Computer Science and Information Systems17(1):117–139 https://doi.org/10.2298/CSIS181118018P

Architecting Business Process Maps

Geert Poels[1], Félix García[2], Francisco Ruiz[2], and Mario Piattini[2]

[1] Faculty of Economics and Business Administration,
Ghent University, Belgium
geert.poels@UGent.be
[2] Institute of Information Technology and Systems
University of Castilla-La Mancha, Spain
{felix.garcia, francisco.ruizg, mario.piattini}@uclm.es

Abstract. Process maps provide a high-level overview of an organization's business processes. While used for many years in different shapes and forms, there is little shared understanding of the concept and its relationship to business process architecture. In this paper, we position the concept of process map within the domain of architecture description. By 'architecting' the concept of business process map, we identify and clarify diverging views of this concept as found in the literature and set requirements for describing process maps. A meta-model for a process mapping language is produced as a result. The proposed meta-model allows investigating the suitability of EA modelling languages as a basis for defining a domain-specific language for process mapping along with the creation of a better understanding of business process architecture in relation to enterprise architecture, which can be beneficial for both BPM and EA professionals.

Keywords: Process map, Business process architecture, Enterprise architecture, Architecture description, Domain-specific modelling.

1. Introduction

Recently, business process architecture design has received attention in Business Process Management (BPM) research [1]. Business process architecture is commonly defined as the organised overview of the processes that exist within an organisation, including their relationships [2]. As the output of the BPM lifecycle process identification phase during which the organisation's business processes are designated and prioritised, the business process architecture provides the basis to single out the processes that will be subjected to further BPM lifecycle activities. Like modelling individual processes is a starting point for any BPM effort [2], modelling the architecture of an organisation's collection of business processes is required for any analysis, design or improvement effort that transcends the level of individual processes (e.g., multi-process analysis [3]). Process architecture has further been positioned as an important instrument for managing large collections of process models in organisations that have already invested heavily in BPM [4]. Process architecture is also essential for process portfolio management [5] and improvement initiatives that concern multiple processes like process standardisation efforts and the identification of shared services [6]. Research in Goal-Oriented Requirements Engineering (GORE) has resulted in a few methods for (re)designing business process architecture in alignment with business goals [7-9], though most GORE research on business processes relates to goal alignment or goal-driven design of individual business processes rather than entire process architectures [10]. In Enterprise Architecture (EA), the business process architecture is considered an integral component of the business layer of an organisation's enterprise architecture, where processes are managed as assets that are vital to the organisation's operations [11, 12]. Meanwhile, different kinds of models have been proposed for representing specific views on an organisation's business process architecture, like business process co-operation models [13], process chain models [14], process landscape models [2] and process maps [6]. In particular, the concept of process map as a holistic and abstract representation of an organisation's business processes, has only recently been investigated [15], while being in use for many years in different shapes and forms. In practice, there is little shared understanding of the concept, related to its contents, form, purpose, and relationship to business process architecture. According to [16], the current variety in process maps that can be observed might be due to the lack of modelling language dedicated to expressing process maps. The need for designing such language, preferably building upon a general-purpose modelling language, has been expressed by many researchers [1, 17, 18].

Although the modelling of business processes, their interrelationships, and their linkages with strategic, operational, informational or infrastructural business and information technology elements is part of several enterprise modelling approaches (e.g., EKD [19], ARIS [20], TBIM [21], MEMO [22], 4EM [23], PGA [24]), these types of models have not been positioned as general solution for articulating process maps as they are part of and make sense in the context of a specific enterprise modelling approach. Some researchers have proposed requirements for a general-purpose process mapping language [6, 18], while Malinova and Mendling [15] have proposed a meta-model for process maps that sets requirements for a process map representation language. Malinova and Mendling [15] further showed that BPMN is not ontologically expressive enough for meeting these requirements, and therefore a process mapping language needs to be designed. Apart from not having a process mapping language, there is lack of clarity in the conceptualisation of the process map in relation to business process architecture. Specifically, in the literature there are substantial differences in conceptualization of business process architecture, process map and their mutual relation. Further, only few works on modelling business process architecture hint at positioning an organization's business process architecture within the broader enterprise architecture, though without elaborating the idea.

The goal of the research presented in this paper is to provide a conceptualization of process maps in the context of enterprise architecture by considering a process mapping language as an architecture description language. As a result, we conceptualize the process map as an enterprise architecture artefact and propose a meta-model for a business process architecture description language that can be used to represent process maps. By 'architecting' the concepts of business process architecture and process map we clarify diverging views of these concepts as found in the literature and set requirements for describing process maps. An integration of the current BPM research on process maps with EA thinking could lead to advancement in the field and increased knowledge sharing, and opens up new possibilities for research on the boundaries of BPM and EA [25]. It could also facilitate the harmonized use of general-purpose modelling languages from both fields (e.g., BPMN and ArchiMate).

The remainder of this paper is structured as follows: Section 2 presents the research methodology. Section 3 presents the background of the research, i.e., the ISO/IEC/IEEE 42010 standard for architecture description, and the related work. Section 4 describes the design of the meta-model based on our conceptualization using the defined requirements as design principles. Section 5 empirically evaluates the meta-model by instantiating it for known classifications of business process architecture descriptions and elaborate examples of process maps found in the literature. The meta-model is also formally evaluated by verifying general requirements for defining domain-specific languages. Finally, Section 6 presents the conclusion and future work.

2. Research Methodology

The development of a new enterprise modelling solution encompassing amongst others a language defined by a meta-model (i.e., abstract syntax and semantics of modelling constructs), a modelling notation (i.e., concrete syntax and notational conventions), and modelling guidelines and tool support (i.e., pragmatics of using the language and notation) – for a full set of requirements see e.g., [22] – can be undertaken as Design Science Research [26]. As several researchers have already noted the lack of a universal process mapping language and have motivated the need for its design, we engaged in an objective-centred initiation of a Design Science Research (DSR) project (Figure 1) [27], where the first research activity is the definition of the objectives of a solution for the identified problem. In this paper, we instantiate these solution objectives as a meta-model for a business process architecture description language that can be used to represent process maps, consisting of modelling constructs, their relationships and constraints – leaving other language requirements (e.g., notational, tool support) outside the scope of the current paper. Following [28], the design of a new meta-model initiates a new research cycle embedded in the overall DSR project.

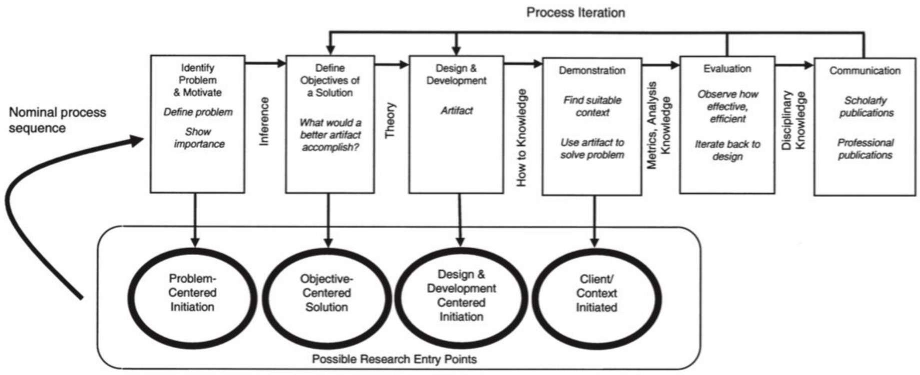


Fig. 1. Design Science Research process [27]

In this embedded DSR project, the following research steps were taken:

(1) Literature review of research on process maps and modelling of business process architecture. Research presenting requirements for process mapping, informal solutions, and reviews of design approaches for business process architecture was analysed. The result was an inference of different perspectives on process maps in relation to business process architecture, indicating a lack of ‘architectural point of view’, which motivated our research (i.e., *identify problem and motivate* in Figure 1) (Section 3).

(2) Analysis of the reviewed process map and business process architecture concepts from an architectural point of view, using the ISO/IEC/IEEE 42010 software and systems engineering international standard for architecture description [29] as conceptual frame of reference. This standard provides a core ontology (i.e., ‘theory’ in Figure 1) for the description of architectures, that we used as a conceptual reference framework for clarifying the relationship between the business process architecture and process map concepts. Contextualising these concepts according to this architecture description standard clarifies their mutual relationship, reveals the diversity that is present in the inferred perspectives from the literature, and guides proper choices regarding assumptions and requirements for business process architecture description (i.e., *define objectives for a solution* in Figure 1) (Section 3).

(3) Guided by the developed contextualization within architecture description, we first recovered a conceptualization of process map as business process architecture model [30]. We critically assess existing proposals of process map conceptualizations, along with proposed requirements for a process mapping language and informal solutions that have been used in the absence of a standard process mapping language. We then developed a new meta-model for business process architecture description that can be used to guide the development of a general representation language for process maps (i.e., *design and development* in Figure 1) (Section 4).

(4) To demonstrate the meta-model’s ability to guide the design of a universal representation language for process maps, we instantiated it to represent a wide array of process maps and other business process architecture descriptions found in the literature which are currently only informally described or represented using different notations (i.e., *demonstration* in Figure 1) (Sub-Sections 5.1, 5.2, and 5.3).

(5) To evaluate the meta-model, we analysed the meta-model instantiations used in the demonstration for evaluating the meta-model’s ability to uniformly represent different kinds of process map (Sub-Sections 5.1, 5.2, and 5.3). In practice, this was an iterative process as representing those process maps allowed us to refine the design of the meta-model until all process maps were valid instantiations of the meta-model. Apart from that, we verified the satisfaction of core requirements for defining domain-specific modelling languages, as identified in the literature [31] (i.e., *evaluation* in Figure 1) (see Sub-Section 5.4).

(6) Presentation of the meta-model and its DSR research process (i.e., *communication* in Figure 1) (i.e., the purpose of this paper).

3. Background

In this Section the background of the presented proposal is described and the related work that we analysed to recover a conceptualization of process map consistent with this background is presented. As conceptual reference framework for imposing an architectural point of view on business process architecture and process maps as models of business process architecture we used the ISO/IEC/IEEE 42010 software and systems engineering international standard for architecture description [29]. This standard presents a conceptual model of architecture description that can be applied to any kind of architecture, including business process architecture. The standard also specifies desired properties for architecture descriptions, which result in requirements for architecture frameworks and architecture description languages such that desired properties are exhibited by the architecture descriptions that are developed using these frameworks and languages. The concepts and requirements provided by the standard can be used to guide the design of a business process architecture description language, which can be used to represent process maps. The mapping to business process architecture description of the standard's architecture description concepts is summarized in Table 1.

Table 1. Process map and process mapping language in terms of business process architecture description

ISO/IES/IEEE 42010 standard- architecture description concept	Mapping to business process architecture description
System	(Collection of) business processes
Environment	Organization Examples company, not-for-profit organization, university, business unit of a company
Stakeholder	Examples operational managers, process/domain architects, CPO, business managers, enterprise architects
System Concern (associates System and Stakeholder)	Examples consistency and completeness, dependencies between processes, responsibilities, performance, strategic fit
Purpose (is a System Concern)	Efficient organization of the work to be performed in the organization
Architecture	Business process architecture
Architecture Description	Business process architecture description
Architecture Viewpoint	Example E2E processes viewpoint
Architecture View	Example E2E processes view
Model kind	Example E2E processes model kind (e.g., meta-model)
Architecture Model	Example E2E processes model (i.e., *process map*)
Architecture Framework	Business process architecture framework
Architecture Description Language	Business process architecture description language (i.e., *process mapping language*)

The standard also defines the concepts of architecture framework and architecture description language (ADL) as mechanisms that can be used for creating and employing architecture descriptions. TOGAF [32] for instance presents an architecture content framework identifying business processes as an architectural artefact, but refers to ArchiMate [13] as possible ADL to provide a notation for modelling business processes. Regarding the definition of business process architecture, we noticed in the literature two main perspectives:

1. The process architecture as the organization of the business processes in terms of their boundaries, dependencies, priorities, criticality, strategic importance, linkage with functional domains, etc. [1, 2]. In this perspective, the business process architecture is used to select processes that will be subjected to analysis and improvement actions (BPM) or to design or align the organization's system of business processes in relation with other organizational assets, goals and strategies (EA, GORE).
2. The process architecture as the organized collection of business process models and their relationships [4]. In this perspective, the business process architecture is used to categorize and manage process models [33] and to maintain the consistency between process models [18].

While these perspectives are not per se incompatible – the process architecture as a guide to initiate BPM and once the BPM initiative is ongoing as an organized overview of the resulting business process models – there are implications for the definition of the process map. In the first perspective, the process map is a model of the business process architecture, while in the second perspective it is part of the business process architecture, which would in that case more appropriately be called the business process models architecture. In the second perspective, the collection of business process models is usually hierarchically structured into several layers of modelling abstraction, resulting in a business process models decomposition tree, starting from the more abstract process models at the top to the more concrete process models at the bottom. While this decomposition can be organized in different ways, either or not ensuring consistency between models at different abstraction levels [34], the process map is typically seen as the entry-level model at the top of the hierarchy, providing a holistic and abstract overview of all or the main processes and their relationships [33].

In the first perspective, the business process architecture can also be hierarchically structured, but now according to increasing levels of granularity. In this context granularity refers to what is being considered as the atomic element of a business process architecture. In a flat (i.e., non-hierarchical) business process architecture, also called process landscape [14], there is only one atomic element and that is the business process. Business processes can be ordered, grouped, decomposed and specialized (whatever type of relation is recognized; see Section 4), but all process steps, process group members, sub-processes and process variants still qualify as business processes. On the contrary, in a hierarchical business process architecture, the atomic element considered at lower levels is more fine-grained than the atomic element considered at higher levels. For instance, the process architecture discussed in [2] defines three levels of granularity with respectively business processes, activities, and tasks as atomic elements. Sometimes the top level of a proposed hierarchy has a more coarse-grained atomic element than the business process. For instance, Van Nuffel and De Backer [18] consider the main business process, representing a process family, as atomic element for the top level, whereas the elementary business processes that are process variants in these process families are only considered at the second level.

Regardless whether the process map is defined as a model of the business process architecture (i.e., first perspective) or as entry-level model of the business process models architecture (i.e., second perspective), other differences surface. These differences emerge as a result of variations in the assumed structure of the business process architecture. Whereas in a flat architecture the scope of the process map is the entire process landscape [6], in a hierarchically structured business process architecture, the scope is typically limited to the top level. For instance, in [18] the process map describes the top level in a five-level process architecture and thus models the main business processes of an organization. On the other hand, in [2] the process map is positioned as a model of the second hierarchical level where it describes the main flow of process activities. Some literature also recognizes that a process map may only partially model the process architecture within its scope and is thus part of a view on the architecture [4]. A process map as part of a view on the business process architecture is an abstraction that serves some purpose. For instance, the requirements for process maps specified in [6] define an abstraction that is useful for identifying sub-processes that can be further investigated for being standardized. Few authors, however, explicate the intended use of process maps.

In summary, the current proposals of desiderata for process mapping are hard to compare and evaluate, unless an architecture viewpoint has been explicated. Purposes listed in the literature can be very general (e.g., representation) or very specific (e.g., identifying functional similarities). We found only one paper (i.e., [18]) that explicitly distinguishes process maps according to several different views, however, without defining the corresponding viewpoints. In general, there is a lack of explicit definition of viewpoints identifying stakeholders in the organization's system of business processes and the concerns of these stakeholders. Furthermore, it is clear that the different proposals for process map representation make their own (and generally implicit) assumptions about architectural viewpoints of the business process architecture description, which along with differing assumptions about the nature and structure of the business process architecture and its relation to the process map, result in lack of consensus on the requirements for and design of a general-purpose process mapping language.

4. Designing the Metamodel

Following [35] and [36] on the difference between ontology and meta-model, we move with the meta-model beyond the conceptualization of process map within the business process architecture description domain (i.e., ontology development), by supporting the computerized representation of business process architecture models (i.e., domain-specific modeling language development). Prior to the development of the metamodel a conceptual analysis of process maps was conducted. Despite the absence of explicitly defined business process architecture viewpoints in the related literature (see Section 3), there is one proposal that is similar to the design of an ADL for business process architecture description as it provides a model kind for process maps. The process map meta-model of Malinova and Mendling [15] is to the best of our knowledge unique in its kind. The proposed meta-model is positioned by its designers as a model of a process map conceptualization rather than a formal meta-model defining a process mapping language, which makes it a valuable starting point for our conceptual analysis. Its embedding in BPM research results, however, in a number of assumptions related to the use of process maps prior to BPM implementation (i.e., process identification) and during BPM implementation (i.e., process model management). Referring to the perspectives discussed in the previous Section, the meta-model conceptualizes the process map as an entry-level model of a hierarchically structured business process models architecture (i.e., the second perspective discussed in Section 3). We therefore complement the conceptual analysis with other related work that positions the process map as a model (i.e., abstraction) of the business process architecture (regardless what view is abstracted). The goal is to arrive at an elaborate conceptualization that covers not only the proposed meta-model but also other proposals even if only informally described or just based on a set of requirements or example notation. As a result of the conceptual analysis of process maps based on the reviewed literature, the following requirements were formulated:

Req. 1: The business process is the atomic element of the process map.

Req. 2: It should be possible to show on a process map the enterprise architecture elements that a business process (composite) is related to.

Req. 3: It should be possible to show on a process map composites of business processes that result from the application of different types of process relations.

- ***(Req. 3a)*** *Sequencing relations*: The execution of business processes may be ordered in time meaning that the execution of a first process is followed by the execution of a second process. Such ordering relations typically indicate that processes are steps in a process chain that serves a higher-level goal. For instance, the requisition process and the purchasing process are steps of the Purchase-to-Pay (P2P) end-to-end process where requisition is performed before purchasing. Identifying ordering relations is important as changes applied to a prior process may affect the design and execution of a subsequent process.
- ***(Req. 3b)*** *Decomposition relations*: A business process can be a sub-process of another business process, like the sales order data entry process that is a sub-process of the sales order handling process. The steps of a process chain are sub-processes of the process chain. Decomposition can also take the form of shared aggregation. For instance, a customer data verification process can be a sub-process of both a sales process and an after-sales service process. Decomposing business processes into sub-processes relates processes hierarchically which is important as BPM actions taken on sub-processes may affect their superordinate processes.
- ***(Req. 3c)*** *Grouping relations*: Business processes can be related through joint membership of a process group. From the moment business processes have something in common, a process group can be defined. For instance, a credit sales process and a cash sales process are members of the group of sales processes. Both processes share the goal of selling products or services to customers, but differ in the manner in which customers pay for their sales. In principle, any property of processes can be used to form process groups. Defining process groups allows abstracting from certain differences between

processes to see ‘the forest through the trees’, which can be important especially for organizations with large numbers of business processes.

- ***(Req. 3d)*** *Specialization relations*: A business process can be a specialized version of another business process, like the job student recruitment process that specializes the personnel recruitment process. A business process and its child processes form together a process family in which the child processes are process variants and the parent process becomes a standard process for these process variants. A process group, like the sales processes group, can be a process family, but is not necessarily so as there might be no standard sales process defined. Identifying specialization relations is important as changes applied to a parent process may have consequences for the child processes. Note that the implementation of specialization is not considered at the abstraction level of the business process architecture. One approach for instance is to define variation points in a standard process, which can be filled differently for the process variants [37].

The solution to these requirements which guided the design of the process maps metamodel can be summarized as follows (detailed explanations are included in [30]):

- We chose to restrict the use of process maps (as a business process architecture model) to *black-box modelling* of organization’s business processes (Req. 1).
- We generalize existing proposals of including process-related elements in a process map by means of an *Enterprise Architecture Element* that can be instantiated in process maps according to needs (depending on business process architecture viewpoint) (Req. 2).
- We recognize the need to represent in a flexible and extensible manner *Business Process Composites* where business processes can be aggregated to higher-level concepts reflecting different internal structures depending on the types of relation between the processes in the composite (Req. 3).

Therefore, our metamodel design philosophy was driven by the conceptualization of the process map as a business process abstraction that provides a black-box view on the organization’s business processes, the search for maximal integration with EA description assuring robustness and extensibility of the meta-model, and the recognition of different kinds of business process composites. In addition, to cater for expressing any business process architecture viewpoint, we need to allow maximum freedom for instantiating the meta-model according to process mapping needs, thereby limiting the number of constraints at the meta-model level. The result of our design is shown in Figure 2 as a UML class diagram.

As it can be observed in Figure 2, the core concept of the meta-model is the ***Business Process Architecture Element***, which is shown as an abstract class. We prefer the term business process architecture element to business process as a business process architecture does not necessarily include all organizational business processes [18], hence only the business processes and their composites that are part of the business process architecture are represented in process maps. Also, according to the architecture description standard, the system’s architecture is what is essential about the system considered by the system stakeholders and their concerns. For instance, while Rosemann and vom Brocke [38] include in their ‘enterprise process architecture’ all processes of an organization, [2] include only those processes that have been identified in the first phase of the BPM cycle [39]. As noted before, the process map conceptualization by Malinova and Mendling [15] is strongly based on the process landscape level of the three-level process architecture in [2], but then seen as entry-level model to the business process models architecture. To allow for different business process architecture viewpoints, we thus prefer the use of the term business process architecture element. The abstract business process architecture element is either an ***Elementary Business Process*** or a ***Business Process Composite***. According to Req. 1, a process map provides a black-box model of the elementary business processes in the business process architecture, meaning that the internal structure and operation of the elementary business processes is hidden [18]. An *Elementary Business Process* is thus an atomic business process architecture element [15]. The concept of business process composite is a new notion that we introduce because of Req. 3 and which is not present in the reviewed literature. We obtain it by applying the composition pattern [40]. A *Business Process Composite* can thus simply be defined as any business process architecture element that is not an elementary business process. By applying the composition pattern, a business process composite can be disaggregated into other business process composites or elementary business processes. A business process composite is thus a non-atomic business process architecture element. Even if non-atomic, a process map may show a business process composite without showing its parts. That is why there is no minimum cardinality constraint on the parts of a business process composite. Through the composition pattern we include the *decomposition* relation (Req. 3b) in the meta-model, in the form of unrestricted shared aggregation. Business process architecture decomposition structures can extend over multiple levels. The only constraint included in the meta-model is that elementary business processes cannot be disaggregated (as this would violate Req. 1). When instantiating the meta-model for developing process maps according to specific business process architecture viewpoints, additional constraints can be imposed. An example of such constraint could be that the leaves in a decomposition structure can only be elementary business processes.

The meta-model (Figure 2) shows three subclasses of business process composite: process group, process chain and process family. The specialization is partial meaning that there can be other business process composites than these three. We include these three composites as a specific type of relation between business process architecture elements defines each of them:

- ***Process Chain*** is an aggregate of business process architecture elements that are related through *sequencing* relations (Req. 3a), meaning that there is a sequential ordering amongst these elements. The roles of prior and subsequent process as steps in the process chain are extended to business process architecture elements to allow, for instance, a process chain to be composed of sequentially ordered ‘sub’ process chains or process families (as represented by their standard processes). Using our meta-model, end-to-end processes can be modelled as process chains.
- ***Process Group*** is an aggregate of business process architecture elements that become members of the same process group as defined by *grouping criteria* (Req. 3c). The members of a process group are related in the sense that they share one or more similar properties. They can, for instance, belong to a same process category, in which case the process group represents a category and the property of fulfilling a similar role in the organization or serving a similar

purpose is used as grouping criterion (see Sub-Section 4.5). The process group may also represent a phase meaning that its member processes are executed at the same time and time of execution is the property that serves as grouping criterion. Also, being related to a same enterprise architecture element is a property that can be used as grouping criterion to define a process group, for instance, all processes having the same business actor as process owner form a process group.

- ***Process Family*** is an aggregate of business process architecture elements based on *specialization* relations (Req. 3d) where a parent process assumes the role of standard process (called main process in [18]) and child processes are variants of the standard process. The specialization relation is defined for the abstract business process architecture elements to allow for maximum freedom when instantiating the meta-model (e.g., one process group specializing another process group, one process chain being a variant of another process chain). We follow Van Nuffel and De Backer [18] by having the process family represented through the standard process, implying that a standard process not only generalizes process variants but also aggregates these variants, i.e., the standard process is not an elementary business process but a business process composite. The meta-model does, however, not impose that all parent processes are standard processes that represent process families.

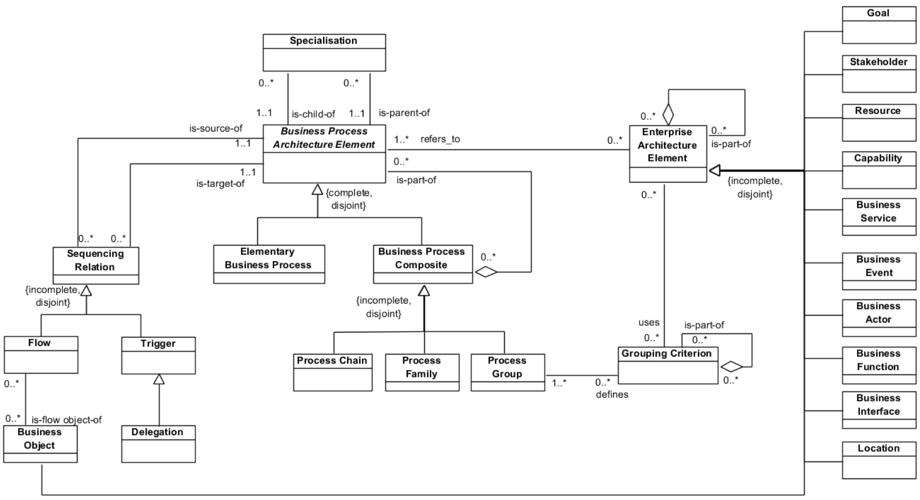


Fig. 2. Meta-model for business process architecture description.

The recognition of different types of business process composite implies that business process architecture elements can be related in different ways. The ***Sequencing Relation*** (Req. 3a) associates a source element to a target element implying a temporal ordering of *source* and *target*. The source and target association ends represent the roles that business process architecture elements fulfil in sequencing relations. In case of sequentially related elementary business processes, the source and target roles are equivalent to the prior and subsequent roles identified in our process map conceptualization [30]. The semantics of the temporal ordering of source and target are defined more precisely in the subclasses of the sequencing relation, yet the specialization is optional to allow for maximum freedom when instantiating the meta-model to describe business process architectures and create process maps. For instance, Dijkman et al. [1] present a process map example where processes are shown as temporally ordered without further specification of the exact nature of the sequential ordering (e.g., Should the prior process be completed before the subsequent process can start? Does the prior process passes information on to the subsequent process?). Scheer's Value Added Diagram [20] has been mentioned in related literature as a framework for designing process maps [41]. The sequential order of the prior and subsequent processes composing the process chain represented on such diagram might be indicated purely by means of secondary notation (i.e., using the chevron symbol for processes and placing sequentially ordered processes adjacently on the map). The only semantics attached to the sequential relation concept in our meta-model is that the business process architecture element that is the target of the relation follows up on something performed by the business process architecture element that is the source of the relation. 'Follows up on' means that it may depend on the type of sequential relation (if further specified) but also on the type of business process architecture elements that are sequentially related. Hence, when instantiating the meta-model, additional constraints might be imposed by the users (e.g., a constraint that process groups cannot be sequentially related). Two subclasses of sequential relation are specified in the meta-model: ***Trigger*** and ***Flow***, which we define based on the formalization in [42]. A Trigger is a sequencing relation in which the source business process architecture element causes the target business process architecture element to be instantiated and to start. Instantiation of business process composites is not further defined as its relevancy and semantics may depend on the chosen business process architecture viewpoint. For instance, the instantiation of a process chain could mean the instantiation of its first process step. For a process family, it could mean the instantiation of any variant of the standard process. For process groups, it probably has no relevancy. We further distinguish delegation as a subclass of trigger. A ***Delegation*** is a trigger in which the source is dependent on the outcome of the target. It is similar to the dependency relation in [18] and the uses relation in [1]. While with trigger it is not required that the source expects a response from the target (and so can end independently of the target), with delegation a response is expected and the source will not end before the target has performed some work whose outcome is needed to successfully complete the source. A ***Flow*** is a sequencing relation in which a business object flows from the source to the target. We define a ***Business Object*** as anything that flows between business process architecture elements and which is considered relevant according to some business process architecture viewpoint to be represented in a process map. This can be information, as in the meta-model of Malinova and Mendling [15], but also physical products or even persons (e.g., a patient). While conceptually every flow has at least one business object as flow object, the process map as an abstraction does not need to show this flow object. On the other hand, a business object can only be a flow object on a process map if it is linked to some flow. As shown in our meta-model, a business object is an enterprise architecture element. The meta-model allows flows between elementary business processes, but also between business process composites. More restrictive rules can be imposed if flows between certain kinds of composites are not meaningful, but for the sake of generality such constraints are not part of the meta-model. Trigger (or delegation) and flow can occur concurrently between a same source and target. Triggering is usually also accompanied by the passing of some information or signal. If meaningful to be shown on the process map, then both relations can be depicted, where the trigger has the meaning of starting the target and the flow has the meaning of passing a business object to the target.

Apart from decomposition and sequencing relations, the meta-model includes ***Specialisation*** which relates a child business process architecture element to a parent business process architecture element (Req. 3d). A parent can have many children while a child can have many parents. The meta-model allows business process composites to specialize other business process composites (e.g., a process group that specializes another process group), but again specialization of business process composites is not further defined as its relevancy and semantics depends on business process architecture viewpoint.

A special kind of relation is that between members of a process group. They are related in the sense that they share a common property. The property on the basis of which the grouping occurs is defined by a ***Grouping Criterion*** (Req. 3c). An example of a grouping criterion is 'the process category is core'. This criterion then groups all the core processes of the organization. A grouping criterion can aggregate other grouping criteria, for instance 'the process category is management and the location of process execution is the company headquarters'. Each of these aggregated criteria defines its own process group, while the aggregate criterion defines the intersection of these groups as a new process group. The type of aggregation shown in the meta-model is shared aggregation, allowing any kind of regular expression of logical operators to compose grouping criteria based on elementary criteria, again allowing maximum freedom when instantiating the meta-model. The meta-model shows that each grouping criterion defines at least one process group (possibly empty), but there can be many groups defined by the same criterion to account for evolution over time (i.e., the set of business process architecture elements that share some property is dynamic). The meta-model also allows that a process map shows process groups without identifying the defining grouping criteria.

The final element on the meta-model is that of ***Enterprise Architecture Element***, which is defined as an element that is part of the enterprise architecture and that is related to a business process architecture element (Req. 2). As discussed in Sub-Section 4.5, the reviewed literature offers a non-exhaustive set of concepts that can be shown in a process map as they relate to business processes or their composites. We believe that most of these can be described as enterprise architecture elements, though it might depend on the EA framework referred to whether they are recognized as such.

The meta-model in Figure 2 shows a number of subclasses of Enterprise Architecture Element. These subclasses are not an exhaustive enumeration of relevant types of enterprise architecture elements that can be included in process maps (as the specialization of Enterprise Architecture Element is partial). They are included in the meta-model for illustrative purposes only, being inspired by the concepts included in the motivation, strategy and business layers of the ArchiMate ADL [13]. We could also add concepts of the application, technology and physical layers of the ArchiMate ADL (e.g., application components, data objects), but we chose not to do so in order not to overload Figure 2. We consider a relationship with an enterprise architecture element as a property of the business process architecture element. Consequently, a primary use of enterprise architecture elements in a process map is to define process groups. Hence, enterprise architecture elements can be used by grouping criteria to define process groups. Enterprise architecture elements can aggregate other enterprise architecture elements. For instance, processes of an international company can be grouped by continent and by country, where a continent is a location aggregating countries as other locations.

The main feature of the meta-model is that through the use of the concept Enterprise Architecture Element, the business process architecture is integrated into the overall enterprise architecture and hence the process map, as a business process architecture model, can be linked to EA models.

5. Testing the Metamodel

To demonstrate and evaluate the meta-model's ability to serve as a conceptual foundation of a universal representation language for process maps, we instantiated it to represent a wide array of example process maps found in the literature. This was a repetitive process providing us with feedback on how to refine the meta-model until all our instantiations were valid. To this end we used three different published classifications of business process architecture description. Sub-Section 5.1 presents meta-model instantiations for the four 'archetypes' of process map used in industry after an extensive empirical study of how process mapping is performed in practice conducted in [4]. Sub-Section 5.2 shows how the meta-model can be instantiated for the example process maps described in [1] based on a systematic literature review of business process architecture design approaches. Next, Sub-Section 5.3 describes how the illustrative process maps that are part of different business process architecture views proposed by Van Nuffel and De Backer [18] can be represented using the meta-model that we designed. We show the instantiation of the meta-model by means of concept maps. To clarify the link between the concept map and the meta-model, concepts are stereotyped with the name of the meta-model class that is instantiated. The relationships between the concepts show the links between these class instances according to the associations and aggregation relationships of the meta-model. The meta-model evaluation against core requirements for domain-specific languages is presented in Section 5.4.

5.1. Representation of Archetypical Process Maps Used in Industry

With regard to the representation of archetypical process maps, in Malinova, Leopold [4] four types are distinguished. Table 2 shows these archetypical process maps (left column) together with their concept map representation using the proposed meta-model (right column). The model shown in the left column of the first row represents two adjacent modelling layers in a *hierarchical process model architecture*. Note that the process models in the layers labelled level 2 and 3 are not black-box models of business processes, showing for instance sequence flows and gateways for non-sequential flow. Following Req. 1, representing this model using our meta-model means abstracting from the internal details of these processes. The corresponding concept map thus shows Level 2 Process as a business process composite that has the two Level 3 Process as parts. These Level 3 Process are modelled as elementary business processes. The second row shows a process map of the *pipeline process architecture* archetype. This process map can be represented without compromise by instantiating the proposed meta-model. At each level a process chain is shown that is decomposed into sequentially related ‘sub’ process chains. The concept map on the right shows the mechanism of this recursive structure of sequentially related process chains. Although not shown in the concept map, for level 4 the business process architecture elements modelled would be elementary business processes. The third row contains the example process map of the *divisional process architecture* archetype. The process map distinguishes between management processes and core processes for which (business) units are responsible. The concept map at the right demonstrates the use of the grouping mechanism. The grouping of management processes is defined by the process category (i.e., management). We chose to explicitly model the grouping criterion, but the grouping can also be implicit and just based on the name of the process group. Core processes are grouped by means of a common property, i.e., their relation to a business unit. Such business units can be represented in an EA model as business actors (e.g., using ArchiMate). The reference to a common business actor thus defines the grouping of the core processes, which is only illustrated for Unit II in the concept map at the right. The plus signs in the process map at the left indicate that each core process is actually a business process composite, which can be further decomposed and specialized if needed, for instance as process chains like in the pipeline process architecture of the second row. Finally, the fourth row shows the example process map of the *service-oriented process architecture* archetype. The process map includes four distinct groups of processes, which we model as process groups, the criterion for grouping being the process category (i.e., management, service, support, measure & analyse). Each process in these groups is modelled as a business process composite (because of the plus sign in the process map). The concept map shows the mechanism of delegation that is exemplified in the process map. A business process (composite) in the service process group delegates part of the work to be performed to processes of other groups. In the example an elementary business process that is part of some business process composite of the support group and an elementary business process that is part of some business process composite that is part of the measure & analyse process group. Note that similar to what is seen in the first row, the right part of the process map (left column) is not a black-box model, hence the instantiation of the meta-model (right column) does not show gateways and sequence flows.

5.2. Representation of Process Maps Resulting from Different Business Process Architecture Design Approaches

Dijkman et al. [1] identified five different approaches for designing business process architectures in the literature. These approaches stand for the academic perspective on business process architecture. Each approach is exemplified by a different process map in an informal notation that is inspired by ArchiMate. Table 3 shows these example process maps (left column) along with their representation as instantiation of the meta-model proposed in Section 4 (right column). The example process maps in the left column of Table 3 differ from those of Table 2 in that they only show business process architecture elements and relationships between those, i.e., no process-related EA elements are shown. All the elements and relationships included in the example process maps can be represented by instantiating the proposed meta-model, in particular by means of business process composites, elementary business processes and is-part-of relationships. In the second row a process chain (Perform Project) is shown that consists of two elementary business processes (Make Project Plan and Approve Project Plan) that are related by a Trigger relation (i.e., Make Project Plan triggers Approve Project Plan). In the third row a process family is shown, which is represented by a standard process (Insurance Application) that generalizes two elementary business processes (Home Insurance Application and Car Insurance Application) that are part of the process family. An alternative representation is to show both processes as children of the parent process Insurance Application, however, the chosen representation emphasizes that Insurance Application represents an entire process family.

Table 2. Meta-model instantiations for archetypical industry process maps.

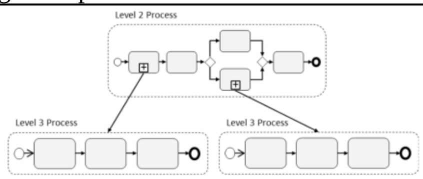
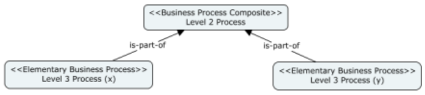
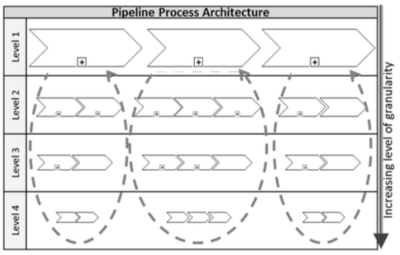
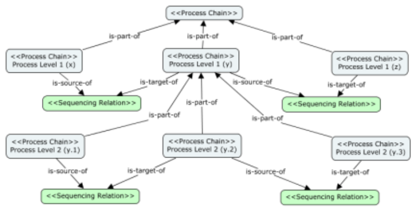
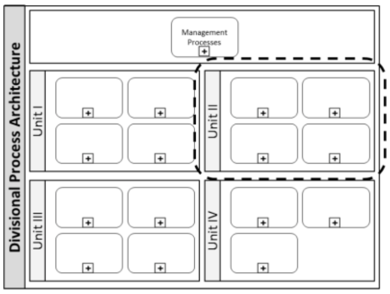
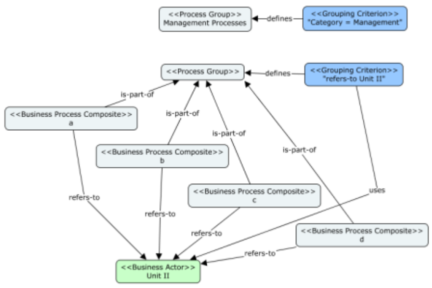
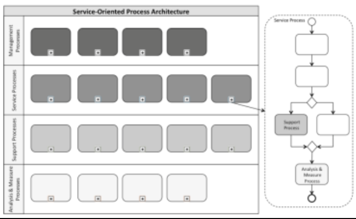
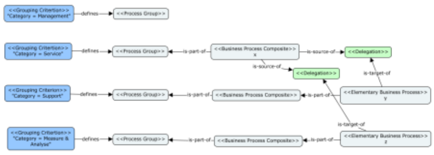
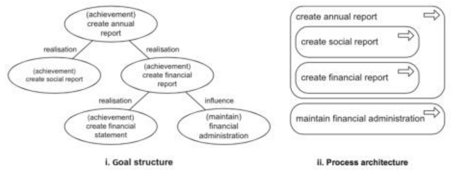
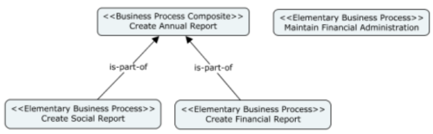
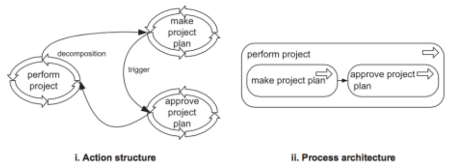
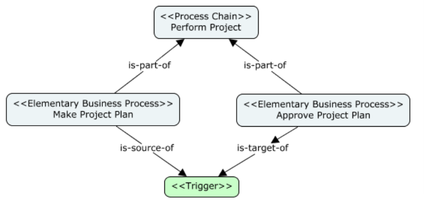
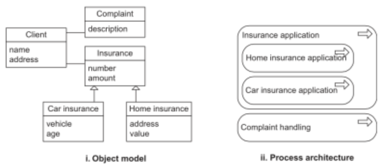
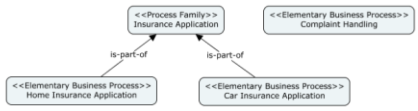
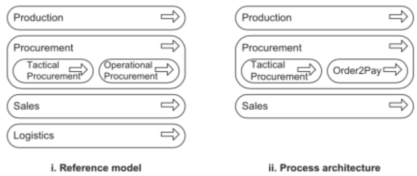
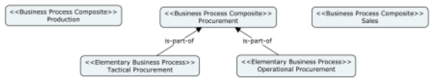
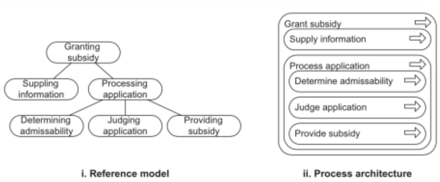
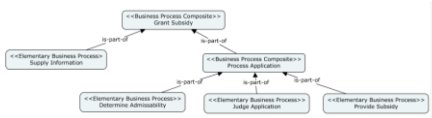
Archetypical industry process map – original representation	Archetypical industry process map – meta-model instantiation as concept map
	
	
	
	

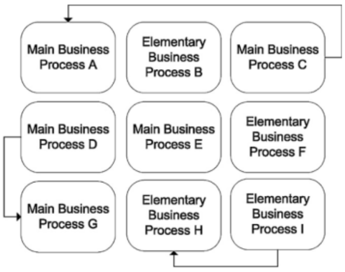
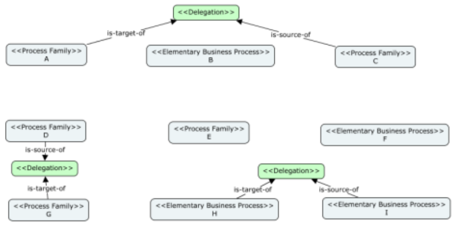
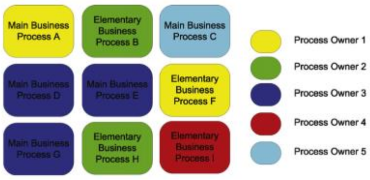
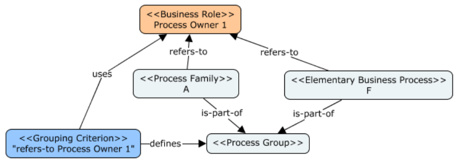
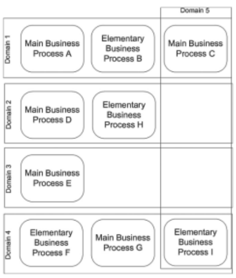
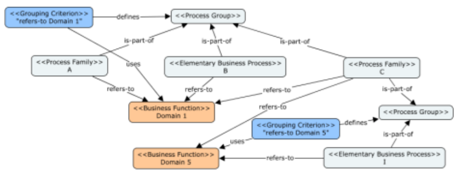
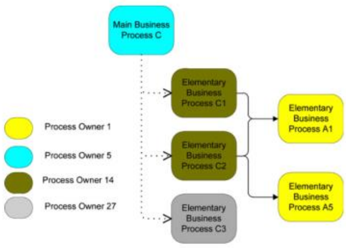
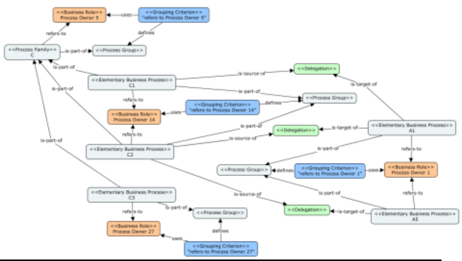
Table 3. Meta-model instantiations of business process architecture design approaches.

Business process architecture design approach – original representation of resulting process map	Business process architecture design approach – meta-model instantiation as concept map
	
	
	
	
	

5.3. Representation of Process Maps that Model Different Business Process Architecture Views

Van Nuffel and De Backer [18] propose six views of business process architecture of which they illustrate three with an example process map in an informal self-crafted notation. Table 4 shows these examples (left column) and how they can be represented using the concepts of the proposed meta-model (right column). The fourth row in the table presents another example taken from [18] which is positioned at a lower level in their business process decomposition structure to show the process variants making up a process family. The first row contains a model that is part of a view that shows main business processes, elementary business processes and dependency relationships between processes. Using the proposed meta-model, these business process architecture elements and relationships are modelled as process families, elementary business processes and delegation relationships, as shown in the concept map (right column).

Table 4. Representation of process maps that model different business process architecture views.

Business process architecture view – original representation	Business process architecture view – meta-model instantiation as concept map
	
	
	
	

According to [18], a dependency relationship means that a 'depender' process depends on the result of a 'dependee' process to perform its task, which corresponds semantically to the delegation relation of our meta-model. The depender is the source of the relation, while the dependee is the target of the relation. The model in the second row uses colour coding to identify the process owner of each main and elementary business process in the process architecture. The concept map demonstrates the grouping mechanism by grouping process families and elementary business process owner that refer to the same process owner, which is modelled as a business role (i.e., specialization of the EA element class of the meta-model). The view portrayed in the third row shows that main and/or elementary business processes can belong to more than one process group. The concept map has process family C referring to functional domains 1 and 5 (modelled as business functions, which specialize the EA element class of the meta-model). Hence, process family C is part of two process groups. Finally, in the fourth row the model shows the process variants (C1, C2 and C3) of a main business process (C). The concept map on the right shows the full model where elementary business processes are part of process families and both kinds of business process architecture elements are further grouped based on common reference to a process owner, modelled as business role like in row two. Process dependencies are modelled using the delegation relation of the meta-model.

5.4. Evaluation of the Meta-Model

As demonstrated in Sub-Sections 5.1 to 5.3, a large variety of notations is used to articulate process maps, regardless whether they originate in practice or in academia. The metamodel was demonstrated to model this variety of situations along with the fulfilling of the stated solution requirements. Furthermore, the meta-model has been evaluated in order to have a first insight about its validity, by considering the core requirements for a domain-specific language (DSL) [31], given that the proposed meta-model is intended to serve as the domain definition meta-model of a potential DSL for process mapping. The meta-model has been tested against these core requirements [31], such as summarized in Table 5.

Table 5. Evaluation of the metamodel

DSL req.	Justification
Conformity	The following general concepts were considered: (i) architecture; (ii) organizational context with regards to business processes; (iii) structure and relations between business processes. Elements at the business level in EA are included (in accordance to the relevant related literature). **(Req. 2)** The meta-model has been applied to represent a wide array of process maps and other business process architecture descriptions found in the literature (see Sub-Sections 5.1, 5.2 and 5.3).
Orthogonality	Each construct in the language was conceived to represent exactly one distinct concept in the domain.
Supportability	The suitability of enhancing EA modelling languages, in particular ArchiMate, will be considered, as we conceptualize the process map as an EA artefact along with the formal meta-model for a business process architecture description language.
Integrability	Conceptualization of the process map is driven as a business process abstraction that provides a black-box view on the organization’s business processes **(Req. 1)**. This can easily be integrated with the

DSL req.	Justification
	white-box perspective, supported by BPMN. A maximal integration with enterprise architecture description is obtained through the alignment with the ISO/IES/IEEE 42010 standard for architecture description. Different kinds of business process composites are considered. **(Req. 3)** Maximum freedom for instantiating the meta-model according to process mapping needs is provided, thereby limiting the number of constraints at the meta-model level.
Longevity	The meta-model was built to be aligned with the relevant standards about enterprise architecture and in the particular domain of business process architecture it generalizes existing proposals to reflect the mostly agreed upon concepts. The usage of generalized concepts (Enterprise Architecture Element, Business Process Architecture Element, etc.) facilitates the extension of the meta-model to be adapted to future situations.
Simplicity	A set of minimal constructs and constraints were considered Some complex mechanisms in process maps (e.g. aggregation) were simplified by applying design patterns (e.g. composition).

6. Conclusion and Future Work

In this paper, a meta-model of a business process architecture description language for representing process maps was presented, based on a process map conceptualization in the context of enterprise architecture. With the aim of testing the meta-model's ability to serve as a foundation of a universal representation language for process maps, a wide sample of process maps were instantiated and it has been shown how the meta-model fulfils core requirements for a domain-specific language.

The contribution of this research is the creation through our meta-model of a better understanding of business process architecture in relation to enterprise architecture, which could promote major advancements in the field and can be beneficial for both BPM and EA professionals and enterprise modelling in general. As a process map is a model of the business process architecture, it is complementary to any model that describes a view of the enterprise architecture (e.g., a goal model, a capability map, an application landscape, an infrastructure landscape). The main novelty in our meta-model, compared to the related work and apart from the generic Business Process Composite concept obtained through applying the composition pattern, was the introduction of the Enterprise Architecture Element as a 'placeholder' for any kind of element in any kind of enterprise architecture model that is related to a Business Process Architecture Element (e.g., goals or capabilities realized by elements of the business process architecture, elements of the business process architecture served by applications that are hosted on IT infrastructure nodes). This way a process map can be effectively integrated into the overall enterprise architecture model of an organization.
Whether this integration is easy to perform is another issue, that will depend on the choice of languages for the different architecture models. For instance, models

expressed in ArchiMate do not allow users to zoom in on the details of specific types of business processes and relationships between them. For this reason, our proposal was designed to overcome this limitation by proposing a new type of model (i.e., the process map as a model of business process architecture) that is aligned with industrial EA standards such as TOGAF and ArchiMate.

The main future work will be the design of a concrete syntax for the meta-model, which considers the suitability of EA modelling languages as a basis for defining a domain-specific language for process mapping. A software tool will be developed and empirical studies will be conducted to test the usability and usefulness of the proposed metamodel and syntax. The resulting feedback can serve to improve the metamodel in a new research cycle by following DSR. In terms of tooling, our proposal will seek to support the navigation between different EA and BPM models. For example, the user could be viewing a model in ArchiMate and by clicking on a business process element, the software would show in another window the detailed process model with the workflow (BPMN). As a result, new possibilities can arise to harmonize the use of general-purpose modelling languages from both fields (e.g., BPMN and ArchiMate).

Acknowledgements. This work is partially supported by: **BIZDEVOPS-Global** (ref. RTI2018-098309-B-C31), "Ministerio de Economía, Industria y Competitividad" (MINECO) & "Fondo Europeo de Desarrollo Regional" (FEDER); and **G3Soft** (SBPLY/17/180501/000150), "Consejería de Educación y Ciencia, Junta de Comunidades de Castilla-La Mancha" and FEDER.

References

1. Dijkman, R., I. Vanderfeesten, and H.A. Reijers, *Business process architectures: overview, comparison and framework.* Enterprise Information Systems, 2016. **10**(2): p. 129-158.
2. Dumas, M., et al., *Fundamentals of Business Process Management*. 2013, Berlin Heidelberg: Springer-Verlag.
3. Soffer, P. and Y. Wand, *Goal-driven multi-process analysis.* Journal of the Association for Information Systems, 2007. **8**(3): p. 175-203.
4. Malinova, M., H. Leopold, and J. Mendling, *An Empirical Investigation on the Design of Process Architectures*, in *11th International Conference on Wirtschaftsinformatik*. 2013: Leipzig.
5. Rosemann, M. *Process Portfolio Management*. BPTrends, 2006.
6. Heinrich, B., et al., *The process map as an instrument to standardize processes: design and application at a financial service provider.* Information Systems and E-Business Management, 2009. **7**(1): p. 81-102.
7. Lapouchian, A., E. Yu, and A. Sturm, *Design Dimensions for Business Process Architecture*, in *ER 2015, Lecture Notes in Computer Science*, P. Johannesson, Editor. 2015a, Springer International Publishing. p. 276-284.
8. Lapouchnian, A., E. Yu, and A. Sturm, *Re-Designing Process Architectures: Towards a Framework of Design Dimensions*, in *7th IEEE International Conference on Research Challenges in Information Science*. 2015b: Athens.
9. Odeh, Y., M. Odeh, and S. Green, *Aligning Riva-based Business Process Architectures with Business Goals Using the i* Framework*, in *3rd International Conference on Business Intelligence and Technology*. 2013: Valencia.
10. Poels, G., et al., *Investigating Goal-Oriented Requirements Engineering for Business Processes.* Journal of Database Management, 2013. **24**(2): p. 35-71.
11. Engelsman, W., et al., *Extending enterprise architecture modelling with business goals and*

requirements. Enterprise Information Systems, 2011. **5**(1): p. 9-36.
12. Nogueira, J., et al., *Leveraging the Zachman Framework Implementation Using Action – Research Methodology – A Case Study: Aligning the Enterprise Architecture and the Business Goals.* Enterprise Information Systems, 2013. **7**(1): p. 100-132.
13. The Open Group, *Open Group Standard. ArchiMate 3.0 Specification*. 2016.
14. Wierda, G., *Mastering ArchiMate. 2nd ed*. 2014.
15. Malinova, M. and J. Mendling, *Why is BPMN not Appropriate for Process Maps?*, in *36th International Conference on Information Systems*. 2015a: Fort Worth.
16. Malinova, M., H. Leopold, and J. Mendling, *An Explorative Study of Process Map Design*, in *CAiSE Forum 2014, Lecture Notes in Business Information Processing*, S. Nurcan and E. Pimenidis, Editors. 2015, Springer International Publishing.
17. Bider, I., et al., *A fractal enterprise model and its application for business development.* Software & Systems Modeling, 2016.
18. Van Nuffel, D. and M. De Backer, *Multi-abstraction layered business process modeling.* Computers in Industry, 2012. **63**(2): p. 131-147.
19. Rolland, C., S. Nurcan, and G. Grosz, *Enterprise knowledge development: the process view.* Information & Management, 1999. **36**(3): p. 165-184.
20. Scheer, A., *ARIS: Business Process Modeling*. 2000, Berlin Heidelberg: Springer-Verlag.
21. Francesconi, F., F. Dalpiaz, and J. Mylopoulos, *TBIM: A Language for Modeling and Reasoning about Business Plans*, in *ER 2013, Lecture Notes in Computer Science*, W. Ng, V. Storey, and J. Trujillo, Editors. 2013, Springer International Publishing. p. 33-46.
22. Frank, U., *Multi-perspective enterprise modeling: foundational concepts, prospects and future research challenges.* Software and Systems Modeling, 2014. **13**(3): p. 941-962.
23. Sandkuhl, K., et al., *Enterprise Modeling. Tackling Business Challenges with the 4EM Method*. 2014, Berlin Heidelberg: Springer-Verlag.
24. Roelens, B., W. Steenacker, and G. Poels, *Realizing strategic fit within the business architecture: the design of a Process-Goal Alignment modeling and analysis technique.* Software & Systems Modeling, 2017.
25. von Rosing, M., et al., *Combining BPM and EA in Complex IT Projects: A Business Architecture Discipline*, in *IEEE 13th Conference on Commerce and Enterprise Computing*. 2011. p. 271-278.
26. Hevner, A.R., et al., *Design science in Information Systems research.* Mis Quarterly, 2004. **28**(1): p. 75-105.
27. Peffers, K., et al., *A design science research methodology for Information Systems Research.* Journal of Management Information Systems, 2007. **24**(3): p. 45-77.
28. Wieringa, R., *Design Science as Nested Problem Solving*, in *4th International Conference on Design Science Research in Information Systems and Technology*. 2009: Philadelphia.
29. ISO/IEC/IEEE, *International Standard 42010. Systems and Software Engineering – Architecture Description*. 2011.
30. Poels, G., et al., *Conceptualizing Business Process Maps*. 2018: CoRR abs/1812.05395.
31. Kolovos, D., et al., *Requirements for Domain-Specific Languages*, in *1st ECOOP Workshop on Domain-Specific Program Development*. 2006: Nantes.
32. The Open Group, *Open Group Standard. TOGAF 9.1 Specification*. 2009.
33. Malinova, M., *A Language for Process Map Design*, in *BPM 2014 Workshops, Lecture Notes in Business Information Processing*, F. Fournier and J. Mendling, Editors. 2015, Springer International Publishing. p. 567-572.
34. Milani, F., et al., *Criteria and Heuristics for Business Process Model Decomposition Review and Comparative Evaluation.* Business & Information Systems Engineering, 2016. **58**(1): p. 7-17.
35. Aßmann, U., S. Zschaler, and G. Wagner, *Ontologies, Meta-models, and the Model-Driven Paradigm*, in *Ontologies for Software Engineering and Software Technology*, C. Calero, F. Ruiz, and M. Piattini, Editors. 2006, Springer-Verlag. p. 249-273.
36. Saeki, M. and H. Kaiya, *On Relationships Among Models, Meta Models, and Ontologies*, in

6th OOPSLA Workshop on Domain-Specific Modeling. 2006: Portland.
37. La Rosa, M., M. Dumas, and A. ter Hofstede, *Modeling Business Process Variability for Design-Time Configuration*, in *Handbook of Research in Business Process Modeling*, J. Cardoso and W. van der Aalst, Editors. 2009, IGI Global. p. 204-228.
38. Rosemann, M. and J. vom Brocke, *The Six Core Elements of Business Process Management*, in *Handbook on Business Process Management 1*, J. vom Brocke and M. Rosemann, Editors. 2015, Springer-Verlag: Berlin Heidelberg. p. 105-122.
39. Malinova, M., B. Hribar, and J. Mendling, *A Framework for Assessing BPM Success*, in *22nd European Conference on Information Systems*. 2014: Tel Aviv.
40. Gamma, E., et al., *Design Patterns: Elements of Reusable Object-Oriented Software*. 1995, Boston: Addison-Wesley Longman Publishing.
41. Malinova, M. and J. Mendling, *Leveraging Innovation Based on Effective Process Map Design: Insights from the Case of a European Insurance Company*, in *BPM – Driving Innovation in a Digital World*, J. vom Brocke and T. Schmiedel, Editors. 2015b, Springer International Publishing: Bern. p. 215-227.
42. Eid-Sabbagh, R.-H., R. Dijkman, and M. Weske, *Business Process Architecture: Use and Correctness*, in *BPM 2012, Lecture Notes in Computer Science*, A. Barros, A. Gal, and E. Kindler, Editors. 2012, Springer International Publishing. p. 65-81.

Geert Poels is head of the Management Information Systems research group at Ghent University (Belgium). He is full professor at the Faculty of Economics and Business Administration, Ghent University where he teaches Computer Science, Information Systems and Management of Information Technology subjects to Business Engineering and Business Administration students. He is member of the Ghent University Research Council. He also teaches, consults and directs master dissertation research at the IC Institute, which is part of the INNOCOM company (Beersel, Belgium). His areas of research are business process management, enterprise modelling, business ontology, digital transformation, agile requirements engineering, and IT governance (as co-developer of COBIT 2019). Mid 2019, he is promoter of 13 completed PhD research projects (11 at UGent and 2 at KU Leuven) and has 118 publications listed in Web of Science. His Google Scholar h-index is 30 with over 3000 citations recorded.

Félix García is Full Professor at the University of Castilla La-Mancha (UCLM), where he received his MSc (2001) and PhD (2004) degrees in Computer Science. He is member of the Alarcos Research Group and his research interests include business process management, software processes, software measurement and agile methods. http://orcid.org/0000-0001-6460-0353.

Francisco Ruiz is a full professor at the University of Castilla-La Mancha (UCLM), Spain. His current interests include enterprise architecture, business process management, information systems, and socio-demographic data analysis. Ruiz received a Master in chemistry-physics from Complutense University of Madrid, and a PhD in computer science from UCLM. Contact him at francisco.ruizg@uclm.es.

Mario Piattini is the director of the Alarcos Research Group and a full professor at the University of Castilla-La Mancha, Spain. His research interests include software and data quality, information-systems audit and security, and IT governance. Piattini received a Ph.D. in computer science from Madrid Technical University, Spain.

Received: November 18, 2018; Accepted: July 30, 2019