Formulating the Enterprise Architecture Compliance Problem

Vytautas ČYRAS[a] and Reinhard RIEDL[b]
[a] *Vilnius University, Vilnius, Lithuania*
[b] *Bern University of Applied Sciences, Bern, Switzerland*

Abstract. We start from the Klaus Julisch's (2008) IT compliance problem definition and make an attempt to formulate the enterprise architecture compliance problem (EACP). The challenging issues comprise the complexity of the law phenomenon, compliance frameworks and methodologies to check EA for non-compliance with laws and regulations. We hold that a compliance methodology should take into account "shared" relevant laws and a requirements engineering framework. We reflect mainly on the view of enterprise architects on legal informatics and a vision driven approach on requirements elicitation in the context of enterprise engineering, which was proposed by Albertas Čaplinskas (2009). Then we raise a question of placing EACP into the Bonazzi-Hussami-Pigneur regulation and IT alignment framework (2009).

Keywords. Regulatory compliance, electronic identity compliance, enterprise systems, transparency engineering, legal informatics, legal visualization

Introduction

This paper attempts to overview some models which could contribute to formulate regulatory compliance problems. However, the field appears too broad to master with a sweep of the arm. A unified "enterprise-wide" compliance process remains an ambition. Thus the authors present reflections on various issues. The message is that a compliance methodology should follow a requirements engineering framework because the latter combines business, IT and legal perspectives.

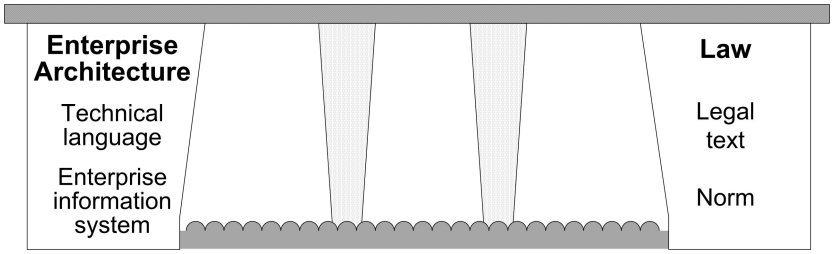

Figure 1. Building a bridge between enterprise architecture and law

This paper is challenged by a (naïve?) question: Does an enterprise architecture comply with the law? It can be compared with Alan Turing's "Can machines think?" [21]. Both cannot be answered simply yes or no. Each challenges to formulate a distinguished problem. Turing replaces the question with the following series: What

does it mean 'think', 'machine' and 'can'? This was discussed in the early artificial intelligence. Similarly we can ask: What enterprise architecture is? and Which law?

The "naive" compliance problem formulation above is similar to bridging enterprise architecture and law (Figure 1). The bridge metaphor is generally used in knowledge visualization. We utilize it because we see a similarity with the bridge between informatics and law, which was proposed by Friedrich Lachmayer to characterize the subject matter of legal informatics [5]. One arch is hardly feasible and therefore the bridge consists of multiple arches. Thus a multiphase transformation process emerges. The transformation is about bridging legal requirements and the enterprise information system.

Transparency optimization is a major purpose in EA. Legal requirements are one of a kind among all requirements tackled by enterprise architects. Different legal issues are concerned in every EA perspective. The enterprise architect's perspective has the task to integrate all the different views on EA, in particular, the business view, the ICT view, and the legal view. Figure 2 shows the key terms within the point of departure.

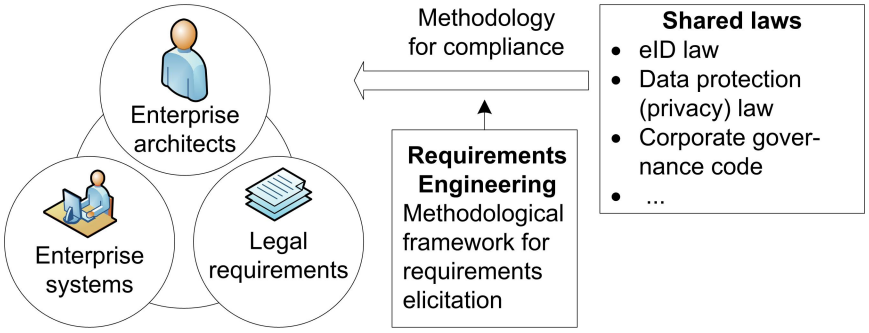

Figure 2. The key concepts tackled in this paper

1. Motivating the Research

1.1. e-Identity and e-Banking Requirements within the STORK 2.0 Project

A practical motivation for academia can be illustrated by the STORK 2.0 project[1] that concerns the design and implementation of the foundations for a unified European identification and authentication space. STORK (Secure idenTity acrOss boRders inKed) [2008-2011] established a European electronic identity (eID) interoperability platform which allows holders of national eIDs to access cross-border electronic services within six pilot applications. Its extension STORK 2.0 [2012-2014] will extend the range of services within four new pilots. One of them is focused on e-banking, mainly retail banking, the scene for secure e-invoicing. The use cases specify three distinct cross-border scenarios using national eIDs: (1) opening a bank account online using eIDs for identity authentication, (2) logging into a cross-border e-banking portal using eIDs, and (3) authorizing payment of electronic bills using eIDs.

[1] Cofunded under EU ICT Policy Support Programme as part of the Competitiveness and Innovation framework Programme (CIP); see also https://www.eid-stork.eu/.

The big picture for the whole project aims at a shift from interoperability for a few national eIDs to the creation of a single European identity space for borderless e-business. All trustworthy eIDs, both national and commercial, should be usable in this space upon the accreditation of the eID providers according to a 4-level quality standard for the trustworthiness of eIDs.

Non-repudiation is a critical requirement for electronic business. It is also a precondition for compliance as otherwise transactions could be repudiated. Both can be based on the authentication of interaction partners and a trustworthy recording of interaction activities. Depending on the criticality of business issues, different levels of quality should be met by the authentication procedures corresponding to different levels of quality for the eIDs used for authentication. But the vision is that it is only the quality level which decides on the width of use of an eID.

The feasibility and the value of the design for a single European identification and authentication space will be validated within STORK 2.0 through four pilots. One pilot is concerned with e-banking. Its key vision is to move identity and access management (IAM) out of the core banking IT system. Authentication should be possible with any eIDs issued by accredited eID providers guaranteeing the highest quality level of trustworthiness. Of course, a solution must include major national electronic identities in Europe, if they comply with highest quality standards. However, even for them legal issues are unclear.

Following is a use case to check for compliance. A company representative with an eID from country *X* (e. g. Germany), working in a company from country *Y* (e.g. Switzerland) logs into a banking platform in country *Z* (e.g. Austria, Lithuania or US). The number of potential customers comprises foreign nationals, for instance, those living in Switzerland and cross border commuters.

Do the requirements for e-banking comply with national eID laws? Can the proof of identity be transferred from the issuing of an eID be transferred to the opening of a bank account with this eID? In many countries this is an open question. For sure, some conflicts exist [14, p. 440]:

> Seen from a European political perspective, eIDs are primarily in potential conflict with privacy protection rights and thus with data protection laws. However, seen from a broader political perspective, the design of a single identification and authentication space also touches the so far hardly discussed eventual right for being recognized by electronic services.

This indicates that compliance is a tricky and much broader issue than it appears to be at first glance. For example, excluding some users may be legally compliant in one country and clearly non-compliant in another.

The STORK 2.0 project also tackles common infrastructure for federated e-Government, in particular in Switzerland. Today's challenges are:

- organizational and business models,
- implementation of a government cloud,
- refinement of the existing enterprise architecture in order to get it "working".

Tomorrow's challenge is enterprise architecture design for the implementation of the Lenk-Schaffroth-Schuppan vision of networked government, which links processes across different public administration organizations [12]. Future challenge is the separation of distribution, execution, and control in order to implement shared service centres for core tasks of the state. Apart from other challenges, there highly complex compliance issues to be considered.

As a final (much more simple) example, we may compare implementations for one-stop government, which depend on the integration of truly independent processes in different government agencies. For such implementations the choice of a tier for integration is critical. It makes a big difference whether it is done in the web-tier (front office integration) or in the application tier (back-office integration, e.g. with WS-BPEL, Web Services Business Process Execution Language). The latter is in many cases not legally compliant because it does not guarantee the immediate registration of incoming requests at every organization.

1.2. Formulating the Enterprise System Compliance Problem

Klaus Julisch (2008) suggests academia a paradigm shift: from "selling" security when organizations seek to "buy" compliance to complementing current security research by additional research into security compliance:

> [A]s long as careers are terminated and people go to jail…for failures in compliance – rather than security – the commercial world will continue to pursue compliance rather than security as their primary goal. [8, p. 71]

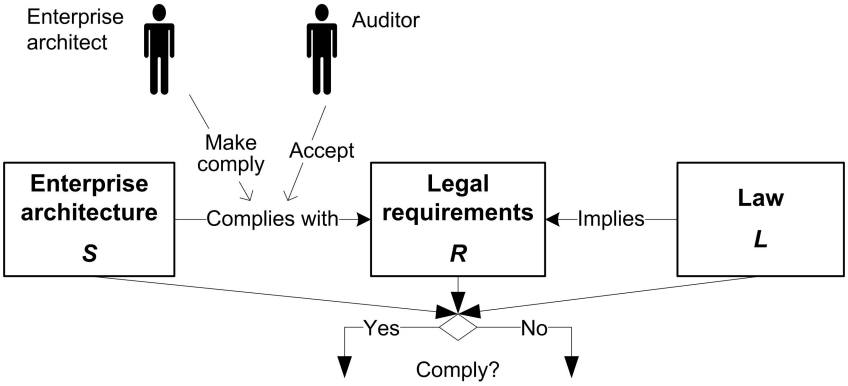

Figure 3. Enterprise system compliance problem

Julisch defines: "security compliance, in IT systems, is the state of conformance with externally imposed functional security requirements and of providing evidence (assurance) thereof." [8, p. 72] He defines the security compliance problem as follows:

> *Definition*: Given an existing IT systems *S* and an externally imposed set *R* of security requirements. The *Security Compliance Problem* is to make system *S* comply with the security requirements *R* and to provide assurance that an independent auditor will accept as evidence of the compliance of system *S* with requirement *R*.

Following the definition above, we would formulate the *Enterprise Architecture Compliance Problem* (EACP). It is (1) to make enterprise architecture *S* comply with requirements *R* that relate to a law *L*, and (2) to provide assurance that an independent auditor will accept this as evidence (Figure 3).

We simply added a law *L* to Julisch's formulation. Semiformal definitions above can only serve as a first iteration. Problem solutions in practice can hardly result with yes or no. Practice involves more elements. Feedback loops would improve *S*, *R* and *L*. Conceptualisations of *L* may involve different elements depending on abstraction level. A legal principle, a whole statute or a specific provision may stand for *L*.

1.3. Complexity Issues when Attempting to Formalise the Law in the Context of EA

Failure to understand the law is one of non-compliance reasons [18, p. 59–61]. This failure can be examined from the management perspective. We comment further from the legal perspective. The texts of laws constitute only a part of the whole legal system. The meaning (*Sinn*) of law – the Ought realm – can hardly be understood from the sole legal text. Therefore a freshman can hardly understand the spirit of law while reading a separate statute. On the other hand, the compliance problem can hardly be reduced to tick the box. The law does not allow to be easily represented for EA developers whose purpose is to enforce the law. The following issues raise difficulties:

1. *Abstractness of norms.* Norms are formulated (on purpose) in abstract terms.
2. *Principle vs. rule.* The difference in regulatory philosophy between the US and other countries [18, p. 46].
3. *Open texture.* This can be illustrated by H. L. A. Hart's example of "Vehicles are forbidden in the park".
4. *The myriad of regulatory requirements.* Compliance frameworks are multidimensional.
5. *Heuristics.* High level concepts are translated into invented low level ones.
6. *Teleology.* The purpose of a legal norm usually can be achieved by a variety of ways. They need not to be listed in a statute and specified in detail.
7. *Legal interpretation methods.* The meaning of a legal text cannot be extracted from the sole text. Apart from the grammatical interpretation, other methods can be invoked, such as systemic and teleological interpretation.
8. *Consciousness of the society.* Modeling it is a tough task.

2. A Variety of Factors to Comply with

Note that judges are allowed to have different opinions. Are auditors allowed too? The COSO framework[2] was issued prior to Sarbanes-Oxley Act of 2002 (SOX)[3]. Deterring fraudulent financial reporting is an aim. The use of the COSO framework by company management shows the scale from 'no extent' to 'large extent' [11]. The Directive 2008/30/EC[4] can be compared with the impact of SOX in the US. Information technology internal controls are not an exclusive concern of COSO.

Anthony Tarantino (2008) devotes the whole book to Governance, Risk and Compliance (GRC). He suggests taking a holistic approach. In particular, he addresses the risk concept [20, p. 15–17, 236–237]. Banking's categorization accords[5] describe seven major areas of operational risk:

1. Internal fraud: unauthorized activities; theft and fraud.
2. External fraud: external security; theft and fraud.

---

[2] The Committee of Sponsoring Organizations (COSO) framework was originally issued in 1992 (entitled Internal Control – Integrated Framework) and updated in 2004 (Enterprise Risk Management), see http://www.coso.org/IC-IntegratedFramework-summary.htm.

[3] See Wikipedia, "Sarbanes–Oxley Act", http://en.wikipedia.org/wiki/Sarbanes%E2%80%93Oxley_Act.

[4] Directive 2008/30/EC of the European Parliament and of the Council of 11 March 2008 amending Directive 2006/43/EC on statutory audits of annual accounts and consolidated accounts. In Lithuania see the Audit Law of 2011-11-17 (originally 15-06-1999) VIII-1227, Gazette 1999, Nr. 59-1916.

[5] See Wikipedia, "Basel II", http://en.wikipedia.org/wiki/Basel_II.

3. Employment practices: employee relations; safe environment; diversity and discrimination.
4. Clients, products, and business processes: suitability, disclosure, and fiduciary aspects; product flaws; improper business or market practices; advisory activities; selection, sponsorship and exposure.
5. Damage to physical assets: disasters and other events.
6. Business disruptions and system failures: systems.
7. Execution, delivery, and process management: transaction capture, execution, and maintenance; monitoring and reporting; incomplete legal documentation; customer account management.

This Level 1 and Level 2 categorization can serve as a framework. It illustrates a variety of risk factors, which have to be faced by auditors and other personnel.

Compliance is a multi-criteria problem. A single framework or standard can hardly be a solution to all compliance and control needs:

> Absolute adherence to a regulation by adopting a basic framework without considering the entire organization and threats that affect it make the organization compliant, but not secure or resilient to operational disruptions. [6, p. 62]

3. Elements of Enterprise System

According to the systems engineering view, an enterprise system consists of three subsystems [4] listed below. However, there is no generally accepted agreement.

1. *Enterprise business system*. It is comprised of business actors, resources and business processes;
2. *Enterprise information system* (IS) "is a whole formed out of organisational memory and sets of information processing actors (IPA), information flows, and interrelated information processing processes implemented in accordance with the enterprise information processing policies and standards" [4]
3. *Enterprise application system*. It is comprised of hardware agents, protocols, knowledge bases and software application programs.

Other elements can be distinguished, too. This depends on an author's view. Ross et al. note that the term 'architecture' has acquired a negative connotation in some companies and quote saying "Architectures, like fondue sets…, are rarely used." [17, p. 47] They make emphasis on distinguishing between enterprise architecture and IT architecture. They also note that the IT unit typically addresses four levels of enterprise architecture [17, p. 48–49]:

1. *Business process architecture*. The activities or tasks composing major business processes identified by the business process owners.
2. *Data or information architecture*. Shared data definitions.
3. *Applications architecture*. Individual applications and their interfaces.
4. *Technology architecture*. Architecture services and the technology standards they are built on.

Subsystems and systems thinking are stressed in [7, p. 29–52]. First a *system* is defined as "a set of discernable, interacting parts or subsystems that form an integrated whole that acts with a single goal or purpose" (p. 29). Then EA is characterised:

> An *Enterprise Architecture* describes the structure of an enterprise, its decomposition into subsystems, the relationships between the subsystems, the relationships with the external environment, the terminology to use, and the guiding principles for the design and evolution of an enterprise. [7, p. 102]

One of the problems right now with enterprise architecture (see [17]) is that for reasons of simplicity those practically usable in real life focus on very few aspects of a real world IS, usually issues which are shared throughout the organization. Depending on the operating model of the organization, this is just technology, or in addition data and/or business processes. In the spirit of EA in the sense of Ross et al., it would make sense to define the "shared" relevant laws and integrate them; cf. [17, p. 12–13]:

> Companies are buffeted by constant changes in regulations, such as Sarbanes-Oxley, Basel II, and HIPAA. As companies become global, they become accountable for increasingly complex reporting requirements. …Companies may not be able to anticipate new regulations, but they can increase the likelihood that needed data is readily available or can easily be accumulated.

Ross et al. suggest encapsulating enterprise architecture in a core diagram, which depicts a high-level view of the process, data, and technologies constituting the desired foundation for execution. Here we raise a question: how to formulate the EA compliance problem once such a one-page core diagram is provided? Writing a list of compliance requirements? An answer should concern a concluding remark that enterprise architecture is not a detailed blueprint of systems, data, and technology, but instead a business vision [17, p. 206].

Enterprise architects check the architecture for potential conflicts with the law. The regulations which influence enterprise architectures, perhaps SOX, can be barely aware [13]. The following relationships can be identified here:

- Architecture descriptions have to leverage checking compliance.
- Legal informatics experts can contribute to legislation, esp. in e-Government.
- Enterprise architects become important partners for legal informatics experts. This is possible in the revision of the law, e.g. in digital identity regulations.
- Contacts with authorities when anticipating ICT perspectives. Regulation of software exchange, for instance, modules in finance informatics.
- Ideas in legal informatics; patterns and anti-patterns.

4. Enterprise Compliance Process

Financial compliance process is an important but not the sole issue of conformance. Enterprise content management (ECM) systems are focused in [9] and a high-level compliance process is provided; see Figure 4. Following is the list of standard requirements that any ECM vendor should provide: library services, repository search, document routing, central user administration, support for all popular text file formats, document imaging. More complex requirements: document-centric collaboration, compound documents support, digital assets management, records management, rule-driven workflow, process management, advanced security, etc. [9, p. 262–264].

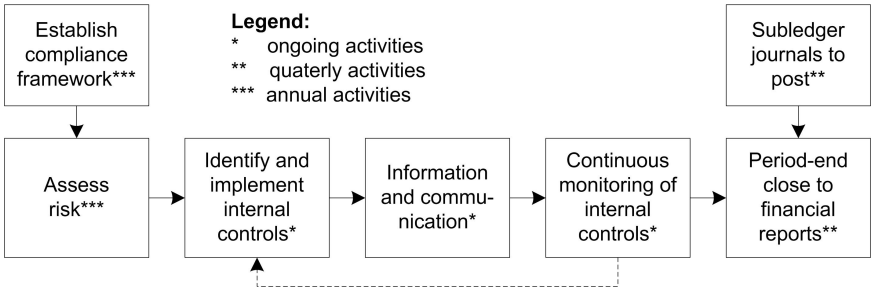

Figure 4. High-level compliance process [9, p. 260]

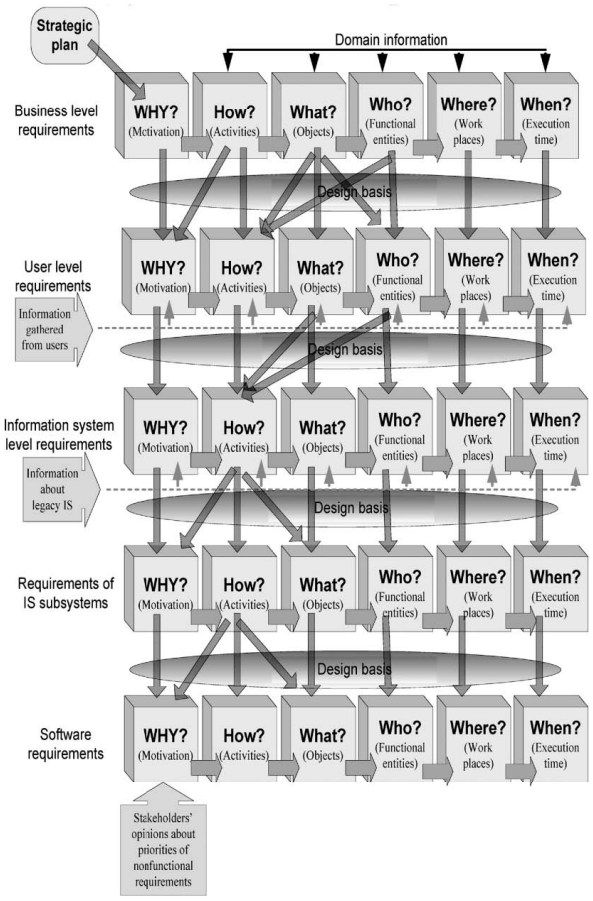

Figure 5. Čaplinskas' methodological framework for requirements elicitation, analysis, specification and validation; see [4]

5. The Legal Perspective in Enterprise Engineering

The only true purpose of the work of enterprise architects is transparency optimization in an organization. Three central perspectives to enterprise systems can be concerned:

1. business perspective,
2. ICT (information and communication technologies) perspective,
3. legal perspective.

Continuing the list above, other potential perspectives can be mentioned:

4. internal communication perspective (a direction tool),
5. public relations and marketing perspective,
6. political and economic perspective (probably in e-Government).

In our research we focus on the legal perspective. More perspectives are concerned in architectural frameworks such as Zachman's one [19]. Zachman's idea to decompose the system into a number of perspectives and focus areas has served as a theoretical basis for the vision driven approach proposed by Čaplinskas. Zachman decomposes each perspective into six focus areas to be answered: what (data)? how (function)? where (network)? who (people)? when (time)? and why (motive)? The following five perspectives (views, levels) are shown in Figure 5:

1. business level requirements (the view of business analyst),
2. user level requirements (the view of stakeholders),
3. IS (information system) requirements (the view of IS analyst),
4. the requirements of IS subsystems (the view of IS engineer),
5. software requirements (the view of software analyst).

Other perspectives, which are out of the Zachman framework (that is an architectural one), are commented in [4, p. 355]:

> To be complete, it should additionally include the requirements of software components (the view of software architect), the implementation requirements (the view of software engineer), the process requirements (the view of process engineer), and the testing requirements (the view of tester). …The first five perspectives differ from corresponding ones provided by the Zachman's framework because they are designed for different purposes.

Each perspective (level) presents a model of the system. Each phase of system's life-cycle is subject to technical standards. The concepts of a to-be-system and requirements are related to law. The requirements document (system specification) is part of the contract with a customer. Every requirement is based on a norm. This norm is present in a technical standard, business rule or other kind of legal source. The difference in the nature of requirements stems from the difference of norms.

6. Towards a Methodology of the Compliance with the Law

The end-to-end enterprise architecture compliance problem is too large and too complex for any one company to tackle. Similar is with compliance auditing, frameworks and good practices. Following are theme aggregates to shape the integration of different recourses compliant with the law [13]:

1. internal arrangement of transparency,
2. methods for the legal architecture view as part of enterprise architecture,
3. design methods for law-triggered changes in the enterprise architecture.

We think that Čaplinskas' approach could provide a framework to shape the methods above. First, it is vision driven. Second, a legal perspective can be added. Further steps face the following problems. An analyst can hardly be aware of legal norms in different branches of law. Therefore methodologies are needed. A trivial idea

might be simply to check the requirements in each cell in Figure 5 for compliance. This would be classified as an ex-ante solution [2] "to design an artefact aimed at avoiding actions that are not compliant". However to lower the risk of violating strategic alignment, a holistic approach has to be undertaken. Ex-post solutions are "to design an artefact to assess the level of compliance".

Risk management for information technology is a growing challenge for GRC [20, p. 18]. The National Institute of Standards and Technology (NIST) provides IT guidelines for U.S. federal agencies. NIST's Special Publication 800-30 (Risk Management Guide for Information Technology Systems) recommendations involve three basic processes: risk assessment, risk mitigation, and evaluation and assessment. Absolute control measures are often cost prohibitive and require IT professionals to weigh the cost versus benefits. This process is complicated by the hundreds of software tool suppliers promising to fix their GRC problems [20, p. 19].

The Bonazzi-Hussami-Pigneur IT compliance framework is worth a special attention. Two dimensions, *Legal* and *IT*, and two kinds of sources of regulation to comply with, *External* and *Internal*, are depicted with squares (Figure 6). Distinguished alignments are represented with arrows. A direction points to the defined artefact [2].

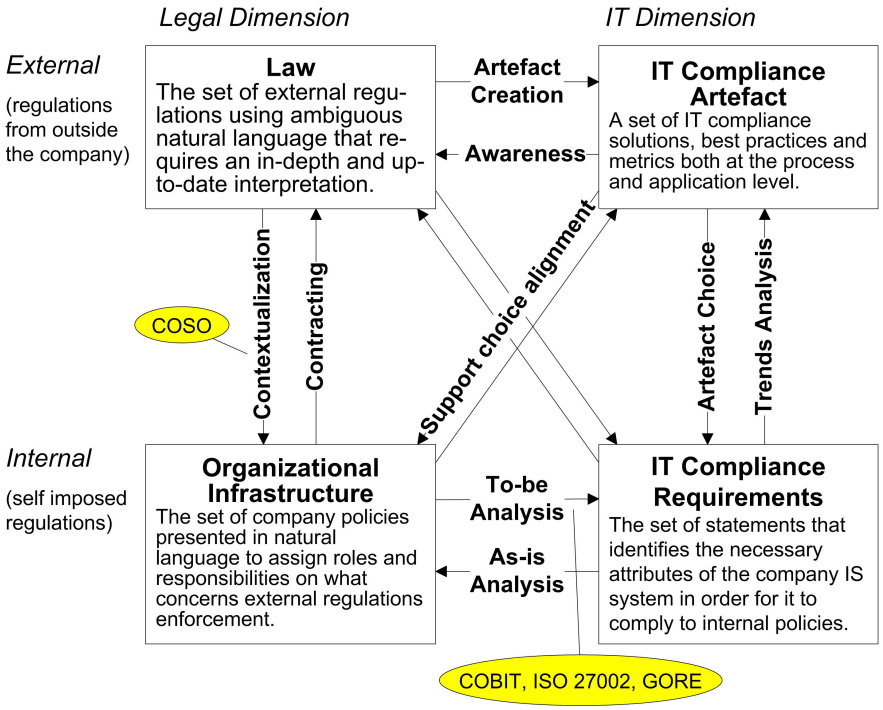

Figure 6. The Bonazzi-Hussami-Pigneur regulation and IT alignment framework; adapted from [2]

Every concept in Figure 6 denotes a broad field. Corporate noncompliance, corruption, etc. are just a few examples of violations. Noncompliance can be civil, criminal, administrative, but also reputational or market based. To analyse for conformance, one analyst can hardly be aware of norms in various branches of law.

To-be Analysis can be treated in different ways, depending on controls or IT risks

(e. g. COBIT[6], ISO 27002[7], GORE [15]). COBIT concerns IT governance and best practices. COBIT view of the implementation of a system's infrastructure can be summarized as follows: plan and organize, acquire and implement, deliver and support, and monitor and evaluate [16, p. 322]. Like its COSO control counterpart, it is the framework for the management of IT processes.

The design of law-compliant information systems is an area on a lower level of abstraction, though falls within the framework above. As noted in [10], representation of legal requirements were lacking in early works. Therefore they propose a framework to model the area which is structured in three dimensions: (1) field of law (flow control, reporting, web applications, etc.), (2) modelling level (analysis level, model level, metamodel level), and (3) research goal (explanation- or design-oriented research).

An interesting perspective for the actual implementation of such a holistic approach is provided by agile software development [1]. Agility in IS solution development was first introduced in the extreme programming (XP) movement. It is nowadays extended to many IS related tasks and applied even in development and design tasks without much IS relationship but with a general multi-disciplinary and multi-stakeholder challenge. The key two ideas of agility is to involve all expertises needed for a good IS solution in the team working on the solution and to develop the solution in short cycles, where at the beginning of each cycle a requirements analysis takes place and at the end of each cycle a working (running) in-between-product is delivered to users. This original concept has been adapted to more conceptual contexts like enterprise architecture management, where no running products can be delivered. In the original setting the key challenge is to design a good basic overall structure, as the spirit of agility contradicts the planning. However, in principle agility proposes an interesting split for a holistic approach. The integration of multidisciplinary perspectives is split into a very rudimentary strategic core concept and a concrete teamwork, where requirements analysis and holistic solution development is done iteratively.

7. Conclusions

The paper presents reflections on different issues of compliance. The authors are influenced by both formal models which are used in computer science and descriptive methods of social sciences (including law). This is all for the best in the present research. However, there is no silver bullet to attack regulatory compliance requirements – no one-off, best-of-breed solution. Similar is with the theoretically formulated Enterprise Architecture Compliance Problem (EACP). Positioning it in an IT alignment framework is a challenge. Though various compliance processes are positioned differently even within the two dimensions of IT and law. Requirements engineering contributes combining business, IT and law.

EACP formulation deals with static artefacts. The idea is to restrict with ex-ante analysis. Dynamics such as management loops and internal controls, which are usually

[6] COBIT is a framework for IT governance and control. See Wikipedia http://en.wikipedia.org/wiki/COBIT and http://www.isaca.org/Knowledge-Center/cobit/Pages/Overview.aspx.

[7] An information security standard, entitled *Information technology – Security techniques – Code of practice for information security management*; see Wikipedia, ISO/IEC 27002 http://en.wikipedia.org/wiki/ISO_17799. ISO 27001 is entitled *Information technology – Security techniques – Information security management systems – Requirements*. It is a specification. It uses words like 'shall'. ISO 27002 is a code of practice, not a specification, cf. [3].

involved in ex-post approaches, would make the formulation too complex to master. Process models involve costs-benefits tradeoffs, for example, fast, lower quality, general but cheap process versus slow, higher quality, specific but expensive one.

Acknowledgements

We are grateful to Friedrich Lachmayer for the encouragement of this joint research. Thanks also to anonymous reviewers for shaping the paper.

References

[1] K. Beck et al., *Manifesto for Agile Software Development*, Agile Alliance, 2001. Available from: http://agilemanifesto.org/ retrieved 11-06-2012.
[2] R. Bonazzi, L. Hussami, and Y. Pigneur, Compliance management is becoming a major issue in IS design. In: A. D'atri, D. Saccà, editors, *Information Systems: People, Organizations, Institutions, and Technologies*, Springer, 2009, 391–398.
[3] A. Calder, ISO 27001 and ISO 17999. In: A. Tarantino, editor, *Governance, Risk, and Compliance Handbook*, John Wiley, 2008, 169–179.
[4] A. Čaplinskas, Requirements elicitation in the context of enterprise engineering: a vision driven approach, *Informatica* **20**(3) (2009), Lithuanian Academy of Sciences, 343–368.
[5] V. Čyras and F. Lachmayer. Multiphase transformation in the legal text-to-program approach. Submitted to: I. Takashi, H. Mori, editors, *Liber amicorum Guido Tsuno*. Chuo University, Japan, 2012.
[6] J. J. Deluccia, *IT Compliance and Controls: Best practices for Implementation*, John Wiley, 2008.
[7] R. E. Giachetti, *Design of Enterprise Systems: Theory, Architecture, and Methods*, CRC Press, 2010.
[8] K. Julisch, Security compliance: the next frontier in security research. In: *Proceedings of the 2008 workshop on New security paradigms NSPW'08*, ACM, 2008, 71–74.
[9] J. Koo, What to look for in enterprise content management for compliance. In: A. Tarantino, editor, *Governance, Risk, and Compliance Handbook*, John Wiley, 2008, 259–266.
[10] R. Knackstedt, M. Heddier, J. Becker, Fachkonzeption Rechtskonformer Informationssysteme als Anwendungsgebiet der Rechtsvisualisierung. In: E. Schweighofer, F. Kummer, editors, *Europäische Projektkultur als Beitrag zur Rationalisierung des Rechts, Tagungsband IRIS 2011*, OCG, 549–558.
[11] T. Leech, COSO – is it fit for purpose? In: A. Tarantino, editor, *Governance, Risk, and Compliance Handbook*, John Wiley, 2008, 65–75.
[12] K. Lenk, T. Schuppan, and M. Schaffroth, *Networked Public Administration: Organisational Concept for a Federal eGovernment Switzerland*. eCH-0126 White Paper. Federal Department of Finance FDF, Switzerland, 2010. Available from: http://www.ech.ch.
[13] R. Riedl, Rechtsinformatik aus Sicht des Unternehmensarchitekten. In: A. Geist, C. R. Brunschwig, F. Lachmayer, G. Schefbeck, hrsg. *Strukturierung der Juristischen Semantik – Structuring Legal Semantics. Festschrift für Erich Schweighofer*, Editions Weblaw, Bern, 2011,257–269.
[14] R. Riedl et al., A multiperspective view of unified identification and authentication spaces. In: E. Schweighofer, F. Kummer, W. Hötzendorfer, editors, *Transformation of Legal Languages, Proceedings of IRIS 2012*, OCG, Wien, 2012, 439–443. Available from: http://jusletter-eu.weblaw.ch/.
[15] A. Rifaut and E. Dubois, Using goal-oriented requirements engineering for improving the quality of ISO/IEC 15504 based compliance assessment frameworks. In: *16th IEEE International Requirements Engineering Conference, RE 2008*, 33–42, IEEE Computer Society, 2008.
[16] I. Rodgers, Internal controls best practices. In: A. Tarantino, editor, *Governance, Risk, and Compliance Handbook*, John Wiley, 2008, 301–323.
[17] J.W. Ross, P. Weill, and D. C. Robertson, *Enterprise Architecture as Strategy: Creating a Foundation for Business Execution*, Harward Business School Press, Boston, 2006.
[18] M. Silverman, *Compliance Management for Public, Private, or Nonprofit Organizations*, McGraw-Hill, New York, 2008.
[19] J. Sowa and J. Zachman, Extending and formalizing the framework for information systems architecture, *IBM Systems Journal* **31**(3) (1992), 590–616.
[20] A. Tarantino, editor, *Governance, Risk, and Compliance Handbook: Technology, Finance, Environmental, and International Guidance and Best Practices*. Introduction, 1–37. Operational risk management in financial services, 233–256. John Wiley, 2008.
[21] A. Turing, Computing machinery and intelligence, *Mind* **59** (October 1950), 433–460.