

# Towards Controlling Lawful States and Events in Business Process Models

Ludmila Penicina

Institute of Applied Computer Systems  
Riga Technical University  
Riga, Latvia  
ludmila.penicina@rtu.lv

Marite Kirikova

Institute of Applied Computer Systems  
Riga Technical University  
Riga, Latvia  
marite.kirikova@rtu.lv

**Abstract**— In enterprises it is important to control whether unlawful events occur or business objects are in unlawful states during business process execution. Existing modelling methods such as BPMN and ArchiMate do not to a full extent support description of conceivable and lawful state and event spaces that can affect execution of the process. This gap jeopardizes compliance of business process models with external and internal regulations. This paper proposes a theoretical solution for closing the gap between business process and enterprise architecture models and conceivable and lawful state and event spaces by using BWW model. BWW model is a generic framework for conceptualization of information system objects.

**Keywords**— BPMN, ArchiMate, Business process, BWW model.

## I. Introduction

Business Process Model and Notation (BPMN) [1] is the de-facto standard for representing in a very expressive graphical way the processes occurring in virtually every kind of organizations [2]. However BPMN has its limitations when it comes to modelling other aspects of organization such as data flow, organizational structure and roles, technical systems, etc. [3]. To provide a uniform representation for diagrams that describe enterprise architecture (EA), ArchiMate modelling language has been developed [4]. The core of ArchiMate language consists of three main types of elements: active structure elements (subjects), behaviour elements, and passive structure elements (objects) [4]. ArchiMate provides graphical representations of EA based on TOGAF standard [5] and defines 3 layers of EA [4]:

1. Business layer offers products and services to external customers.
2. Application layer supports the business layer with application services.
3. Technology layer offers infrastructure services needed to run applications.

In ArchiMate language the existence of business processes model is depicted. However, ArchiMate does not prescribe to list the flow of activities in detail [4]. BPMN and ArchiMate have different purposes and advantages within a specific modelling domain. The core of ArchiMate modelling language describes 3 layers of enterprise supporting reasoning about the structural and behavioural properties of it [4], BPMN supports detailed modelling of business processes and their execution. Linkage between BPMN models and ArchiMate models provides possibilities to complement BPMN models with

enterprise aspects and ArchiMate models with detailed process descriptions. Some tools like ARIS [6] and QPR [7] allow linking BPMN and ArchiMate models in their modelling environments.

Wand and Weber [8] built a set of models for the evaluation of modelling techniques based on an upper ontology defined by Bunge [9]. They extended Bunge's ontology and applied it to the modelling of information systems (BWW model) [8]. BWW model consists of constructs present in the real world that must be represented in information systems. BWW model contains general concepts that are necessary for description of information systems [10]. A number of studies have been dedicated to evaluate different versions of BPMN against BWW model and have revealed limitations of BPMN to depict some important aspects of business processes: e.g., describing lawful states and events in BPMN models (see [11], [12], [13] for detailed analysis). BWW model allows straightforwardly addressing: (1) states of things, (2) lawful state space and lawful event space of things, (3) conceivable state space and conceivable event space of things, (4) state law that restricts values of the properties of things to a lawful subset, and (5) lawful transformations that define which events in things are lawful. The previous research has shown that these aspects are supported to the full extent neither in BPMN nor in ArchiMate (see [11], [12], [13]). Thus there is a gap between BPMN-ArchiMate models and a set of conceivable and lawful states and events that motivates the research presented in this paper: we state that BWW aspects missing from BPMN and ArchiMate meta-models are essential for supporting compliance of business process models and must be addressed in meta-models.

The importance of providing means for embedding and visualizing different controls in business process models for transparency and compliance is also discussed in [14] where limitations of existing modelling languages (including BPMN 2.0) with regard to their capability of embedding controls for compliance are described. According to the results presented in [14], BPMN 2.0 is the most suitable modelling language for modelling controls in comparison to other business process modelling languages. The results presented in [14] motivate the choice of BPMN as a business process modelling language for supporting compliance of business process models that can be achieved (1) by providing linkage with ArchiMate, and (2) by complementing interlinked BPMN-ArchiMate models with BWW model elements describing conceivable and lawful state and event spaces. To introduce the proposed approach we start

with the discussion of the definitions of states and events, conceivable and lawful event and state spaces and how these spaces can be applied to business process model elements.

Every business process includes not only behavioural aspects but also structural aspects: physical or virtual elements which represent: (a) data objects with a specific lifecycle that represents how data objects change states during the process, and (b) subjects that perform tasks in the process. In ArchiMate [4] structural aspect is divided in passive structure elements (objects) and active structure elements (subjects). An active structure element is defined as an entity that is capable of performing behaviour and a passive structure element is defined as an object on which behaviour is performed [4].

According to [15] data object can be formalized as a set of states and transitions between data states (object life cycle). In BWV model *Thing* is an elementary unit. BWV model provides no further distinction into active and passive structure elements. According to BWV model *Things* possess *Properties* which define *States* of *Things*. Based on this, we here address not only states of passive structure elements but also states of active structure elements, e.g., Application Component can have a set of properties that define the states of it.

According to BWV model - a change in *State* of *Thing* is *Event* [11]. *Events* are part of business process behavioural aspect. According to BPMN specification [1] - events in business processes can occur in the beginning of a process, between execution of activities, during execution of an activity or at the end of a process, also BPMN specifies a number of different types of events. In ArchiMate the notion of event is also depicted, but only in the Business layer and without providing a detailed description.

Based on aforementioned definitions, the summary of notions of *State* and *Event* in a business process model follows. Every business process has behavioural and structural aspects. The structural aspect of a business process model includes active and passive structure elements and according to BWV these are *Things* that possess *Properties* and can assume specific *States*. BWV model defines that a change in *State* of *Thing* is *Event*, and BPMN and ArchiMate meta-models support the notion of *Event*, however, defining a set of conceivable and lawful event space there is not possible. Described gap between BPMN-ArchiMate models and a set of conceivable and lawful states and events is depicted in Figure 1.

To be able to control whether an unlawful event has occurred in a business process, or a data object or a subject (business actor) has assumed an unlawful state, it is necessary: (1) to define allowed (lawful) and not allowed (unlawful) events and states, (2) to monitor what states active and passive elements are in, and (3) to monitor whether an unlawful event has occurred in a business process as a result of an unlawful state. For example, an unlawful state of a data object or subject is a state that is in conflict with external and internal regulations. Here we are not talking about the soundness of the process – correctness criteria that a process model has to fulfil, e.g., without deadlock or livelock patterns. We are talking about different factors that change over time and can affect execution of business process models, but using existing modelling methods cannot be depicted in a business process model, e.g., internal regulations or legislation.

For instance, consider a real life example that describes self-archiving service for researchers at University, in which a researcher uploads his publication to university repository. Uploaded publication can assume several states based on the set of its properties, e.g., lawful state will be when a version of a publication's full text is the pre-print and publisher has permitted archiving this publication. Lawful event will be allowing showing a full text of this publication publicly. Unlawful event will be when a publisher has not allowed archiving but a full text is made available publicly.

The aim of this paper is to provide a theoretical solution for defining and controlling conceivable and lawful state and event spaces in linked BPMN and ArchiMate models. We state that linkage between BPMN and ArchiMate models will contribute to ability to view at which enterprise layer unlawful state or event has occurred, and to indicate on what EA elements lawful events and states depend on. We are using the notions from BWV model: *State Law*, *Conceivable State Space*, *Lawful State Space*, *Conceivable Event Space*, *Lawful Event Space*, and *History* to define lawful and conceivable state spaces for data objects and subjects and lawful and

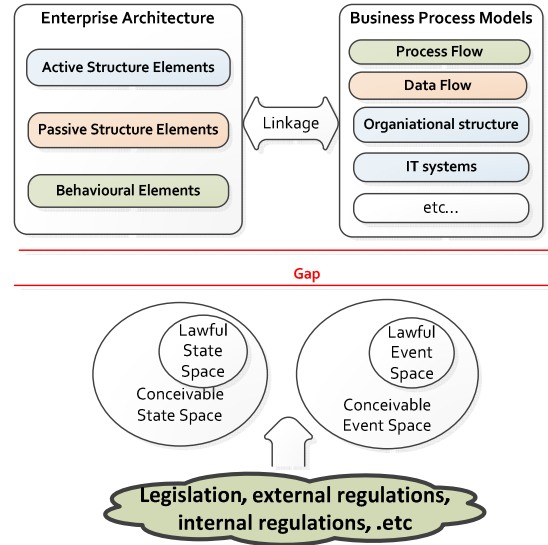


Fig. 1. Existing Gap between BPMN-ArchiMate Models and a Set of Conceivable and Lawful States and Events.

conceivable event spaces.

The paper is structured as follows. In Section 2 the related work is outlined and core elements of BWV model are presented. In Section 3 the research method is presented. In Section 4 requirements for addressing lawfulness of BPMN and ArchiMate models are discussed. A brief example of defining lawfulness in BPMN-ArchiMate models is discussed in Section 5. Conclusions and future work are presented in Section 6.

## II. RELATED WORKS

The lack of consistent theoretical foundation for building information systems urged Wand and Weber [8] to build a set of models for the evaluation of modelling techniques. Wand and Weber have extended the ontology presented by Mario

Bunge [9] and developed a formal foundation called BWV model for modelling information systems [8] consisting of the constructs present in the real world that must be represented in information systems. BWV model contains general concepts that are necessary for description of information systems [10]. Elements in BWV model (in the text shown in *italics*) can be organized in the following groups (adapted from [10]):

1. *Thing* - including *Properties*, *Classes* and *Kinds* of *Things*. *Thing* is an elementary unit in BWV. *Things* possess *Properties*, which defines *States* of a *Thing*. *Things* can belong to *Classes* or *Kinds* depending on a number of common *Properties*. A *Thing* can act on another *Thing* if its existence affects the *History* of the other *Thing*. *Things* are coupled if one *Thing* acts on another.
2. *State of Thing* - *Properties* of *Things* define their *States*. *State Law* restricts *Values* of *Properties* of *Things*. *Conceivable State Space* is a set of all *States* a *Thing* can assume. *Lawful State Space* defines *States* that comply with *State Law*. *Stable State* is a *State* in which *Thing* or a *System* will remain unless forced to change by *Thing* in the *System Environment*. *Unstable State* is *State* that will be changed into another *State* by the *Transformations* in the *System*. *History* is the chronologically-ordered *States* of *Thing*.
3. *Transformation* – transformation between *States* of *Things*. *Transformation* is a mapping from one *State* to another. *Lawful Transformation* defines which *Events* in *Thing* are lawful.
4. *Event* - event is a change in *State* of *Thing*. *Conceivable Event Space* is a set of all *Events* that can occur to *Thing*. *Lawful Event Space* is a set of all *Events* that are lawful to *Thing*. *Events* can be *Internal Events* and *External Events*. *Events* can be *Well-Defined* - *Event* in which the subsequent *State* can be predicted - or *Poorly-Defined* - *Event* in which the subsequent *State* cannot be predicted.
5. *System* – a set of coupled *Things*. *System Composition* are *Things* in the *System*. *System Environment* is *Things* outside the *System* interacting with the *System*. *System Structure* is a set of couplings that exists among *Things*. *Subsystem* is *System* whose composition and structure is a subset of the composition and structure of another *System*. *System Decomposition* is a set of *Subsystems*. *Level Structure* is an alignment of the subsystems.

The BWV model has been used in a number of studies for evaluation of modelling techniques. The authors of [10] report on the outcomes of an ontological analysis of BPMN and explore identified issues by reporting on interviews conducted with BPMN users in Australia. As a result [10] defines few potential shortcomings in BPMN - such as existence of some ambiguous elements in its specification.

The authors of [11] examine how process modelling techniques have developed and compare modelling techniques using BWV model as a benchmark used for the analysis of grammars that purport to model the real world and the interactions within it. The authors of [16] propose an approach for developing a conceptual model that represents the structural, relational and behavioural elements of the computing systems based on the BWV model.

The authors of [15] propose a notion of “weak conformance” which checks conformance of a process model with respect to data objects that its tasks access. This notion can be used to tell whether in every execution of a process

model each time a task needs to access a data object in a particular state, it is ensured that the data object is in the expected state or can reach the expected state and, hence, the process model can achieve its goals.

UEML (Unified Enterprise Modelling Language) [17] was developed during a European project [18] and aims at supporting integrated use of enterprise and information systems models expressed using different languages by providing a hub through which modelling languages can be connected. Initially UEML was based on BWV concepts and consisted of around 25 concepts but since then UEML has grown to comprise 130 concepts from 10 different modelling languages [17].

Tools ARIS [6] and QPR [7] allow augmenting BPMN and ArchiMate models in their modelling environments, however, these are just tool vendor initiatives and there exist no standardized linkage between BPMN and ArchiMate.

This paper presents a continued research based on the previous published papers ([12], [13], and [19]) concerning BPMN, ArchiMate and BWV linkage, and gaps between BWV, ArchiMate and BPMN that must be addressed for lawful business process and enterprise architecture models. The original contribution of this paper is the method for defining conceivable and lawful state and event spaces in linked BPMN-ArchiMate models.

### III. RESEARCH METHOD

To provide a solution for defining and controlling conceivable and lawful state and event spaces of BPMN-ArchiMate models we follow the research method that is depicted in Figure 2 and described below in detail.

1. *Evaluating to which extent BPMN and ArchiMate elements*

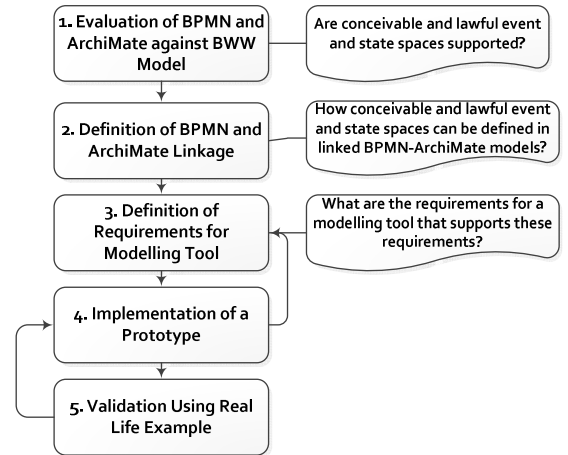


Fig. 2. Research Method Steps

support BWV elements that are aimed to describe conceivable and lawful states and events. We define what BPMN and ArchiMate elements refer to which BWV model elements. We evaluate how BPMN and ArchiMate elements support BWV model using definitions from BPMN specification [1], ArchiMate specification [4], and

definition of BWW elements from [11], and we base this evaluation on previous research presented in [10] and [11]. This step extends the results presented in [12] and [13] by contributing a more detailed BWW-BPMN and BWW-ArchiMate reference.

2. To be able to provide a view of conceivable and lawful state and event spaces in the context of EA and to assess on what EA elements lawful events and states of a process depend, we dedicate the next step to *defining linkage between BPMN and ArchiMate*. We define linkage relationships for BPMN and ArchiMate elements based on the relationships BWW model defines for its elements, e.g., in BWW model *Thing* is linked with *System*, this means that BPMN element referred to BWW *Thing* can have a relationship with ArchiMate element referred to BWW *System* element and vice-versa. We propose to use the following relationships for linking BPMN-ArchiMate model elements:

- a. At first we define relationships between BPMN and ArchiMate elements referred to the same BWW element. This relationship will represent reflexive relationship IS EQUAL TO. For example, BPMN and ArchiMate elements referred to BWW *Thing* will have IS EQUAL TO relationship between them, e.g., BPMN Lane and ArchiMate Business Actor will have a linkage relationship IS EQUAL TO.
- b. Then we define relationships between BPMN and ArchiMate elements referred to different BWW model elements. These relationships are of 2 types:
  - i. ASSOCIATION relation – two elements are associated.
  - ii. EXTENSION relation – element is the extension of the related element.
- c. Also based on BWW model we state that:
  - i. *State Law (SL)*, *Conceivable State Space (CSS)*, *Lawful State Space (LSS)*, and *History (H)* can be defined for BPMN and ArchiMate elements that are referred to BWW *Things*, since *Things* possess *Properties* and therefore can assume different *States*. So based on that we propose to define < SL, CSS, LSS, H> for *Things* only.
  - ii. We propose to define *Conceivable Event Space (CES)*, and *Lawful Event Space (LES)* for BPMN and ArchiMate elements that are referred to BWW *Transformations* and *Events*, since *Events* are happening during the process flow.

The results of this research method step are presented in Section IV.

3. The next step is dedicated to *defining requirements for the solution (the modelling tool) that will enable describing conceivable and lawful state and event spaces of BPMN-ArchiMate models using BWW model elements*. This research step also includes defining an illustrative example of the theoretical solution. The results of this research method step are also presented in Section IV.
4. The next step is *implementation of a solution for defining and controlling lawful states and events in linked BPMN-ArchiMate models in meta modelling development and*

*configuration platform* [20]. This step is a part of further work.

5. *Validation of the implementation using a real-life example*. This step is a part of further work, however, the way how research results could be evaluated by real-life example is partly illustrated in Section V.

#### IV. REQUIREMENTS FOR CONTROLLING LAWFULNESS OF BPMN MODELS

BWW model defines elements of information system that are supported by BPMN and ArchiMate modelling languages as well as a set of elements that are not supported by these standards (see [11] and [13] for more details). It indicates that complementary to BPMN and ArchiMate models it is necessary to address the missing elements in order to be able to define and control lawful state and event spaces in business process models. Majority of BPMN and ArchiMate core elements can be mapped to BWW constructs. However, still, there exist six elements that cannot be represented using these two modelling languages, namely, *State Law (SL)*, *Conceivable State Space (CSS)*, *Lawful State Space (LSS)*, *History (H)*, *Conceivable Event Space (CES)*, and *Lawful Event Space (LES)*. The missing BWW model elements have to be added to BPMN-ArchiMate models in order to track lawful and unlawful states and events in BPMN-ArchiMate models to support business process model compliance with regulations. The proposed solution for controlling lawful states and events in business process models requires a repository-based modelling tool that:

1. Accommodates all three modelling methods used, namely, BPMN, ArchiMate, and BWW.
2. Provides possibility to define in BPMN and ArchiMate models BWW elements that are not supported by BPMN and ArchiMate, namely:
  - a. *State Law (SL)* – defining restrictions of the properties of BPMN and ArchiMate structural elements to a subset that is deemed lawful, e.g., because of internal or external regulations or legislation.
  - b. *Conceivable State Space (CSS)* – possibility to define all states of BPMN and ArchiMate structural elements that they could ever assume, this helps preventing that models contain unrealistic (unconceivable) states. States are defined as a vector of values for all property functions of model elements.
  - c. *Lawful State Space (LSS)* – possibility to define the set of states of BPMN and ArchiMate structural elements that comply with state laws of these elements.
  - d. *History (H)* – possibility to maintain the chronologically-ordered states of BPMN and ArchiMate elements that they traverse in time.
  - e. *Conceivable Event Space (CES)* – possibility to define the set of all possible events that can occur in BPMN and ArchiMate models, this helps preventing creation of models that contain unrealistic (unconceivable) events.
  - f. *Lawful Event Space (LES)* – possibility to define the set of all events that are lawful in BPMN and ArchiMate models.

3. Provides a possibility to link BPMN and ArchiMate model elements using relationship types that are partially described in Table 1.
4. Allows defining algorithms, mechanisms, and queries to execute analysis of the business process models:
  - a. Performing analysis of BPMN-ArchiMate models that shows: (a) which are lawful states, (b) which are lawful events, and (c) which are unlawful states and events.
  - b. Analysing whether unconceivable states and events are present in the model.
  - c. Analysing whether unlawful states and events are present in the normal business process flow.

The necessity for BPMN and ArchiMate models linkage was recognized empirically during industrial research project that included the case study of creating an enterprise model with process patterns for the Accounting Law [21]. The obtained results showed the limitations of BPMN concerning structural modelling, e.g., limitations of BPMN to model organizational roles, actors and structure, limitations of BPMN to model application landscape, etc. (see [21] for details). The experiments showed that in order to deliver a meaningful result it was required to use a repository-based modelling tool that allowed linking BPMN to EA depicted in ArchiMate. The main requirement was to link BPMN models with ArchiMate in a way that BPMN models are the extension of ArchiMate layers, but the structural properties and enterprise layers are depicted in ArchiMate models. BPMN and ArchiMate models linkage solution must provide a possibility to define relationships from both sides – it should be possible to define linkage to ArchiMate model elements from BPMN model elements and vice versa.

Another important requirement concerns the relationship types between BPMN and ArchiMate model elements. The question is: what types of cross-element relationships are we required to use? We have addressed the scientific literature and found the possible answer in [22]. We propose to define BPMN and ArchiMate cross-element relationships based on correspondence links defined in [22], namely, using IS EQUAL TO, ASSOCIATION and EXTENSION relationships types.

Linkage of BPMN and ArchiMate is based on using BWW model as a meta-structure that maintains the relationships between BPMN and ArchiMate models. Here we continue the work presented in [13]. In [13] we identified the semantic distance between concepts in BPMN, ArchiMate, and BWW and discussed the gaps between BPMN, ArchiMate, and BWW. In [12] and [13] we mainly addressed the issues of evaluating to what extent BPMN and ArchiMate support BWW model and whether BWW model can be a meta-

structure for linking BPMN and ArchiMate. The contribution of this paper is defining particular relationship types between BPMN and ArchiMate elements. We use IS EQUAL TO, ASSOCIATION, and EXTENSION relationships.

1. IS EQUAL TO relationship – means that equivalence relation exists between BPMN and ArchiMate elements, e.g., BPMN Lane is defined as IS EQUAL TO ArchiMate Business Actor because both elements are referred to *Thing* in BWW model. This relationship is reflexive, symmetric and transitive.
2. ASSOCIATION relationship – means that a relationship exists between elements, that is not covered by IS EQUAL TO or EXTENSION relationship, e.g., Business Actor in ArchiMate and Task in BPMN have ASSOCIATION relationship, because Business Actor is referred to BWW *Thing* and Task is referred to *Transformation*, and *Thing* and *Transformation* are related in BWW model. This relationship is reflexive and transitive.
3. EXTENSION relationship – means that an element is extended by the related element, e.g., Business Event in ArchiMate can be extended with Intermediate Message Throwing event in BPMN.

Due to the space limitations it is not possible to present the complete set of relationships between BPMN and ArchiMate based on BWW model. Table 1 depicts an excerpt of BPMN and ArchiMate elements linkage with BWW elements that are for defining conceivable and lawful state and event space for these BPMN and ArchiMate elements. The full version of Table 1 consists of approximately 80 rows.

## V. THE ELECTRONIC SUBMISSION EXAMPLE

In this section we illustrate the proposed theoretical solution for defining and controlling conceivable and lawful state and event spaces in linked BPMN and ArchiMate models by using electronic submission case mentioned in Introduction. The electronic submission system provides a possibility for researchers to publish their works with metadata and full texts in Open Access online repository. Figure 3 depicts a fragment of a simplified ArchiMate model for the use of electronic submission system. This model fragment contains 3 layers of EA, namely, (1) business layer with actors, assigned roles to the actors and a high-level business process description, (2) application layer with application components and application services used by the business layer elements, and (3) technological layer with API service and systems software used by the application layer elements. The business layer of the example is represented by one main business process called Self-Archiving Process, during which a publication is added to the repository and reviewed by the system administrator.

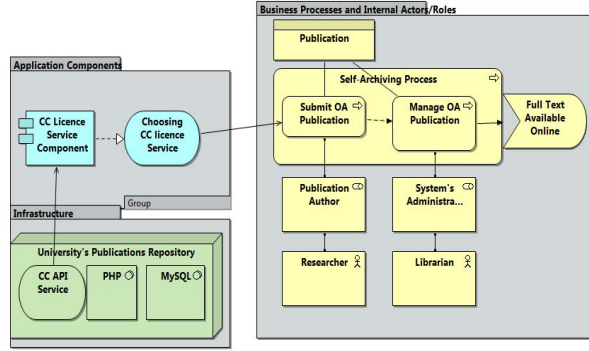


Fig. 3. ArchiMate model of electronic submission.

Figure 4 presents Self-Archiving business process in detail using BPMN notation.

In the remainder of this section we evaluate whether the proposed approach can close the gap (see Figure 1) between BPMN-ArchiMate models and a set of conceivable and lawful states and events that are defined outside of the business process (in this case lawful states and events are defined in Publisher Archiving policy). In Table 2 BWW elements describing a set of conceivable and lawful states and events for BPMN-ArchiMate models (shown in Figure 3 and Figure 4) are depicted in column “BWW Elements for Defining States and Events”. These BWW elements describe a set of conceivable and lawful states and events that cannot be described to the full extent using BPMN and ArchiMate modelling methods. Thus, in the electronic submission example the gap between BPMN-ArchiMate Models and a set of conceivable and lawful states and events can be closed in the following way (see Figure 5):

1. Using the tuple of BWW elements that describe conceivable and lawful aspects <State law, Conceivable State Space, Lawful State Space, History, Conceivable Event Space, and Lawful Event Space> for BPMN and ArchiMate elements in electronic submission process we are able to define lawful states and events that otherwise

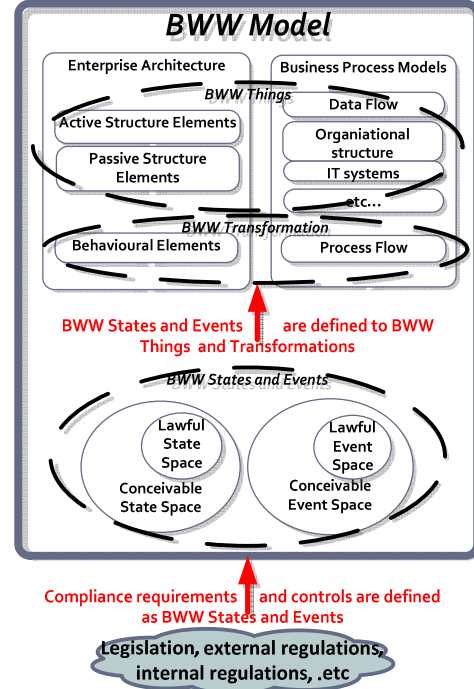


Fig. 5. Closing the Gap between BPMN-ArchiMate Models and a Set of Conceivable and Lawful States and Events.

are not straightforwardly depicted in the models. For instance, *Lawful* and *Conceivable* states of the Publication for ensuring lawful archiving in the University Repository to avoid DRM trials.

2. By linking BPMN and ArchiMate model elements we are able to see on what EA elements and at what EA levels the lawful states and events depend in business process models.

The case of electronic submission illustrated in this section shows that the gap reflected in Figure 1 can be closed by defining and controlling conceivable and lawful state and event spaces of BPMN-ArchiMate models (see Figure 5).

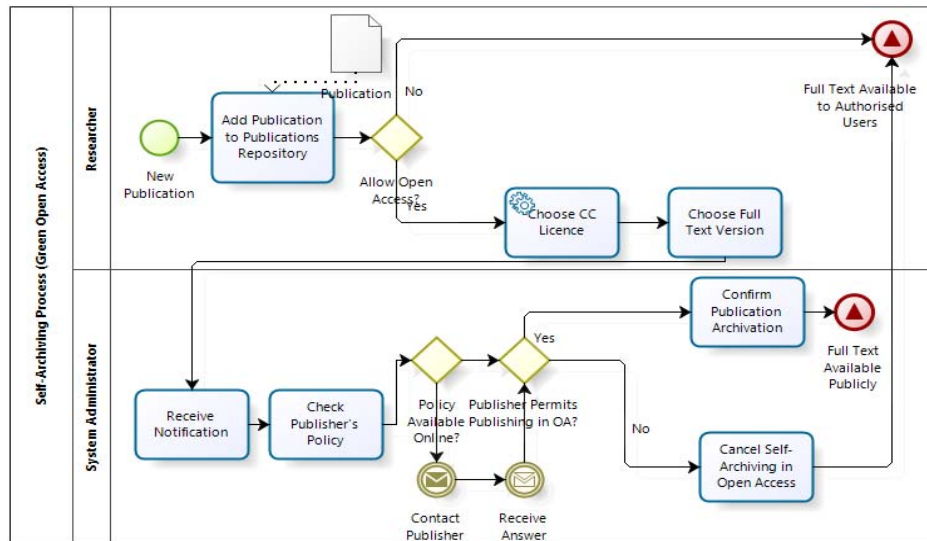


Fig. 4. BPMN Model of Self-archiving Business Process.

TABLE 1. EXCERPT OF RELATIONSHIPS BETWEEN BPMN AND ARCHIMATE BASED ON BWW MODEL.

BWW Element	ArchiMate Element	Linkage Relationship Type	BPMN Element	BWW Elements for Defining Lawfulness
<i>Thing</i>	Business actor, Business role, Business interface, Location, Application component, Application interface, Node, Device, Infrastructure interface, System software, Business object, Contract, Product, Data object, Artifact	IS EQUAL TO	Lane, Participant (Pool), Resources, Data Object , Data Store, Data Input, Data Output , Message	<i>State law, Conceivable State Space, Lawful State Space, History</i>
<i>Transformation</i>	Business process, Business function, Business interaction, Business service, Application function, Application interaction, Application service	IS EQUAL TO	Process, Send Task, Receive Task, User Task, Manual Task, Business Rule, Empty Task, Embedded Sub-Process, Event Sub-Process, Call Activity, Service task , Script Task	<i>Conceivable Event Space, Lawful Event Space</i>
<i>External event</i>	Business Event	EXTENSION	Message Start event, Conditional Start event, Signal Start event, Multiple start event, Message intermediate event, Compensation intermediate event, Conditional intermediate event, Signal intermediate event, Multiple intermediate event	<i>Conceivable Event Space, Lawful Event Space</i>
<i>Transformation to Thing/ Thing to Transformation</i>	Business process, Business function, Business interaction, Business service, Application function, Application interaction	ASSOCIATION	Lane, Participant (Pool), Resources, Data Object , Data Store, Data Input, Data Output , Message	<i>For Things:</i> <i>State law, Conceivable State Space, Lawful State Space, History</i>  <i>For Transformation:</i> <i>Conceivable Event Space, Lawful Event Space</i>
	Business actor, Business role, Business interface, Location, Application component, Application interface, Node, Device, Infrastructure interface, System software, Business object, Contract, Product, Data object, Artifact	ASSOCIATION	Process, Send Task, Receive Task, User Task, Manual Task, Business Rule, Empty Task, Embedded Sub-Process, Event Sub-Process, Call Activity, Service task , Script Task	

TABLE 2. EXAMPLE OF ELECTRONIC SUBMISSION PROCESS.

BWW Element	ArchiMate Element	Linkage Relationship Type	BPMN Element	BWW Elements for Defining States and Events
<i>Transformation</i>	<b>Business Process “Self-Archiving” at Business Layer</b>	IS EQUAL TO	<b>Pool “Self-Archiving”</b>	<p><b>Conceivable Event Space:</b></p> <ol style="list-style-type: none"> <li>1. Systems Administrator does not receive a notification about a new Publication.</li> <li>2. Notification is sent out to the wrong employee.</li> <li>3. Publisher does not reply to System’s Administrator request to permit publishing.</li> </ol> <p><b>Lawful Event Space:</b></p> <ol style="list-style-type: none"> <li>1. Author submits a Publication.</li> <li>2. System’s Administrator receives a notification about a new Publication.</li> <li>3. Publisher replies to the System’s Administrator request.</li> </ol>
<i>Thing</i>	<b>Business Object “Publication” at Business Layer</b>	IS EQUAL TO	<b>Data Object “Publication”</b>	<p><b>State law</b></p> <ol style="list-style-type: none"> <li>1. Property “Version of the Full Text” with values {Pre-print, Post-print, Published version}.</li> <li>2. Property “Status” with values {Full Text Available to Authorised Users, Full Text Available Publicly}, etc.</li> </ol> <p><b>Conceivable State Space</b></p> <ol style="list-style-type: none"> <li>1. Author has submitted post-print version of the full text but Publisher permits publishing only pre-prints – this state is Conceivable but it is Unlawful.</li> </ol> <p><b>Lawful State Space</b></p> <ol style="list-style-type: none"> <li>1. Author has submitted post-print version of the full text and Publisher permits publishing post-prints.</li> </ol> <p><b>History</b></p> <ol style="list-style-type: none"> <li>1. Full Text Available to Authorised Users, Full Text Available Publicly.</li> </ol>
<i>Thing to Transformation</i>	<b>Application component “CC Licence Service Component” at Application Layer</b>	ASSOCIATION	<b>Service Task “Choose CC Licence”</b>	<p><b>State law for “CC Licence Service Component”</b></p> <ol style="list-style-type: none"> <li>1. Property “Enabled” with values {True, False}.</li> <li>2. Property “Status” with values {Up, Possible Service Trouble, Likely Service Disruption, Disruption}.</li> <li>3. Property “Version of CC” with values {Past, Current}.</li> </ol> <p><b>Conceivable State Space for “CC Licence Service Component”</b></p> <ol style="list-style-type: none"> <li>1. Author chooses the CC Licence but service component has Status “Disruption”.</li> <li>2. Author chooses the CC Licence but service is disabled.</li> <li>3. Service component provides Past versions of CC Licences..</li> </ol> <p><b>Lawful State Space for “CC Licence Service Component”</b></p> <ol style="list-style-type: none"> <li>1. Author chooses the CC Licence and service component has Status “Up”.</li> <li>2. Author chooses the CC Licence and service is enabled.</li> <li>3. Service component provides Current versions of CC Licences..</li> </ol> <p><b>History for “CC Licence Service Component”</b></p> <ol style="list-style-type: none"> <li>1. Chronologically ordered states of service component.</li> </ol> <p><b>Conceivable Event Space for “Choose CC Licence”</b></p> <ol style="list-style-type: none"> <li>1. Author has chosen the wrong CC Licence.</li> <li>2. If a publication includes rights held by others, author has not gotten permission to sublicense those rights under the CC license.</li> </ol> <p><b>Lawful Event Space for “CC Licence Service Component”</b></p> <ol style="list-style-type: none"> <li>1. Author has chosen the right The CC Licence.</li> <li>2. Author has gotten permission to sublicense rights held by others.</li> </ol>



## VI. CONCLUSIONS

This paper presents the results of the research aimed at supporting defining and controlling conceivable and lawful state and event spaces in linked BPMN-ArchiMate models. The paper (1) indicates the existing gap between EA and business process models and externally defined set of conceivable states and events that affect execution of business processes, and (2) proposes the solution for closing the indicated gap between BPMN-ArchiMate models and a set of conceivable and lawful states and events using BWV model. The paper contributes (1) the method for defining conceivable and lawful state and event spaces of business process models based on BWV model and BPMN-ArchiMate linkage (Table 1); and (2) illustrates the proposed method with the example from the electronic publishing (Table 2). The proposed method may enable enterprises (1) to define conceivable and lawful state and event spaces in business process models; and (2) to assess on what EA elements the lawfulness of business processes depends. BWV based linkage of BPMN and ArchiMate facilitates transparent representation of lawful aspects of BPMN models at different EA layers, and gives an opportunity to care for model quality by evaluating BPMN and ArchiMate models against BWV model.

The paper presents the theoretical part of the research in progress. Further research involves implementation of modelling environment capable of defining and controlling conceivable state and event spaces in models represented in linked BPMN and ArchiMate models; as well as validation of concepts presented in the paper on models of real enterprises.

## ACKNOWLEDGMENT

This work is partly supported by the research grant of Latvian Council of Science, Project No. Z12.0342.

## REFERENCES

- [1] OMG, "Business Process Model and Notation 2.0," *Business Process Model and Notation 2.0*, 2011. [Online]. Available: [www.bpmn.org](http://www.bpmn.org). [Accessed: 30-Nov-2012].
- [2] M. Chinosi and A. Trombetta, "BPMN: An introduction to the standard," *Comput. Stand. Interfaces*, vol. 34, no. 1, pp. 124–134, Jan. 2012.
- [3] B. Silver, *BPMN Method and Style with Implementer's Guide*, 2nd Editio. Cody-Cassidy Press, 2011.
- [4] The Open Group, "ArchiMate 2.0 Specification," 2012. [Online]. Available: <http://goo.gl/7gC5B>.
- [5] The Open Group, "TOGAF 9.1." [Online]. Available: <https://www2.opengroup.org/ogsys/catalog/g116>.
- [6] Softwareag, "ARIS ArchiMate Modeler." [Online]. Available: <http://goo.gl/WI76E>.
- [7] "Enterprise Architecture modeling software QPR EnterpriseArchitect." [Online]. Available: <http://goo.gl/8Ots6h>.
- [8] Y. Wand and R. Weber, "On the ontological expressiveness of information systems analysis and design grammars," *Inf. Syst. J.*, vol. 3, no. 4, pp. 217–237, 1993.
- [9] M. Bunge, *Treatise on Basic Philosophy: Vol. 4: Ontology II: A World of Systems*. 1979, p. 309.
- [10] J. Recker, M. Indulska, M. Rosemann, and P. Green, "Do Process Modelling Techniques Get Better ? A Comparative Ontological Analysis of BPMN," *Campbell, Bruce,*

- Underwood, Jim, Bunker, Deborah 16th Australas. Conf. Inf. Syst.*, 2005.
- [11] M. Rosemann, J. Recker, M. Indulska, and P. Green, "A Study of the Evolution of the Representational Capabilities of Process Modeling Grammars," *Adv. Inf. Syst. Eng.*, pp. 447–461, 2006.
- [12] L. Penicina, "Linking BPMN, ArchiMate, and BWV: Perfect Match for Complete and Lawful Business Process Models?," *Short Pap. Proc. 6th IFIP WG 8.1 Work. Conf. Pract. Enterp. Model. (PoEM 2013)*, vol. Vol-1023, pp. 156–165, 2013.
- [13] L. Penicina and M. Kirikova, "Towards Completeness and Lawfulness of Business Process Models," *Perspect. Bus. Informatics Res. Lect. Notes Bus. Inf. Process.*, vol. Vol.158, pp. 63–77, 2013.
- [14] H. Betke, K. Kittel, and S. Sackmann, "Modeling Controls for Compliance -- An Analysis of Business Process Modeling Languages," in *2013 27th International Conference on Advanced Information Networking and Applications Workshops*, 2013, pp. 866–871.
- [15] A. Meyer, A. Polyvyanyy, and M. Weske, "Weak Conformance of Process Models with respect to Data Objects," *ZEUS*, pp. 74–80, 2012.
- [16] C. Goumopoulos and A. Kameas, *Theory and Applications of Ontology: Computer Applications*. Dordrecht: Springer Netherlands, 2010, pp. 463–485.
- [17] V. Anaya, G. Berio, M. Harzallah, P. Heymans, R. Matulevičius, A. L. Opdahl, H. Panetto, and M. J. Verdecho, "The Unified Enterprise Modelling Language—Overview and further work," *Comput. Ind.*, vol. 61, no. 2, pp. 99–111, Feb. 2010.
- [18] "List of public deliverables of the INTEROP project." [Online]. Available: <http://goo.gl/QIPQB3>.
- [19] L. Penicina, "Choosing a BPMN 2.0 Compatible Upper Ontology," *Fifth Int. Conf. Information, Process. Knowl. Manag.*, pp. 89–96, 2013.
- [20] BOC GROUP, "ADOxx.org." [Online]. Available: <http://www.adoxx.org/live/>.
- [21] L. Businska, M. Kirikova, L. Penicina, I. Buksa, and P. Rudzajs, "Enterprise Modeling for Respecting Regulations," in *PoEM Short Papers 2012 "Emerging Topics in the Practice of Enterprise Modelling": The 5th IFIP WG8.1 Working Conference on the Practice of Enterprise Modelling (PoEM 2012)*, 2012, pp. 106–118.
- [22] A. Etien and C. Rolland, "Measuring the fitness relationship," *Requir. Eng.*, vol. 10, no. 3, pp. 184–197, Aug. 2005.