stated by the argument $A3.3$ ("Duplication admitted if no other choices available"). In addition, it assumes the role of data controller of the healthcare professional/bodies is clearly defined (see argument $A3.2$, "Delegate duplication not cited as a solution") and that delegating the storage of the sensitive data at the data controller is not cited as a solution in the legislation, as stated by the argument $A3.1$ ("Data Controller role undefined").

A second round of discussions with the domain experts and designers of the systems brings to another level of argumentations answering and attacking the first set. In particular:

- argument $A3.1.1$ supports the initial task on the base of law Dlgs 196/2003 n.26(1)(2) [3] clearly defining the responsibility of entities to be considered as data controller, so attacking the argument $A3.1$;
- argument $A3.2.1$ ("No duplication outside data controller boundaries") makes explicit the assumption of the designers that the law does not allow duplication outside the boundaries of the data controller, so that delegating the storage of the sensitive information to the Data Controller actor is an admissible solution;
- argument $A3.3.1$ ("Evaluated different solutions") states that different solutions have been evaluated before the final design of the system, so attacking $A3.3$.

A third round of discussions have been executed with the experts. In this case no other arguments emerged except for the argument $A3.2.1.1$ ("Duplication allowed but Agreement Needed") that partially attacked the argument $A3.2.1$ as shown in Figure 7. The argument $A3.2.1.1$ states that the duplication is allowed but an agreement is needed.

Justification procedure

The execution of the algorithm for justification for all the

TABLE II
ARGUMENTATIONS.

Name	Description
A1.1	EHR does not know what is sensitive and what is public
A2.1	EHR is not said to be a trusted party
A3.1	Data Controller role undefined
A3.2	Delegate duplication not cited as solution
A3.3	Duplication admitted if no other choices available
A3.1.1	Healthcare professional/bodies are data controller by law 26(1)(2) of Dlgs196/2003
A3.2.1	No duplication outside data controller boundaries
A3.2.1.1	Duplication allowed but agreement needed
A3.3.1	Evaluated different solutions

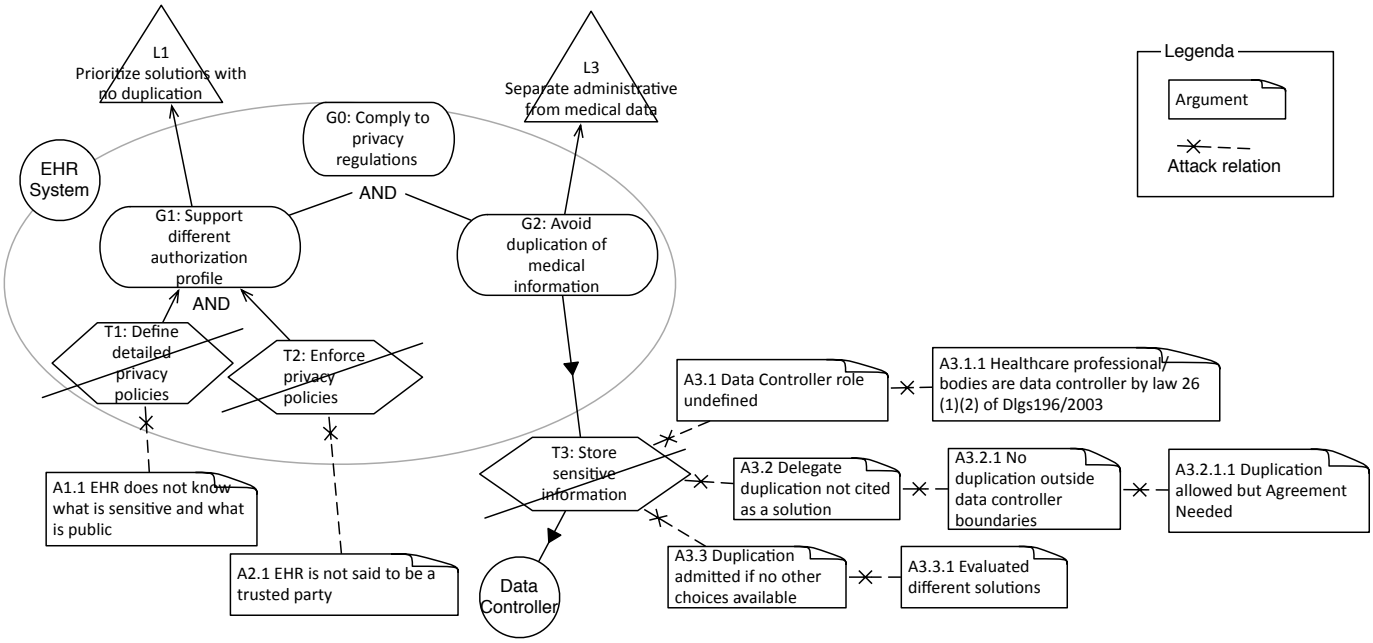

Figure 7. Attacks modeling. The existing *Nòmos* model and the set of arguments attacking model entities and other arguments.

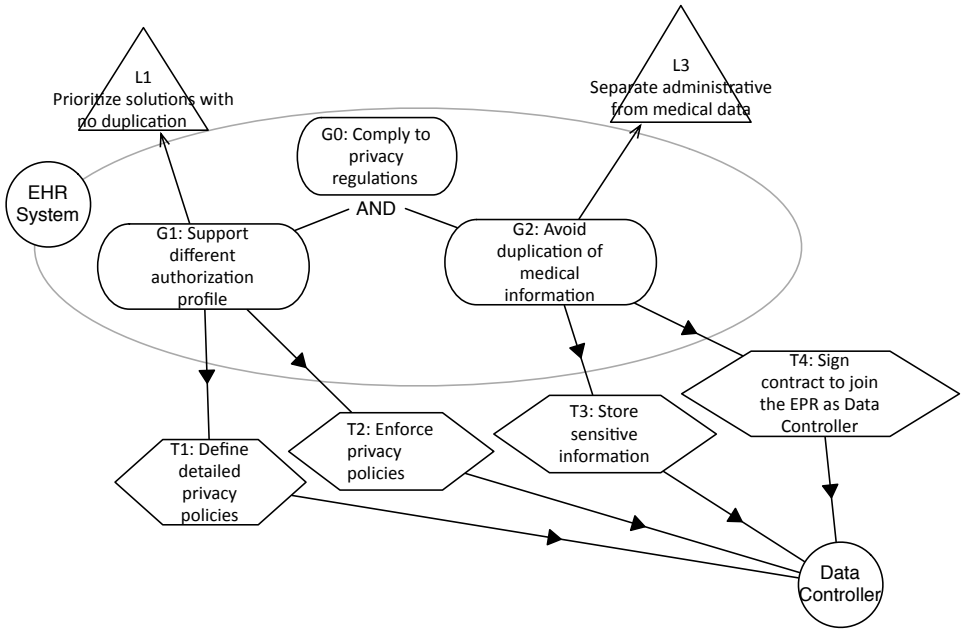

Figure 8. Extended model. The evolution of the model on the bases of the arguments related to the existing *Nòmos* model.

dialectical trees in Figure 8 brings the following results:

- arguments $A1.1$ and $A2.1$ succeded in attacking the current model; in fact, referring to $A1.1$ and following the justification algorithm, initially $A1.1$ (a leaf in the dialectical tree) is labelled "undefeated", the task $T1$ is then labelled "defeated", so it is successfully attacked by the arguments; in this case a modification in the model is requested;
- following the same reasoning, the arguments $A3.1$ and $A3.3$ are rejected thanks to the arguments $A3.1.1$ and $A3.3.1$ respectively, so $T3$ is not attacked by these dialectical trees;
- the line of arguments $A3.2$, $A3.2.1$, $A3.2.1.1$ produces a valid attack to the model (in particular to $T3$) related to the need of an agreement to consider the Data Controller a valid point of data duplication; also this third argument calls for the modification of the model.

Model evolution

Based on this output the analyst restructures the model, obtaining the model version depicted in Figure 8, to derive a specification that is compliant to the legislation and properly answers the weaknesses highlighted in the argumentation process.

The changes considered in this case are:

- delegation of some tasks to the data controller: in particular, tasks $T1$ is delegated because the data controller is the only entity capable of defining in a correct way the privacy policies on its data; by delegating task $T2$ we make sure only the data controller has full control on the data accessed;
- extension of the model with further details delegating the realization of goal $G2$ to two tasks: $T3$, ([Store sensitive information]) already existing in the previous version of the model (see task $T3$ in Figure 7) and adding the task $T4$ ([Sign contract to join the EHR as Data Controller]) to "cover" the attack $A3.2.1.1$. This changes assure, respectively, that the sensitive information is maintained at the data controller and that the data controller agrees to retain the responsibility in storing its data and in applying the privacy policy to make sure sensitive data is accessed only by authorized consumers.

C. Evaluation and discussion

Qualitative analysis The experience received a positive feedback from the designers primarily because it saves time in defining the set of requirements. Typically, the reasons leading to a certain requirement are not traced and consequently people waste their time re-discussing again decisions that have already been taken. By tracing the requirements together with their argumentations it is easier to get an approval with the users and to reach a more stable model faster.

Furthermore, the analysis approach allows to understand core requirements that have high impact on the design and implementation early. In our reference scenario changing the way privacy requirements are enforced from the EHR to the data controller has a serious impact on the design of the system and the subsequent development phases.

The case study presented in this paper focuses on the more critical parts of the project and in particular on the aspects that are more important from the legal point of view that is how to design and prove the EHR is compliant to the privacy regulations. These aspects are also the more controversial and debated with the users since the project is new and there is no prior agreed solutions on which to base the decisions.

A perplexity emerged from the designers regarding the scalability of the approach applied on a wider and more complex scenario. The manageability of the model requirements enriched with the argumentations depends on the complexity of the argumentation chains, that is primarily on their length and also on their numerousness in the model. Intuitively, a model with many, long argumentation chains is more complex to deal with and to evolve in the future. Another dimension of complexity to consider is the length of the lifetime of an argumentation before it is absorbed in the model itself in form of new elements or in transformations of the existing model.

Quantitative analysis. In this case study we showed how the model (and consequently the design of the system) can change reasoning on argumentations.

Given $c \in A(M)$ one of the argumentation chains of model M we indicate with $L(c)$ its length. For example, the chain $c = (A3.1, A3.1.1)$ has length 2. We define the *weight of a chain* as: $W(c) = L(c)(L(c) + 1)$. Intuitively the weight of a chain reflect its length as, in general, longer chains are more costly than the shorter ones.

The weight of a chain is used to define the cost associated to the entire argumentation model. In particular, the cost of a certain argumentation scenario is defined as follows: $W(M) = k * \sum_{c_i \in A(M)} W(c_i)$ where k is a parameter to be tuned during the experimentation and that in our case was set to $1/8$.

The weight associated to the model produced during the argumentation process is 2.5. Notice how the same model with all argumentation chains of length 1 will have a lower cost (in this case 1). In this case study we saw that we introduce 9% of tasks w.r.t to the existing tasks in the model.

We also saw that the algorithm for the propagation of argumentations allows to keep under control the complexity of the argumentation chains and to prune the elements that are no more valid.

VI. Related Works

Law has been investigated in the past as an application of AI techniques for performing automatic reasoning and deductions [14]. In recent years, there has been various efforts to deal with law-related issues from the requirements elicitation phase. Antòn and Breaux have developed a systematic process, called semantic parameterisation [5], which consists of identifying in legal text restricted natural language statements (RNLSs) and then expressing them as semantic models of rights and obligations (along with auxiliary concepts such as actors and constraints). Secure Tropos [8] is a framework for security-related goal-oriented requirements modelling that, in order to

ensure access control, uses strategic dependencies refined with concepts such as: trust, delegation of a permission to fulfill a goal, execute a task or access a resource, as well as ownership of goals or other intentional elements. Along similar lines, Darimont and Lemoine have used KAOS as a modelling language for representing objectives extracted from regulation texts [6]. Such an approach is based on the analogy between regulation documents and requirements documents. Ghanavati et al. [7] use GRL to model goals and actions prescribed by laws. This work is founded on the premise that the same modelling framework can be used for both regulations and requirements. Likewise, Rifaut and Dubois use *i** to produce a goal model of the Basel II regulation [12]. A goal-only approach has also be experimented in the Normative *i** framework [15], in which the notion of compliance was not considered. Finally, much work has been done in AI on formalizing law, e.g. [11], [13]. We use some of this work as a foundation for our framework. However, our software engineering task of having a person check for compliance between a model of law and another of requirements is different from that of formalizing law for purposes of automatic question-answering and reasoning.

VII. Conclusions

In this paper, we have presented a method and an exploratory study on the use of argumentation as theoretical solution for a practice problem concerning law compliance of information systems.

The method mixes *Nòmos*, a Goal-Oriented requirements engineering technique with an argumentation framework. The *Nòmos* modeling language is intended to reason about laws and strategies and to offer model-based evidence that a set of given requirements is compliant with a particular law. The argumentation framework is applied on the *Nòmos* models to build the knowledge base needed to support the analysis of the quality of the models, allowing to support their controlled evolution in order to increase their compliance to laws.

The method has been profitably applied by analysts in an industrial health care project, where the legal issues, in particular those relate to the privacy and management of data, are critical.

As future work on the methodological side, we plan to refine the method in order to specify more detailed guidelines for the analysts, especially to assure the scalability of the approach to larger case studies.

References

[1] Medical privacy - national standards to protect the privacy of personal health information. Office for Civil Rights, US Department of Health and Human Services, 2000.

[2] Online news published in dmreview.com, november 15, 2004.

[3] Personal data protection code. In *Legislative Decree no. 196*, June 2003.

[4] Guidelines on the electronic health record and the health file. Italy's Official Journal n. 71, 2009.

[5] Travis D. Breaux, Annie I. Antón, and Jon Doyle. Semantic parameterization: A process for modeling domain descriptions. *ACM Trans. Softw. Eng. Methodol.*, 18(2):1–27, 2008.

[6] Robert Darimont and Michel Lemoine. Goal-oriented analysis of regulations. In Régine Laleau and Michel Lemoine, editors, *ReMo2V, held at CAiSE'06*, volume 241 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2006.

[7] Sepideh Ghanavati, Daniel Amyot, and Liam Peyton. Towards a framework for tracking legal compliance in healthcare. In John Krogstie, Andreas L. Opdahl, and Guttorm Sindre, editors, *CAiSE*, volume 4495 of *Lecture Notes in Computer Science*, pages 218–232. Springer, 2007.

[8] Paolo Giorgini, Fabio Massacci, John Mylopoulos, and Nicola Zannone. Requirements engineering meets trust management: Model, methodology, and reasoning. In *ITRUST-04*, volume 2995 of *LNCS*, pages 176–190. SVG, 2004.

[9] Wesley Newcomb Hohfeld. *Fundamental Legal Conceptions as Applied in Judicial Reasoning*. Yale Law Journal 23(1), 1913.

[10] Ivan Jureta, Stéphane Faulkner, and Pierre-Yves Schobbens. Clear justification of modeling decisions for goal-oriented requirements engineering. *Requir. Eng.*, 13(2):87–115, 2008.

[11] Vineet Padmanabhan, Guido Governatori, Shazia Wasim Sadiq, Robert Colomb, and Antonino Rotolo. Process modelling: the deontic way. In Markus Stumptner, Sven Hartmann, and Yasushi Kiyoki, editors, *APCCM*, volume 53 of *CRPIT*, pages 75–84. Australian Computer Society, 2006.

[12] Andre Rifaut and Eric Dubois. Using goal-oriented requirements engineering for improving the quality of iso/iec 15504 based compliance assessment frameworks. In *Proceedings of RE 2008*, pages 33–42, Washington, DC, USA, 2008. IEEE Computer Society.

[13] Giovanni Sartor. Fundamental legal concepts: A formal and teleological characterisation. *Artificial Intelligence and Law*, 14(1-2):101–142, April 2006.

[14] M. J. Sergot, F. Sadri, R. A. Kowalski, F. Kriwaczek, P. Hammond, and H. T. Cory. The british nationality act as a logic program. *Commun. ACM*, 29:370–386, May 1986.

[15] Alberto Siena, Neil A. M. Maiden, James Lockerbie, Kristine Karlsen, Anna Perini, and Angelo Susi. Exploring the effectiveness of normative i* modelling: Results from a case study on food chain traceability. In *Proceedings of CAiSE 2008*, pages 182–196, 2008.

[16] Alberto Siena, John Mylopoulos, Anna Perini, and Angelo Susi. Designing law-compliant software requirements. In *Conceptual Modeling - ER 2009*, pages 472–486, 2009.

[17] Eric Siu-Kwong Yu. *Modelling strategic relationships for process reengineering*. PhD thesis, Toronto, Ont., Canada, Canada, 1996.