Toward Benchmarks to Assess Advancement in Legal Requirements Modeling

Ivan Jureta[1], Travis Breaux[2], Alberto Siena[3], David Gordon[2]
[1] University of Namur, Namur, Belgium
ivan.jureta@unamur.be
[2] Carnegie Mellon University,Pittsburgh, Pennsylvania, USA
{breaux,dggordon}@cmu.edu
[3] University of Trento, Trento, Italy
a.siena@unitn.it

***Abstract*—As software engineers create and evolve information systems to support business practices, these engineers need to address constraints imposed by laws, regulations and policies that govern those business practices. Requirements modeling can be used to extract important legal constraints from laws, and decide how, and evaluate if an information system design complies to applicable laws. To advance research on evaluating requirements modeling formalisms for the representation of legal information, we propose several benchmarks that we believe represent important challenges in modeling laws and requirements governing information systems, and evaluating the compliance of these requirements with laws. While incomplete, the proposed set of benchmarks covers a range of challenges in modeling laws and requirements that we observed in privacy and security law: from the possibility to trace model fragments to law fragments, to the ability to distinguish modalities in law, and to model relations between requirements and law fragments, needed when evaluating compliance. Benchmarks can be used as a checklist when designing and discussing requirements formalisms that support legal requirements modeling. Each benchmark is motivated by related work, a brief legal excerpt, and our experience in modeling regulations.**

***Index Terms*—legal requirements, requirements modeling, benchmarks, evaluation**

I. INTRODUCTION

Requirements engineering (RE) research has increasingly recognized the importance of laws and regulations and their impact on system design. Early efforts in RE focused on law include the study of privacy policies [1] and a repository of privacy requirements aimed at developing legally compliant systems [2]. Since, new methods have been introduced to make requirements extraction more systematic [3], to help engineers trace business processes to regulations using contribution links [4], and to enable reasoning over legal norms [5], among others. While other disciplines have long been interested in modeling laws and regulations [6], the unique focus of requirements engineering aims to help business analysts and engineers capture legal knowledge for the purpose of assessing whether their system designs may comply with government laws, regulations and policies. Therefore, as RE matures in this respect, the research challenge transitions from wholly exploratory research aimed at discovering novel methods and notations, to scientific evaluation focused on how well a particular method or notation performs against some standard or benchmark. In this paper, we present candidate benchmarks that result from our analysis in prior work and our experiences in modeling legal requirements. Throughout the paper, we cite prior evidence, some of which is empirically based, to demonstrate our observations that motivate the desire for each benchmark. Several examples that are used to demonstrate the benchmarks are taken from the Health Insurance Portability and Accountability Act (HIPAA), which is a U.S. law corresponding to several regulations that govern electronic health information. Finally, we believe these benchmarks can be used to as a reference point to assess the ability of a proposed modeling language to address important aspects of developing legally compliant systems.

Requirements modeling and languages to support this effort have been adapted to address new problems in specific domains, such as security [7]. These models serve several purposes, such as to surface ambiguity and inconsistency and to create a concrete representation that can be used by designers as an early system specification. In a system engineering project, RE aims to find design specifications of the system-to-be, which we simply call "the system"; the specification must satisfy all requirements and not violate any assumptions about the environment, in which the system will run [8]. When the environment is regulated by laws and policies, and stakeholders believe that the system behavior is affected by these laws and policies, then a responsible party must determine if and how the laws and policies relate to the system requirements and environmental assumptions, and the system design. Legal RE research is further concerned with developing new theories, methods, and technology to guide and support engineers in addressing these issues.

To help requirements engineers assess whether laws and policies affect their system requirements, environmental assumptions and so on, we identify three critical questions that we believe legal RE researchers seek to answer with new theories, methods and tools: (i) how to extract requirements and other software artifacts from laws; (ii) how to model extracted information to analyze system

requirements, environmental assumptions, and designs; and (iii) what does it mean to say that a system design specification complies with applicable laws, and how to evaluate if this is the case? In this paper, we primarily focus on questions (i) and (ii) as they relate to requirements modeling.

In the remainder of this paper, we begin with a review of related work on benchmarking in Section II, before introducing our proposed benchmarks in Section III and summarizing with future work in Section IV.

II. Background on RE Benchmarking

Hagge and Kreutzkamp conducted an early requirements benchmark study, wherein a benchmark scenario was developed to evaluate how well two systems performed with respect to satisfying user requirements [9]. The evaluation was based on 4-point scale to measure a "requirements conformity" level; these levels were mapped to percentages with weighted averages. Since, others have sought to compare various requirements notations, such as UML Activity Diagrams and Business Process Modeling Notation (BPMN) [10] and multiple feature diagram notations [11, 12].

Wasson warns that, due to the complex socio-technical nature of requirements and associated processes, "the objects of measurement are likely to be compound variables, corresponding to emergent properties" [13]. She identifies several key challenges to which we must attend: benchmarks must be properly instrumented to conform with scientifically acceptable standards for data collection; choreographing human subjects must address issues of participant motivation, expertise and competing demands; and underpinning measurement is the challenge of transferring domain-specific knowledge to individuals who are not experts [13].

Wasson further argues that metrics are typically identified in small studies, wherein local factors such as the accessibility of study observables and subjectivity of the analysts can influence the early success of a newly discovered metric [14]. She notes that standardization requires understanding subtle variations in how we operationalize the measurable constructs; for example, she cites that requirements completeness has been defined along two different dimensions: the *sum of known requirements* or the *sum of needed requirements*. In legal RE, we can relate this distinction to various ways that legal compliance has been defined: as maintaining a defensible position in a court of law [15], or as an attestation to an auditor [16].

III. Benchmarks for Legal RE Modeling

In this section, we present proposed benchmarks to evaluate extracting and modeling legal information for systems engineering. Each benchmark identifies a specific challenge that requirements modeling researchers need to consider when assessing the scope and effectiveness of their approach. This includes formal and semi-formal languages, combined with inference rules for computing conclusions (i.e., implicit information) from given information.

A. Extensible Domain Ontology

Laws contain terms-of-art or terminology with domain-specific meanings, which depend on how the law has been conceived and on the domain being regulated. Ideas described in law can be specified as concepts and relations, which together define a (formal) ontology. Defining an ontology involves making decisions about (i) how broad the ontology should be, that is, what its coverage is, (ii) choosing how detailed it should be, i.e., its depth, and (iii) how it relates to ontologies of other laws, or to higher-level ontologies that include concepts and relations specialized in different laws. An ontology of legal concepts should be as accurate as possible in capturing the underlying legal concepts while being extensible across domains. This distinction has been divided into the *upper ontology*, which is reusable, and the *lower ontology*, which is grounded in the domain [17]. To address this challenge, we introduce the following benchmark:

Benchmark SE (Separation and Extensibility): A legal RE formalism should separately distinguish the concepts shared across different laws, from the concepts that are unique to a specific law.

Example: In the Health Insurance Portability and Accountability Act (HIPAA), a covered entity or business associate must, in accordance with § 164.306: (...) (d)(1) (...) Implement policies and procedures that govern the receipt and removal of hardware and electronic media that contain electronic protected health information into and out of a facility, and the movement of these items within the facility.

The above legal fragment refers to several entities that have legal definitions, such as a covered entity and business associate, and electronic protected health information; these are specific to this law. However, we might categories these entities into general classes that recur across multiple laws, such as Actors and Information (or Objects). The former are in the domain (lower ontology) and the latter are reusable (upper ontology).

Consider how Benchmark SE can influence the design of a legal RE formalism. To represent classes such as "business associate" and their instances in the example, the formalism requires a mechanism that resembles predicates in first- and higher-order logic. In propositional logic, one could write the proposition *ba* to be true, if and only if, the actor (e.g., a central stakeholder) is a business associate. This approach may work if there is only one stakeholder, but it fails when we need to reason about interdependencies among multiple stakeholders or entities. In this case, we would use a predicate logic by writing $ba(x) \wedge ce(y)$ to be true, if and only if, a stakeholder x is a business associate and their

counterparty *y* is a covered entity. In this respect, a predicate logic is necessarily more expressive.

B. Axioms of Policy and Law

Laws, regulations and policies express provisions by means of modal verbs such as "must", "may" and so forth. The legal RE formalism needs to be capable of distinguishing modes conveyed by the various modal verbs. This results in two issues: (i) choosing the minimal set of modal verbs to cover in the formalism (e.g., can "may" be formulated using "must" and negation, or are these clearly distinct modes?), and (ii) how to define (e.g., by giving axioms) each modal verb in the formalism, so as to ensure that it plays the same or similar role in representations made with the formalism, and in the legal text in which it is used.

For example, deontic logic uses two operators, one for obligation and another for permission; a third operator, for prohibition, is defined as a negation of a permission [18]. Alternatively, Hohfeld's taxonomy of legal concepts describes a set of eight operators based on the analysis of how juries reasoned about court decisions in the early 20th century [19]. These concepts have since been formalized into a representation language [20]. Benchmarking a requirements modeling notation depends upon a precise set of axioms as follows:

Benchmark MC (Minimality and Conservativeness): With respect to modalities, the formalism for legal RE modeling should be (A) minimal in the number of modal verbs in the laws it intends to model, and (B) conservative in the inferences it allows from its representations of law.

In part A of benchmark MC, the minimality of the set of modalities, has itself two effects on the design of a legal RE formalism. Firstly, there is so-called internal minimality, which is that only primitive modalities should be included in the definition of the formalism. A modality is *primitive*, if and only if, it cannot be defined using other modalities; a modality is *derived*, if and only if, it can be defined using existing modalities, concepts, and relations in the formalism. For example, if a formalism has negation, and two modalities, obligation and permission, then prohibition can be defined as the negation of a permission. In that case, the formalism would fail to achieve MC-A if it was defined so as to suggest that prohibition is a primitive modality. Secondly, there is external minimality, which is that the formalism should not include modalities for which there is no need in the modeling of law. If the formalism does include such modalities, then they should not be used if they result in the violation of MC-B, which we now discuss.

Part B of benchmark MC concerns the content of the set of all conclusions that can be computed from models made using the formalism. MC-B is violated, if and only if, it is both possible to have a representation of a law made with that formalism, and to deduce from that representation more information than one can conclude from the reading of the law. In other words, MC-B is violated, if the act of modeling a law results in the expansion of the information available from that law (e.g., the modeler or inferences yield new information that cannot be concluded from the law). There is an important nuance in MC-B: if there is a set of formulas which all come from law (i.e., represent statements from law), then the computed inferences from that set alone should be conservative. But if one computes inferences from both that set *and* a set of formulas that represent statements a lawyer gave when interpreting that law, then these inferences can include new information, rather than be conservative with regards to the law.

MC-B not only influences the variety of modalities to include in a formalism, but also how these modalities are used when modeling, and what inference rules are allowed on the model of law. A trivial example is if the formalism has inference rules that satisfy *ex falso quodlibet* principle, that anything can be concluded from an inconsistent set of formulas. Suppose also that the formalism does *not* have a priority relation that would say which, among conflicting obligations, applies when that conflict is detected. If conflict is equated with inconsistency, then a model of law, which is inconsistent, and in which there is no way to state priority among inconsistent obligations will also produce any conclusion, and thereby the formalism will clearly violate MC-B. Benchmark MC-B fits Belnap's notion of the conservativeness of a definition [26], and here, it is both the definition of the formalism and the models of law made with the formalism that should be conservative.

C. Entity Classification and Requirements Coverage

Laws apply to various actors, objects and events (e.g., transactions), which may include the actor responsible for achieving compliance, their clients or third party service providers, and government actors responsible for enforcement. Objects may include regulated information, computer processes and devices. Actors and objects can be named directly, or defined indirectly by the actor or object's actions, possessed attributes, or their relationships to other actors and objects that are responsible for satisfying legal conditions. For example, in the HIPAA excerpt in section III.B, a "group health plan" is given the responsibility for ensuring that its documentation provides that another stakeholder, a plan sponsor, appropriately safeguards electronic protected health information when that information is created, received, maintained, or transmitted. Conditions, may be explicitly described, such as "[when the information is] created, received, maintained, or transmitted..." or inferred from rights granted to other stakeholders. Therefore, we propose the following benchmark:

Benchmark EC (Entity Classification): The formalism for legal RE modeling should enable the modeler to (A) identify different entity classes or categories present in the

regulation as well as relationships between classes; (B) determine the requirements that a class is responsible for satisfying (forward mapping); and (C) conversely, determine which class or classes are responsible for satisfying a particular requirement (backward mapping).

Benchmark EC-A demonstrates that a modeling formalism can represent the various stakeholder roles a regulation contains in addition to any relevant relationships between roles. Consider the following excerpt from the HIPAA, which supplements that from Section III.B:

> "SEC 160.103. DEFINITIONS.
> Covered entity means: (1) A health plan. (2) A health care clearinghouse. (3) A health care provider who transmits any health information in electronic form in connection with a transaction covered by this subchapter.
> ...
> Group health plan... means an employee welfare benefit plan (as defined in section 3(1) of the Employee Retirement Income and Security Act of 1974 (ERISA), 29 U.S.C. 1002(1))...
>
> (1) Health plan includes the following, singly or in combination:
> (i) A group health plan, as defined in this section.
> (ii) A health insurance issuer, as defined in this section."

Though some actor roles may have commonly accepted meanings, many are given explicit definitions, such as covered entity being defined as a health plan, health care clearinghouse, or health care provider. Definitions for actor roles can be hierarchical [21], in that a single definition may refer to a number of sub-roles simultaneously, which themselves may refer to a number of sub-roles. Figure 1 visually depicts the above excerpt: solid-line arrows point from sub-roles to super-roles or broader classes; double-arrows depict equivalent classes; and dotted-line arrows point from one role that is excluded from another role. In this example, the definition for covered entity includes health plans, which includes group health plans, etc. In addition, definitions may explicitly exclude certain actor roles, abdicating certain groups from requirements that they would otherwise be expected to perform. This may be done wholly within a particular law or by referencing definitions from other laws, such as the definition for "employee welfare benefit plan" drawn from Employee Retirement Income and Security Act of 1974, which excludes pension plans that fund retirement from welfare plans that fund health care. Similar to actors, object hierarchies have been observed to comprise software and hardware features used to classify covered entities [17].

Benchmark EC-B refers to the capability of the model to map entity categories to the requirements that those categories are responsible for demonstrating. These requirements may be unconditional, or they may be linked to pre-conditions that must first be satisfied before a requirement is expected of the entity. For example, consider a group health plan who must ensure that plan documents for plan sponsors will safeguard electronic information, when that information *is created, received, maintained, or transmitted*. This statement also includes a requirement for plan sponsors, which is that they are responsible for reasonably and appropriately safeguarding the information.

Let *a* be the actor role "group health plan", *p* be a condition "information [is] created, received, maintained, or transmitted", *q* the condition "ensure the documentation provides...", and *r* the condition "electronic protected health information... is disclosed pursuant to §164.504(f)(1)(ii) or (iii)." To demonstrate benchmark EC-2, the modeling formalism must enable determining which conditions *q* a role *a* must satisfy, and if there are additional conditions (e.g., *p*) that must be met prior to linking the responsibility *q* to the role. Demonstrating this benchmark also requires satisfying conditions under which the actor does not have the responsibility, e.g., when condition *r* is true.

Benchmark EC-C is symmetric with EC-B as it demonstrates the ability to present the entity categories that are linked to satisfying a given condition. Using the same representations for *a*, *p*, *q*, and *r* from the previous paragraph, the model must be able to determine, from a condition q, which actor has the responsibility for satisfying the condition (e.g., *a*). Thus, EC-B and EC-C provide a bi-directional mapping to exploring coverage.

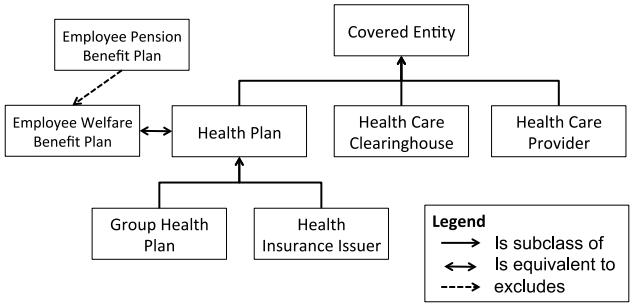

Figure 1. Entity classification that depicts actor roles in three types of relationships: sub-roles, equivalent roles, and excluded roles

D. Traceability among Legal and Software Artifacts

The lifecycle stage of software changes how we perceive legal compliance*: design-time compliance* concerns decisions that a designer makes about how to interpret a law in the context of their system design (e.g., where does data flow, how is data protected, etc.); *run-time compliance* concerns whether a system behaves correctly, e.g., by using self monitoring and self-control; and *legal compliance* is typically enforced by judges and, to avoid fines or sanctions, organizations regularly need evidence that demonstrates that correct design-time decisions were made and that the system behaved correctly or took corrective action. This need for evidence, in turn, requires traceability.

Default representations of law are natural language texts that use specific legal terminology and follow rules intended to facilitate reference to law fragments. Such rules suggest, for example, that a law has sections, and sections have

paragraphs. To reference a law fragment, it is enough to give its section, and when applicable, paragraph identifiers, already of which are provided in the text of the law.

When modeling law with requirements, the modeler herself, or after consulting with law experts, produces propositions that convey relevant legal information from the paragraphs and sections in the law, and she carries these propositions into the model. Whether these propositions convey exactly what the law states or an interpreted variant thereof (see Benchmark MC), she must preserve the ability to trace fragments of the model back to the originating law text. This leads to the following benchmark:

Benchmark RT (Reference Tracing): The formalism for legal RE modeling enables the modeler to preserve the relationship between the model fragments and legal text as follows: (A) precisely mapping atomic model fragments to the smallest, originating legal text fragment; (B) preserving to hierarchical paragraph structure of the legal text.

Benchmark RT has three parts. First, RT-A demonstrates the ability to find the minimal set of keywords, phrases or sentences in the text that each proposition in a model refers to; the strings are minimal if they contain all the information that is needed to produce the proposition(s) and no more. Historically, this mapping has been maintained at three levels of textual abstraction, from coarse to fine-grained: *document-level*, which maps a proposition to a legal document; *paragraph-level*, which maps a proposition to an indexed paragraph in a legal text; and *phrase-level*, which maps a preposition to a specific phrase in the legal text. Benchmark RT-A requires that all three levels should be demonstrated.

Suppose that the model includes the representation of the following proposition: "If a person desires to conduct a transaction, then the insurance plan should not delay that transaction." The question that RT-A raises is "Does a proposition x in the model originate from law? If yes, which law text fragment does it originate from?" A simple way to enable modeling formalism to answer this question is to have, on every proposition in the model, the identifier of the section, paragraph, or otherwise, in the law that gave this proposition. The proposition in the example is from HIPAA, and it would need to be annotated with "SEC. 1175 (a) (C)" which it originates from:

> "SEC. 1175. (a) CONDUCT OF TRANSACTIONS BY PLANS.--"(1) IN GENERAL.--If a person desires to conduct a transaction referred to in section 1173(a)(1) with a health plan as a standard transaction-- "(A) the health plan may not refuse to conduct such transaction as a standard transaction; "(B) the insurance plan may not delay such transaction, or otherwise adversely affect, or attempt to adversely affect, the person or the transaction on the ground that the transaction is a standard transaction; and "(C) the information transmitted and received in connection with the transaction shall be in the form of standard data elements of health information.

When designing a modeling formalism, two important questions are: what are the primitives, and how can they be combined? This is the issue of compositionality. For example, in classical propositional logic, primitives, usually denoted by lowercase Latin letters, are considered as the smallest elements, i.e., the primitives that refer to atomic propositions. In first-order classical logic, some of the primitives are terms, and it is instead grounded predicates that refer to atomic propositions. In legal text, paragraphs, sections, and any other mechanism for structuring law induce part-of or part-whole relations over fragments of legal text, and more generally, suggest rules for compositionality of legal text.

The presence part-of relations between law fragments leads to RT-B. When carrying over propositions from law to a legal RE model, it seems appropriate to maintain the structure imposed by that law. Consider an abstract example: let p and q refer to two propositions that originate in a law, and let id(p) and id(q) be the annotations of paragraph and section indices from the originating law. Let p be part of q in law, because id(p) identifies a paragraph in the section that is identified by id(q). In the above excerpt, if id(q) mapped to "(1)", then id(p) would map to "1175". The modeling formalism will not demonstrate RT-B, if the modeler can say that p is part of q, *and* cannot conclude that that q is not part of p.

E. Prioritization / Pre-emption

Laws and regulations may include exceptions, which create alternative conditions that may apply to one kind of covered entity and not another kind [21]. These exceptions may be based on their classification (see Benchmark EC), or they may be based on specific events that have or will occur. In addition to exceptions as conditions, exceptions can appear between two legal statements: if an entity is covered by two obligations, the exception prescribes which obligation must be discharged from which it follows that the other obligation would not apply to the entity.

Benchmark EX (Exceptions): If two or more law fragments place constraints on the same actions, and cannot all be satisfied together, the ability to identify which subset of these actions to satisfy.

Consider the following excerpt from the HIPAA Privacy Rule; section (b)(1) describes an exception, which is contained in the second fragment for (f)(1):

> **§ 164.314 Organizational requirements.**
> (b)(1) Standard: Requirements for group health plans. Except when the only electronic protected health information disclosed to a plan sponsor is disclosed pursuant to §164.504(f)(1)(ii) or (iii), or as authorized under §164.508, a group health plan must ensure that its plan documents provide that the plan sponsor will reasonably and appropriately safeguard electronic protected health information created, received, maintained, or transmitted to or by the plan sponsor on behalf of the group health plan.

> **(f)(1) Standard: Requirements for group health plans.**
> (ii) The group health plan, or a health insurance issuer or HMO with respect to the group health plan, may disclose summary health information to the plan sponsor, if the plan sponsor requests the summary health information for the purpose of: (A) Obtaining premium bids from health plans for providing health insurance coverage under the group health plan; or (B) Modifying, amending, or terminating the group health plan.

In (b)(1), a group health plan is generally required to require the plan sponsor to safeguard health information. Under (f)(1), the group help plan may disclose summary health information to the plan sponsor for two specific purposes. Because (f)(1) is an exception to (b)(1), we assume that summary health information shared under these purposes is not subjected to the same level of safeguarding as protection health information, in general. To demonstrate benchmark EX, the formalism must answer this question in the negative: given that summary healthcare information was received for obtaining premium bids, must the plan sponsor reasonably and appropriately safeguard the information?

Unlike the above example, there may be situations wherein conflicts arise between norms or requirements and no course of action will satisfy the conditions in the law. In this situation, a technique similar to goal satisficing [22] may be necessary to maximize compliance and legal expert advice will absolutely be necessary to weigh how to proceed. In some situations, regulatory enforcement may be an appropriate metric that can be used to prioritize which norms or requirements to comply with.

F. Identifying the Paths to Compliance

Complying with a law does not necessarily equate to satisfying all conditions stated in the law. While a law may contain an abstract rule to satisfy, such as to report revenue, the same law may provide different ways to report revenue, each applying in different conditions; for example, rules are different for individuals and companies, for nationals and foreigners, for holders of intellectual property rights, etc.

The existence of alternative sets of conditions, such that satisfying every condition in a single set gives reason to believe that the designer is taking steps toward legal compliance, leads to two questions: (i) how to find which set among alternatives is relevant to the system, given its requirements and assumed environment conditions, and (ii) deciding which of the alternatives to satisfy. This leads to the following benchmark:

Benchmark CA (Compliance Alternatives): The formalism for legal RE modeling should enable the modeler to: (A) represent alternative sets of conditions that a law expresses; (B) identify those alternative sets which are applicable to the system; and (C) compare applicable alternatives, so as to help choose the most appropriate alternative.

Benchmark CA-A requires that the modeling formalism be able to represent that some sets of propositions should not be satisfied together. This is a relatively simple requirement on a formalism, and will be satisfied if there is a notion of disjunction and/or exclusive disjunction. Exclusive disjunction is a consequence of Benchmark EX (certain legal exceptions), whereas various forms of disjunction can arise from entity categories in Benchmark EC (multiple actor roles).

Part B of benchmark CA reflects the idea that it is the requirements, environmental assumptions, and the design of the system that determine which laws need to be complied with. When a law offers alternative paths to satisfy it, then the same idea applies: the requirements, environmental assumptions, and system design will provide evidence to satisfy some paths, over other alternatives. Benchmark CA-B is predicated on part A, and leads to understanding the various types of evidence required to satisfy a condition; we discuss various types of evidence, including how evidence is used in reasoning, later in Benchmark DR.

In addition, benchmark CA-B requires that a modeling formalism demonstrate the distinction between applicability and satisfaction of law. A condition that originates from law applies, if it is believed that the system needs to satisfy the condition. For example, if a system will handle electronic protected health information on individuals, then HIPAA may apply. Whether that system satisfies conditions originating from HIPAA is a separate question that needs to be answered, sometimes by adjusting the system design. Consider the following law fragment:

> "(2) SAFEGUARDS.--Each person described in section 1172(a) who maintains or transmits health information shall maintain reasonable and appropriate administrative, technical, and physical safeguards--
> (A) to ensure the integrity and confidentiality of the information;
> (B) to protect against any reasonably anticipated—
> (i) threats or hazards to the security or integrity of the information; and
> (ii) unauthorized uses or disclosures of the information; and
> (C) otherwise to ensure compliance with this part by the officers and employees of such person. "

Let *p* refer to the proposition “if a person transmits health information, then she must ensure the integrity and confidentiality of health information.” Benchmark CA-B requires that it be possible, using the modeling formalism, to determine if the system design is expected to perform actions that result in states in which a person’s health information is transmitted, and if yes, the formalism should indicate that the design must satisfy *p*. In other words, the modeling formalism should have means to determine which law or law fragments should be satisfied, i.e., which law fragments are applicable.

The notion of applicability introduced with CA-B suggests that there is interdependency between the decisions of which system requirements to satisfy and which laws to comply to. Different system requirements could trigger the applicability of different laws. It follows that the desirability of a system design no longer depends on the requirements alone, but can be influenced by the which laws are

applicable and thus systems may be redesigned to avoid applicability of specific laws.

G. Dialectical Reasoning

The formalism for legal RE modeling may include a relation for saying that, if some requirements are satisfied, then some propositions from law will be complied with. Another useful relation would be the one for saying the opposite: that when some requirements are satisfied, some propositions from law will be violated, and so not complied with. Let R+ denote the former relation, and R- the latter.

Suppose that there is a set of requirements and law propositions, and that *r* is some requirement, and *o* some law proposition in that set. The modeler may then assert that there is R+ from *r* to *o*, this would be interpreted as, if *r* is satisfied, then *o* will be satisfied as well, or the modeler may assert that there is R- from *r* to *o*, which means that, if *r* is satisfied, then *o* will not be satisfied.

The issue in asserting that there is R+ or R- between *r* and *o*, is that the existence of that relation is based on assumptions made by the modeler. The cautious approach is to treat these assumptions as defeasible: as propositions that are assumed true, unless and until there is evidence to the contrary. When new evidence does become available, it may itself be defeasible. As a result, the legal RE formalism would need to be capable of dialectic reasoning, which is that the truth of a proposition depends on the truth of the propositions attacking it, that the truth of these attackers depends on the truth of other propositions attacking these attackers, and so on. In other words, the formalism would need to support reasoning analogous to that in formal argumentation [23]. This gives the following benchmark:

Benchmark DR (Dialectical Reasoning): The formalism for legal RE modeling should enable the modeler to: (A) represent evidence that supports or attacks modeling choices made in applying the formalism; and (B) compute the evidence which is acceptable, given all available evidence.

Benchmark DR-A concerns the representation of evidence in favor and against a modeling choice. For example, the modeler may encounter evidence against her choice to consider a law fragment as applicable to some requirements; a legal expert, may disagree, and provide evidence against modeler's evidence. According to dialectical reasoning, and roughly speaking, the new evidence counter-argues the already available evidence, and the modeler would need to conclude that the law fragment applies.

It follows that the notion of evidence, or more generally, of argument is needed in the formalism, as well as at least two relations: one to indicate that an argument supports a modeling choice or another argument, and another to indicate the opposite, that an argument counters a modeling choice, or another argument. As Breaux and Anton categorize conditions based on the type of evidence needed to satisfy them [21]. Some conditions are non-ephemeral, such as a stakeholder role (e.g., an organization that is a hospital is unlikely to change this role too frequently), whereas other conditions are ephemeral, transactional or otherwise non-persistent (e.g., treating a patient). In addition, the measurement taken to assess conditional satisfaction may be a variable, such as the number of days before which a privacy notice must be sent, or a psychological construct, such as legal or medical knowledge, or another form of subjective belief. To demonstrate this benchmark, a formalism should interface and account for the various types of evidence needed to satisfy corresponding types of conditions.

Part B of Benchmark DR, or Benchmark DR-B requires that the inference rules in the formalism support dialectical reasoning. Here, it is necessary to determine the status (truth or acceptability, for example) of all arguments supporting and countering a proposition in order to evaluate the status of that proposition.

H. Applicability & Verification Benchmark

The ultimate purpose for having a formal representation of legal provisions is to use formal analysis techniques to check the models for desired properties. So, modeling formalisms can be compared with respect to the kind of questions they support answering. This is particularly true of design-time compliance, which is the legal conditions that system requirements have to meet; thus, a modeling formalism has to be evaluated with respect to it's capability to answer design-time compliance questions.

Considering a law model $L = \{n_1, \ldots n_l\}$, representation of a legal text, where n_k is a normative proposition extracted from the law, and a system design, expressed in terms of its requirements R, a generic definition of compliance may be defined as the condition that, for each n_k in L, if n_k applies to at least one requirement r_a in R, then a requirement r_s must also exist in R, such that r_s satisfies n_k. The formalism for legal requirements should support the analyst in verifying that: (i) which norms apply to the system; (ii) whether the system requirements satisfy the applicable legal norms; and (iii) what is missing in the system specification to satisfy the applicable norms. We can draw some important benchmarks from this.

Benchmark V1 (Applicability): Given a system design, the formalism must exhibit the ability to verify if one or more fragments of the law apply to the system design.

> Example. § 164.502(a) A covered entity may not use or disclose protected health information (...) A covered entity is permitted to use protected health information (...) For treatment, payment, or health care operations. (b) When using PHI (Protected Health Information) a CE must make reasonable efforts to limit PHI to the minimum necessary.

The obligation to limit PHI to the minimum necessary is not a requirement for the system, unless the system has been designed for treatment, payment or health care operations. When this condition applies, the CE can use patients' PHI, but a further requirement has to be satisfied, since the minimality of PHI has to be ensured. Benchmark G establishes the need for a mechanism to fill truth values of law propositions. Benchmark V1 states that, given a truth value to some law proposition, e.g. p1 = "use protected health information", the formalism must be able to identify a relation to some other law proposition, e.g. p2 = "must make reasonable efforts to …". The nature of the relation must be such that it informs the modeler that the given value of the first proposition(s) implies the need to ensure a certain value of the second proposition(s).

Benchmark V2 (Verification): Given a set of norms that apply to the system design, the formalism must exhibit the ability to verify if that design satisfies the norms, or if compliance is considered as being a matter of degree, the ability to evaluate the level to which the design satisfies the norms.

For each normative proposition in L, and given the truth values established as described in benchmark G, the formalism must be able to mark the normative propositions as satisfied, if there is a requirement satisfying it; violated, if there is a requirement explicitly in contrast with it; or missing, if no requirement exists, which satisfies the proposition.

IV. Discussion and Summary

In this paper, we propose several benchmarks to summarize accomplishments to date in the legal RE community and to motivate future research directions in developing legal requirements modeling languages. These benchmarks were demonstrated in the context of the laws and regulations that govern information privacy and security. While we believe that some of these benchmarks will apply to other domains, such as accessibility or safety, we do not assume the list is complete or that all of these benchmarks will apply to all domains. The benchmarks are intended to represent milestones that researchers and developers can use to demonstrate advancement in the field of legal RE. While this list is by no means complete, we strived to conservatively summarize advancements from prior work, and which we consider as particularly relevant for future legal RE research. This includes empirical case studies or simple examples that describe the problem and a corresponding solution. We envision that the proposed benchmarks may be extended or even subsumed as legal RE continues to develop existing and new languages with validation across multiple, legal jurisdictions.

At present, there are some requirements modeling techniques, languages and notations that at least partially demonstrate a few of these benchmarks. The legal requirements specification language (LRSL) [24] maintains traceability to the legal document structure and formalizes cross-references to demonstrate Benchmark RT (reference traceability). Nomos [5] can model alternative ways to comply with applicable norms, in line with Benchmark CA. Nomos can further distinguish applicable from non-applicable norms, as in Benchmark V1. Ghanavati et al. describe a method that uses i* diagrams to construct arguments to support compliance with regulations [4]. This technique may be later augmented with an inference system to demonstrate benchmark DR-A, in which contribution links map supporting evidence back to encoded legal requirements. Alternatively, Maxwell et al.'s work [25] to analyze cross-references describes results that may be partially used to demonstrate Benchmark DR-B, in particular, by illustrating specific argumentation strategies to remove requirements conflicts with laws.

Underpinning several, if not all, of the benchmarks are various notions of validity. Benchmark MC, for example assumes that the minimal set of concepts is fundamentally representative (or valid constructs) and that their logical meanings are consistent with the meanings acquired by legal experts from the same legal text. Even among legal experts, there may exist statistical variance in the conclusions that they draw from a single text. Similarly, Benchmark DR may depend on contextual information to determine when an argument is satisfactory; as this information changes, a standing argument may become invalid. Thus, the challenge Benchmark DR may seem achievable under one set of assumptions about how much context is required, and later invalid after new, missing context is presented. Therefore, in addition to formalization of legal theories, researchers should consider their assumptions that underpin these formalisms to understand when the theory may become inconsistent with the viewpoints of legal experts. We envision a new line of human subject studies in RE to measure how legal experts (or their protégés) draw conclusions from legal texts. The results of these studies would be used to reinforce or challenge the above benchmarks or to establish new ground rules for reasoning about requirements specifications and law.

In future work, we aim to conduct a more comprehensive study to expand upon our examples and establish data sets that can be used to demonstrate each of our proposed benchmarks. We hope to conduct this work in collaboration with other researchers and practitioners in legal RE.

References

[1] A.I. Antón, J.B. Earp, C. Potts, T.A. Alspaugh. "The Role of Policy and Stakeholder Privacy Values in Requirements Engineering," *IEEE Int'l Symp. Req'ts Eng.*, pp. 138-145, 2001.

[2] A. Toval, A. Olmos, M. Piattini. "Legal Requirements Reuse: A Critical Success Factor for Requirements Quality and Personal Data Protection," *IEEE Joint Int'l Conf. Req'ts Eng.,* pp. 95-103, 2002.

[3] T. D. Breaux, M. W. Vail, A. I. Antón. "Towards Regulatory Compliance: Extracting Rights and Obligations to Align Requirements with Regulations." *IEEE Int'l Req'ts Engr. Conf.,* pp. 46-55, 2006.

[4] S. Ghanavati, D. Amyot, L. Peyton. "Compliance Analysis Based on a Goal-oriented Requirement Language Evaluation Methodology," *IEEE Int'l Req'ts Engr. Conf*. pp. 133-142, 2009.

[5] A. Siena, I. Jureta, S. Ingolfo, A. Susi, A. Perini, J. Mylopoulos. "Capturing variability of law with Nomós 2," 31st Int'l Conf. Conc. Mod., pp. 383-396, 2012.

[6] P. N. Otto, A. I. Antón. "Addressing Legal Requirements in Requirements Engineering." *IEEE Int'l Req'ts. Engr. Conf.* pp. 5-14, 2007.

[7] H. Mouratidis, P. Giorgini. "Secure Tropos: A Security-Oriented Extension of the Tropos Methodology." *Int'l J. Soft. Engr. & Know. Engr.,* 17(02): 285, 2007.

[8] P. Zave, M. Jackson. "Four dark corners of requirements engineering," *ACM Trans. on Soft. Engr. & Meth.* 6.1 (1997): 1-30.

[9] L. Hagge, J. Kreutzhamp, "A Benchmarking Method for Information Systems," IEEE Req'ts Engr. Conf., pp. 245-253, 2003.

[10] A. Gross, J. Jukiewicz, J. Doerr, J. Nawrocki. "Investigating the Usefulness of Notations in the Context of Requirements Engineering," IEEE 2nd Int'l W'shp Empricial Req'ts Engr., pp. 9-16, 2012.

[11] O. Djebbi, C. Salinesi. "Criteria for Comparing Requirements Variability Modeling Notations for Product Lines," *4th Int'l W'shp on Comparative Eval. in Req'ts Eng.,* Minneapolis, Minnesota, 2006.

[12] J.-C. Trigaux, P. Heymans, P.-Y. Schobbens, A. Classen "Comparative semantics of Feature Diagrams: FFD vs. vDFD," *4th Int'l W'shp on Comparative Eval. in Req'ts Eng.,* Minneapolis, Minnesota, 2006.

[13] K. Wasson, "Comparative Evaluation: Implications from the Multidisciplinary Nature of Requirements," *1st Int'l W'shp on Comparative Eval. in Req'ts Eng.,* Monterey Bay, California, 2003.

[14] K. Wasson, "Comparative Evaluation: Implications from the Multidisciplinary Nature of Requirements," *2nd Int'l W'shp on Comparative Eval. in Req'ts Eng.,* Kyoto, Japan, 2004.

[15] T.D. Breaux, A.I. Antón, C.-M. Karat, J. Karat. "Enforceability vs. Accountability in Electronic Policies." IEEE 7th Int'l W'shp on Pol. for Dist. Sys. & Nets. pp. 227-230, Jun. 2006

[16] A. Siena, J. Mylopoulos, A. Perini, A. Susi. From Laws to Requirements. 1st Int'l W'shp on Req'ts Engr. & Law (Relaw'08), held at the RE 2008. September, 2008. Barcelona, Spain.

[17] T.D. Breaux, Legal Requirements Acquisition for the Specification of Legally Compliant Information Systems. Ph.D. Thesis, North Carolina State University, Apr. 2009.

[18] Georg Henrik von Wright. Deontic logic. Mind, 60:1–15, 1951.

[19] W.N. Hohfeld. Some fundamental legal conceptions as applied in judicial reasoning. *The Yale Law Journal*, 23(1):16–59, 1913.

[20] L.E. Allen, C.S. Saxon. "Better language, better thought, better communication: the a-hohfeld language for legal analysis." *5th Int'l Conf. AI & Law*, pp. 219–228, 1995.

[21] T. D. Breaux, A. I. Antón. "Analyzing Regulatory Rules for Privacy and Security Requirements." *IEEE Trans. Soft. Engr.,* 34(1):5-20, Jan/Feb 2008

[22] J. Mylopoulos, L. Chung, and E. Yu, "From Object-Oriented to Goal-Oriented Requirements Analysis," Comm. ACM, vol. 42, no. 1, pp. 31-37, 1999.

[23] P. M. Dung. "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games." *Artificial intelligence* 77(2): 321-357, 1995.

[24] T. D. Breaux, D. G. Gordon, "Regulatory Requirements Traceability and Analysis Using Semi-Formal Specifications," *19th Wrk'ing Conf. Req'ts Engr.: Fnd. for Soft. Qual.*, Essen, Germany, Apr. 2013.

[25] J. Maxwell, A.I. Anton, P. Swire, "Discoverying conflicting software requirements by analyzing legal cross-references," In Submission: *IEEE Int'l Conf. Req'ts. Engr.*, 2011.

[26] N. Belnap. "On rigorous definitions." *Philosophical studies* 72(2): 115-146, 1993.