Managing Changing Compliance Requirements by Predicting Regulatory Evolution

An Adaptability Framework

Jeremy C. Maxwell[1,2], Annie I. Anton[3], and Peter Swire[4]
[1]College of Engineering, North Carolina State University, Raleigh, NC, USA
[2]Allscripts Healthcare Solutions, Raleigh, NC, USA
[3]School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA, USA
[4] Moritz School of Law, Ohio State University, Columbus, OH, USA

jcmaxwe3@ncsu.edu, aianton@cc.gatech.edu, swire@osu.edu

***Abstract*—Over time, laws change to meet evolving social needs. Requirements engineers that develop software for regulated domains, such as healthcare or finance, must adapt their software as laws change to maintain legal compliance. In the United States, regulatory agencies will almost always release a proposed regulation, or rule, and accept comments from the public. The agency then considers these comments when drafting a final rule that will be binding on the regulated domain. Herein, we examine how these proposed rules evolve into final rules, and propose an Adaptability Framework. This framework can aid software engineers in predicting what areas of a proposed rule are most likely to evolve, allowing engineers to begin building towards the more stable sections of the rule. We develop the framework through a formative study using the Health Insurance Portability and Accountability (HIPAA) Security Rule and apply it in a summative study on the Health Information Technology: Initial Set of Standards, Implementation Specifications, and Certification Criteria for Electronic Health Record Technology.**

Keywords-Healthcare IT, Regulatory Compliance, Requirements Engineering, Requirements Evolution

I. INTRODUCTION

Requirements engineers need tools and techniques to adapt to regulatory evolution. A regulatory text can change as often as once a year [21], requiring potentially critical modifications to relevant software. Such changes may be legally mandatory, leading to expensive rework of legacy systems that were designed prior the new legal requirements being announced. Additionally, businesses have strong incentives to be first to market when these changes occur.

In the United States, regulations are issued by federal agencies that regulate certain domains. For example, the Department of Health and Human Services (HHS) regulates healthcare related industries. When a regulatory agency seeks to issue a new regulation, the agency will first issue a proposed rule or a notice of proposed rule making. Except in emergencies, the public will then be given the opportunity to comment on the proposed rule. The regulatory agency then issues a final rule that is binding on the regulated domain.

Because market forces so often compel software organizations to be the first to market, they must begin complying with regulations before the final rule is published. For example, the American Recovery and Reinvestment Act of 2009 [1] (ARRA) created the Meaningful Use (MU) program that makes $23 billion in incentives available for healthcare providers that adopt certified Electronic Health Record (EHR) technology and use it in a meaningful way. The incentives are paid out over three Stages that require providers to meet increasingly intensified clinical quality criteria. For example, one of the clinical quality criteria concerns patient engagement; for each stage of MU, providers must engage a greater portion of their patients and in more ways. As part of the criteria, EHR technology must be updated during each stage of MU to enable physicians to document, track, and submit the clinical quality criteria.

The proposed rule for MU Stage 1 was released on January 13, 2010, whereas the final rule was issued on July 28, 2010. Eligible providers and hospitals could begin applying for Stage One incentives on January 1, 2011. Engineers that waited until the final rule was released were left with less than six months to adapt their EHRs to meet the MU Stage 1 requirements, have their EHR certified, and installed at physician practices and hospitals—it is unlikely that these engineers met this tight deadline.

In this paper, we present our Adaptability Framework. The framework helps requirements engineers to identify: why regulations change (rationale); how regulations change (classifications); and which portions of a proposed rule that are most likely to change when the final rule is issued (heuristics). In addition, we propose an initial set of rationales, classifications, and heuristics for the healthcare domain. The framework allows engineers to focus primarily on analyzing and specifying compliance requirements from the more stable areas of the law, while the less stable areas of the law are being clarified during final rulemaking. We developed the Adaptability Framework through a multiple case study. In the multiple case study we employed the proposed and final versions of the Health Insurance Portability and Accountability Act (HIPAA) Security Rule, and the rule that specifies the requirements for certified EHR technology for MU Stage 1, called the Initial Set of Standards, Implementation Specifications, and Certification Criteria for Electronic Health Record Technology (hereafter referred to as the EHR Certification Rule). Using our

1 http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=111_cong_bills&docid=f:h1enr.pdf

Adaptability Framework, we accurately predicted nine areas of the proposed EHR Certification Rule that were likely to change when the final EHR Certification Rule was released.

The remainder of this paper is organized as follows: Section II reviews related work and provides a legal background primer; Section III outlines our multiple case study design; Section IV describes the Adaptability Framework developed during our formative study; Section V describes our summative study in which we applied the framework; Section VI discusses threats to validity; and Section VII discusses our multiple case study and future work.

II. BACKGROUND

In this section, we describe related work and provide a legal background.

A. Related Work

Prior work addressing the evolution of regulatory requirements falls into one of two categories: legal compliance and requirements evolution as we now discuss.

1) RE & Legal Compliance Literature: Otto and Antón observe that regulations may undergo frequent updates and amendments, sometimes as often as once a year [21]. They go on to state that addressing these updates and changes is a key element of regulatory compliance in requirements engineering [21].

Other work in legal compliance and requirements engineering has focused on: identifying conflicts introduced by cross-references from one legal text to another [17]; specifying compliance requirements from regulations using a frame-based method [3], a goal-based method [22], and a production rules-based method [15, 16]; modeling business process' compliance with the law [7]; determining the legal implementation readiness of existing requirements [14]; and ensuring compliance with organizational privacy policies [1, 23]. In our work, we identify which areas of the law are more stable, allowing engineers to focus on analyzing these sections using these existing techniques in the literature.

Ghanavati et al state that their work can assist requirements engineers handle regulatory evolution, but do not discuss details [7]. Islam et al present a framework that includes modeling evolving regulatory requirements [11]. However, their method requires that engineers model the legal text before analysis [11]—a luxury that engineers building software for rapidly changing environments may not have. In addition, they focus on eliciting requirements from the amended legal text [11], not on predicting regulatory evolution as we discuss herein.

2) Requirements Evolution: Managing changing requirements is difficult, the stakes are high, and engineers need better ways of handling change. Zowghi and Offen use a logic-based approach to reason about changing requirements [25], but requirements engineers that face short compliance deadlines usually do not have the luxury of creating formal logic models. Carter et al. develop an evolutionary prototype model that helps engineers address the challenges associated with requirements creep [5], but is targeted towards smaller development teams [5]. Jones recommends several techniques and processes for making requirements changes less impactful [12], for example, by creating prototypes or following rapid application development [12]. In our work, we do not examine the impact of regulatory changes on software systems, rather, we explore how and why regulations change. Antón and Potts examine how telephony features have evolved over time [2], but they use an after-the-fact analysis. In our work, we develop heuristics that engineers can use to predict future change.

Nurmuliani, Zowghi, and Williams use a card sorting technique to explore how software practitioners view and organize requirements changes [19]. Given a set of 52 requirements changes written on index cards, they asked practitioners to group the cards according to their own criteria [19]. Nurmuliani et al. observe that practitioners view and organize requirements change according to their role; for example, a project manager views requirements changes according to the impact upon the project schedule whereas developers are more likely to organize requirements changes according to effort [19].

Previous researchers have documented the ways that requirements changes occur. McGee and Greer developed a taxonomy that classifies the source of requirements change [18]. In their taxonomy, the *market change domain* includes requirements changes due to regulatory changes [18]. Our work goes beyond McGee and Greer's work by classifying why and how regulations change. In addition, we propose a set of heuristics that predict areas of a proposed regulatory rule that are likely to change. Harker et al. classify requirements as either stable or changing [10]. They further classify changing requirements into several categories [10] in which evolving legal requirements are mutable requirements—requirements that evolve due to environmental change. They also recommend several techniques to manage requirements change [10]. Our work enables engineers to follow the spirit of their recommendations to identify the minimal set of stable requirements, and build for those requirements first [10].

B. Legal Background

In the United States, the Administrative Procedure Act sets forth the steps in the creation of a federal regulation, also called a rule[2]. Regulations are binding legal requirements that are issued by a federal agency. Regulations implement a statute that has been passed by the Congress. The normal process is that the agency first issues a Notice of Proposed Rulemaking[3]. The public is then given a period to comment on the rule, with the comment period generally being at least 60 days[4]. The agency is required to review the public

[2] The Administrative Procedure Act, 5 U.S.C. §553 (2006).

[3] 5 U.S.C. §533(b) (2006).

[4] Executive Order 12866, Sept. 30, 1993: Regulatory Planning and Review, 58 Fed. Reg. 51,735 (Oct. 4, 1993).

comments and take them into account in drafting a Final Rule[5]. The Final Rule has binding effect, such as the HHS MU rules.

In cases of emergency, an agency can issue a Final Rule without prior public comment[6]. In such instances, the agency often solicits public comments after the Rule is issued.[7] Parties affected by a Rule in most instances have a right to appeal the Rule to federal court[8]. The court may uphold the Rule, or find that the Rule is substantively illegal (e.g., the Rule did not accurately implement the statute) or procedurally illegal (e.g., the Rule and accompanying explanation provided by the agency did not adequately respond to the public comments).

C. Terminology

In our work, we use the following terms:

- A *legal statement* is a sentence or sentence fragment in a regulation.
- A *compliance requirement* is a software or organizational requirement that enables an organization to comply with a government regulation.

III. Multiple Case Study Design

In our multiple case study, we performed a formative study followed by a summative study. In the formative study, we developed our Adaptability Framework. In our summative study, we applied the Adaptability Framework to predict how a legal text will evolve. We now describe our multiple case study design.

A. Research Questions

In our multiple case study, we sought to answer the following research questions:

RQ_1: Can we model the changes that are made to two proposed rules—the HIPAA Security Rule and the EHR Certification Rule—during final rulemaking?

RQ_2: Can requirements engineers predict changes that will be made to proposed rules?

B. Materials

The inputs for our study are the interim and final versions of two regulations. For our formative study, we employ the proposed and final versions of the HIPAA Security Rule. The Security Rule is 5,563 words long and describes security controls that certain organizations, called covered entities, have to have in place to protect electronic health information (PHI). Covered entities include physician practices, health plans, and healthcare clearinghouses. In our summative study, we employ the interim and final versions of the Initial Set of Standards, Implementation Specifications, and Certification Criteria for Electronic Health Record Technology (hereafter the EHR Certification Rule). The EHR Certification Rule is 4,736 words long and describes requirements that an EHR must satisfy in order to be certified under the Meaningful Use program—allowing eligible physicians and hospitals to qualify for government incentives when they use the EHR. Both Rules are available on the HHS website[9]. Both of these regulations are issued by the same agency—HHS—and regulate the same domain—healthcare IT.

C. Formative Study Design

In this section, we describe the design of our formative case study. Due to renumbering and reorganization changes, the topic of a particular section in a proposed rule may differ from the topic in the matching section number of a final rule. Thus, we first perform a topical mapping from the proposed HIPAA Security Rule to the final HIPAA Security Rule. For example, §142.308(a)(3) in the proposed HIPAA Security Rule describes business continuity and disaster recovery plans that a covered entity must maintain. However, in the final HIPAA Security Rule, these plans are discussed in §164.308(a)(7)(i). Where a topic that appeared in one section in the proposed rule is discussed in multiple sections in the final rule, we add a mapping for each section.

As we map each section, we examine the text of the proposed and final Security Rule, documenting changes to the regulation using a spreadsheet. For each identified change, we examine HHS's commentary that accompanies the final Security Rule to reason about the stated rationale for why a change was made. We use grounded theory analysis [8, 9] to classify changes and the rationale behind the change. In grounded theory analysis, theory is developed from the systematic study of a data set [8, 9]. The developed theory is "grounded" in the data, in that it is applicable only to the given data set [8, 9]. Future studies will allow us to make claims about the generalizability of our results. Grounded theory contrasts with the traditional scientific method, where hypotheses are formulated then tested through experiments. Researchers have previously used grounded theory analysis for requirements engineering research [6, 13] and when analyzing legal and policy requirements [1, 3, 4].

Upon completing our analysis of the Rules, we formulated the Adaptability Rationale and Adaptability Taxonomy. We then developed heuristics based on the rationale and taxonomy. The heuristics help requirements engineers predict that changes that will occur in a proposed regulation as it is updated in final rulemaking.

D. Summative Study Design

Our summative study had two phases. First, we used the heuristics developed during our formative study to predict the areas of the proposed EHR Certification Rule that were likely to be changed in final rulemaking. Second, to validate

[5] 5 U.S.C. §§603-604.

[6] 5 U.S.C. §608.

[7] 5 U.S.C. §608.

[8] 5 U.S.C. §553(e).

9 http://healthit.hhs.gov/portal/server.pt/community/healthit_hhs_gov__standards_ifr/1195 and http://www.hhs.gov/ocr/privacy/hipaa/administrative/securityrule/

our predictions, we analyzed the rules using the same grounded theory analysis on the EHR Certification Rule that we performed in our formative study described in Section III.C.

IV. The Adaptability Framework

The Adaptability Framework identifies three components of each regulatory change (see Figure 1): adaptability rationale that capture *why* regulations change; an adaptability taxonomy that capture *how* regulations change; and a set of adaptability heuristics that requirements engineers can use to predict *that* a regulation will change.

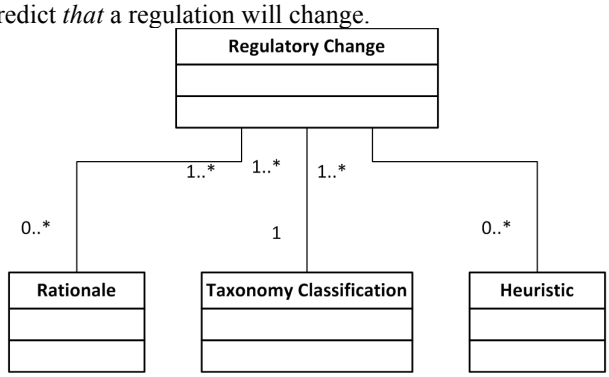

Figure 1. Adaptability Framework

A. Adaptability Rationale

An adaptability rationale is the stated reason why legal drafters make changes to a proposed rule before issuing the final rule. Rationales are stated in the commentary that accompanies a final ruling. The framework contains a total of ten rationales that we identified during our study. Table I lists each rationale as well as the number of times we identified the rationale in each case. It is important to note that a single change in a regulation may have multiple rationales.

TABLE I. Adaptability Rationale

Rationale	# in HIPAA Case Study	# in EHR Cert-ification Case Study
Ambiguity	14	14
Format & Organization	5	2
Technology-Specific Elements	4	2
Inappropriate for Domain	4	6
Potential Conflict	2	2
Resources Lacking to Implement	3	5
Concerns about Over-Regulation	3	4
Change in Another Regulation	1	2
Redundant	11	0
Unknown	17	5

1) Ambiguity: Regulations may be amended when public comments reveal that they are ambiguous or unclear. For example, the proposed HIPAA Security Rule requires covered entities to secure their workstation locations, and provide several examples such as "not placing a terminal used to access patient information in any area of a doctor's office where the screen contents can be viewed from the reception area" (§142.308(b)(5)). Commenters note, however, that the examples are presented in a way that makes them appear to be required. In addition, the proposed rule used the terms "workstation" and "terminal" interchangeably, and commenters said it is ambiguous whether covered entities have to secure other types of workstations such as laptops. In the final HIPAA Security Rule, this statement is generalized to require that covered entities "implement physical safeguards for all workstations that access electronic protected health information, to restrict access to authorized users" (§164.310(c)) The examples are removed from the final rule and the term "terminal" is removed.

2) Format & Organization: As we discussed in Section III.C, regulations are subject to format & organizational changes as they are updated. For example, the HIPAA Security Rule was renumbered from 45 CFR 142 to 45 CFR 160-164.

3) Technology-Specific Elements: When regulations mandate that certain technologies be adopted, industry innovation is stifled because businesses are disincentivized to improve the state of the art. In addition, government mandated technologies can actually lead to security vulnerabilities in critical systems such as EHRs if security flaws are identified in the mandated technologies. For example, the proposed HIPAA Security Rule mandates that covered entities adopt access controls from a list of technologies that include context-based access control, role-based access control, or user-based access control. However, this discourages innovation of other types of access controls. Thus, the final HIPAA Security Rule removes this list of access control technologies.

4) Innappropriate for Domain: Sometimes, a regulation will specify requirements that are inappropriate for the domain it regulates. For example, the proposed EHR Certification Rule contains requirements that would require EHRs to perform administrative functions such as submitting insurance claims. However, this functionality is typically performed by a practice management system, not by an EHR. Thus, HHS removed these requirements before issuing the final rule.

5) Potential Conflict: Regulations can contain requirements that conflict with each other [17]. Notice and comment rulemaking provides legal drafters with an opportunity to identify potential conflicts. For example, the EHR Certification Rule requires that EHRs record patient smoking status using a set of predefined values (i.e., current smoker, former smoker, and never smoked). Commenters determined that the predefined smoking status values in the proposed EHR Certification Rule were inconsistent with the

smoking status values used by the Centers for Disease Control (CDC). Thus, HHS adopts the same list of values used by the CDC in the final EHR Certification Rule.

6) Resources Lacking to Implement: A regulated industry may lack the resources to implement requirements in a proposed rule. For example, the proposed EHR Certification Rule contains a requirement that certified EHRs record disclosures of PHI made for treatment, payment, or healthcare operations. However, commenters noted that industry lacked the resources to implement the requirement in time for MU Stage 1. In response to these comments, HHS made the requirement optional for Stage 1.

7) Concerns about Over-Regulation: Regulatory agencies may receive comments indicating that certain requirements in a proposed rule are unnecessary and over-regulate the domain. For example, the proposed HIPAA Security Rule would require that covered entities maintain procedures & policies to follow when an employee is terminated. However, commenters point out that this may be unnecessary in certain settings, such as a small rural provider whose only employee is their spouse.

8) Change in Another Regulation: Regulatory agencies update rules to keep up-to-date with changes in other laws. For instance, as we described in Section I, the EHR Certification Rule places requirements on certified EHRs to allow providers to meet the MU clinical quality criteria. The clinical quality criteria are specified in a separate rule, the Medicare & Medicaid EHR Incentive Program Rule[10]. Before the final EHR Certification Rule was issued, the EHR Incentive Program Rule was updated to include an additional clinical quality criteria: providers must provide patient-specific education resources to patients. This led HHS to add a new requirement in the final EHR Certification Rule that certified EHRs have the capability to provide patient-specific education.

9) Redundant: A proposed rule sometimes contains redundant requirements. For example, the proposed HIPAA Security Rule contains the following two requirements: (1) covered entities must ensure that personnel have are authorized when viewing PHI (§142.308(a)(7)(iii)), and (2) covered entities must maintain personnel clearance procedures (§142.308(a)(7)(iv)). Commenters pointed out that these requirements seemed redundant, and they were combined in the final rule.

10) Unknown: Some changes that are made to regulations go unexplained in the final rule commentary; the rationale for these changes is unknown. For example, the proposed EHR Certification Rule requires that certified EHRs have the ability to record, retrieve, and transmit immunization information to immunization registries. In the commentary that accompanies the final EHR Certification Rule, HHS agrees with a commenter that recommended that "modify" be included in the list of operations the EHR can perform on immunization information, but neither the commenter nor HHS provide reasoning why this action should be included in the final rule. Thus, the rationale for this change is unknown.

10 http://edocket.access.gpo.gov/2010/pdf/2010-17207.pdf

B. Adaptability Taxonomy

The adaptability taxonomy classifications describe the changes that legal drafters make to proposed rules when they issue final rules. There are eight classifications in the adaptability taxonomy, displayed in Table II, along with the number of times we identified each classification within the HIPAA Security Rule case study and the EHR Certification case study. In the remainder of this section, we describe each classification in detail.

TABLE II. ADAPTABILITY TAXONOMY CLASSIFICATIONS

Taxonomy Classifications	# in HIPAA Security Rule Case Study	# in EHR Certification Case Study
Reorganization	4	1
Elaboration	11	5
Introduction of a term	2	1
Removal	21	21
Generalization	12	2
Addition	12	8
Requirement made optional	0	1
Introduced cross-reference	0	3

1) Reorganization: Regulatory texts may be reorganized when they are published as final rules; this reorganization often results in renumbered sections as well. For example, in the interim HIPAA Security Rule, the security requirements are placed in a single section entitled "Security Standard". In the final Security Rule, security requirements are reorganized among several sections based on theme, for example, "technical safeguards", "physical safeguards", and "administrative safeguards". These changes do not impact the meaning of the requirements. As we discussed in Section III.B, to maintain traceability, we map requirements in proposed rules to the matching requirements in final rules because they are sometimes reorganized and renumbered.

2) Elaboration: Legal statements may be updated to provide greater clarity and reduce ambiguity. For example, in the proposed HIPAA Security Rule, covered entities are required to "assess potential risks and vulnerabilities to the individual health data in its possession and develop, implement, and maintain appropriate security measure" (§142.308). In the final HIPAA Security Rule, HHS describes several comments that they received asking to clarify the term "risk assessment", along with several comments expressing confusion as to how the assessment should be performed. In response to these comments, HHS elaborated this requirement. The new requirement states at

§164.308(a)(1) and §164.306(b) that the risk assessment must be used to ensure the confidentiality, integrity, and availability of ePHI (electronic PHI), and that security measures should protect against reasonably anticipated threats based, among other things, on the covered entity's size, complexity, technical infrastructure, and threat probability.

3) Introduction of a Term: Sometimes, legal drafters introduce a term to provide a vocabulary for common concepts. For example, the term "covered entity" is not used in the proposed HIPAA Security Rule. Instead, each organization covered by the proposed Security Rule is defined in §142.302. Throughout the remainder of the proposed Rule, these organizations are referred to as an "entity designated in §142.302". In the final HIPAA Security Rule, the term 'covered entity' is introduced as "short hand" for this language.

4) Removal: Legal statements may be removed from a final rule for a variety of reasons. For example, the actions needed to comply with a particular compliance requirement may be too expensive, industry may lack the appropriate infrastructure to support the requirement, or a particular requirement may be out of scope for the domain in question. For example, as we discussed in Section IV.A.4, the proposed EHR Certification Rule contained the requirement that certified EHRs submit insurance claims. However, this functionality is typically performed by practice management systems, not EHRs. As such, HHS removed this requirement before publishing the final EHR Certification Rule.

5) Generalization: A generalization occurs when a legal statement's scope is broadened. For example, §142.308(a)(7)(i) of the proposed HIPAA Security Rule requires that covered entities oversee maintenance workers working in locations that house PHI. In the final HIPAA Security Rule at §164.308(a)(3)(ii)(A), this legal statement is broadened to require covered entities to supervise all workforce members that work in locations that house PHI.

6) Addition: Additions are legal statements that are added to a final rule. For example, the final HIPAA Security Rule added statements requiring that covered entities keep up to date documentation about security policies and procedures. This requirement was not in the proposed HIPAA Security Rule.

7) Requirement Made Optional: Regulatory agencies may make a requirement optional based upon feedback they receive from the public. For example, as discussed in Section IV.A.6, HHS made the requirement that certified EHRs record disclosures of PHI made for treatment, payment, or healthcare operations optional for MU Stage 1.

8) Introduced Cross-Reference: Cross-references are citations from one legal text to another [17]. Cross-references may add constraints and exceptions to compliance requirements, may be outside of the scope of the software system being developed, or may even introduce conflicting requirements that must be addressed [17]. When issuing final rules, a regulatory agency may introduce a cross-reference that did not previously exist in the proposed rule. For example, the proposed EHR Certification Rule defines what actions are considered a disclosure of PHI. In the final EHR Certification Rule, this definition is replaced with a cross-reference to the definition of disclosure in HIPAA.

C. Adaptability Heuristics

The adaptability heuristics aid requirements engineers in identifying which areas of the law are likely to change. Legal drafters may use varying strategies when making changes to the law. For example, ambiguity in a proposed rule may lead drafters to elaborate the rule to resolve the ambiguity; alternatively, they may remove the ambiguous requirement altogether. Thus, a 1-to-1 ratio does not exist between the heuristics and the adaptability taxonomy classifications. The heuristics predict *that* a section of a proposed rule may change, not *how* that section will change.

Not all changes to a proposed rule can be predicted. For example, the adaptability rationale *Resources Lacking to Implement* is difficult to predict without deep knowledge of the domain, the resources, and associated infrastructure available to organizations.

We now introduce our adaptability heuristics.

H_1: Ambiguous requirements suggest that the law may be disambiguated and therefore subject to change.

We employ the Inquiry-Cycle Model to identify ambiguity [21]. Unanswered Inquiry-Cycle Model questions indicate an ambiguous compliance requirement that needs to be clarified in the final rule, because they represent compliance requirements that cannot be operationalized as software requirements without clarification. For example, consider the proposed EHR Certification Rule, which requires that certified EHRs use a hashing algorithm that is SHA-1 or higher for integrity protection (§170.210(c)). It is not clear who determines that a hashing algorithm is higher than SHA-1 or how they would make such a determination—unanswered *who* and *how-to* Inquiry-Cycle Model questions. In the final EHR Certification Rule, HHS revises this requirement to read "A hashing algorithm with a security strength equal to or greater than SHA–1 (Secure Hash Algorithm (SHA–1) as specified by the National Institute of Standards and Technology (NIST) in FIPS PUB 180–3 (October, 2008)) must be used to verify that electronic health information has not been altered" (§170.210(c)). By referring to the FIPS PUB 180-3 document published by NIST, the unanswered *who* and *how-to* questions are addressed.

H_2: A repeated concept suggests that the concept may formally be defined in the final rule.

When a concept is repeated in a proposed rule, it may be formally defined in the final rule. In the proposed HIPAA Security Rule, §142.302 describes the entities that must comply with the rule, including health plans, healthcare clearinghouses, and healthcare providers. §142.102 contains

similar language. Throughout the remainder of the proposed rule, these entities are referred to as "entities designated in §142.302". In the final rule, the term "covered entity" is introduced, and the definitions at §142.102 and §142.302 are combined into the definition of a covered entity.

H_3: Duplicate concepts may be combined or disambiguated.

When a proposed rule uses multiple terms for the same concept, they are likely to be combined into one concept, or their differences defined. For example, the proposed HIPAA Security Rule uses the terms "health information pertaining to an individual", "health information", "data", and "information" interchangeably. In the final HIPAA Security Rule, these terms are replaced with the term electronic protected health information (ePHI).

H_4: Technology-specific requirements may be removed.

Technology-specific requirements in a proposed rule may be removed in favor of requirements that fosters industry innovation and avoids implementation and design bias. For example the proposed HIPAA Security Rule requires that covered entities adopt an access control mechanism from a list defined in the regulation. However, as commenters point out, defining a list of acceptable access control techniques stifles innovation—if an improved access control technique is developed, it may not meet the requirements of the regulation until the regulation is updated. In the final HIPAA Security Rule, the requirement is restated to require policies and procedures that restrict access to PHI to individuals that have been assigned access rights—without requiring covered entities to adopt specific access control technologies.

H_5: Specific requirements subsumed by a broader requirement may be removed.

When requirements are duplicated in a proposed regulation, detailed requirements may be removed in favor of the requirements that can be more broadly applied. In the proposed HIPAA Security Rule, covered entities are required to have: (a). visitor access control procedures, and (b). access control procedures. These requirements express duplication—access control procedures would necessarily have provisions around visitor access control. In the final HIPAA Security Rule, the visitor access control requirement is removed.

V. Applying the Adaptability Framework

Our summative case study examined the EHR Certification Rule and was conducted in two phases. In the first phase, we employed the adaptability heuristics to predict which areas of the proposed EHR Certification Rule would change. The first phase took 15 person hours. The Appendix lists the 14 areas in the proposed EHR Certification Rule that we predicted would change.

In the second phase of our summative study, we analyzed the proposed and final EHR Certification Rules to validate our predictions and identify unpredicted changes. The second phase took 20 person hours. Table III displays the results of this analysis. We found:

- 9 changes that we accurately predicted (true positives),
- 5 changes we predicted that were not accurate (false positives),
- 104 legal statements for which we predicted no change and for which no change occurred (true negatives), and
- 33 legal statements for which we predicted no change and which changed in the final rule (false negatives).

This yields an accuracy of 0.75 (the ratio of predictions that were correct), a precision of 0.64 (the ratio of predictions that were accurate), and a recall of 0.21 (the ratio of the regulatory changes we identified).

TABLE III. Actual Changes to the EHR Certification Rule

	Accurate	Inaccurate
Predicted	11 (true positives)	5 (false positives)
Not Predicted	104 (true negatives)	33 (false negatives)

VI. Threats to Validity

When designing any case study, care should be taken to mitigate threats to validity. We make no causal inferences as a result of our study, so internal validity is not a concern [24]. The Adaptability Rationale may appear to make causal inferences because they document the reasons why a regulation changes, but we did not make these inferences. Rather, we document HHS's stated reasons for why they changed the regulation.

External validity is the ability of a case study's findings to generalize to broader populations [24]. We employ grounded theory analysis; thus, our Adaptability Framework taxonomy is currently applicable to the healthcare regulations that we examined. Future studies will explore the applicability of our framework to other types of laws, such statues, and in other domains. We anticipate additional rationale, taxonomy classifications, and heuristics will be identified as we study additional domains.

Construct validity addresses the degree to which a case study is in accordance with the theoretical concepts used [24]. Three ways to reinforce construct validity are: use multiple sources of reliable evidence; establish a chain of evidence; and have key informants review draft case study reports [24]. To establish a chain of evidence, we carefully documented the Adaptability Rationale, Adaptability Taxonomy, and Adaptability Heuristics when performing our analyses; these classifications became the Adaptability Framework in Section IV. Finally, our draft case study report was reviewed by several ThePrivacyPlace members as well as by the law professor co-author who was a senior official during the drafting of the HIPAA Security Rule.

Reliability is the ability to repeat a study and observe similar results [24]. To reinforce our study's reliability, we carefully document each rationale, taxonomy classification, and heuristic using our grounded theory approach. Moreover, by adopting a multiple case study approach, we relied on multiple sources of evidence; herein, our EHR Certification study benefited from our prior development of our Adaptability Framework during our HIPAA study.

VII. Discussion & Future Work

To date, the literature has lacked tools for software engineers to identify and prepare for changes in legal rules. This paper is the first to attempt to predict what areas of a proposed legal rule will change in the final rule. Progress in this task will assist in legal compliance, reduce the costs of adapting legacy systems to changes in legal requirements, and assist a company to be first to market with compliant software.

In this paper, we describe a multiple case study in which we examine how proposed regulatory rules evolve into final rules. We developed our adaptability framework through a formative case study on the HIPAA Security Rule and applied the framework in a summative study on the EHR Certification Rule. The framework consists of three components: adaptability rationales that describe *why* regulations change; adaptability taxonomy classifications that describe *how* regulations change; and adaptability heuristics that predict *that* a regulation will change. This framework aids requirements engineers in predicting areas in a proposed rule that are likely to change in final rulemaking, allowing engineers to begin working towards more stable areas of the regulation. To the best of our knowledge, we are the first software engineering researchers to examine how regulatory requirements evolve.

The paper shows overall progress in identifying portions of a proposed rule that are likely to change. For the EHR Certification Rule, our framework correctly predicted 11 true positives and 104 true negatives for changes, with 5 false positives and 33 false negatives, or 115 correct predictions out of 153 total predictions, or 75% correct. This first analysis correctly predicted 11/16 (68%) of the areas of change, and 104/109 (95%) of the areas of no change, assisting software engineers to prioritize areas for development prior to release of the final rule.

Our research study had two research questions (see Section II). Our first research question (Can we model the changes that are made to two proposed rules—the HIPAA Security Rule and the EHR Certification Rule—during final rulemaking?) is addressed by the adaptability rationale and taxonomy that model how the HIPAA Security Rule and EHR Certification Rule evolved, respectively. We had hoped the adaptability heuristics would address our second research question (Can requirements engineers predict changes that will be made to proposed rules?). As previously discussed, not all changes that are made to a proposed rule can be predicted. For example, the results here indicated that it was particularly challenging to predict how legal drafters may reorganize and renumber legal statements within a rule, although the consequence for requirements engineers may be to make it easy to change legal cross-references. As discussed in Section V, even with this understanding, we identified only 21% of the actual regulatory changes in our study. Our study's recall highlights the challenges that face requirements engineers developing software for regulated domains. Thus, the current set of adaptability heuristics does not fully address our second research question. We position our adaptability framework as a preliminary effort to address the challenges introduced by evolving compliance requirements.

There were several adaptability rationales for which we did not predict any changes. For example, we did not predict any changes that had the rationale *Inappropriate for Domain* or *Resources Lacking to Implement*. To identify these changes likely requires deep domain knowledge. In our future studies, we plan to analyze public comments that are submitted for a proposed rule. Comments are publicly available online[11]. By analyzing these comments we seek to determine whether they can be used to identify potential changes that will be made to the proposed rule before it is finalized.

In our summative study, we made fourteen predictions, of which nine were accurate (true positives) and five were inaccurate predictions (false positives). Our precision (0.64) suggests that requirements engineers are able to predict which areas of a proposed regulation are likely to change. By predicting which areas of a regulation are likely to change, requirements engineers can focus on the stable areas of a regulation first, waiting for the less stable areas of a regulation to be clarified in final rulemaking. Of the five inaccurate predictions:

- Two of the predictions are related to technology-specific requirements (Prediction #3 and #6 in the Appendix). However, one goal of the EHR Certification Rule is to foster interoperability between healthcare IT systems. Thus, HHS specified specific standards in the rule to encourage this interoperability. For example, the EHR Certification Rule specifies HL7[12] version 2.5.1 as the standard when submitting lab results to public health agencies.
- Two of the predictions were made because we identified ambiguity in the proposed rule (Predictions #8 and #10 in the Appendix). For example, in §170.302(g)(1), EHRs are required to receive lab results in a structured format. This format is not defined, and this ambiguity remains in the final rule.
- We made another prediction because of ambiguity in the proposed rule (Prediction #14 in the Appendix). HHS acknowledged that the requirement in §170.304(d) is ambiguous. However, they did not change the regulation, but rather pointed to another law that provides clarification.

Our previous work revealed that financial regulations are less likely to contain conflicts due to cross-references because: (1) the financial industry has a long history of regulation accompanied by strong enforcement mechanisms, and (2) the financial industry has the highest lobbyist expenditure [17]. In our future work, we plan to examine how rules evolve in the financial domain to determine if changes in financial rules can be predicted with more accuracy than in healthcare rules. Also, the proposed EHR certification rule for Meaningful Use Stage 2[13] was recently

11 http://www.regulations.gov/

12 http://www.hl7.org/

13 http://www.ofr.gov/OFRUpload/OFRData/2012-04430_PI.pdf

issued by HHS, and is currently in the comments phase. We plan on performing a case study on this regulation using the same design as our summative study.

ACKNOWLEDGMENTS

We thank ThePrivacyPlace reading group for their helpful comments, and Kenesa Ahmad for her legal research assistance.

REFERENCES

[1] A.I. Antón, J.B. Earp, "A Requirements Taxonomy for Reducing Web Site Privacy Vulnerabilities", *Requirements Engineering Journal*, 9(3), 2004, pp. 169-185.

[2] A.I. Antón, C. Potts, "Functional Paleontology: The Evolution of User-Visible System Services", *IEEE Trans. on Software Engineering*, 29(2), 2003, pp. 151-166.

[3] T.D. Breaux, A.I. Antón, "Analyzing Regulatory Rules for Privacy and Security Requirements", *IEEE Trans. on Software Engineering*, 34(1), Jan.-Feb. 2008, pp. 5-20.

[4] T.D. Breaux, M.W. Vail, A.I. Antón, "Towards Regulatory Compliance: Extracting Rights and Obligations to Align Requirements with Regulations", *14th IEEE Intl. Requirements Engineering Conf.*, 2006, pp. 46-55.

[5] R.A. Carter, A.I. Antón, A. Dagnino, L. Williams, "Evolving Beyond Requirements Creep: A Risk-Based Evolutionary Prototyping Model", *5th IEEE Intl. Requirements Engineering Conf*., 2001, pp. 94-101.

[6] D.E. Damian, D. Zowghi, "Requirements Engineering Challenges in Multi-site Software Development Organizations", *Requirements Engineering Journal*, 2003, pp. 149-160.

[7] S. Ghanavati, D. Amyot, L. Peyton, "Towards a Framework for Tracking Legal Compliance in Healthcare", *19th Intl. Conf. on Advanced Information Systems Engineering*, 2007, pp. 218-232.

[8] B.G. Glaser, *Theoretical Sensitivity*, Sociology Press, 1978.

[9] B.G. Glaser, A.L. Strauss, *The Discovery of Grounded Theory*, Aldine Transaction, 1967.

[10] S.D.P. Harker, K.D. Eason, J.E. Dobson, "The Change and Evolution of Requirements as a Challenge to the Practice of Software Engineering", *IEEE Intl. Symposium on Requirements Engineering*, 1993, pp. 266-272.

[11] S. Islam, H. Mouratidis, S. Wagner, "Towards a Framework to Elicit and Manage Security and Privacy Requirements from Laws and Regulations", *17th Intl. Working Conf. on Req. Engineering: Foundation for Software Quality*, 2010, pp. 255-261.

[12] C. Jones, "Strategies for Managing Requirements Creep", *IEEE Computer,* 29(6), 1996, pp. 92-94.

[13] L. Karlsson, A.G. Dahlstedt, B. Regnell, J. Natt och Dag, A. Persson, "Requirements Engineering Challenges in Market-Driven Software Development-An Interview Study with Practioners", *Information and Software Technology,* 49, 2007, pp. 588-604.

[14] A.K. Massey, B. Smith, P.N. Otto, A.I. Antón, "Assessing the Accuracy of Legal Implementation Readiness Decisions", *19th IEEE Intl. Requirements Engineering Conference*, 2011, pp. 207-216.

[15] J.C. Maxwell, A.I. Antón, "Developing Production Rule Models to Aid in Acquiring Requirements from Legal Texts", *17th Intl. IEEE Requirements Engineering Conf.*, 2009, pp. 101-110.

[16] J.C. Maxwell, A.I. Antón, "The Production Rule Framework: Developing a Canonical Set of Software Requirements for Compliance with Law", *1st ACM Intl. Health Informatics Symposium*, 2010.

[17] J.C. Maxwell, A.I. Antón, P. Swire, M. Riaz, C. McCraw, "A Legal Cross-References Taxonomy for Reasoning about Compliance Requirements", *Requirements Engineering Journal*, 17(2), 2012, pp. 99-115.

[18] S. McGee and D. Greer, "A Software Requirements Change Taxonomy", *4th Intl. Conf. on Software Engineering Advances*, 2009, pp. 51-58.

[19] N. Nurmuliani, D. Zowghi, S.P. Williams, "Using Card Sorting Technique to Classify Requirements Change", *12th Intl. IEEE Requirements Engineering Conf.*, 2004, pp. 240-248.

[20] P.N. Otto, A.I. Antón, "Addressing Legal Requirements in Requirements Engineering", *15th IEEE Intl. Requirements Engineering Conf.,* 2007, pp. 5-14.

[21] C. Potts, K. Takahashi, A.I. Antón, "Inquiry-Based Requirements Analysis", *IEEE Software*, 11(2), Mar. 1994, pp. 21-32.

[22] A. Siena, A. Perini, A. Susi, J. J. Mylopoulos, "A Meta-Model for Modelling Law-Compliant Requirements", *2nd Intl. Workshop on Requirements and Law*, 2009.

[23] J. D. Young. "Commitment Analysis to Operationalize Software Requirements From Privacy Policies". *Requirements Engineering Journal*, 16(1):33–46, 2011.

[24] R.K. Yin, *Case Study Research: Design and Methods, in Applied Social Research Methods Series*, Vol. 5, 2003, 3rd ed.

[25] D. Zowghi, R. Offen, "A Logical Framework for Modeling and Reasoning About the Evolution of Requirements", *3rd Intl. IEEE Requirements Engineering Conf.,* 1997, pp. 247-257.

APPENDIX: PREDICTED CHANGES TO THE EHR CERTIFICATION RULE

In-dex	Proposed Text	Final Text	Heuristic	Acc-urate?
1	170.102: Certified EHR Technology means a Complete EHR or a combination of EHR Modules, each of which: (1) Meets the requirements included in the definition of a Qualified EHR; and (2) Has been tested and certified in accordance with the certification program established by the National Coordinator as having met all applicable certification criteria adopted by the Secretary.	170.102: Certified EHR Technology means: (1) A Complete EHR that meets the requirements included in the definition of a Qualified EHR and has been tested and certified in accordance with the certification program established by the National Coordinator as having met all applicable certification criteria adopted by the Secretary; or (2) A combination of EHR Modules in which each constituent EHR Module of the combination has been tested and certified in accordance with the certification program established by the National Coordinator as having met all applicable certification criteria adopted by the Secretary, and the resultant combination also meets the requirements included in the definition of a Qualified EHR.	H1-unresolved *what–is* question	Yes
2	170.202(a-b): The Secretary adopts the following standards […] (a) Standard. The Organization for the Advancement of Strutured Information Standards	<removed>	H4	Yes

In-dex	Proposed Text	Final Text	Heuristic	Acc-urate?
	(OASIS) Simple Object Access Protocol (SOAP) […] (b) Alternative Standard. A stateless, client-server, cacheable communications protocol that adheres to the principles of Representational State Transfer (REST) must be used.			
3	170.205: <requires that certified EHRs adopt various standards such as LOINC and SNOWMED>	170.205-207: <requires that certified EHRs adopt various standards such as LOINC and SNOWMED>	H4	No
4	170.210(a)(1): The Secretary adopts the following standards to protect electronic health information created, maintained, and exchanged: (a) Encryption and decryption of electronic health information. (1) General. A symmetric 128 bit fixed-block cipher algorithm capable of using a 128, 192, or 256 bit encryption key must be used.	170.210(a)(1): The Secretary adopts the following standards to protect electronic health information created, maintained, and exchanged: (a) Encryption and decryption of electronic health information—(1) General. Any encryption algorithm identified by the National Institute of Standards and Technology (NIST) as an approved security function in Annex A of the Federal Information Processing Standards (FIPS) Publication 140–2 (incorporated by reference in §170.299).	H4	Yes
5	170.210(b): The date, time, patient identification, and user identification must be recorded when electronic health information is created, modified, deleted, or printed; and an indication of which action(s) occurred must also be recorded.	170.210(b) The date, time, patient identification, and user identification must be recorded when electronic health information is created, modified, accessed, or deleted; and an indication of which action(s) occurred and by whom must also be recorded.	H1-unresolved *what-if* question	Yes
6	170.210(c): A secure hashing algorithm must be used to verify that electronic health information has not been altered in transit. The secure hash algorithm (SHA) used must be SHA–1 or higher.	170.210(c): A hashing algorithm with a security strength equal to or greater than SHA–1 (Secure Hash Algorithm (SHA–1) as specified by the National Institute of Standards and Technology (NIST) in FIPS PUB 180–3 (October, 2008)) must be used to verify that electronic health information has not been altered.	H4	No
7	170.210(c): <see prediction #6>	170.210(c): <see prediction #6>	H1 - unresolved *who* and *how-to* questions	Yes
8	170.302(g)(1): Electronically receive clinical laboratory test results in a structured format and display such results in human readable format.	170.302(h)(1): Electronically receive clinical laboratory test results in a structured format and display such results in human readable format.	H1 – unresolved *what-is* question	No
9	170.302(h): Enable a user to electronically select, sort, retrieve, and output a list of patients and patients' clinical information, based on user-defined demographic data, medication list, and specific conditions.	170.302(i): Enable a user to electronically select, sort, retrieve, and generate lists of patients according to, at a minimum, the data elements included in: (1) Problem list; (2) Medication list; (3) Demographics; and (4) Laboratory test results.	H1 – unresolved *what-is* question	Yes
10	170.302(h)(i)(1): Calculate and electronically display quality measures as specified by CMS or states.	170.302(n): For each meaningful use objective with a percentage-based measure, electronically record the numerator and denominator and generate a report including the numerator, denominator, and resulting percentage associated with each applicable meaningful use measure.	H1 – unresolved *what-is* question	No
11	170.302(m)(2): Electronically record, retrieve, and transmit immunization information to immunization registries in accordance with the applicable state-designated standard format.	<removed>	H1 – unresolved *what-is* question	Yes
12	170.302(r)(1): Record actions related to electronic health information in accordance with the standard specified in §170.210(b). (2) Provide alerts based on user-defined events.	170.302(r)(1): (r) Audit log. (1)—Record actions related to electronic health information in accordance with the standard specified in § 170.210(b).	H1 – unresolved *what-is* question	Yes
13	170.302(u)(1): Encrypt and decrypt electronic health information according to user-defined preferences in accordance with the standard specified in § 170.210(a)(1).	170.302(u)(1): Encrypt and decrypt electronic health information in accordance with the standard specified in § 170.210(a)(1), unless the Secretary determines that the use of such algorithm would pose a significant security risk for Certified EHR Technology.	H1-unresolved *what-if* question	Yes
14	170.304(d): Electronically generate, upon request, a patient reminder list for preventive or follow-up care according to patient preferences based on demographic data, specific conditions, and/or medication list.	170.304(d): Enable a user to electronically generate a patient reminder list for preventive or follow-up care according to patient preferences based on, at a minimum, the data elements included in: (1) Problem list; (2) Medication list; (3) Medication allergy list; (4) Demographics; and (5) Laboratory test results.	H1-unresolved *how-to* question	No