L-SQUARE: Preliminary Extension of the SQUARE Methodology to Address Legal Compliance

Aaron Alva, Lisa Young
Software Engineering Institute, Carnegie Mellon University
Pittsburgh, PA, USA
aalva@uw.edu; lry@cert.org

***Abstract*—Laws and regulations must be considered in the requirements engineering process to help ensure legal compliance when developing software or engineering systems. To incorporate legal compliance considerations into the requirements engineering process, we introduce a preliminary extension of the SQUARE methodology, called L-SQUARE. In this paper, we develop L-SQUARE by discussing legal compliance concerns at each of the traditional nine steps in SQUARE. Then, we link existing research in requirements engineering and the law to each step, emphasizing where compliance concerns can be addressed. This preliminary extension of SQUARE sets existing research into an established methodology for requirements engineering, creating a framework for situating current research in legal compliance, and identifying gaps for future work.**

Index Terms—Legal compliance, laws, regulations, methodology

I. Introduction

Legal compliance is an important factor to understand when developing software or systems engineering requirements, especially in highly regulated fields such as critical infrastructure sectors including financial services, healthcare, energy, and others. Legal compliance requires an organization to derive engineering requirements from laws or regulations and demonstrate traceability should the organization be subject to a compliance audit.

In this paper, we provide a preliminary extension of the SQUARE methodology—called L-SQUARE—to address requirements engineering for legal compliance. L-SQUARE is designed to operate separately from SQUARE, except in specified steps where we indicate that use jointly with SQUARE may be warranted. We walk through each step of the original SQUARE methodology when formulating L-SQUARE to connect a non-exhaustive set of current research in requirements engineering and the law, and emphasize legal compliance concerns that may arise.

The development of L-SQUARE provides the field with a high-level, flexible framework that organizations can tailor for legal compliance requirements engineering.

II. Extending Established Methodologies for Requirements Engineering

A. SQUARE Methodology Overview

The Security Quality Requirements Engineering (SQUARE) methodology, developed at Carnegie Mellon University in 2005, provides a step-by-step process for "eliciting and prioritizing" security requirements so that such requirements are baked-into software and systems early in the life cycle. [1] The SQUARE methodology helps ensure security is not an afterthought when designing or engineering a new system.

SQUARE involves nine steps: definitions agreement; identify assets and security goals; develop artifacts to support security requirements definition; perform risk assessment; select elicitation techniques; elicit security requirements; categorize and label level and type of security requirement; prioritize requirements; and inspect requirements.

B. SQUARE Methodology Extensions

The SQUARE methodology is flexible and has been extended, making it useful as a framework underlying L-SQUARE. SQUARE has been extended twice in order to address two specific issues—acquisition and privacy.

SQUARE for Acquisition, or A-SQUARE, was developed for organizations that acquire software but do not have control over the entire systems development life cycle. Those organizations may find the A-SQUARE method useful to develop or enhance requirements for the RFP process or identify security gaps in COTS products. [2] SQUARE for Privacy, or P-SQUARE, modifies SQUARE to consider privacy requirements engineering in new software development. P-SQUARE uses the same nine-step process as the original SQUARE method with elaborations or adaptations in each step to consider privacy laws and regulations instead of security goals. [3]

III. SQUARE for Legal Compliance

In this paper, we extend SQUARE to address legal compliance requirements. We call our extension L-SQUARE, or SQUARE for Legal Compliance. In part A, we address why a separate L-SQUARE is needed in some cases, focusing particularly on highly regulated fields.

A. Why L-SQUARE is Needed

Organizations in regulated fields should address legal compliance when developing software, engineering systems or acquiring software. While all systems may not require a legal review if there is no risk of legal non-compliance, when legal

978-1-4799-6339-3/14/$31.00 © 2014 IEEE

compliance concerns arise, L-SQUARE can be used as a methodology to develop requirements that are traceable to a specific legal compliance requirement. This methodology's main purpose is to ensure due diligence that the system or software being developed complies with applicable laws or regulations. Organizations should strive for legal compliance, though conducting an extensive requirements engineering process for legal compliance may be most practical where the risk of legal non-compliance is greater than the effort needed to conduct this process.

The L-SQUARE method is targeted at organizations that are in regulatory-heavy industries that wish to ensure a robust consideration of legal requirements over and above what is required for security or privacy. The critical infrastructure sectors, in particular finance, healthcare, and energy, are examples where organizations must comply with a variety of laws and regulations.

There are several examples of laws and regulations, and other legally binding practices that should be considered in the requirements engineering process, depending on what is relevant to the organization and the sector in which it operates. [4] The U.S. healthcare sector must consider the Healthcare Portability and Accountability Act (HIPAA).[1] HIPAA is promulgated through regulations, particularly the Privacy Rule and the Security Rule. The U.S. energy sector must consider a series of regulations, depending on the facet of the business in which they operate. The energy sector's legal compliance requirements are controlled by an independent industry group—the North American Electric Reliability Corporation (NERC). NERC creates and distributes the standards, while the U.S. government makes such standards legally binding through an independent agency—the Federal Energy Regulatory Commission (FERC).[2] The financial sector must comply with various financial regulations including the Gramm-Leach-Bliley Act (GLBA).[3] This paper is not intended to be an authoritative source for legal compliance regulations or standards but merely to point out that there are myriad legal standards that may need to be considered for compliance.

The risk of legal non-compliance can lead to possible financial or reputational impacts. In some instances, particularly in the financial sector, organizational leaders can be held directly accountable for non-compliance. For example, the GLBA can result in civil or criminal actions against organizational leaders for non-compliance.[4] Fines can also be severe, ranging from $100K per occurrence in the Gramm-Leach-Bliley Act (GLBA), and up to $250K in HIPAA.[5]

IV. Developing each step in L-SQUARE

This section will proceed step-by-step through the current SQUARE methodology, and describe preliminary extensions for L-SQUARE using the existing order of steps. For each step we will briefly describe the step in SQUARE, discuss legal compliance challenges that should be addressed, and incorporate current requirements engineering and the law research that addresses the particular step. Then, we will propose a potential reordering of steps in L-SQUARE to develop a more effective legal compliance requirements engineering methodology.

A. Agree on Definitions (Step 1)

In Step 1 stakeholders agree on definitions that will become consistent throughout the requirements engineering process. This step serves the purpose of reducing ambiguity so that stakeholders can clearly communicate particular legal compliance requirements. Stakeholders in this process would include representatives from legal, privacy, compliance, security, and software engineers.

Several legal compliance challenges arise in this step. For laws and regulations, defining legal terms in the context of requirements engineering may be difficult. Legal terms may be domain-specific and used differently across multiple laws or regulations. [5] Legal compliance requirements may arise from case law that is difficult to interpret. [6] Additional complexity is added when an organization must comply with multiple related laws across different jurisdictions. [7]

Researchers have supported the need for consistent definitions in the legal requirements engineering context. Otto and Antón have called for a data dictionary for all domain-specific terms, and a contextual picture of such terms within a particular regulatory context. [6] Researchers have also developed languages and methodologies to explicitly define legal requirements. This work will be discussed in Step 5.

For legal compliance, starting with a definitions agreement step may be premature. There is a prerequisite question to answer prior to this step in the context of compliance: What laws or regulations must the organization comply with?

B. Identify Assets & Legal Compliance Goals (Step 2)

In Step 2 stakeholders articulate and formally agree on the goals of the project. This step serves the purpose of creating a common understanding of the legal goals that must be met as the output of this step. Stakeholders in this step would include representatives from the business or mission, legal, privacy, compliance, security, and software engineers.

The broadest goal identified at this step related to legal compliance is likely to be simple: be compliant. But the complexity of multiple applicable laws requires an inquiry into which laws or regulations apply. Identifying what laws and regulations an organization must be compliant with is difficult, especially for organizations that operate across multiple geographic regions, states, or countries. [6, 8]

In this step, stakeholders should preliminarily identify laws and regulations that are known to apply to the organization. This would provide a broad scope by which the requirements engineering process could proceed. For example, the stakeholders could generally identify that HIPAA applies because the system being developed is for healthcare uses. For this step, stakeholders need not specify particularly what section within HIPAA may apply, only that it generally does apply.

[1] Pub. L. No. 104-191, 110 Stat. 1936 (1996).

[2] 18 C.F.R. §§ 1-399 (2012).

[3] Pub. L. No. 106-102, 113 Stat. 1338 (1999).

[4] 15 U.S.C. § 6823 (2012).

[5] 42 U.S.C. § 1320d-5 (2012).

If the organization operates in multiple legal jurisdictions then this step would require that all laws be identified for which compliance is required. For example, the United States has 47 data breach notification laws, as required separately by states. This step would require the organization to identify each of the 47 laws in preparation for eliciting requirements from those laws, as well as combining those laws together into fewer, legally defensible requirements. Gordon and Breaux have developed a methodology for comparing requirements from multiple-jurisdictions, which can be employed in step 6, but would require input of applicable laws that are gathered during this step. [7]

If possible, it would be helpful to involve legal stakeholders at this step so that the general regulatory context can be identified prior to the elicitation steps. While legal stakeholders may not be readily available in smaller organizations, checking for applicable laws and policies in advance of requirements elicitation can help avoid requirements engineering efforts incorrectly directed at inapplicable laws. Researchers have begun to evaluate tradeoffs in efficiency and effectiveness of technical professionals interpreting legal texts compared to legal professionals. [9, 10]

From the initial set of identified laws that are applicable to the organization, the requirements analysts should link the legal goals to security & privacy goals expressed by the organization. In doing so, L-SQUARE links to traditional SQUARE as well as P-SQUARE methodologies. It serves to align necessary compliance goals with desired organizational goals. [1]

C. Develop Artifacts (Step 3)

In Step 3 stakeholders provide input to the RE team to develop use case scenarios including misuse case development and relevant case law that would provide a robust consideration of legal compliance requirements. In the case of legal compliance, the primary misuse case we have identified is non-compliance. In the misuse case of non-compliance the organization identifies and documents defensible alternatives to compliance with the actual legal text, such as compensating controls. [11] This decision would arise particularly where an organization has competing regulations with which they must comply.

In traditional SQUARE, this step may be conducted simultaneously with Step 2 in the form of a work session or facilitated workshop as the output of Steps 2 and 3 are inputs to Step 4. Stakeholders in this step would include representatives from the business or mission, audit, legal, privacy, compliance, security, and software engineers. Including the audit viewpoint in this step may provide insight into what auditors or regulators look for when conducting a software audit for legal compliance. [11]

For legal compliance, this SQUARE step may not be best positioned in this order. Moving this step after prioritization (Step 8) would functionally allow for identification and decision making on the compliance alternatives an organization choses. After Step 8, the organization should have sufficient information in order to provide traceability to its decisions.

D. Perform Risk Assessment (Step 4)

Step 4 expands the SQUARE risk assessment to include legal risks of non-compliance. This step serves the purpose of connecting the security requirements goals to the legal risks that have the greatest potential impact on the business or mission. Stakeholders in this process would include representatives from the business or mission, audit, risk managers, legal, privacy, compliance, security groups, and software engineers. In some cases, legal experts knowledgeable in a particular area (e.g. patent law, trademark law, etc.) may be necessary in order to aid in difficult legal issues.

As a preliminary analysis in creating L-SQUARE, this step may need to be performed first. If the risk of legal non-compliance is low or the organization can show the security or privacy goals meet legal compliance goals sufficiently, then the remainder of the L-SQUARE process may be unnecessary.

However, as regulations evolve it may be pertinent for an organization to research whether pending regulations or potential case law would suggest new regulations.

E. Select Elicitation Technique (Step 5)

In Step 5 the various elicitation techniques that can be used are selected. There may be many viable techniques to ensure a robust and complete coverage of legal compliance requirements. It is unlikely that a single technique will work for all projects or circumstances. [1]

This step requires that one or more elicitation techniques be selected that provide a robust consideration of legal compliance requirements that are specific to the business or mission needs depending on the sector in which the organization operates. Stakeholders in this step would include representatives from the business or mission, legal, privacy, compliance, security, and software engineers.

Researchers have developed languages and methods for modeling laws, which can be employed to develop artifacts necessary to demonstrate compliance. The following examples of current elicitation techniques organizations can use for eliciting requirements based on legal texts.

Breaux developed a Frame-Based Requirements Analysis Method to acquire, specify, and analyze legal requirements derived from regulatory texts. [12] Breaux and Antón extended Breaux's method to develop the Legal Requirements Specification Language (LRSL), a systematic methodology to derive rights and obligations from regulatory texts. [13] The semantic models derived from the texts can help clarify ambiguities in order to create requirements that are more defensible. [14] Breaux et. al. have extended the LRSL, applying it to additional uses, including water marking, [15] requirements coverage modeling, [11] and more. [15]

Siena, Perini, et. al. have created a meta-model for modeling law-compliant requirements [16], which is based on the Nòmos framework. Siena's Nòmos framework provides a language for modeling requirements and legal impact on them; guidelines to analyze for compliance; and a systematic process for generating legally compliant requirements. [17]

El Kharbili has extended the CoReL modeling language to visually represent compliance requirements. [18]

Each of these elicitation techniques are examples of some of the existing techniques discussed in current requirements engineering literature. These techniques can be employed by an organization to develop legal requirements necessary for compliance.

SQUARE does not dictate which elicitation technique should be selected. The selection of a particular technique may depend of a number of factors including the type of legal text (law, regulation, standard, contract, or more); the way the legal text is written (prescriptive, goals-based, standards-based); whether the legal texts are omnibus (such as HIPAA) or are multi-jurisdiction (such as the 47 different state data breach notification laws); and more.

F. Elicit Legal Compliance Requirements (Step 6)

In Step 6 the requirements analysts executes the elicitation techniques selected to ensure all of the relevant legal compliance requirements are considered. This step serves the purpose of documenting a complete set of legal requirements that will be able to be verified once the project has been implemented. The requirements gathered or generated in this step will be reviewed and prioritized in subsequent steps. Stakeholders in this step would include representatives from the business or mission, legal, privacy, compliance, security, and software engineers.

In this step, the requirements analysts would execute the elicitation technique identified in step 5. The output of this step is a list of all relevant requirements the organization may need to implement in order to be legally compliant.

G. Categorize Requirements (Step 7)

In Step 7 all of the gathered and elicited requirements are categorized in such a way they can be affinity grouped for prioritization. This step serves the purpose of allowing the RE team to provide to the stakeholders a set of essential requirements for systems and software that are distinct from any architectural constraints that may have become evident in the process.

Stakeholders in this step would include representatives from the business or mission, audit, risk managers, legal, privacy, compliance, security, and software engineers.

There are minimally five groupings used in SQUARE to categorize requirements: 1) essential; 2) non-essential; 3) software requirements; 4) system requirements; or 5) architectural or infrastructure requirements. [1] SQUARE is typically for software or system requirements, not architectural requirements.

For multi-jurisdiction requirements, this step would also involve the process of joining, disjoining, or providing a minimum requirement that incorporates the multiple laws. From there, the organization could identify high-water mark or low-water mark requirements in preparation for the next step, prioritization. [7]

There are a variety of ways to solve conflicts between legal requirements that, if unresolved, may result in non-compliance. Maxwell, Antón, et. al. discussed five ways for resolving non-compliance between conflicting HIPAA legal texts. Their techniques were to follow the most restrictive law; store data separately; obligations supersede legal privileges; and consult legal domain experts. [8] Gordon and Breaux's water marking methods also provide a way to resolve conflicts. [19] Elicitation techniques have also incorporated proposed ways to resolve conflicts. The SQUARE methodology lists a number of requirements solicitation techniques, many of which discuss ways to resolve conflicts. [1]

H. Prioritize Legal Compliance Requirements (Step 8)

In Step 8 stakeholders agree on the priority of the requirements to be implemented. This step serves the purpose of decisively selecting which requirements are to be included in the project and which requirements are excluded from the project. Stakeholders in this process would include representatives from legal, security, privacy, compliance, security, and software engineers.

The need to be legally compliant should be delineated from the need to have a system properly designed to be secure and privacy-preserving. We caution that during the prioritization process, step 8, the legal compliance requirements be considered alongside security and privacy requirements by taking into account the identified risks, costs, and benefits. Researchers have noted that legal domain experts and engineers will likely differ when prioritizing requirements. [19, 20, 21]

There are at least four possible scenarios for the relationship between a legal compliance requirement and a security requirement. These scenarios could be developed by using L-SQUARE jointly with SQUARE in this particular step in order to elicit security requirements that seek to functionally achieve the organization's stated security goals. These scenarios would require prioritization between the legal compliance and security/privacy requirement.

First, the legal compliance requirement and the security requirement can match. This can occur when the legal compliance requirement is equally sufficient to the security requirement for the organization's desired security goals.

Second, the legal compliance requirement can fall below the security requirement. In this case, the legal compliance requirement is the minimal baseline security necessary, but is not sufficient to meet the organization's security goal. In this case, prioritization should favor the security requirement.

Third, the legal compliance requirement can be greater than the identified security requirement. In this case, the legal compliance requirement should be provided greater priority.

Fourth, legal compliance requirements may not match security requirements or be necessary for achieving security goals. In these cases, the legal compliance requirements are misfits from the system or software being designed, but nevertheless are required by law for compliance.

An example of the fourth possibility is Washington State's PCI law.[6] In Washington State, for a processor, business, or vendor to be protected from liability after a breach, they must be certified as PCI compliant via a safe harbor provision in the code. [22] But the law also applies if there is a breach of data

[6] R.C.W. 19.255.020 (2014) http://apps.leg.wa.gov/Rcw/default.aspx?cite=19.255.020

from an “identification device” defined as “an item that uses radio frequency identification technology or facial recognition technology." [22] So, an organization developing software or systems that involve these identification devices must certify as PCI compliant even though they process no credit card payments. In this case, PCI certification is the legal compliance requirement, but is unnecessary to meet security goals, as they relate to “identification devices” being developed. In these cases, organizations may be bound by law to become compliant in the proscribed way.

These four possible scenarios may be useful in characterizing legal prioritization. In a different approach, Massey, Otto, and Antón have developed a legal prioritization approach that includes two steps: 1) determining legal implications for each requirement and 2) calculating a prioritization score. [23] In their approach, they note that requirements engineers may decide to include other criteria beyond legal implications alone when calculating overall priority. [23]

There is an open research question as to how to prioritize similar legal compliance requirements compared to security and privacy requirements.

I. Requirements Inspection (Step 9)

In Step 9 the stakeholders inspect the prioritized requirements to ensure consensus on the legal requirements for compliance. This step serves the purpose of ensuring the validity and feasibility of each legal requirement in relation to security and compliance goals and provides a documented set of requirements to the development team. Stakeholders in this step would include representatives from the business or mission, audit, risk managers, legal, privacy, compliance, security, and software engineers.

The main goal of this step is to ensure that each requirement elicited from laws or regulations maintains traceability back to the goal of legal compliance. By including this step, SQUARE reinforces the need for traceability between legal texts and elicited legal requirements. For traceability to be achieved, ambiguities in legal texts, when translated to legal requirements, should be checked and made explicit when possible. [15]

Researchers have identified traceability as a challenge [24, 25], and developed several approaches to ensuring traceability. For example, Massey, Smith, et. al. have evaluated several U.S. laws to determine whether elicited requirements meet or exceed legal compliance obligations. [9] Cleland-Huang, Czauderna, et. al. have developed machine learning for traceability between regulatory texts and requirements, as applied to HIPAA. [26]

J. SQUARE step reordering/linking for L-SQUARE

In order for L-SQUARE more effectively elicit legal compliance requirements, it may be useful to reorder the existing nine steps, or to explicitly link such steps during the requirements engineering process.

We recommend that before embarking on the entire requirements engineering process for legal compliance, it would be wise to undertake a thorough assessment of the risk of non-compliance with legal requirements. If the risk assessment validates that privacy and security requirements are sufficient to cover the baseline legal obligations then there is no need to go further with requirements elicitation. This would move the existing SQUARE step 4 to the first step in L-SQUARE.

We also recommend that agreeing on definitions (SQUARE step 1) is not a logical first step in a requirements engineering process that is directed toward legal compliance. At a minimum, the organization must first identify which laws or regulations may be applicable to a system or software development. This must occur before there is an agreement on the definition of particular legal terms. So, we recommend that current SQUARE step 1—agree on definitions—be moved to some point in the process after SQUARE step 2 (identification assets and security goals). More work must be done to identify the most effective location within the process for it.

V. Future Research for L-SQUARE

This is a preliminary extension of the SQUARE method. There is additional work to be done in order to solidify this extension; evaluate how this *ex ante* requirements engineering methodology can account for *ex post* changes in legal compliance requirements; and validate it using case studies.

A. Situating

More work must be done on where and how to merge compliance requirements with security or privacy goals identified by the organization and its stakeholders in traditional SQUARE step 2. In practice it is likely most effective to align compliance goals with particular security and privacy goals. Future work on L-SQUARE should also address how to identify additional requirements necessary for compliance, and place them within the context of security and privacy requirements.

Additionally, future work must be done to incorporate requirements engineering research not preliminarily addressed in this paper’s non-exhaustive survey. Additional work should also compare L-SQUARE with additional legal compliance frameworks such as Legal-URN [27] and Nòmos. [17]

B. Evolving

Requirements evolve—including in legal compliance. [28] We have alluded in the risk assessment section (SQUARE step 4) that an organization may want to conduct a risk assessment of the legislative and regulatory environment to determine whether regulations may be looming. The question not answered in this paper is whether and how the evolution of security and privacy requirements should be considered within L-SQUARE.

C. Validating

Validation can be done using a real-world case, and can be evaluated according to criteria consistent with the SQUARE analysis conducted in [29]. To evaluate SQUARE as applied to smart grid advanced metering infrastructure, ten criteria were used: flexibility, sampling, analyst permissibility, interpretiveness, data sufficiency, coherence, repeatability, completeness, usability, and credibility. [29]

VI. Conclusion

We have presented a preliminary extension of the SQUARE methodology called L-SQUARE. The L-SQUARE extension's goal is to help organizations understand where and how legal compliance requirements can be planned for in a traditional requirements engineering process. Future work on L-SQUARE will include solidifying the methodology specific to legal compliance; accounting for evolving legal compliance requirements; and validating our approach. SQUARE has provided requirements analysts with a flexible approach to conducting the requirements engineering process. In L-SQUARE, we have shown that SQUARE can be extended to provide the same flexible approach for legal compliance requirements, an approach that provides a broader view of current research in the field, and can be used to identify gaps for future research.

References

[1] N. Mead, E. Hough, T. Stehney II, "Security Quality Requirements Engineering," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, Technical Report CMU/SEI-2005-TR-009, 2005. http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=7657

[2] N. Mead, "Adapting the SQUARE Method for Security Requirements Engineering to Acquisition" Software Engineering Institute, Pittsburgh, PA, Tech. Paper Feb. 2010. Available: http://resources.sei.cmu.edu/asset_files/WhitePaper/2010_019_001_51613.pdf

[3] A. Bijwe. and N. Mead, "Adapting the SQUARE Process for Privacy Requirements Engineering," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, Technical Note CMU/SEI-2010-TN-022, 2010. http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=9357

[4] T. D. Breaux and T. A. Alspaugh, "Governance and accountability in the new data ecology," *Fourth International Workshop on Requirements Engineering and Law (RELAW)*, pp. 5–14, 2011.

[5] P. N. Otto and A. I. Antón, "Addressing Legal Requirements in Requirements Engineering," *15th IEEE Int'l Req'ts Engr. Conf. (RE'07)*, pp. 5–14, 2007.

[6] S. Bruninghaus and K.D. Ashley, "Predicting outcomes of case based legal arguments," *9th International Conference on Artificial Intelligence and Law,* New York, NY, USA, 2003, pp. 233–242.

[7] D. G. Gordon and T. D. Breaux, "Comparing requirements from multiple jurisdictions," *Fourth International Workshop on Requirements Engineering and Law (RELAW),* pp. 43–49, 2011.

[8] J. C. Maxwell, A. I. Antón, P. Swire, M. Riaz, and C. M. McCraw, "A legal cross-references taxonomy for reasoning about compliance requirements," *Requirements Eng*, vol. 17, no. 2, pp. 102, Apr. 2012. [9] A. K. Massey, B. Smith, P. N. Otto, and A. I. Anton, "Assessing the accuracy of legal implementation readiness decisions," 19th *IEEE Int'l Req'ts Engr. Conf. (RE'11)*, pp. 207–216, 2011.

[10] David G. Gordon, Travis D. Breaux. The Role of Legal Expertise in Interpretation of Legal Requirements and Definitions. Accepted To: *22nd IEEE International Requirements Engineering Conference (RE'14)*, Karlskrona, Sweden, Aug. 2014.

[11] I. Jureta, T. Breaux, A. Siena, and D. Gordon, "Toward benchmarks to assess advancement in legal requirements modeling," *Sixth International Workshop on Requirements Engineering and Law (RELAW),* pp. 25–33, 2013.

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

[13] T. D. Breaux and A. I. Anton, "Analyzing Regulatory Rules for Privacy and Security Requirements," *Software Engineering, IEEE Transactions on*, vol. 34, no. 1, pp. 5–20, 2008.

[14] T. D. Breaux, M. W. Vail, and A. I. Anton, "Towards Regulatory Compliance: Extracting Rights and Obligations to Align Requirements with Regulations," *14th IEEE Int'l Req'ts Engr. Conf. (RE'06)*, Minneapolis, Minnesota, pp. 49–58, Sep. 2006.

[15] T. D. Breaux and D. G. Gordon, "Preserving traceability and encoding meaning in legal requirements extraction," *Sixth International Workshop on Requirements Engineering and Law (RELAW),* pp. 57–60, 2013.

[16] A. Siena, A. Perini, A. Susi, and J. Mylopoulos, "A Meta-Model for Modeling Law-Compliant Requirements," *Second International Workshop on Requirements Engineering and Law (RELAW)*, pp. 45–51, 2009.

[17] A. Siena, *Engineering Law-Compliant Requirements: The Nomos Framework*. PhD Dissertation, University of Trento, Jan. 2010.

[18] M. El Kharbili, "Applying CoReL to an excerpt of HIPAA: a critical discussion," *Sixth International Workshop on Requirements Engineering and Law (RELAW),* pp. 61–64, 2013.

[19] D. G. Gordon and T. D. Breaux, "A cross-domain empirical study and legal evaluation of the requirements water marking method," *Requirements Eng*, vol. 18, no. 2, pp. 147–173, Apr. 2013.

[20] A. Bobkowska and M. Kowalska, "On efficient collaboration between lawyers and software engineers when transforming legal regulations to law-related requirements," presented at the 2nd International Conference on Information Technology (ICIT), pp. 105–109, 2010.

[21] A. Siena, J. Mylopoulos, A. Perini, and A. Susi, "From Laws to Requirements," *Requirements Engineering and Law, 2008. RELAW '08*, pp. 6–10, 2008.

[22] David Navetta, "FAQ on Washington State's PCI Law," *InfoLawGroup blog*, Mar. 2010, accessed May 29, 2014, http://www.infolawgroup.com/2010/03/articles/payment-card-breach-laws/faq-on-washington-states-pci-law/.

[23] A. K. Massey, P. N. Otto, and A. I. Anton, "Prioritizing Legal Requirements," *Second International Workshop on Requirements Engineering and Law (RELAW),* pp. 27–32, 2009.

[24] O. C. Z. Gotel and A. C. W. Finkelstein, "An analysis of the requirements traceability problem," *1st IEEE Int'l Req'ts Engr. Conf. (RE'94)*, pp. 94–101, 1994.

[25] N. Kiyavitskaya, A. Krausova, and N. Zannone, "Why Eliciting and Managing Legal Requirements Is Hard," *Requirements Engineering and Law, 2008. RELAW '08*, pp. 26–30, 2008.

[26] J. Cleland-Huang, A. Czauderna, M. Gibiec, and J. Emenecker, "A machine learning approach for tracing regulatory codes to product specific requirements," *32nd Annual ACM/IEEE International Conference on Software Engineering*, New York, New York, USA, 2010, vol. 1, pp. 155–164.

[27] S. Ghanavati. *Legal-URN Framework for Legal Compliance of Business Processes*. Ph.D. Thesis, University of Ottawa, 2013.

[28] J. C. Maxwell, A. I. Anton, and P. Swire, "Managing changing compliance requirements by predicting regulatory evolution," presented at the *20th IEEE Int'l Req'ts Engr. Conf.*, pp. 101–110, 2012.

[29] H. Suleiman and D. Svetinovic, "Evaluating the effectiveness of the security quality requirements engineering (SQUARE) method: a case study using smart grid advanced metering infrastructure," *Requirements Eng*, vol. 18, no. 3, p. 270, Apr. 2012.

.