Integrating Regulatory Requirements into Information Systems Design and Implementation

Research-in-Progress

Joerg Becker
University of Muenster
Leonardo-Campus 3,
48149 Muenster, Germany
becker@ercis.uni-muenster.de

Marcel Heddier
University of Muenster
Leonardo-Campus 3,
48149 Muenster, Germany
marcel.heddier@ercis.uni-muenster.de

Sebastian Braeuer
University of Muenster
Leonardo-Campus 3,
48149 Muenster, Germany
sebastian.braeuer@ercis.uni-muenster.de

Ralf Knackstedt
University of Hildesheim
Marienburger Platz 22,
31141 Hildesheim, Germany
ralf.knackstedt@uni-hildesheim.de

Abstract

Information systems for supporting collaborative activities (e.g., business process modeling tools, workflow management tools, or project management tools) should explicitly address regulatory requirements. Integrating a legal perspective into such information systems would contribute to a better legal compliance and a more effective collaboration. This paper introduces a meta-design for integrating regulations into the design and implementation of information systems. A prototypical implementation of the meta-design shows its general applicability.

Keywords: Regulatory Compliance, Meta-Design, IT Governance, Software Prototype

Introduction

Regulation and the related regulatory compliance risks are getting more and more important for companies who are involved in (global) collaborations. To reduce the risks, collaborations are often subject to severe regulation. Collaboration contracts, e.g., define the specific circumstances and details of inter-organizational collaborations (Giannoccaro and Pontrandolfo 2009; Mowery et al. 1996). They can ensure objectivity, certainty, and consistency (Jamieson 2010). Besides contracts, there are also other (legal) regulations that companies have to comply with. This is, e.g., relevant (regional) law, relevant jurisdiction, or internal company guidelines, policies, and agreements. From an IT perspective, this means that companies should adequately integrate regulatory requirements into the design and implementation of their respective information systems.

A lot of information systems (IS) research has been conducted on IS that support inter-organizational and intra-organizational collaborative activities, like collaboration platforms (e.g Carstensen and Schmidt 1999; Ellis et al. 1991), knowledge management systems (e.g. Alavi and Leidner 2001; Bhatt 2001), or workflow management systems (e.g. Leymann and Roller 2000; van der Aalst and van Hee 2004). Numerous software tools were developed to support these activities. There is tool support for the design and modeling of collaborative activities (e.g., modeling tools or project management tools) as well as support for the execution and control of collaborative activities (e.g., workflow management tools, enterprise wikis, or computer-supported cooperative work (CSCW) systems). However, these tools often insufficiently (and mostly just implicitly) address the regulatory requirements of collaborative activities.

The goal of our paper is to provide a concept for integrating the legal (regulatory) perspective and the IS perspective of IT-supported collaboration. An insufficient alignment of these perspectives may lead to the following problems: Inter-organizational collaboration may suffer or even fail if certain issues are either not or insufficiently treated in a collaboration contract (Berger-Walliser et al. 2011). Therefore, it is crucial that the legal perspective is considered as early as the design phase of collaborative activities. Agile principles for software development or modern interpretations of information systems such as Yoo's concept of digitally enabled generativity (i.e. information systems are designed in an open-ended fashion to support beforehand unforeseen applications (Yoo 2012)) demand an iterative design and implementation process that is accompanied by constant adjustments or clarifications of the legal perspective.

Inappropriate or insufficient legal and contractual regulation may reduce trust and enhance risks in collaborative situations (Lui and Ngo 2004; Malhotra 2009). If the legal imperative is not transparent and comprehensible for a person who is actually conducting collaborative activities, these activities may be executed differently than prescribed, which may violate inter-organizational contractual agreements or intra-organizational service-level agreements, guidelines, or policies. Legal consequences like contractual penalties may result from such behavior.

In this paper, we follow a design science approach and develop a concept (called meta-design) and a software prototype that serve as structure and guidance for the development or extension of information systems in order to address regulatory requirements during the design (e.g., with modeling tools) and/or execution (e.g., with workflow management tools) of collaborative activities. Such systems should be able to support interdisciplinary collaboration, since there are actors from different disciplines (e.g., managers, engineers, or lawyers) involved in designing and implementing a collaboration (Argyres and Mayer 2007). Of course, as already Parnas et al. note, the sole existence and even the wide dissemination of such a concept will not lead to an automatic adaption in practice, since actual IS development projects seldom follow a (fully) rational process, although such a process might be faked (Parnas and Clements 1986). Consequently, suggestions for the integration of this concept in an overall development process with defined outcomes and a detailed evaluation are still required.

The remainder of the paper is structured as follows: We present related work on the integration of legal regulations and (collaborative) information systems. After this, we introduce a concept (meta-design) for integrating legal regulations and IS for designing and executing collaborative activities. To demonstrate the applicability of the meta-design, we used the meta-design as a basis for a prototypical software implementation. Finally, we discuss our contribution to research and practice and the need for further research (which is basically the evaluation of the meta-design).

Related Work

The idea to integrate a legal perspective into information systems (design) is addressed by several research approaches in IS and mainly discussed in the context of Business Process Compliance (BPC). BPC comprises approaches that integrate legal requirements and business process management so that the design and execution of the respective business processes is legally compliant. Kharbili (2012) provides an overview on existing regulatory compliance management frameworks and derives a set of criteria that should be met by BPC solutions. His work shows that most approaches in this area focus on modeling, verification, and formal semantics of business processes and compliance rules. However, they often have unsatisfactory business user orientation (e.g., 'user-friendliness') and are often hardly applicable in domains other than business process management (e.g., collaboration design) (Kharbili 2012). Sadiq et al. (2007) introduce the term 'compliance by design' in order to describe how to ensure compliance as early as the design of business processes. The term describes "the ability to capture compliance requirements through a generic requirements modeling framework, and subsequently facilitate the propagation of these requirements into business process models and enterprise applications" (Sadiq et al. 2007, p. 161).

Other BPC approaches address the regulatory perspective of single aspects of business process management, such as process modeling methods or the formalization of processes and compliance rules. Governatori and Rotolo (2010) develop a formal language for describing business processes including regulations (normative constraints) and propose an automated approach for compliance checking. Speck et al. (2011) propose an extension of the Event Process Chain (EPC) modeling language with Computational Tree Logic (CTL) in order to specify rules and constraints for business processes. This can be used to define business rules and (legal) regulations for business processes. In business process management systems, legal compliance can, e.g., be ensured with semantic constraints (Ly et al. 2009). Zur Muehlen et al. (2007) use business rule modeling languages (e.g., SRML) for the definition of compliance rules. Witt et al. (2012) show how to include and validate privacy regulations in process models. Becker et al. (2011a) propose to combine pattern-checking approaches with the Semantic Business Process Modeling Language (SBPML) in order to allow for automated compliance checking in process models. Contract management systems are designed to manage contractual rights and obligations and to integrate them into business processes (Chieu et al. 2007).

The integration of regulations and business on a more holistic level is also discussed under the headword(s) Governance, Risk, and Compliance (GRC). Racz et al. (2010) provide a literature review on the different aspects of this research area. Becht et al. (2003) elaborate on regulations and corporate governance, taking different actors and different organizations into account.

Some approaches in IS research address the integration of legal requirements into modeling methods other than business process modeling methods. For example, Siena et al. (2009) develop a goal-oriented modeling technique to model legal requirements and to integrate them into IS design. Ghanavati et al. (2009) are using a goal-oriented modeling approach to model the impact of legal regulations on business goals in a healthcare environment. Mahler (2009) proposes a (modeling) method for legal risk management in order to improve "the quality for proactive contractual decisions" (Mahler 2009, p. 9). Knackstedt et al. (2012) develop an extension to a data-warehouse modeling language to incorporate legal requirements into report design and generation.

Overall, a lot of research has been conducted on the integration of regulations into business process-related information systems (design). In the context of collaboration information systems (design) in particular an integrated perspective on regulation and collaboration is beneficial. Further, we think that the consideration of legal regulations during the transition between design and execution (e.g., business process modeling and workflow-management) should receive more attention. Lastly, most approaches pursue the goal of formalizing regulations and processes in order to enable automated compliance checking. We think that especially in collaborative environments with actors from different disciplines (e.g., business, information systems, or law), the aspects of comprehensibility, communication, and target-group-specific functionality are getting more relevant. Therefore, we assume that the development of a generic concept for integrating regulations into information systems (design) would contribute to the field. It would provide structure and guidance for designing and extending collaborative IS with respect to legal regulations.

A Meta-Design for Integrating Regulations and Collaborative Information Systems

The Methodology

We use the concept of a meta-design for structuring and presenting our results. A meta-design on a meta-level "provides a philosophy – a set of beliefs and guidelines – that helps to select appropriate methods and procedures. It is substantiated by theoretical insights and by concrete empirical examples" (Fischer and Herrmann 2011, p. 7). A meta-design is appropriate because it "is focused on objectives, techniques and processes to allow users to act as designers. In doing so, it does not provide fixed solutions but frameworks within which all stakeholders can contribute to the development of technical functionality" (Fischer and Herrmann 2011, p. 1).

Based on previous literature, we develop a meta-design that describes how the regulatory perspective and information systems can be integrated. The proposed meta-design provides structure and guidance for the design and implementation of information systems that comply with legal regulations. Thereby, it can be applied for inter- and intra-organizational collaboration, for different types of information systems (CSCW systems, BPM systems, knowledge management systems, groupware systems, etc.), and for different types of collaboration (structured process, semi-structured process, ad-hoc).

In our research, we draw from the design science guidelines from Hevner et al. (2004). The proposed meta-design, thereby, constitutes the first IT artifact, which addresses the problems that come along when neglecting the legal perspective in collaborative information systems. The actual prototypical implementation constitutes the second IT artifact. In the next step of this research, both artifacts, the meta-design and the software-prototype, need to be evaluated (especially in a practical application scenario) and communicated.

The Meta-Design

The meta-design is depicted as a four-field matrix (Figure 1). The first dimension of this matrix is the layer of information systems development, which can either be the design layer or the implementation layer. The second dimension of this matrix is the direction of the perspective on integrating legal aspects into information systems. The perspective is either starting from a regulation and looking at an IS element, or starting from an IS element and looking at a regulation. The transition from IS design to IS implementation is depicted as a model-instance-bridge. The four cells of the matrix, therefore, describe four different viewpoints that should be considered by information systems in order to also account for the regulatory perspective. In the following, the dimensions and the fields of the meta-design are developed and supported by literature.

The Information Systems Development Layer

The meta-design distinguishes between the design layer and the implementation layer of information systems. This distinction is a reduced view on existing layer models in this context. The ARIS architecture of integrated information systems, e.g., distinguishes between 'Concept', 'Data Processing Concept', and 'Implementation' layer (Scheer 1992). The two layers of our meta-design correspond with the 'Concept' and 'Implementation' layers of the ARIS architecture. The (unmodified) waterfall model roughly distinguishes between the layers 'Requirements', 'Design', 'Implementation', 'Verification', and 'Maintenance' (Royce 1970). 'Requirements' and 'Design' fall within the design layer of our meta-design. 'Implementation', 'Verification', and 'Maintenance' belong to the implementation layer of the meta-design. In more modern agile software development approaches, e.g. Scrum or Extreme Programming, a strongly iterative development process leads to a short-cycled repetitive pass through these layers and allows for a much faster reaction to altered (legal) requirements and an earlier determination of their impact (Highsmith and Cockburn 2001).

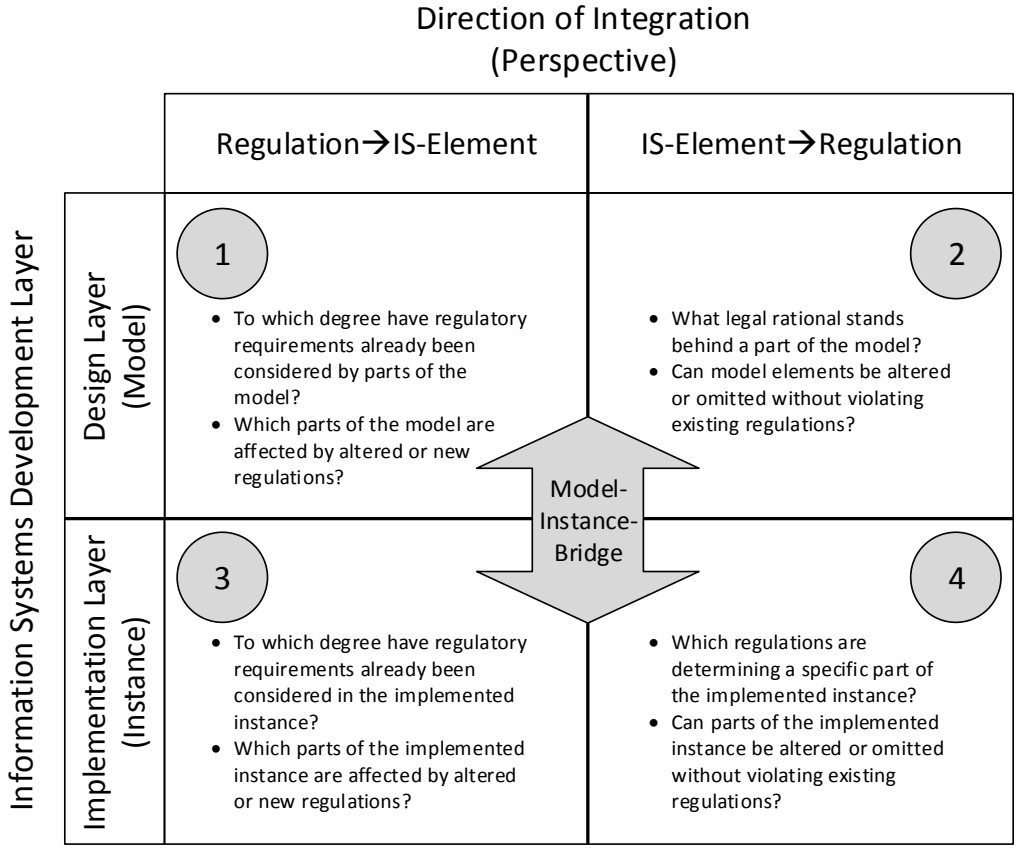

Figure 1. Meta-design for integrating legal regulations into collaborative information systems

The design layer contains conceptual modeling techniques and modeling tools that allow for a conceptual design of, e.g., processes, collaboration designs (structures), document structures, or even legal requirements. Conceptual modeling is a well-established field in IS research (e.g., Wand and Weber 2002) and practice (e.g., Fettke 2009).

The implementation layer contains the implemented instance of a designed model. In case of a process model, this could be the implementation of the process in a workflow management tool. In case of a (model of a) project plan, this could be the implementation of the project plan in a project management tool. In case of a (model of a) document structure, this could be the implementation of the structure in a document management tool, and so on. The implementation layer also contains concrete instances of implementations (e.g., a concrete workflow instance in a workflow management tool).

The Direction of Integration (Perspective)

The second dimension of the meta-design is the perspective on the integration of legal regulations and information systems elements. Thereby, we distinguish between two different viewpoints. The first perspective is called Regulation→IS-Element and the second perspective is called IS-Element→Regulation. We define 'Regulation' as a generic concept for laws and legal regulations, contractual rights and obligations, internal or external guidelines and policies, etc. Likewise, we define 'IS-Element' as a generic concept for elements that are part of (collaborative) information systems like process models, process model elements, document structures, documents, information flows, chat rooms, project plans, wiki entries, etc.

The perspective Regulation→IS-Element means starting from a regulatory element and looking at related IS elements at the design and implementation layer (①+③). For a specific regulation (e.g., a concrete paragraph in data protection law) or for a specific set of rules (e.g., a contractual agreement), it is of interest what IS elements are affected by this and how these elements should be designed to comply with the regulation(s). For example, a company's internal privacy policy could affect the design and implementation of the company's internal wiki by prohibiting the use of the employees' full names in the wiki. Furthermore, it is of interest what regulations are already considered in the design and implementation of an information system and which regulations still need to be implemented. If a regulatory element changes (e.g., an amendment of existing data protection law or a contractual amendment), it is of interest which IS elements are affected by this change. Such questions basically come from the (research) field of (business process) compliance (e.g., Awad et al. 2010; Damianides 2005; Ly et al. 2009; Rinderle-Ma et al. 2008; Smith et al. 2010). The need for deriving business process definitions from contractual agreements is also stated, e.g., by Kabilan (2005) and Karlapalem et al. (2001). The proposed meta-design can be applied not only for business processes but also for other elements that are relevant in collaborative situations.

The perspective IS-Element→Regulation means starting from a IS element and looking at the related regulatory elements at the design and implementation layer (②+④). Here, it is of interest what regulations are relevant for a specific IS element (e.g., a model element of a project plan at the design layer or the implementation of a workflow activity at the implementation layer). It is further of interest what (legal) rationale is behind certain IS elements. If an IS element is going to be changed or omitted, it is important to know if there are (legal) regulations that might prevent such changes or impose restrictions to such changes (or maybe even should be modified because of such changes). Such questions also mainly come from the (research) field of business process compliance but are also highly relevant questions in business process reengineering projects (e.g., Becker et al. 2011b). In such projects, it is crucial to monitor the changing environment continuously (which also includes the legal situation) and to adapt business processes accordingly. Again, the generic nature of the proposed meta-design is not only covering business process elements but also other types of IS elements.

The Model-Instance-Bridge

Finally, the proposed meta-design addresses the consistency of design layer and implementation layer. However, researchers and practitioners have noted that an increasing complexity at the implementation layer draws the attention away from solving the actual business problem (design layer) towards solving problems with software code (implementation layer) (e.g. Kleppe et al. 2003). As a consequence, the relation between the originally recorded requirements specification and the finally produced output is often lost in the course of a project. Model-driven software development (MDSD, or just MDD for model-driven development) approaches such as the standardized model-driven architecture (MDA) framework (Frankel 2003; Kleppe et al. 2003), model-integrated computing (Gokhale et al. 2002; Sztipanovits and Karsai 1997), and software factories (Greenfield and Short 2003) propose possible ways for bridging the gaps between model and implementation (instance).

The model-instance-bridge of the meta-design suggests using such approaches to ensure consistency of design and implementation (in both directions!) and to allow for focusing on the design aspects of 'regulation-enhanced' collaborative information systems. According to Kleppe et al. (2003), such an approach should include three desired features: 'Tunability' (Transformation rules should allow a parameterization for ensuring a higher flexibility and individualization without completely redefining the transformation rule), 'Traceability' (It should be possible to trace back elements of the target model or software code to the source model), and 'Incremental Consistency' (This feature demands the persistence of information, i.e. that information that has been added to a target model or software code should still be available after altering the source model and after regenerating the target model or software code). In addition, some researchers have proposed even stricter requirements for transformations to allow a bidirectional model synchronization, i.e., that changes to the model or code are automatically forwarded to the other layer (Angyal et al. 2007; Giese and Wagner 2009), which is also referred to as round-trip engineering (Antkiewicz and Czarnecki 2006; Hettel et al. 2008).

Application of the Meta-Design

Figure 2 shows one possibility of how the meta-design can be instantiated by an actual implementation. This implementation proposal addresses a collaboration modeling tool on the design layer and a customized enterprise wiki with project management functionalities on the implementation layer. The proposal covers all four viewpoints of the meta-design and the model-instance-bridge. It is of course possible to create alternative instantiations of the meta-design with different tools on the design and implementation layer. The concrete implementation of the four viewpoints would then differ.

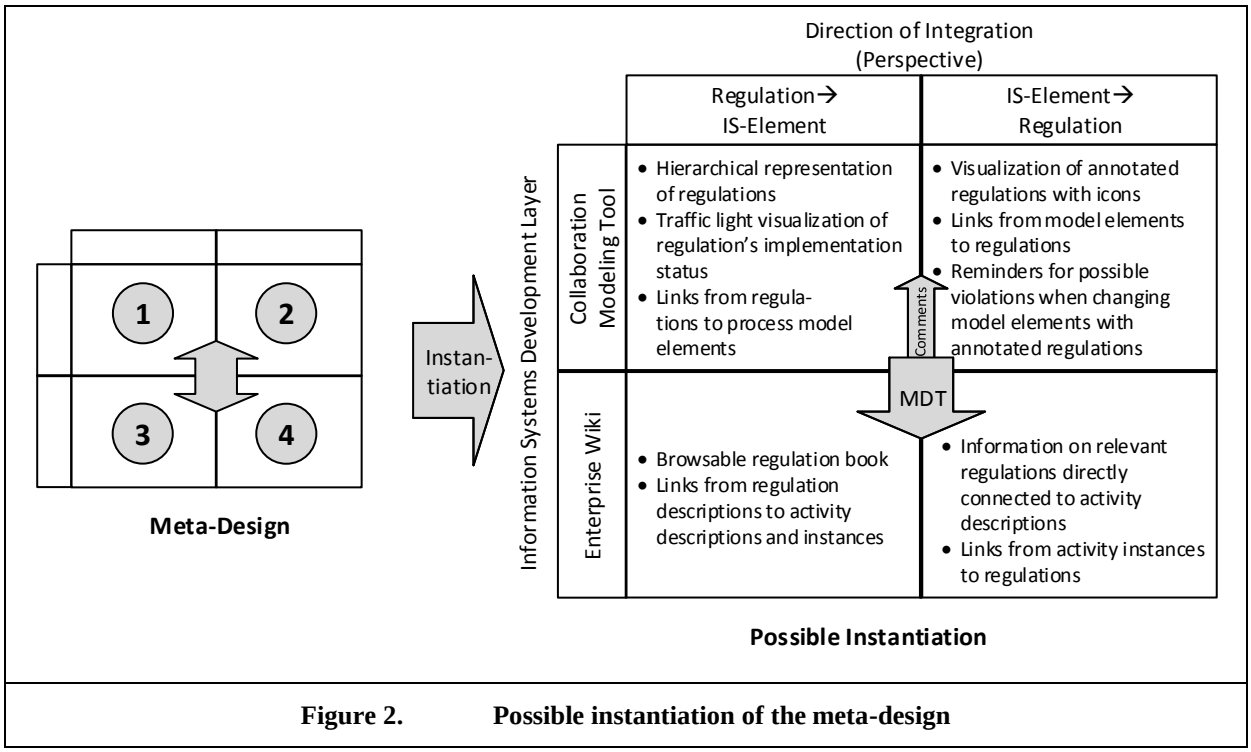

Figure 2. Possible instantiation of the meta-design

For fulfilling the requirements of the perspective Regulation→IS-Element on the design layer (①), a process manager needs to be provided with an easy access to relevant regulations during the design of the model. We have covered this requirement by including a module in our modeling tool that allows to define hierarchical structures, e.g., for representing sales agreements, non-disclosure agreements or service-level agreements as part of service contracts (Figure 3, screenshot A). Additionally, for monitoring the status of regulations, e.g., their up-to-dateness, we have included color coded flags as status symbols next to groups of regulations or single regulations (Figure 3, screenshot B). A grey flag, e.g., highlights outdated regulations. For allowing the process manager to recognize those model elements that are affected by a regulation, it is possible to link model elements to rules (Figure 3, screenshot B, 'Assignment').

When considering the perspective IS-Element→Regulation on the design layer (②), it is required that the process manager is informed about model elements with annotated regulations directly in the model. For this purpose, we have included a small icon next to all accordingly affected model elements. Clicking on this icon directly forwards the user to the annotated regulation and its description (Figure 3, screenshot C). Furthermore, if the process manager tries to change a model element with annotated regulations, a pop-up window informs him or her about a possible regulatory compliance violation and provides cross references to the respective regulations.

For reducing the gap between the process manager's specifications on the design layer and the final implementation in the target application, a model-instance-bridge is used that holds the opportunity to perform instantiations of the model in a (semi-)automated way and ensures consistency between design and implementation layer. In our current implementation, the model-instance-bridge is implemented as a

model-driven transformation (MDT) by following the general principle of the MDA framework. When exporting a model, a web service is contacted that translates the model into wiki code that is then used to update the structures and contents of the wiki. Inverse transformations from the wiki to the model are currently supported in the form of predefined commentary boxes that allow a wiki user to inform the process manager about required changes to the model.

Regarding the perspective Regulation→IS-Element on the implementation layer (③), it is important to keep employees informed about relevant regulations (e.g., for ensuring compliant behavior or just as a source of information). In our exemplary implementation, we consequently mirror the regulation hierarchies and contents in the enterprise wiki by creating interlinked and thus explorable wiki structures. Further, links are included that forward the user to the description pages for the affected model elements or the actual model element's instance in a concrete project.

At the perspective IS-Element→Regulation on the implementation layer (④), when a wiki user, e.g., executes a task and performs modifications to a process instance, our implementation ensures that regulatory requirements can be directly viewed and accessed by the respective user. Following the generated links, the wiki user is able to retrieve further information on relevant regulation elements.

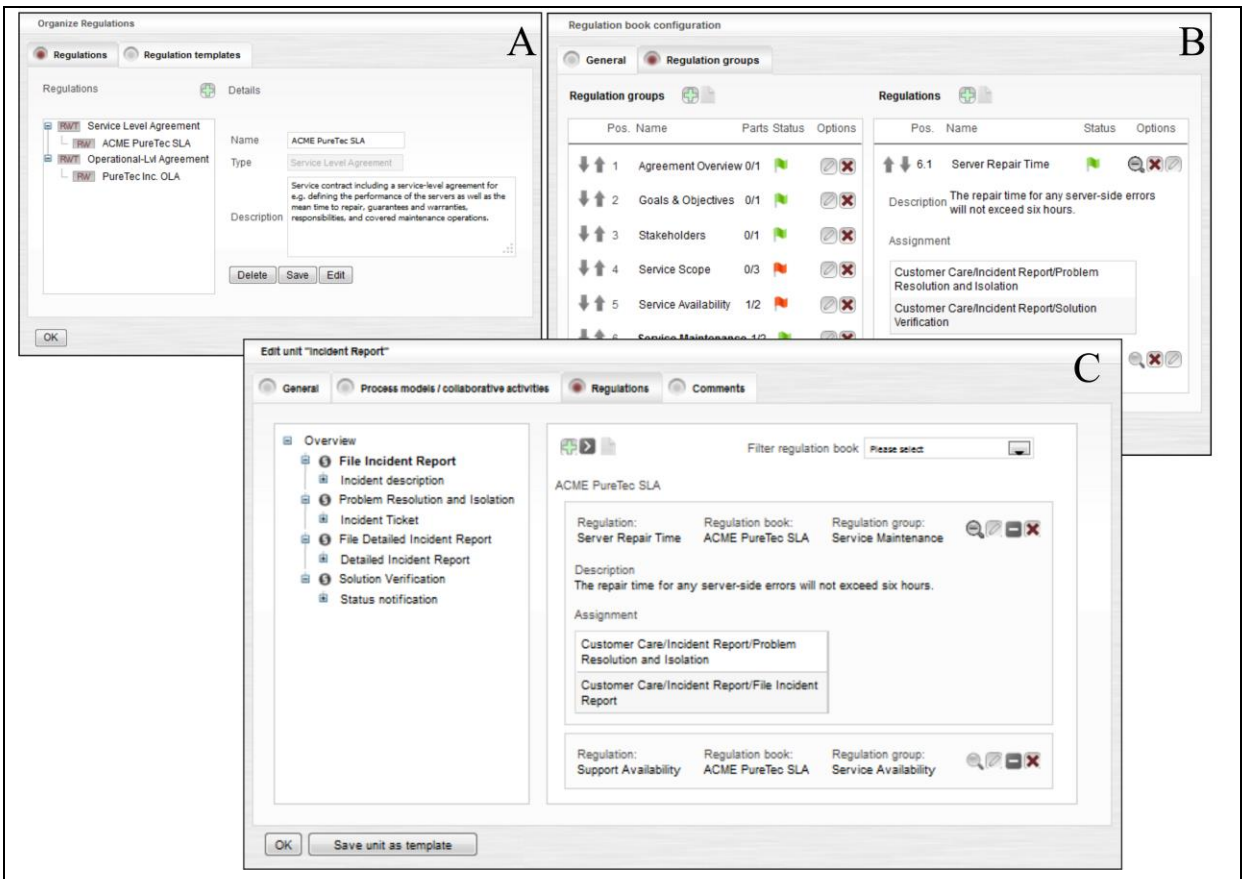

Figure 3. Regulation management (A, B) and annotation of regulations (C) within the modeling tool

Conclusion and Outlook

Of course, we cannot raise the claim that the presented concept forms a universal remedy to solve a plethora of compliance issues in IS design and implementation, many of them caused by the characteristics of IS design and development projects and the nature of the human being themselves. However, the proposed meta-design may serve as structure and guidance for IS designers who aim to explicitly integrate a legal perspective into information systems. Thus, it is ensured that all four perspectives as well as the model-instance-bridge are considered in the solution. A concrete implementation of functionality is not in the scope of the meta-design, because this depends on the specific implementation scenario.

By applying the meta-design to an actual software implementation we demonstrated its applicability for a specific use case (design with a collaboration modeling tool; implementation with an enterprise wiki). The meta-design may also be instantiated by using other platforms (e.g., workflow management tools, document management tools, CSCW tools, etc.). It may even be applicable for other classes of information systems (e.g., the integration of a legal perspective into data warehouse design and implementation; especially data protection law) but the feasibility and relevance of doing so need to be further examined. The proposed meta-design now needs further evaluation (especially in practice). This constitutes our next step in this line of research.

Besides evaluating the meta-design, it should be the task of future research to advance general interdisciplinary research at the intersection of IS and law in order to better understand the specific requirements of both, the legal and the IS perspective. Usually, legal experts communicate with the respective information systems experts on a textual basis and in a legal jargon. This may lead to typical problems of interdisciplinary communication, like misunderstandings or high coordination efforts. Information systems that result from the application of the meta-design (and, therefore, integrate both perspectives) may provide a basis for interdisciplinary communication between the legal experts and IS or business experts. For this paper it means to also evaluate the meta-design from a jurisprudential perspective. This would contribute to the interdisciplinary body of research in the field of IS and law.

References

Alavi, M., and Leidner, D.E. 2001. "Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues," *MIS Quarterly* (25:1), pp. 107-136.

Angyal, L., Lengyel, L., and Charaf, H. 2007. "Novel Techniques for Model-Code Synchronization," *ECEASST* (8).

Antkiewicz, M., and Czarnecki, K. 2006. "Framework-Specific Modeling Languages with Round-Trip Engineering," in: *21th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, OOPSLA 2006*, P.L. Tarr and W.R. Cook (eds.). Portland, Oregon, USA: ACM, pp. 692-706.

Argyres, N., and Mayer, K.J. 2007. "Contract Design as a Firm Capability: An Integration of Learning and Transaction Cost Perspectives," *The Academy of Management Review* (32:4), pp. 1060-1077.

Awad, A., Weidlich, M., and Weske, M. 2010. "Consistency Checking of Compliance Rules," *Business Information Systems - Lecture Notes in Business Information Processing* (47:4), pp. 106-118.

Becht, M., Bolton, P., and Roell, A. 2003. "Corporate Governance and Control," in *Handbook of the Economics of Finance*, G.M. Constantinides, M. Harris and R.M. Stulz (eds.). Elsevier, pp. 1-109.

Becker, J., Bergener, P., Delfmann, P., Eggert, M., and Weiss, B. 2011a. "Supporting Business Process Compliance in Financial Institutions – a Model-Driven Approach," *Internationale Tagung Wirtschaftsinformatik (WI2011)*, Zurich, Switzerland.

Becker, J., Kugeler, M., and Rosemann, M. (eds.). 2011b. *Process Management*. Berlin: Springer.

Berger-Walliser, G., Bird, R.C., and Haapio, H. 2011. "Promoting Business Success through Contract Visualization," *Journal of Law, Business, and Ethics* (17:1).

Bhatt, G. 2001. "Knowledge Management in Organizations: Examining the Interaction between Technologies, Techniques, and People," *Journal of Knowledge Management* (5:1), pp. 68-75.

Carstensen, P., and Schmidt, K. 1999. "Computer Supported Cooperative Work: New Challenges to Systems Design."

Chieu, T.C., Nguyen, T., Maradugu, S., and Kwok, T. 2007. "An Enterprise Electronic Contract Management System Based on Service-Oriented Architecture " *IEEE International Conference on Services Computing*, pp. 613-320.

Damianides, M. 2005. "Sabanes-Oxley and It Governance: New Guidance on It Control and Compliance," *Information Systems Management* (Winter 2005), pp. 77-85.

Ellis, C.A., Gibbs, S.J., and Rein, G. 1991. "Groupware: Some Issues and Experiences," *Commun. ACM* (34:1), pp. 39-58.

Fettke, P. 2009. "How Conceptual Modeling Is Used," *Communications of the Association for Information Systems* (25:43), pp. 571-592.

Fischer, G., and Herrmann, T. 2011. "Socio-Technical Systems - a Meta-Design Perspective," *International Journal for Sociotechnology and Knowledge Development* (3:1), pp. 1-33.

Frankel, D.S. 2003. *Model Driven Architecture : Applying Mda to Enterprise Computing*. Wiley: Indianapolis, Ind.

Ghanavati, S., Siena, A., Perini, A., Amyot, D., Peyton, L., and Susi, A. 2009. "A Legal Perspective an Business: Modeling the Impact of Law," *Lecture Notes in Business Information Processing* (26:1), pp. 267-278.

Giannoccaro, I., and Pontrandolfo, P. 2009. "Negotiation of the Revenue Sharing Contract: An Agent-Based Systems Approach," *International Journal of Production Economics* (122:1), pp. 558-566.

Giese, H., and Wagner, R. 2009. "From Model Transformation to Incremental Bidirectional Model Synchronization," *Software & Systems Modeling* (8:1), 2009/02/01, pp. 21-43.

Gokhale, A., Schmidt, D.C., Natarajan, B., and Wang, N. 2002. "Applying Model-Integrated Computing to Component Middleware and Enterprise Applications," *Commun. ACM* (45:10), pp. 65-70.

Governatori, G., and Rotolo, A. 2010. "A Conceptually Rich Model of Business Process Compliance," *APCCM*, Brisbane.

Greenfield, J., and Short, K. 2003. "Software Factories: Assembling Applications with Patterns, Models, Frameworks and Tools," in: *Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*. Anaheim, CA, USA: ACM, pp. 16-27.

Hettel, T., Lawley, M., and Raymond, K. 2008. "Model Synchronisation: Definitions for Round-Trip Engineering," in *Theory and Practice of Model Transformations*, A. Vallecillo, J. Gray and A. Pierantonio (eds.). Springer Berlin Heidelberg, pp. 31-45.

Hevner, A.R., March, S.T., Park, J., and Ram, S. 2004. "Design Science in Information Systems Research," *MIS Quarterly* (28:1), pp. 75-105.

Highsmith, J., and Cockburn, A. 2001. "Agile Software Development: The Business of Innovation," *Computer* (34:9), pp. 120-127.

Jamieson, N.J. 2010. "Codes, Contracts, and Commerce: The Contractual Mistakes Act Part Ii: Releasing More Light Than Heat," *Statute Law Review* (31:2), pp. 107-150.

Kabilan, V. 2005. "Contract Workflow Model Patterns Using Bpmn," *Proceedings of the 10th International Workshop on Exploring Modeling Methods in Systems Analysis and Design (EMMSAD 05)*, Porto, Portugal, pp. 557-568.

Karlapalem, K., Dani, A.D., and Krishna, P.R. 2001. "A Frame Work for Modeling Electronic Contracts," *Proceedings of the 20th International Conference on Conceptual Modeling: Conceptual Modeling* H.S. Kunii, S. Jajodia and A. Sølvberg (eds.), pp. 193-207.

Kharbili, M.E. 2012. "Business Process Regulatory Compliance Management Solution Frameworks: A Comparative Evaluation," *Asia-Pacic Conference on Conceptual Modelling*, A. Ghose and F. Ferrarotti (eds.), Melbourne, Australia.

Kleppe, A.G., Warmer, J., and Bast, W. 2003. *Mda Explained: The Model Driven Architecture: Practice and Promise*. Addison-Wesley Longman Publishing Co., Inc.

Knackstedt, R., Eggert, M., and Fleischer, S. 2012. "The Legal Perspective on Business to Government Reporting – a Conceptual Modeling Approach and Its Application in the Financial Sector," *45th Hawaii International Conference on System Sciences (HICSS 2012)*, Maui, Hawaii, USA.

Leymann, F., and Roller, D. 2000. *Production Workflow: Concepts and Techniques*. Prentice Hall PTR.

Lui, S.S., and Ngo, H.-Y. 2004. "The Role of Trust and Contractual Safeguards on Cooperation in Non-Equity Alliances," *Journal of Management* (30:4), pp. 471-485.

Ly, L.T., Rinderle-Ma, S., Göser, K., and Dadam, P. 2009. "On Enabling Integrated Process Compliance with Semantic Constraints in Process Management Systems," *Information Systems Frontiers*), pp. 1-25.

Mahler, T. 2009. "Visualising Legal Risk," *Jusletter IT* (1st September 2009).

Malhotra, D. 2009. "When Contracts Destroy Trust," *Harvard Business Review* (87:5), p. 25.

Mowery, D.C., Oxley, J.E., and Silverman, B.S. 1996. "Strategic Alliances and Interfirm Knowledge Transfer," *Strategic Management Journal* (17:Special Issue: Knowledge and the Firm), pp. 77-91.

Parnas, D.L., and Clements, P.C. 1986. "A Rational Design Process: How and Why to Fake It," *IEEE Trans. Softw. Eng.* (12:2), pp. 251-257.

Racz, N., Weippl, E., and Seufert, A. 2010. "A Frame of Reference for Research of Integrated Governance, Risk and Compliance (Grc)," *Lecture Notes in Computer Science* (6109:1), pp. 106-117.

Rinderle-Ma, S., Ly, L.T., and Dadam, P. 2008. "Business Process Compliance (Aktuelles Schlagwort)," *EMISA Forum* (28:2), pp. 24-29.

Royce, W. 1970. "Managing the Development of Large Software Systems," *Proceedings of IEEE WESCON*, pp. 328-338.

Sadiq, S., Governatori, G., and Namiri, K. 2007. "Modeling Control Objectives for Business Process Compliance," *International conference on business process management*, G. Alonso, P. Dadam and M. Rosemann (eds.): Springer, pp. 149-164.

Scheer, A.W. 1992. *Architecture of Integrated Information Systems: Foundations of Enterprise Modelling*. Berlin: Springer-Verlag.

Siena, A., Mylopoulos, J., Perini, A., and Susi, A. 2009. "Designing Law-Compliant Software Requirements," *International Conference on Conceptual Modeling (ER)*, Gramado, Brazil, pp. 472 - 486.

Smith, S., Winchester, D., Bunker, D., and Jamieson, R. 2010. "Circuits of Power: A Study of Mandated Compliance to an Information Systems Security De Jure Standard in a Government Organization," *MIS Quarterly* (34:3), pp. 463-486.

Speck, A., Feja, S., Witt, S., Pulvermüller, E., and Schulz, M. 2011. "Formalizing Business Process Specifications," *Computer Science and Information Systems* (8:2), pp. 427-446.

Sztipanovits, J., and Karsai, G. 1997. "Model-Integrated Computing," *Computer* (30:4), pp. 110-111.

van der Aalst, W., and van Hee, K. 2004. "Workflow Management - Models, Methods, and Systems." MIT Press.

Wand, Y., and Weber, R. 2002. "Research Commentary: Information Systems and Conceptual Modeling - a Research Agenda," *Information Systems Research* (13:4), pp. 363-376.

Witt, S., Feja, S., Speck, A., and Prietz, C. 2012. "Integrated Privacy Modeling and Validation for Business Process Models," *Joint EDBT/ICDT Workshops* New York, USA, pp. 196-205.

Yoo, Y. 2012. "The Tables Have Turned: How Can the Information Systems Field Contribute to Technology and Innovation Management Research?," *Journal of the Association for Information Systems* (14:5), pp. 227-236.

zur Muehlen, M., Indulska, M., and Kamp, G. 2007. "Business Process and Business Rule Modeling Languages for Compliance Management: A Representational Analysis," *ER '07 Tutorials, posters, panels and industrial contributions at the 26th international conference on Conceptual modeling*, J. Grundy, S. Hartmann, A.H.F. Laender, L. Maciaszek and J.F. Roddick (eds.), pp. 127-132.