

Towards compliance requirements modeling and evaluation of E-government inter-organizational collaborative business processes

Laura González, Andrea Delgado
Instituto de Computación, Facultad de Ingeniería,
Universidad de la República, Uruguay
 {lauragon,adelgado}@fing.edu.uy

Abstract

Business process compliance requirements are becoming increasingly important to organizations as they focus on their processes and the way they are carried out in their daily operation. Being compliant with regulations and laws, among others, is mandatory in an e-government environment where these processes are to be transparent to citizens and subject to public audits. Also, in e-government domain, as these processes are mostly inter-organizational collaborative processes that are spread within several organizations, compliance definition, monitoring and evaluation becomes more complex. In this paper we present an approach to deal with compliance requirements modeling and evaluation for such processes. We focus on modeling compliance requirements over BPMN 2.0 and evaluating this specification against process execution traces, in order to detect compliance violations. We present a Compliance Requirements Modeling Language (CRML) and its connection with BPMN 2.0 elements, a Compliance Requirements Model (CRM) specific for business processes, and an initial view on post-mortem compliance evaluation with process mining.

1. Introduction

Business Process Management (BPM) [1, 2, 3] provides the basis to support the business process lifecycle from modeling, configuration and enactment to evaluation. Modeling business processes (BPs) is a key input for this lifecycle, which is mandatory in organizations where Business Process Management Systems (BPMS) [4] are in place to enact processes. Although there are several languages and approaches for modeling BPs, in organizations with BPMS settings, the preferred one is the Business Process Model and Notation (BPMN 2.0) [5]. Business process compliance requirements are becoming increasingly important to organizations as they focus on their processes and

the way they are carried out in their daily operation. Modeling Compliance Requirements over BPs poses several challenges, since there are few modeling languages for specifying compliance [6], and being the main focus controlling compliance itself, they are usually not completely suitable for modeling complete BPs with extra elements for compliance.

Being compliant with regulations and laws, among others, is mandatory in an e-government environment where these processes are to be transparent to citizens and subject to public audits. Also, as these processes are mostly inter-organizational collaborative processes that are spread within several organizations, compliance definition, monitoring and evaluation becomes more complex. For these inter-organizational collaborative processes we identified mainly two interaction scenarios between participants to consider [7]: (i) closed, in which interactions between organizations are explicitly defined and agreed as collaborative BPs (e.g., e-government domain); or (ii) open in which organizations offer capabilities for integration, not explicitly agreeing on their BPs but mainly on the contract of the capabilities they provide or require to be able to participate in the collaboration. Despite the fact that e-government is a closed environment, the execution of processes can be difficult to trace and to reconstruct between different participant organizations.

This fact can be a barrier when trying to evaluate compliance requirements violations for such processes, based on process execution traces. Process Mining [8] provides the means for: i) discovering business process models from event logs where process execution traces are provided, ii) checking conformance of execution data against an (existing or discovered) process model, and iii) enhancing process models with other execution information such as resources involved. Our approach is based on process mining to evaluate compliance violations using post-mortem process execution data i.e. event logs, but also considering compliance requirements modeling. Although some existing compliance requirements evaluation approaches also

propose to check for compliance violations in a post-mortem way, most of them focus on the compliance monitoring stage [6]. As part of a continuous improvement effort, it is mandatory for organizations to be able to analyze their execution data, detecting improvement opportunities and carrying out improvement efforts in order to provide better process models that are compliant with regulations and laws.

In [9] we defined the Process and Data sCience for oRganizational improvEment (PRICED) framework, to provide support for business process execution data analysis and evaluation with data and process mining techniques, including aspects of data quality and compliance requirements. In this paper we present the approach to deal with compliance requirements modeling and evaluation for e-government inter-organizational collaborative BPs, which is part of the PRICED framework. We focus on modeling compliance requirements over BPMN 2.0 and evaluating this specification against process execution traces, in order to detect compliance violations.

Our main contributions are: i) a metamodel-based Compliance Requirements Modeling Language (CRML), to specify compliance requirements models, and its connection to BPMN 2.0 to specify compliance requirements over process models; ii) a Compliance Requirements Model (CRM) specific for BPs integrating several common existing compliance requirements (compliance patterns), iii) an initial view on post-mortem compliance evaluation with process mining, and iv) tool support for i) as Eclipse plug-ins.

The rest of the document is organized as follows: in Section 2 we briefly present the research methodology we followed, in Section 3 we present the Compliance Requirements approach and in Section 4 the Compliance Requirements Model (CRM) we specified for BPs. In Section 5 we present an example of application and in Section 6 we discuss other proposals related to ours. Finally in Section 7 we draw some conclusions and ongoing work.

2. Research methodology

To address the research we follow Design Science guidelines [10, 11] as research methodology, applying a combination of research methods suited to each research phase. In Design Science, knowledge and understanding of a problem and its solution are based on two main processes: building and assessment (of the application) of an artifact. An artifact can be a methodology, an algorithm, a tool, among others. Given a problem, useful artifacts are built to solve it, and then evaluated regarding their usefulness [10].

For the evaluation of artifacts, and depending on the artifact to be evaluated, approaches such as empirical strategies as case study or experiments [12, 13] can be used, both human-oriented i.e. carried out by humans over the artifacts, or technology-oriented, i.e. on algorithms or tools and their results, among others. We are working with a team from the e-government in our country, the Uruguayan e-Government Agency (Agencia de Gobierno Electrónico y Sociedad de la Información y del Conocimiento, AGESIC)¹ which has real processes and organizational data for us to apply our approach for compliance requirements.

At the beginning of our research, an exhaustive literature review [14] was conducted in order to find existing work on compliance requirements modeling, monitoring and evaluation for business processes, including methodologies, frameworks, languages and tools. As main artifacts generated within this work, we present the general approach for business process compliance requirements (c.f. Section 3.1), the Compliance Requirements Modeling Language (CRML) and corresponding metamodel (c.f. Section 3.2), and the Compliance Requirements Model (CRM) for business processes (c.f. Section 4).

All these artifacts were developed based on an extensive analysis of existing standards and approaches for compliance requirements and business process management, as well as previous works on conceptualization and metamodels for e-government inter-organizational business processes [15], compliance requirements and management [14, 16], as well as BPs execution measurement [17], which were evaluated and validated with real case studies from organizations we worked with.

3. Compliance Requirements approach

In this section we present key elements of our Compliance Requirements approach proposal. We describe the Compliance Requirements Modeling Language (CRML) to specify such requirements over BPs. It is based on previous works where we have presented Compliance Requirements monitoring [14, 16] and BPs Execution measurement [17] approaches, and e-government inter-organizational BPs metamodel [15]. We also present tool support for our approach.

3.1. General Description

Our compliance approach proposal envisions a step by step process and guides for going from compliance

¹<https://www.gub.uy/agencia-gobierno-electronico-sociedad-informacion-conocimiento/>

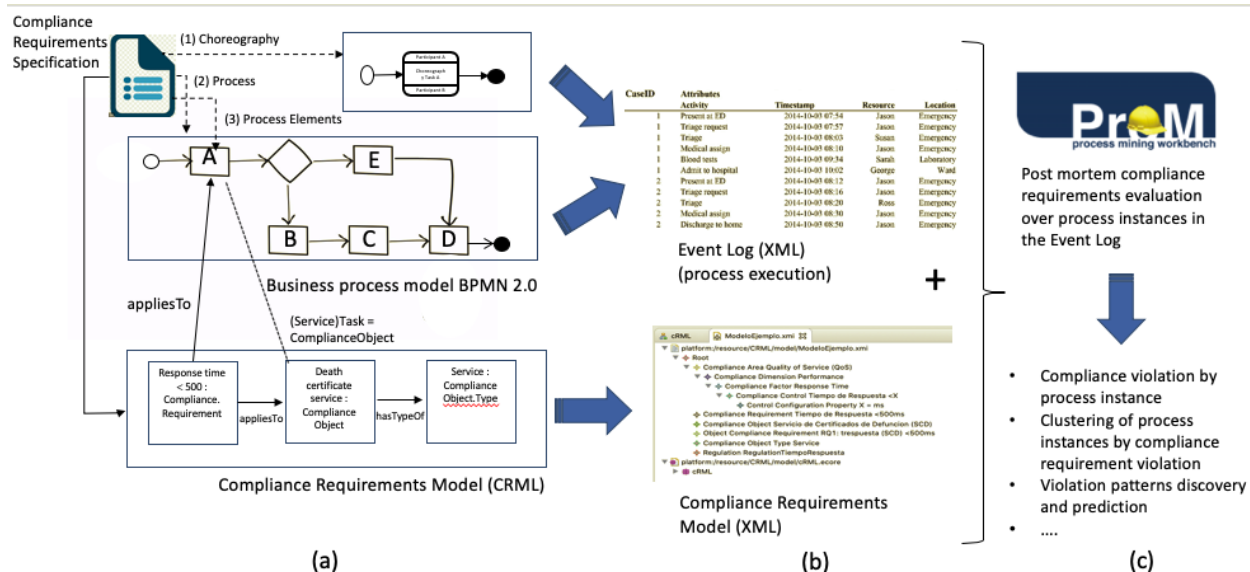


Figure 1. General approach for compliance Requirements modeling and evaluation over Business Processes

requirements specification over business processes to compliance requirements evaluation based on process execution. It is focused on the specification of compliance requirements over business process models (i.e. Design time), the monitoring of compliance requirements over business process execution (i.e. Run time), and the post mortem evaluation of those compliance requirements based on process execution (i.e. Evaluation time). We present the general view on Design and Evaluation, as the Run time proposal was already presented in previous work. In Figure 1 the general compliance approach for modeling and evaluation over BPs is presented.

Figure 1 (a) presents the compliance requirements modeling stage, where both an independent Compliance Requirements Model (CRM), and a business process model extended with compliance requirements information are defined.

As shown on the top left of Figure 1 (a), compliance requirements can be specified over different elements in a business process model. From the Compliance Requirement Specification (CRS) to (1) choreography, (2) process and (3) specific elements. For example, for a collaborative process, we can specify a requirement over the message interactions (choreography), such as *if message B was received by participant 1 from participant 2, then a message A must have been sent first from participant 1 to participant 2*, as shown by arrow (1). A compliance requirement involving only one participant (orchestration), such as *process Cycle time (or Throughput time) must be less than 10 days* can be specified over the complete process, as shown by

arrow (2). Also, compliance requirements over specific elements of the process, such as Service Task or a Business rule, e.g. *Response Time of a Service Task must be less than 500ms*, can be directly specified over the element, as shown by arrow (3).

On the bottom of Figure 1 (a) an example excerpt of the compliance requirements model is shown, from the CRS on the top left of Figure 1 (a), specified in the Compliance Requirement Modeling Language (CRML) we have defined (c.f. Sub Section 3.2). Concepts as Compliance Requirement, Compliance Object and Compliance Object Type are central to the language, and allow us to link the compliance requirements model to the different business process elements as described before. In the example CRM in Figure 1 (a), the requirement *Response Time less than 500ms* is specified over the Service Task corresponding to the Death certificate service (SCD).

Figure 1 (b) shows the collection of business process execution data (orchestration and collaborative processes) in the form of an Event log [8] in the XES format (XML file), and the compliance requirement model specified in CRML (also a XML file). These two files are the input for the evaluation stage shown in Figure 1 (c), where process instances (traces) in the event log are analyzed in a post mortem way against the compliance requirements specified over the BP model, looking for violations. The Process Mining framework ProM² provides the perfect environment for adding a compliance requirements evaluation plug-in.

²<http://www.promtools.org/>

Several analysis can be performed from there, such as clustering of process instances by means of compliance requirement violation e.g. process instances where the requirement *Response Time less than 500ms* specified over the Service Task corresponding to the Death certificate service is violated, and others where this requirement is violated and a second requirement too, and so on. Another analysis that can be performed is discovering violation patterns from the behaviour of the process instances analyzed, helping to identify elements in the process execution that can lead to compliance requirements violations. This can also be used in runtime for monitoring compliance over executing process instances, alerting when a violation pattern is detected.

3.2. Compliance Requirements Modeling Language (CRML)

The Compliance Requirements Modeling Language (CRML) is a Domain Specific Language (DSL) that allows the specification of compliance requirements as shown in Figure 1. CRML is defined by means of a metamodel which was conceptualized in previous works [14] and now formalized with corresponding Abstract and Concrete syntax. Figure 2 presents the metamodel defined with its main concepts and relationships.

First of all, on the left side of Figure 2, it can be seen that Compliance is defined within a `ComplianceArea`, e.g. Quality of Service (QoS) or Business Processes (BPs), which is composed of several `ComplianceDimension`, e.g. Performance or Control Flow, which in turn is composed of several `ComplianceFactor`, e.g. Response Time or Sequence Flow. Then, for each `ComplianceFactor`, several specific `ComplianceControl` can be defined e.g. Response Time less than X, or Message A received before than Message B, providing existing knowledge on compliance specification as part of the Compliance Requirements Model. What is more, CRML provides the means of specifying usual regulations and restrictions that apply over business process by defining compliance requirements patterns (CRPs), in the form of several `ComplianceControl` elements specific for BPs.

A `ComplianceRequirement` is based on a `ComplianceControl`, and can be seen as a specific instantiation (i.e. a specific value) of it over a `ComplianceObject` or a `ComplianceObjectType` e.g. an instance or an element of type service, task, message, process, or choreography, etc. This is a crucial element of our approach, since based on this fact, we were

able to specify a Compliance Requirements Model (CRM) (c.f. Section 4) defining Compliance Areas, Dimensions, Factors and Controls to support the specification of compliance requirements over BPs. A `ComplianceRequirement` originates on an existing `Regulation`, and presents a `ControlConfigurationPropertyValue` which configures a `ControlConfigurationProperty` of a `ComplianceControl`.

As `ComplianceRequirement` are applied to `ComplianceObject` elements, to link the CRML metamodel with the BPMN 2.0 metamodel, we matched the `ComplianceObject` element to the `BaseElement` of BPMN 2.0, in order to enable the specification of `ComplianceRequirement` to several process elements at different levels such as process, choreography, collaboration, message, task, among others. In [9] we introduced this connection in the context of the PRICED framework definition.

The metamodel in Figure 2 presents the Abstract syntax of the language i.e. its concepts and relationships between them, and for the Concrete syntax we defined a visual notation with corresponding elements, which we show with an example model from the CRML modeling plug-in in Section 5.

3.3. Compliance Requirements evaluation

We envision the post-mortem evaluation of compliance requirements based on the event log that contains the process execution traces, and the compliance requirements model which specifies the compliance requirements for the process. Although there are some proposals that also evaluate compliance violations over process execution traces from the event log, there are some extra challenges regarding inter-organizational collaborative processes. For instance, process traces are more complex to build into the event log, since different parts of the process execute in different organizations.

Consider two organizations (A and B) that interact with each other via message exchanges. This interaction can be implemented in several ways (web services, message queue, etc.) and registered in different ways and locations. In an e-government setting ((i) closed interaction scenario), we identified two main sub-scenarios for interactions between processes (choreographies): 1) in a direct way, organization A invokes a service or sends a message directly to organization B, and 2) with an intermediate integration platform (e.g. the Uruguayan setting), organization A invokes organization B via the platform, where metadata of interaction messages are logged.

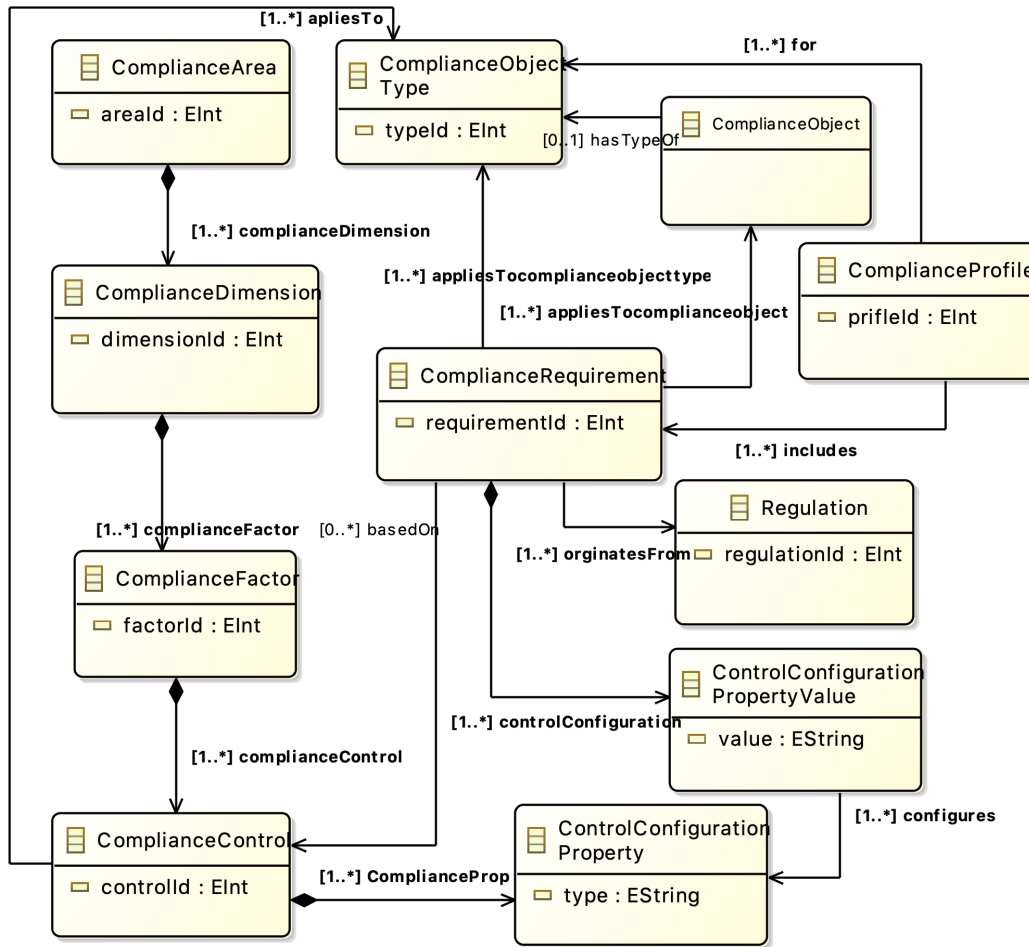


Figure 2. Compliance Requirements Modeling Language (CRML) metamodel

In scenario 1) Organization A will register sending a message to Organization B and receiving a message from Organization B, and organization B will register the same for the messages it sends and receives. In this case, the choreography between them can be reconstructed from each organization's logs (if we can obtain them) and integrating it with each internal process, the collaborative BP can be discovered. In scenario 2) these interactions are registered within an intermediate integration platform so the choreography can be reconstructed from these logs, but it must also be integrated with the records from each internal process to discover the collaborative BP.

We are working on compliance evaluation with traditional event logs for intra-organizational processes, and extending the event log for inter-organizational collaborative processes, to apply the compliance evaluation over other elements such as choreographies.

3.4. Tool support

To support our Compliance Requirements approach we envision two plug-ins for the modeling stage: one we have already developed for creating Compliance Requirements models based on the CRML, and other to define specific Compliance Requirements over BPMN 2.0 models. For the evaluation stage we are working on the ProM plug-in for compliance assessment.

The CRML modeler prototype is an Eclipse plug-in developed with Sirius³ which is an Eclipse project for creating domain-specific modeling workbenches. The CRML plug-in prototype is based on the definitions presented in Figure 2 and the concrete syntax we defined as notation elements, and is available here⁴.

³<https://www.eclipse.org/sirius/>

⁴<https://gitlab.fing.edu.uy/open-coal/priced-crml>

We are also working in a BPMN 2.0 CRML modeler as an extension of the Eclipse BPMN 2.0 modeler⁵ plug-in, to integrate selected elements from the CRML into BPMN 2.0 models, in order to specify the CRML compliance requirements over BPMN 2.0 elements.

4. Compliance Model for BPs

This section proposes a compliance model for business processes, which is based on the metamodel described in Section 3.2 as well as on the compliance perspectives proposed in [18]. The compliance model aims to provide a library of built-in compliance elements in order to support the specification of compliance requirements over business processes.

Section 4.1 presents an excerpt of the model, by providing a high-level view of its dimensions and factors. In addition, Section 4.2 and Section 4.3 describe controls for two factors of the model: Interaction and Time, respectively. Controls are inspired by related work [19][20] and are formally specified using extended compliance rule graphs (eCRG) [18].

4.1. General Description

As shown in Table 1, the model comprises an area (i.e. Business Processes) as well as five dimensions (e.g. control flow, interaction) and twenty factors (e.g. sequence flow, message flow), which correspond to the compliance perspectives and concepts proposed in [18].

The Control Flow dimension comprises compliance aspects related to the occurrence and order of tasks [18]. It includes five factors: Tasks, Sequence Flow, Parallel Flow, Alternative and Exclusive. An example of compliance requirement in this dimension may state that "after the occurrence of a task, another must occur".

The Interaction dimension comprises aspects related to inter-organizational message exchanges [18]. It includes two factors: Send / Receive Messages and Message Flow. An example of compliance requirement in this dimension may state that "if a specific message is exchanged, another message must not be exchanged".

The Time dimension comprises compliance aspects related to time [18]. It includes three factors: Points in Time, Interval and Duration. An example of compliance requirement in this dimension may state that "the temporal distance between two messages must be less than 30 minutes".

The Resources dimension comprises compliance aspects related to the resources which are used within processes [18]. It includes six factors: Roles, Staff Members, Groups, Organizational Units, Resource

Table 1. Compliance Model for Business Processes

Area	Dimension	Factor
Business Processes	Control Flow	Tasks Sequence Flow Parallel Flow Exclusive Alternative
	Interaction	Send / Receive Messages Message Flow
	Time	Points in Time Interval Duration
	Resources	Roles Staff Members Groups Organizational Units Resource Relations Performer Relations
	Data	Data Objects Data Containers Data Relations Data Flow

Relations, and Performer Relations. An example of compliance requirement in this dimension may state that "a given task must be performed by a given role".

The Data dimension comprises compliance aspects related to the data elements used within processes [18]. It includes four factors: Data Objects, Data Containers, Data Relations and Data Flow. An example of compliance requirement in this dimension may state that "a given task must write a data object to a data container".

As mentioned before, in order to support the specification of compliance requirements over different process elements, `ComplianceObject` are set to be high-level elements such as process and choreography, as well as all elements inheriting (directly or indirectly) from `BaseElement` as defined in BPMN 2.0 (e.g. Activity, Choreography Activity and Sequence Flow).

4.2. Interaction Controls

Table 2 presents compliance controls defined for the interaction dimension, which are formally specified with eCRG as shown in Figure 3.

4.3. Time Controls

Table 3 presents compliance controls defined for the time dimension, which are formally specified with eCRG as shown in Figure 4.

⁵<https://www.eclipse.org/bpmn2-modeler/>

Table 2. Controls for Interaction Dimension

Factor	Control	Description	Applies to	Properties
Send / Receive Messages	M exchanged (a)	Message M must be exchanged within the interaction.	Choreography	message name, sender, receiver
	M not exchanged (b)	Message M must not be exchanged within the interaction.	Choreography	message name, sender, receiver
Message Flow	M exchanged after N (c)	Message M must be exchanged after message N.	Choreography	message names, senders, receivers
	M exchanged before N (d)	Message M must be exchanged before message N.	Choreography	message names, senders, receivers

Table 3. Controls for Time Dimension

Factor	Control	Description	Applies to	Properties
Point in Time	M exchanged after D (a)	Message M must be exchanged after date D.	Choreography	message name, sender, receiver, date
	M exchanged before D (b)	Message M must be exchanged before date D.	Choreography	message name, sender, receiver, date
Interval	M exchanged after N within T (c)	Message M must be exchanged after message N within the interval T.	Choreography	message names, senders, receivers, interval
	M and N exchanged with distance less than T (d)	Message M and message N must be exchanged with a temporal distance less than T.	Choreography	message names, senders, receivers, distance

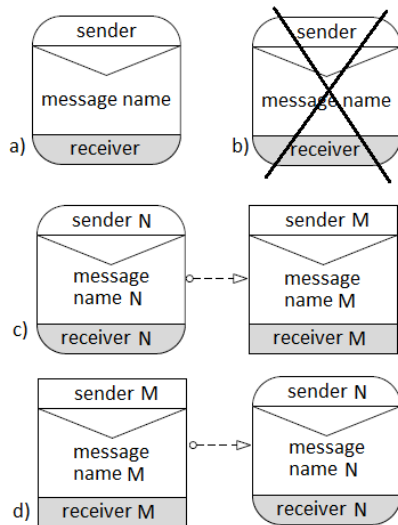


Figure 3. Controls for Interaction Dimension

5. Example of application

This section presents how the proposed approach may be applied in order to deal with compliance issues of collaborative business processes within an e-government scenario. In particular, the example leverages the compliance controls proposed in Section 4 in order to define compliance requirements over a collaborative business process.

5.1. E-government Scenario

The scenario is inspired by the Uruguayan e-Government Interoperability Platform [21], which enable government organizations to offer business services leveraging the web services technology. In this context, two or more organizations may carry out collaborative business processes (CBPs) by using the services available in the platform. For example, a Passport Application CBP would enable citizens to get or renew their passport [16].

The passport CBP is carried out by three government

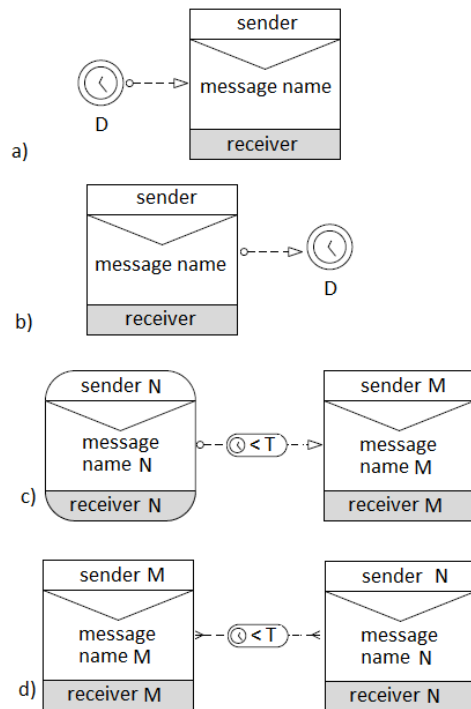


Figure 4. Controls for Time Dimension

organizations: the Technical Police National Directorate (Dirección Nacional de Policía Técnica, DNPT), the Uruguayan e-government agency (AGESIC), and the Civil Identification National Directorate (DNIC). In particular, DNPT returns a judicial records response to DNIC (N) in order to decide if the appointment can be carried out (M) or not.

Figure 5 presents the message exchanges involved in this CBP using a BPMN2 choreography diagram.

5.2. Defining a Compliance Requirement

This section shows the definition of a compliance requirement for the CBP by leveraging the proposed approach. In particular, a requirement stating that the Appointment Result has to be exchanged after the Judicial Records response is going to be defined.

To this end, the compliance control "M exchanged after N" from the Interaction dimension is going to be used. Table 4 presents the definition of this compliance requirement, specifying values for all the configuration properties of the compliance control.

In addition, Figure 6 presents how the requirement is modeled using the CRML language. In particular, the requirement applies to the compliance object Passport Choreography, which is of type Choreography.

Table 4. Compliance Requirement Definition

Name	Appointment Result after Judicial Records
Control	M exchanged after N
message name M	Appointment Result
sender M	DNIC
receiver M	AGESIC
message name N	Judicial Records
sender N	DNPT
receiver N	DNIC
Regulation:	Passport Law

6. Related Work

Different proposals address the specification of compliance requirements in order to control their fulfillment (at design time, runtime or after execution), usually leveraging formal approaches such as the deontic and temporal families of logic [22].

The COMPAS project defines DSLs [23] which enable compliance specialists to specify requirements at a high level of abstraction and hiding technical details. The SeaFlows [24] and C³Pro [25] projects enable the visual specification of compliance requirements by means of compliance rule graphs (CRG) and extended CRG (eCRG), respectively. Other proposals include pattern-based solutions [19] and the concept of compliance scopes for business processes [26].

In addition, policy languages have been proposed aiming at specifying compliance requirements and establishing actions to be performed. Some examples are the CoReL policy language [27] and the Business Contract Language (BLC) [28].

Our approach leverages some of these proposals in order to enable the agile specification of compliance requirements over business processes, by providing a compliance model with reusable controls and associated tools to link them to BPMN process specifications.

Regarding the evaluation of compliance requirements, although there are some proposals which evaluate them against the event log [29] [6], to the best of our knowledge none of them focus on business process models in BPMN 2.0 extended with specific compliance requirements elements.

7. Conclusions and Future Work

In this paper we have presented a proposal towards Compliance Requirements modeling and evaluation for e-government inter-organizational business processes. The approach includes elements for

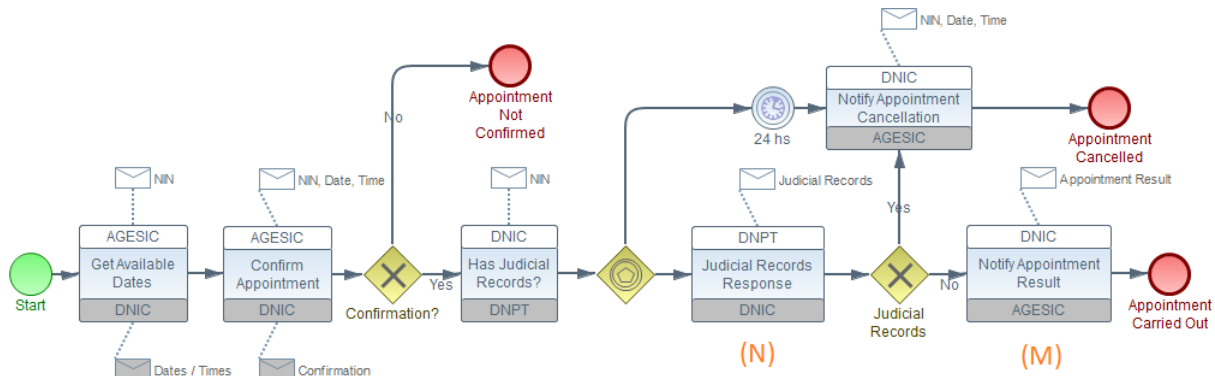


Figure 5. Passport Application Collaborative Business Process (BPMN2 Choreography Diagram) [16]

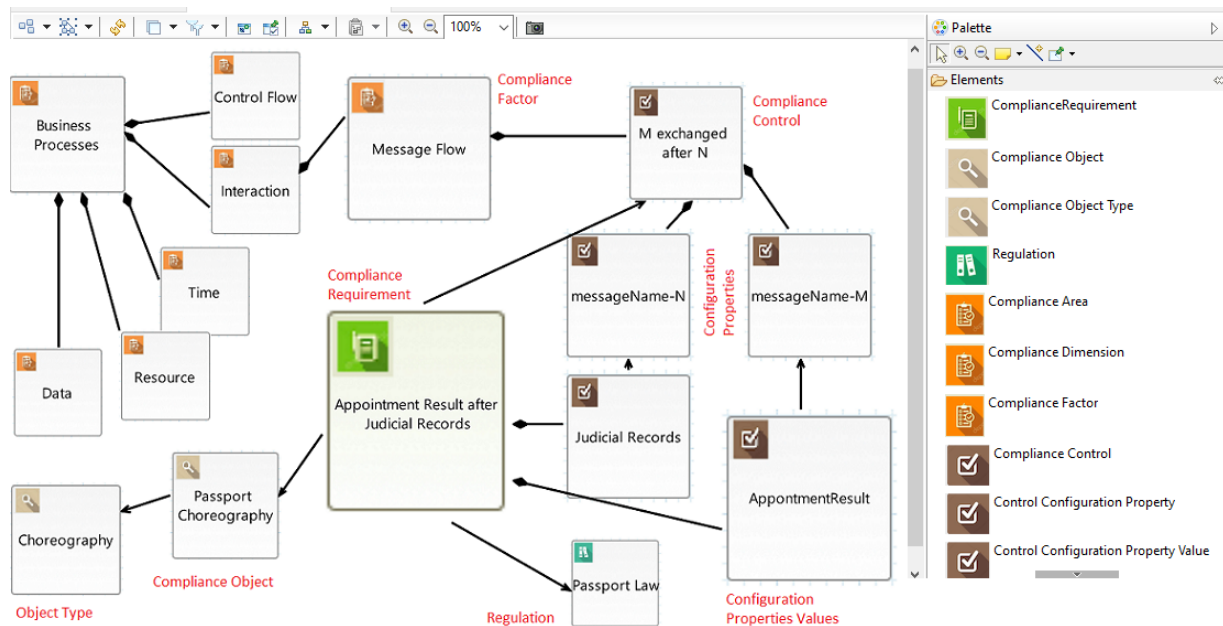


Figure 6. Compliance Requirement Definition in the CRML Eclipse plug-in

the modeling stage: a metamodel-based Compliance Requirements Modeling Language (CRML) to model such requirements and how to link those to BPMN 2.0 models, and Compliance Requirements Model (CRM) specific for business process which integrates several common existing compliance requirements (patterns) that can be used to define specific requirements over specific processes. For the evaluation stage the proposal envisions the application of process mining to analyze process execution traces in order to find compliance violations, based on the compliance requirements specified in the modeling stage for the process.

We presented preliminary results on the modeling stage, and we are working on the evaluation stage, since the focus of the work are inter-organizational e-government processes which adds several levels

of complexity in dealing with process traces and reconstructing the corresponding event log. To the best of our knowledge, there are no other proposals that defines a compliance requirements model and its relation with BPMN 2.0 in order to specify requirements over BPMN 2.0 model elements, as a key input to detect compliance violations in a post-mortem way over event logs, by applying process mining techniques.

We believe our proposal can help organizations to explicitly model compliance requirements over their processes, helping in the reasoning and control over applicable regulations and laws for e-government processes, and also to evaluate violations occurrences based on these specifications. As it is integrated in a general framework for applying data and process mining to the complete execution data of

organizations, it provides another view on process execution requirements that should be taken into account, even more in an e-government context.

Acknowledgments

This work was supported by project "Minería de procesos y datos para la mejora de procesos en las organizaciones" funded by Comisión Sectorial de Investigación Científica (CSIC), Universidad de la República (UdelAR), Uruguay.

References

- [1] W. M. P. van der Aalst, A. H. M. ter Hofstede, and M. Weske, "Business process management: a survey," in *International Conference on BPM, 2003*, pp. 1–12, 2003.
- [2] M. Weske, *Business Process Management - Concepts, Languages, Architectures, 3rd Edition*. Springer, 2019.
- [3] M. Dumas, M. L. Rosa, J. Mendling, and H. Reijers, *Fundamentals of Business Process Management, 2nd edition*. Springer, 2018.
- [4] J. Chang, *Business Process Management Systems: Strategy and Implementation*. CRC Press, 2016.
- [5] OMG, "Business process model and notation (BPMN) v2.0," tech. rep., OMG, 2011.
- [6] M. Hashmi, G. Governatori, H.-P. Lam, and M. T. Wynn, "Are we done with business process compliance: state of the art and challenges ahead," *Knowledge and Information Systems*, vol. 57, pp. 79–133, 1 2018.
- [7] A. Delgado, L. González, and D. Calegari, "Towards setting up a collaborative environment to support collaborative business processes and services with social interactions," in *Service-Oriented Computing - ICSOC Workshops and Satellite Events*, vol. 10797 of *LNCSS*, pp. 308–320, Springer, 2017.
- [8] W. M. P. van der Aalst, *Process Mining - Data Science in Action, Second Edition*. Springer, 2016.
- [9] A. Delgado, A. Marotta, L. González, L. Tansini, and D. Calegari, "Towards a data science framework integrating process and data mining for organizational improvement," in *15th International Conference on Software Technologies (ICSOFT)*, pp. 492–500, 2020.
- [10] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Quarterly*, vol. 28, no. 1, p. 75–105, 2004.
- [11] R. J. Wieringa, *Design Science Methodology for Inf. Systems and Software Engineering*. Springer, 2014.
- [12] R. K. Yin, *Case Study Research: Design and Methods, 5th edition*. SAGE Publications, Inc., 2014.
- [13] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, and B. Regnell, *Experimentation in Software Engineering*. Springer, 2012.
- [14] L. González and R. Ruggia, "A comprehensive approach to compliance management in inter-organizational service integration platforms," in *13th Int. Conference on Software Technologies*, SCITEPRESS, 2018.
- [15] A. Delgado, D. Calegari, L. González, A. Montarnal, and F. Benaben, "Towards a metamodel supporting e-government collaborative business processes management within a service-based interoperability platform," in *The 53rd Hawaii International Conference on System Sciences (HICSS-53)*, January 2020.
- [16] L. González and R. Ruggia, "Controlling compliance of collaborative business processes through an integration platform within an e-government scenario," in *53rd Hawaii International Conference on System Sciences, HICSS 2020*, pp. 1–10, ScholarSpace, 2020.
- [17] A. Delgado, B. Weber, F. Ruiz, I. G. R. de Guzmán, and M. Piattini, "An integrated approach based on execution measures for the continuous improvement of business processes realized by services," *Information & Software Technology*, vol. 56, no. 2, pp. 134–162, 2014.
- [18] D. Knuplesch, M. Reichert, L. T. Ly, A. Kumar, and S. Rinderle-Ma, "Visual modeling of business process compliance rules with the support of multiple perspectives," in *Conceptual Modeling*, pp. 106–120, Springer, 2013.
- [19] O. Turetken, A. Elgammal, W. van den Heuvel, and M. P. Papazoglou, "Capturing compliance requirements: A pattern-based approach," *IEEE Software*, vol. 29, no. 3, pp. 28–36, 2012.
- [20] D. Knuplesch and M. Reichert, "A visual language for modeling multiple perspectives of business process compliance rules," *Software & Systems Modeling*, vol. 16, pp. 715–736, apr 2016.
- [21] L. Gonzalez, R. Ruggia, J. Abin, G. Llambias, R. Sosa, B. Rienzi, D. Bello, and F. Alvarez, "A service-oriented integration platform to support a joined-up e-government approach: The uruguayan experience," in *Advancing Democracy, Government and Governance*, vol. 7452 of *LNCSS*. Springer, 2012.
- [22] M. P. Papazoglou, "Making business processes compliant to standards and regulations," in *15th Int. Enterprise Distributed Object Computing Conf.*, IEEE, 8 2011.
- [23] E. Mulo, U. Zdun, and S. Dustdar, "Domain-specific language for event-based compliance monitoring in process-driven SOAs," *Service Oriented Computing and Applications*, vol. 7, pp. 59–73, 9 2012.
- [24] L. T. Ly, S. Rinderle-Ma, and P. Dadam, "Design and verification of instantiable compliance rule graphs in process-aware information systems," in *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, pp. 9–23, Springer, 2010.
- [25] D. Knuplesch, M. Reichert, and A. Kumar, "Visually monitoring multiple perspectives of business process compliance," in *International Conference on BPM, 2016*, pp. 263–279, Springer, 2015.
- [26] D. Schleicher, F. Leymann, D. Schumm, and M. Weidmann, "Compliance scopes: Extending the bpmn 2.0 meta model to specify compliance requirements," in *Int. Conf. on SOCA*, pp. 1–8, 2010.
- [27] M. El Kharbili, Q. Ma, P. Kelsen, and E. Pulvermueller, "Corel: Policy-based and model-driven regulatory compliance management," in *Int. Enterprise Distributed Object Computing Conference*, pp. 247–256, 2011.
- [28] G. Governatori and Z. Milosevic, "Dealing with contract violations: formalism and domain specific language," in *9th Int. Enterprise Comp. Conf. (EDOC05)*, IEEE, 2005.
- [29] W. M. P. van der Aalst, K. M. van Hee, J. M. van der Werf, and M. Verdonk, "Auditing 2.0: Using process mining to support tomorrow's auditor," *Computer*, vol. 43, no. 3, pp. 90–93, 2010.