Legal Goal-Oriented Requirement Language (*Legal GRL*) for Modeling Regulations

Sepideh Ghanavati
CRP Henri Tudor
Luxembourg City, Luxembourg
sepideh.ghanavati@tudor.lu

Daniel Amyot
EECS, University of Ottawa
Ottawa, Canada
damyot@uottawa.ca

André Rifaut
CRP Henri Tudor
Luxembourg City, Luxembourg
andre.rifaut@tudor.lu

ABSTRACT

Every year, governments introduce new or revised regulations that are imposing new types of requirements on software development. Analyzing and modeling these legal requirements is time consuming, challenging and cumbersome for software and requirements engineers. Having regulation models can help understand regulations and converge toward better compliance levels for software and systems. This paper introduces a systematic method to extract legal requirements from regulations by mapping the latter to the Legal Profile for Goal-oriented Requirements Language (GRL) (*Legal GRL*). This profile provides a conceptual meta-model for the anatomy of regulations and maps its elements to standard GRL with specialized annotations and links, with analysis techniques that exploit this additional information. The paper also illustrates examples of Legal GRL models for *The Privacy and Electronic Communications Regulations*. Existing tool support (jUCMNav) is also extended to support Legal GRL modeling.

Categories and Subject Descriptors

D.2.1 [**Software Engineering**]: Requirements / Specifications

General Terms

Legal Aspects, Design, Management

Keywords

Goal Modeling, Legal Requirements, Software Compliance

1. INTRODUCTION

Nowadays, software companies and developers need to comply with a large number of regulations and ensure that their business and system requirements are aligned with relevant legal requirements. However, these regulations are complex and vague by nature, and they are constantly evolving. In order to help requirements and software engineers to better understand and align their requirements with regulations, much work has been done to model the legal requirements, at times in the same notation as the software requirements. Goal-oriented requirements engineering methods have been used to model regulations in recent years [1, 6, 11, 15, 17, 18, 19]. The majority of these approaches apply i^*-based or TROPOS-based notations such as i^* itself, *Nòmos*, SecureTropos, Secure i^* or the Goal-oriented Requirement Language (GRL). Having such models can help requirements and software engineers to better align legal requirements with system requirements, leading to software applications that are compliant with regulations while avoiding the high cost of non-compliance (e.g., fines, loss of reputation, and even prison). In 2012 for example, the Federal Trade Commission (FTC) charged Google a record of 22.5 million USD as civil penalty [4]. The *Commission Nationale de l'Informatique et des Libertés* (CNIL), in France, also fined Google with 150,000 EUR [12] for non-compliance with the "French data privacy law" [3].

A recent systematic literature review of goal-oriented modeling frameworks for compliance [7] showed that although using goal models helps software engineers, requirements analysts, and organizations in analyzing and understanding regulations and achieving compliance, there is still some gaps between current approaches and the needs of organizations. One of the problems identified in this review is that existing approaches do not provide systematic guidelines on how to extract and model legal statements with goal modeling notations. This problem can lead to difficulties in reusing or recreating these models in multiple contexts, and it makes it hard for legal experts and compliance analysts to validate such models.

This paper contributes partial solutions to this problem. It first identifies major elements of regulation structures and proposes a meta-model that captures them. Then, it provides guidelines on how to extract these elements and maps them to goal models. The goal modeling language used here, GRL, was chosen because this is the only standardized goal modeling available so far, it is combined with Use Case Maps (enabling one to capture business processes) in the User Requirements Notation (URN) standard, it provides strategies and various tool-supported evaluation algorithms, and it supports integrated mechanisms for profiling the language to specific domains. GRL is tailored here through a lightweight legal profile to capture the elements of law, including deontic modalities. This new profile is called Legal GRL. Legal GRL is part of the LEGAL-URN framework, which includes Legal UCM as well [5]. The paper also

MiSE'14, June 2 – June 3, 2014, Hyderabad, India
Copyright 2014 ACM 978-1-4503-2849-4/14/06...$15.00
http://dx.doi.org/10.1145/2593770.2593780

presents examples from a case study in the privacy laws in UK, and finally provides some discussion about future work and conclusion.

2. FROM REGULATIONS TO LEGAL GRL

The Goal-oriented Requirement Language (GRL) is combined with the UCM notation in the User Requirements Notation standard [10]. GRL's main concepts come from management and from socio-technical systems and include actors, which have intentional elements (goals, softgoals, tasks, and resources) and indicators, linked through various relationships (AND/OR decompositions, dependencies, and weighted contributions). The bottom part of Figure 1 shows the notation used for the main GRL intentional elements and links. GRL strategies provide initial satisfaction values to some of the intentional elements, which are then propagated to the other elements and their actors through these relationships. GRL has no specific concepts for regulations, nor does it understand deontic modalities (i.e., obligations and permissions) commonly found in legal documents. However, GRL/URN supports lightweight extension mechanisms, such as metadata (name-value pairs and stereotypes) for annotating any element and generic typed URN links between any pair of elements, which enable the language to be *profiled* to a specific domain. Evaluation strategies and profiling abilities turn GRL into a unique and attractive option as a goal-oriented language for our work.

This section explains our method for creating Legal GRL models from regulations. In our method, we use the well-known Hohfeldian classification of rights [8]: Duty-Claim, Privilege-NoClaim, Power-Liability and Immunity-Disability. Through refinements, we transform these Hohfeldian classifications into a format that exploits deontic modalities at the GRL level. Our method contains two steps:

- **Step 1.** Developing a Hohfeldian model of law – We classify each statement of the legal document based on Hohfeld's classes of rights and annotate it accordingly. We also identify "subject", "verb", "actions", "preconditions", "exceptions" and "cross-references" parts in each legal statement.
- **Step 2.** Developing Legal GRL models – We refine the Hohfeldian classes of rights into deontic *Permission* and *Obligation* goals, develop the goal model of the law, and annotate the intentional elements with «Permission», «Obligation», «Precondition», «Exception», and «XRef» stereotypes.

Each of these steps is discussed in detail in the following two sub-sections.

2.1 Developing a Hohfeldian Model of Law

This intermediate model uses Hohfeldian concepts for classifying legal statements, in addition to legal statement structural elements such as "subject", "modality","verb", "actions", "preconditions", "exceptions" and "cross-references". To extract Hohfeldian statements from legal documents and build the Hohfeldian model, we consider the following rules:

- **Rule 1** – Each legal statement shall be atomic. This means that each legal statement contains one <actor> (the subject) and one <modal verb> (modality). However, the statement can also have one to many <clause> (<verb> and <actions>), 0 to many <cross-reference>, 0 to many <precondition> and 0 to many <exception>.
- **Rule 2** – If a legal statement contains more than one modal verb, it must be broken down into atomic statements.
- **Rule 3** – Exceptions are treated as separate statements.
- **Rule 4** – If there is an internal cross-reference in a legal statement, we replace the referencing part of the statement with the referenced statement and break the statement into atomic statements. External cross-references also break into atomic statements but they are mapped to the original legal statement via links.

The structure of a Hohfeldian model is captured as a metamodel in the top part of Figure 1. A Hohfeldian model consists of one to many atomic Hohfeldian statements structured as described in Rule 1. Each such statement also has a Hohfeldian type. We identify Privilege-NoClaim, Duty-Claim, Power-Liability and Immunity-Disability statements based on the potential modalities. Table **??** shows a non-exhaustive list of natural language modalities mapped to their respective Hohfeldian classes. The first column represents the classes, the second column represents the modalities for Duty, Privilege, Power and Immunity (one side of the dual Hohfeldian relationships) and finally the last column shows the modalities for Claim, NoClaim, Liability and Disability (the other side of the dual relationships).

Duty-Claim statements can have keywords such as *must, shall, will, have to, is necessary to*, while Privilege-NoClaim statements include keywords as *may, might, should*. For all classes, other synonyms (including expressions) also exist. Power-Liability and Immunity-Disability are statements that can override the two above statements and they are mainly decomposed into either of the first two types. The Power-Liability gives a power to the actor to perform an action that was not allowed in a regular situation. This means that the Power-Liability statements may contain the keywords *must allow actor to, shall authorize an actor to*, etc. Immunity-Disability disables an actor from performing an action and prevents him from being penalized. Immunity-Disability keywords can be *must not allow an actor to, shall not permit an actor to.*Note again that the lists provided in this table are not exhaustive and there can be many more modalities that can match these classes. This tagging is currently performed manually.

2.2 Developing Legal-GRL Models

Once the Hohfeldian model is available, the four groups of Hohfeldian rights must be refined into permissions and obligations. Note that a modal interdiction is represented as the negation of a permission. We hence specialize GRL to capture two new types of legal goals. This GRL model is called *Legal GRL*. To build the Legal GRL model, we create *Obligation* and *Permission* (soft-)goals as derivatives of the Hohfeldian model and then refine these goals until they can be expressed in terms of operationalized tasks.

In addition to these two types of goals and in order to capture preconditions, exception and external cross-references, we introduce three other types of goals, also mapped to the Hohfeldian model: precondition goals/softgoals, exception

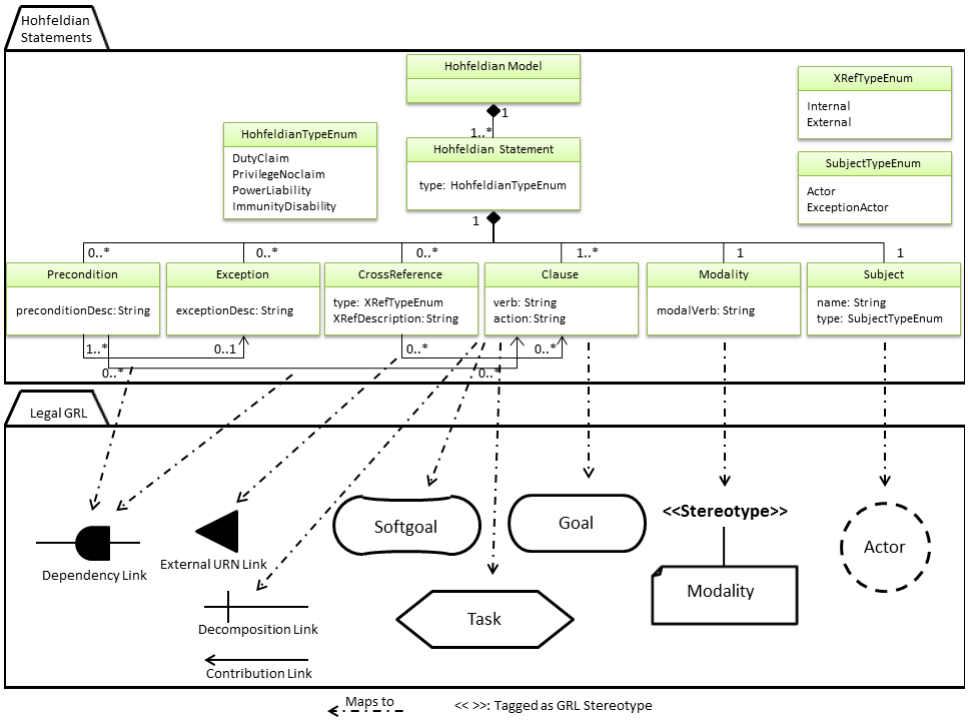

Figure 1: Mapping from the Hohfeldian Meta-Model to Legal GRL

goals/softgoals and XRef goals/softgoals. In Legal GRL, each of the intentional elements (IEs) or actors is annotated with a stereotype, with suitable support from jUCMNav, an Eclipse tool for URN modeling, analysis and transformations [13]. The new stereotypes for Legal GRL are: «Permission», «Obligation», «Precondition», «Exception», «XRef» and «ExceptionActor». A set of user-selectable rules in jUCMNav (implemented with the Object Constraint Language) ensure the well-formedness of Legal GRL models [5].

As in conventional GRL models, intentional elements are connected together through *AND/OR/XOR-decomposition*, weighted *Contribution*, *Dependency*, or typed *URN* links. However, in Legal GRL models, *Dependency* links and *External URN* links are defined more specifically:

- *Dependency* links are links from a «Precondition» to an intentional element (usually a softgoal or a goal) and are implemented as standard GRL dependency links.
- *External URN links* (▶) are also used to map an «XRef» goal to an external cross-reference intentional element in another legal model.

Actors in the Legal GRL model can be divided into two types: regular Actors and ExceptionActors. ExceptionActors only need to be satisfied if the preconditions for their exception goals are satisfied. The ExceptionActors have exception goals bound to them and are annotated as «ExceptionActors» in Legal GRL.

To build the Legal GRL model of a regulation from its Hohfeldian model, we follow these steps:

1. Refine Hohfeldian statements into obligation and permission statements.
2. Any Duty - Claim statement is refined to obligation whereas any Privilege - NoClaim statement is refined to permission.
3. Power - Liability and Immunity - Disability are first refined to Duty - Claim and Privilege - NoClaim statements and then to permission and obligations.
4. Map obligation and permission statements to obligation/permission goals in the Legal GRL model.
5. Map preconditions, external cross-reference and exceptions statements to precondition, XRef, and exception goals (respectively) in the Legal GRL model.
6. Internal cross-reference statements are modeled as regular obligation/permission goals. There will be a GRL link from the internal cross-reference goal to the main goal to which it is cross-referenced.
7. For an external cross-reference statement, treat the statement as a regular statement, model it in a separate GRL model, and link the high-level XRef goal to the goal that had it cross-referenced.
8. Exception goals must have at least one precondition goal. If the precondition is satisfied, then it triggers the exception goal.
9. Each subject in the Hohfeldian statements is modeled as an actor in the Legal GRL model whereas each exception subject in the Hohfeldian statement is modeled as an exception actor in the Legal GRL model.
10. Map actions in the Clause part of an Hohfeldian statement to tasks in the Legal GRL model.
11. Provide links between the goals and tasks in the Legal GRL model as follows:

    - If obligation and permission goals are connected to softgoals, the links are either contribution or

Table 1: Mapping Hohfeldian Models to Legal GRL

Hohfeldian Model Element	GRL Element
Subject	Actor, ExceptionActor
Modality	Obligation/Permission
Clause	Intentional Elements (IEs)
Crossreference	XRef IEs
Precondition	Precondition IEs
Exception	Exception IEs

decomposition links (i.e., similar to conventional GRL models).

- If obligation or permission goals are refined into more obligation or permission goals, they are connected through decomposition links.
- Tasks are connected to obligation and permission goals through AND, OR or XOR decomposition links.
- Preconditions are connected to intentional elements through dependency links.
- Exception and XRef goals are treated similarly to obligation and permission goals.

Table 1 provides a summary of the mapping between the Hohfeldian model's elements and the Legal GRL model's elements, while Figure 1 illustrates this mapping.

Once the Legal GRL model is created, it can be used as (or traced to) a set of requirements for software development. This model can be combined with the GRL model created for business requirements that the software needs to satisfy.

3. EXAMPLES OF APPLICATION

In this section, we provide examples from The Privacy and Electronic Communications (EC Directive) Regulations 2003 [9], a regulation in UK, to show how to create a Legal GRL model from such text.

Article 14 (2) (Restrictions on the processing of location data) states: *Location data relating to a user or subscriber of a public electronic communications network or a public electronic communications service may only be processed,(a)where that user or subscriber cannot be identified from such data; or(b)where necessary for the provision of a value added service, with the consent of that user or subscriber.*

This statement includes an actor, a modal verb, a clause and some preconditions. The statement says that "The location data of a user can be processed (by the system), if the preconditions mentioned in the statement are satisfied." The subject of the statement is not explicitly mentioned. However, by breaking down the statement, we can deduct that the subject of the statement is the system or software that uses the location data or a person that uses the location data via an electronic communications means.

This statement contains the keyword *may* and can be classified as a *Privilege-NoClaim* statement. Any Privilege-NoClaim statement is a permission. Thus, this statement is further mapped to a *Permission* goal, Process Location Data. The actor in this statement is the subject we identified above, which is System. Table 2 illustrates elements of this Hohfeldian model. The statement includes three preconditions, which all need to be satisfied for the *Permission* goal to be achieved. To be able to show the dependency between the three preconditions and the *Permission* goal Process Location Data, we create an intermediate precondition

Table 2: EC - Article 14 (2)

Actor	System
Modal Verb	may
Clause	process location data of a user
Precondition 1	that user or subscriber cannot be identified from such data
Precondition 2	necessary for the provision of a value added service
Precondition 3	with the consent of that user or [..]

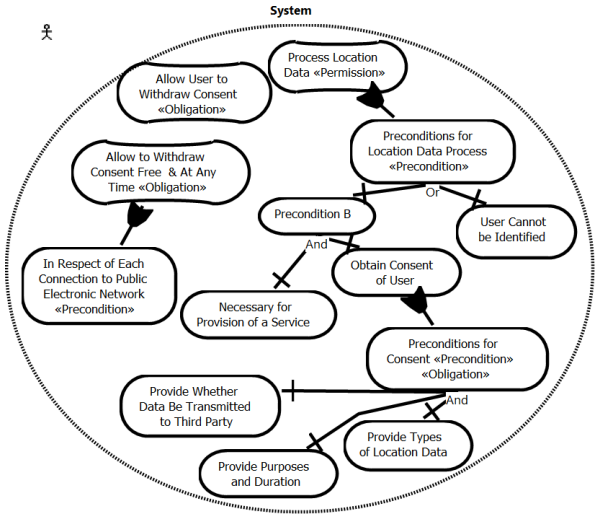

Figure 2: EC - Article 14

Preconditions for Location Data Process, which is connected to the goal with a dependency link and has the three child preconditions connected via AND-decomposition links. Figure 2 shows the Legal GRL model of this article.

Article 14(3) states that *Prior to obtaining the consent of the user or subscriber under paragraph (2b), the public communications provider in question must provide the following information to the user or subscriber to whom the data relate (a)the types of location data that will be processed; (b)the purposes and duration of the processing of those data; and (c)whether the data will be transmitted to a third party for the purpose of providing the value added service.*

This statement contains *must* keywords. The public communications provider has to provide information about location data to the user prior to obtaining the consent. This duty can be done via the software or system. This statement is of kind *Duty-Claim* and explains the duty of the communication provider and its software. This Duty-Claim statement will be refined to an *Obligation* goal. In addition, this statement with its subparts need to be satisfied before the consent can be obtained. Thus, this statement and its subparts become the *Precondition* of the goal Obtain Consent of User. To show these dependencies, we again add an intermediate "Precondition-Obligation" goal Preconditions for Consent with three child goals linked via AND-decompositions. Table 3 and Figure 2 present the elements of this statement and their Legal GRL model.

Article 14 (4) indicates that *A user or subscriber who has given his consent to the processing of data under paragraph (2b) shall (a)be able to withdraw such consent at any time,*

Table 3: EC - Article 14 (3)

Actor	Public communications provider via system
Modal Verb	must
Clause 1	provide types of location data that [..]
Clause 2	provide purposes and duration of [..]
Clause 3	provide whether the data will be transmitted to a third party for [..]
Precondition	-

Table 4: EC - Article 14 (4)

Actor	System
Modal Verb	shall allow user to
Clause 1	withdraw such consent
Clause 2	withdraw such consent, using a simple means and free of charge
Precondition	in respect of each connection to [..] or each transmission of [..]

and (b)in respect of each connection to the public electronic communications network in question or each transmission of a communication, be given the opportunity to withdraw such consent, using a simple means and free of charge.

In this article, the user has a claim to withdraw the consent from the system at any time, with using a simple means and free of charge. Thus, the system has a duty over the user to provide the means for user to be able to withdraw the consent at any time. This statement is of type *Duty-Claim*. Since, we are dealing with the duties of the system, we provide the elements of the Hohfeldian model based on the system as an actor (Table 4 and Figure 2). The two *Obligation* goals are Allow User to Withdraw Consent and Allow to Withdraw Consent Free & At Any Time, which are mapped to clauses 1 and 2 of the Hohfeldian model.

4. DISCUSSION

As shown in the previous sections, we extracted legal requirements from a regulation with the help of an intermediate Hohfeldian model. Then, we created the Legal GRL model, a formalized goal model with deontic modalities that captures legal system requirements. Such a model can be combined with more operational or business-level models of software systems (with UCM scenarios [14], UML models, or others), leading to a more complete, coherent, traceable, and relevant set of requirements for software applications. Such legal models are also reusable across multiple applications.

With deontic modalities, precondition and exceptions, it also becomes possible to analyze the compliance of the software with its relevant regulations. When a legal goal is of type *Obligation*, it must be satisfied completely. Thus, there must be some functionality in the software that satisfies this goal. For example, the software shall provide users the means to withdraw consent in order to satisfy the *Obligation* goal Allow User to Withdraw Consent.

When an *Obligation* or *Permission* goal depends on a precondition, the latter must be met before the goal is achieved. If the precondition is not first met, the goal linked to it cannot be achieved either, as otherwise the legal statement would be violated. In our example, the *Permission* goal Process Location Data, from Article 14(2), can only be satisfied if is preconditions are satisfied. Thus, if the software processes location data without satisfying the three preconditions, then the system is violating the regulation.

When a goal is of type of *Permission*, the software may or may not satisfy this goal. If the software embeds a functionality that can achieve such goal (e.g., if it processes location data (Article 14(2)), then all of the sub-goals or preconditions related to this Process Location Data goal shall be satisfied. Otherwise, the software will be in breach of the regulation. Yet, if the software is not meant to include functionalities that would meet that *Permission*, then the latter does not apply to this system and can be removed from the analysis. In the Legal-URN framework [5], such specialization of the legal model for a particular context can be done by adding a «No» stereotype to the *Permission* goals that are not relevant to the software. For example, if the software is not meant to process location data, this goal can be tagged with «No» and be formally removed from the analysis.

Note that the Legal-URN analysis scenarios discussed above are automated through a specialized goal evaluation algorithm recently added to jUCMNav [13]. Such analysis works best when the software functionalities can be summarized in terms of another (standard) GRL model, connected to the Legal GRL model of the regulation through contribution and traceability links. This also scales to multiple regulations, even in the presence of unexpected conflicts in regulations from different jurisdictions [5].

The approach we proposed here aims to help companies understand relevant legal requirements and achieve compliance. However, this approach does not prevent the need to have lawyers validate the models. The Legal GRL models are also linked to the original text to avoid mis-interpretation of the law. In addition, it is inevitable to have several interpretations of a legal statement by different lawyers, auditors or judges. With the help of alternatives (modeled with tasks in Legal GRL), one can however capture many interpretations of a legal statement in one model and analyze these alternatives via GRL analysis methods to identify the solution that satisfies both regulation and organization needs.

This approach has also been tested in a case study with a hospital in Ontario, Canada. In this case study, we identified four regulations related to health care in Ontario. First, we started with the Personal Health Information Protected Act (2004) and we modeled 58 PHIPA statements with Legal GRL. Next, we modeled 20 more relevant legal statements from three additional regulations. The details of these models are discussed in [5]. This extensive case study from the health care domain together with the examples provided in this paper from the privacy and electronic communication domain, suggest that Legal GRL can be used in modeling legal requirements, across different domains.

5. CONCLUSIONS

A recent systematic literature review [7] identified the need for guidelines to model legal statements with goal modeling languages. In this paper, we provided a method to extract structured elements from regulations and model them with Legal GRL. Since Legal GRL models use standard GRL, supplemented with stereotypes that can be exploited during automated compliance analysis, this approach helps software developers capture legal requirements at the same time as they are capturing other software requirements. This

can help developers avoid costly non-compliance issues. Moreover, goal-oriented methods have been recognized for providing support through the complete software engineering lifecycle, which is important for the compliance problems [2]. The cost of creating a Legal GRL model can be amortized as it can be developed once per regulation/law, and then made available and reused elsewhere. Legal GRL models can also be specialized to the specific context of a software project through the use of «No» stereotypes, which indicate the parts of a regulation that do not apply to a system. The larger Legal-URN framework also provides links to legal sources and design models (goal and business/scenario models), which can help provide traceability and claim compliance during software certification.

The meta-model introduced here could possibly lead to standardized GRL extensions for modeling business rules, policies and internal controls. Business rules contain preconditions, exceptions and cross-references, which are not captured by standard GRL. Re-using these concepts in the goal modeling of business or software rules could help developers create legally-compliant specifications, to better analyze alternative design decisions (in terms of their impact on compliance), and derive rule-based descriptions for software.

For future work, we aim to provide better support for the Compliance Function, especially in the financial sector, which needs to capture and analyze the set of regulations covering their business domain, and support the management of (non-) compliance. In order to handle a set of regulations, we aim to adapt our Hohfeldian meta-model and Legal GRL framework to support the method presented in a related measurement framework [16]. This method uses system-based and performance-based descriptions for the set of regulations to improve the analysis of convergence and divergence between regulations and business processes. These descriptions could also be mapped to compliance measures used for the management of (non-) compliance. In addition, we want use Legal GRL to explore methods for identifying inconsistencies and incompleteness within legal text.

6. ACKNOWLEDGMENTS

This work was partially funded by AFR - PDR grant #5810263, and by previous NSERC scholarships.

7. REFERENCES

[1] T. D. Breaux and A. I. Antón. Analyzing goal semantics for rights, permissions, and obligations. In *RE*, pages 177–188. IEEE Computer Society, 2005.

[2] A. Cleven and R. Winter. Regulatory compliance in information systems research - literature analysis and research agenda. In *Enterprise, Business-Process and Information Systems Modeling*, volume 29 of *LNBIP*, pages 174–186. Springer, 2009.

[3] CNIL. Loi Informatique et Libertés. `http://bit.ly/1bqyJdf`, 2009. [Online; accessed January 2014].

[4] Federal Trade Commissions. Google Will Pay $22.5 Million to Settle FTC Charges it Misrepresented Privacy Assurances to Users of Apple's Safari Internet Browser. `http://1.usa.gov/19VRGcD`, 2012. [Online; accessed January 2014].

[5] S. Ghanavati. Legal-URN framework for legal compliance of business processes. `http://hdl.handle.net/10393/24028`, 2013. Ph.D. thesis, University of Ottawa, Canada.

[6] S. Ghanavati, D. Amyot, and L. Peyton. Towards a framework for tracking legal compliance in healthcare. In *19th Int. Conf. on Advanced Information Systems Engineering (CAiSE'07)*, volume 4495 of *LNCS*, pages 218–232. Springer, 2007.

[7] S. Ghanavati, D. Amyot, and L. Peyton. A systematic review of goal-oriented requirements management frameworks for business process compliance. In *Fourth Int. Workshop on Requirements Engineering and Law (RELAW'11)*, pages 25–34, 2011.

[8] G. Governatori, A. Rotolo, and G. Sartor. Temporalised normative positions in defeasible logic. In *Proc. of the 10th Int. Conf. on Artificial Intelligence and Law*, ICAIL '05, pages 25–34, USA, 2005. ACM.

[9] Government of UK. The Privacy and Electronic Communications (EC Directive) Regulations 2003. `http://www.legislation.gov.uk/uksi/2003/2426/contents/made`, 2003. [Online; accessed January 2014].

[10] International Telecommunication Union. Recommendation Z.151 (10/12), User Requirements Notation (URN) - Language definition. `http://www.itu.int/rec/T-REC-Z.151/en`, 2012.

[11] F. Ishikawa, R. Inoue, and S. Honiden. Modeling, analyzing and weaving legal interpretations in goal-oriented requirements engineering. In *Proc. of Second Int. Workshop on Requirements Engineering and Law*, RELAW '09, pages 39–44, USA, 2009.

[12] C. Miley. France data authorities fine Google for non-compliance with privacy policy. `http://bit.ly/1jSKXBi`, 2014. [Online; accessed January 2014].

[13] G. Mussbacher and D. Amyot. Goal and scenario modeling, analysis, and transformation with jUCMNav. In *ICSE Companion*, pages 431–432. IEEE, 2009.

[14] A. Pourshahid, D. Amyot, L. Peyton, S. Ghanavati, P. Chen, M. Weiss, and A. Forster. Business process management with the user requirements notation. *Electronic Commerce Research*, 9(4):269–316, 2009.

[15] A. Rifaut and E. Dubois. Using goal-oriented requirements engineering for improving the quality of ISO/IEC 15504 based compliance assessment frameworks. In *16th Int. Conf. on Requirements Engineering (RE'08)*, pages 33–42, Spain, 2008.

[16] A. Rifaut and S. Ghanavati. Measurement-oriented comparison of multiple regulations with grl. In *5th Int. Workshop on Requirements Engineering and Law*, pages 7–16. IEEE, 2012.

[17] A. Shamsaei, D. Amyot, and A. Pourshahid. A systematic review of compliance measurement based on goals and indicators. In *CAiSE Workshops*, volume 83 of *LNBIP*, pages 228–237. Springer, 2011.

[18] A. Siena, J. Mylopoulos, A. Perini, and A. Susi. Designing law-compliant software requirements. *Conceptual Modeling-ER*, pages 472–486, 2009.

[19] A. Siena, A. Perini, A. Susi, and J. Mylopoulos. A Meta-Model for Modelling Law-Compliant Requirements. In *2nd Int. Workshop on Requirements Engineering and Law*, pages 45–51, USA, 2009.