

Representation and analysis of enterprise models with semantic techniques: an application to ArchiMate, e3value and business model canvas

Artur Caetano^{1,2} · Gonçalo Antunes² · João Pombinho^{1,2,3} · Marzieh Bakhshandeh² · José Granjo^{2,4} · José Borbinha^{1,2} · Miguel Mira da Silva^{1,2}

Received: 22 June 2014 / Revised: 23 November 2015 / Accepted: 3 March 2016 /
Published online: 19 March 2016
© Springer-Verlag London 2016

Abstract Enterprise models assist the governance and transformation of organizations through the specification, communication and analysis of strategy, goals, processes, information, along with the underlying application and technological infrastructure. Such models cross-cut different concerns and are often conceptualized using domain-specific modelling languages. This paper explores the application of graph-based semantic techniques to specify, integrate and analyse multiple, heterogeneous enterprise models. In particular, the proposal described in this paper (1) specifies enterprise models as ontological schemas, (2) uses transformation mapping functions to integrate the ontological schemas and (3) analyses the integrated schemas with graph querying and logical inference. The proposal is evaluated through a scenario that integrates three distinct enterprise modelling languages: the business model canvas, e3value, and the business layer of the ArchiMate language. The results show,

✉ Artur Caetano
artur.caetano@tecnico.ulisboa.pt

Gonçalo Antunes
goncalo.antunes@tecnico.ulisboa.pt

João Pombinho
jtmp@tecnico.ulisboa.pt

Marzieh Bakhshandeh
marzieh.bakhshandeh@tecnico.ulisboa.pt

José Granjo
jose.granjo@screeninteraction.com

José Borbinha
jose.borbinha@tecnico.ulisboa.pt

Miguel Mira da Silva
mms@tecnico.ulisboa.pt

¹ IST, University of Lisbon, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal

² INESC-ID, Rua Alves Redol 9, 1000-029 Lisboa, Portugal

³ NOS Comunicações, Rua Actor António Silva 9, 1600-404 Lisboa, Portugal

⁴ Screen Interaction AB, Arne Beurlings Torg 3, 164 40 Kista, Stockholm, Sweden

on the one hand, that the graph-based approach is able to handle the specification, integration and analysis of enterprise models represented with different modelling languages and, on the other, that the integration challenge resides in defining appropriate mapping functions between the schemas.

Keywords Enterprise modelling · Model analysis · Model integration · Semantic techniques · Ontology · ArchiMate · e3value · Business model canvas

1 Introduction

Enterprise models play an important role in the governance and transformation of digital organizations where the business is supported in some degree by information systems and information technology [1–4]. Enterprise model analysis examines the artefacts, properties and dependencies of a model to generate information that can be used to assess, transform or redesign organizational systems [5–7]. Model analysis uses information within the models or information about the models to drive the overall analysis process [8].

A *business model* describes how an organization creates and delivers value, and depicts the organization from a high level of abstraction [9]. A *value network* describes how a group of organizational entities, such as suppliers, producers and clients, exchange value across the value network, and how value flows between the entities, while each implements its own business model [10]. Since a business model abstracts the organization from a high-level strategic perspective, it does not detail how the business is operated or automated. To achieve such a goal, an enterprise model describes business processes and how processes are realized by applications and technology. An *enterprise modelling language*, such as ArchiMate [11], can be used for this purpose.

Business models, value networks and enterprise architecture models address different organizational concerns and provide different views on an organization: a business model describes the organization's strategy in terms of value propositions and value creation; an enterprise architecture model describes how an organization operates, often through the description of the relationships between its processes, resources, applications and technology; and a value network model describes the role of each organization within the wider value chain. When the system of interest is a single organization, then modelling these three different concerns entails specifying the three underlying models for that organization. But when the system of interest includes multiple organizations, the model landscape comprises several models: there will be a shared value chain model that represents all the organizations, along with at least one business model and one enterprise architecture model for each individual organization that pertains to the value chain. Although these models address different concerns, being able to jointly represent and analyse them is attractive because it contributes to understand the relationships and dependencies between the concepts that are otherwise distributed across different models. Thus, the integrated models enable to check how the business model of an organization fits into the wider value network, and how the business operations are actually realizing the business model [12–15].

There are different approaches to deal with the specification, integration and analysis of enterprise models. These include multi-perspective enterprise modelling [16–19], model mapping [14, 20, 21], federated enterprise modelling [3, 22–25], as well as the application of knowledge representation techniques to enterprise modelling [26–28]. This paper uses semantic techniques to represent and integrate enterprise models through the definition of mapping functions that relate conceptual schemas. The content and structure of the integrated

models can then be analysed with graph querying and logical inference. In other words, this paper assesses the suitability of *applying ontologies to represent and analyse federated enterprise models*. In particular, the research here reported explores the following avenues:

- The application of ontologies to *represent* enterprise models. The paper represents the business model canvas [9], e3value [29], and a subset of ArchiMate's business layer [11].
- The application of ontologies to *integrate* enterprise models. Integration is accomplished through the specification of mapping functions.
- The application of computational semantic techniques (in particular, graph querying and logical inference) to *analyse* the integrated enterprise models.

The contribution of this paper is presented as a design artefact [30,31], containing a set of ontological schemas designed to represent, integrate and analyse federated enterprise models. The artefact is demonstrated through the specification, integration and analysis of a model landscape comprising three enterprise modelling languages: the business model canvas (BMC), e3value and ArchiMate. To produce the design artefact, the different models were specified as individual ontological schemas and then integrated through mapping. The artefact was evaluated with a scenario that analyses the integrated models. The main contribution of the paper is assessing the application of semantic techniques with regard to enterprise model specification, integration and analysis. These techniques provide support for creating and validating models for different but related perspectives of the enterprise [7, 15, 32].

The remainder of this paper is structured as follows. The next section introduces the BMC, e3value and the ArchiMate modelling languages, along with a short introduction to ontology engineering. Section 3 describes the proposed approach to the representation, integration and analysis of enterprise models. Section 4 applies the approach to the integration of the BMC, e3value and ArchiMate languages. Section 5 evaluates the design of the artefact with the analysis of an electronic health service business model. Finally, Sect. 6 concludes the article.

2 Background

This section introduces the business model canvas, e3value and ArchiMate. The combination of these enterprise modelling languages emphasizes their collective capability to represent the complementary perspectives of *teleology* and *ontology*. The concept of teleology originates from the Greek *telos*, which means “end”, “purpose” or “goal”. It concerns the explanation of a thing in terms of the end for which the thing exists or was produced for. The teleological perspective of an enterprise is presented in duality with its ontological perspective. Ontology is defined as the study or description of things as they are. As explained in the following subsections, the business model canvas and e3Value languages mainly focus on the teleological perspective, i.e. on the “purpose” or “goal” of an organization and thus abstract from detailed enterprise operations, whereas ArchiMate focusses on the ontological perspective.

2.1 Business model canvas

A business model describes how an organization creates, delivers and captures value [9]. Since organizations depend on the value they create and on how they capture it, the development of business models is often considered a vector for innovation. In the words of Chesbrough [33], “a company has at least as much value to gain from developing an innovative new business model as from developing an innovative new technology”. The business model canvas (BMC) is a “visual template that represents the business model of an organization and translates it

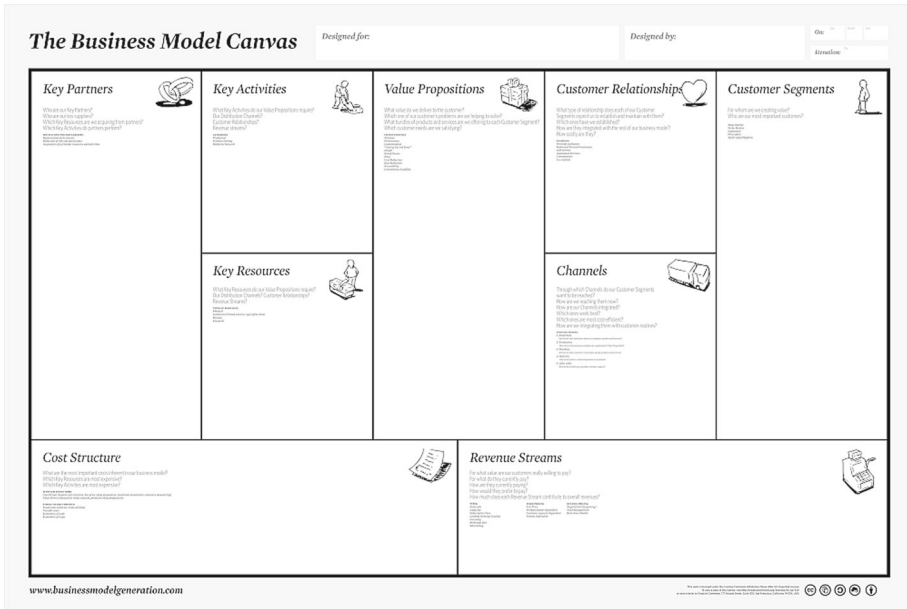


Fig. 1 The business model canvas (from [9])

Table 1 The concepts of the business model canvas (adapted from [9])

Concept	Domain	Definition
Value proposition	Offering	A bundle of services or products that create value for a Customer Segment
Customer segment	Customers	A group to whom the organization delivers Value Proposition
Channel	Customers	The mechanism used by the organization to deliver a Value Proposition to a Customer Segment
Customer relationship	Customers	The type of relationship towards a Customer Segment (e.g. customer acquisition, customer retention, upselling)
Key activities	Infrastructure	The core activities that create or support the creation of the Value Proposition
Key resources	Infrastructure	The assets required for the organization to operate its business model. These resources are often used by the Key Activities.
Key partnerships	Infrastructure	The external entities (e.g. suppliers, distributors) that the organization relies on to operate its business model
Cost structure	Finances	The costs incurred while creating and delivering the Value Proposition and maintaining Customer Relationships
Revenue streams	Finances	How each Customer Segment generates revenue

into explicit knowledge” [9]. The focus of the BMC is describing a single organization from a high-level strategic perspective. As depicted in Fig. 1, the canvas comprises nine building blocks, whose content is described in Table 1.

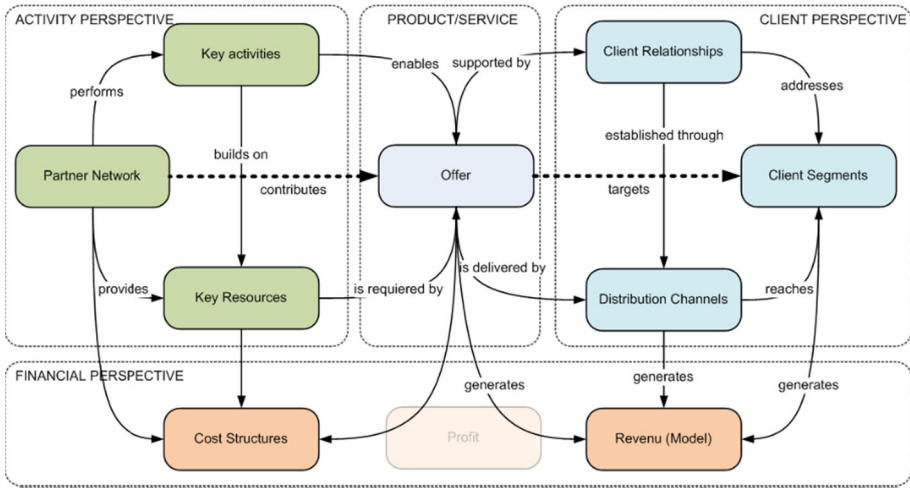


Fig. 2 The business model ontology (adapted from [34])

The BMC is often populated through collaborative and visual thinking techniques, such as ideation, brainstorming, prototyping and storytelling [9]. These techniques promote communication between the stakeholders and to foster innovation. Although the BMC is presented as a visual template, it is grounded on the business model ontology (v. Fig. 2) that establishes the foundations behind its concepts and relationships [34].

2.2 e3value

A value network provides a common understanding of a business idea that is executed by a network of actors that jointly create, distribute and consume value [12]. The e3value considers that value transactions are economically reciprocal, i.e. an actor providing a value object expects to receive a reciprocal value object in return. Value models communicate the role of a single company within the value network, and make possible analysing how the network creates and delivers value as the result of inter-organizational cooperation[10]. Figure 3 depicts the e3value meta-model, and Table 2 describes its main concepts.

Value networks are abstracted as a set of value transactions occurring between actors. While the business model canvas focuses on a single node of the value network, the e3value encompasses the overall value network [13]. BMC is designed to convey and discuss subjective business ideas while lowering the adoption barriers to its users [9]. The e3value is designed not only to visually communicate a value network model but also to support model checking and numerical computation, e.g. to simulate the economic viability of different value networks. Therefore, combining BMC and e3value models makes possible describing and analysing the same scenario according to complementary perspectives [13, 14, 36, 37].

2.3 ArchiMate

ArchiMate is an enterprise architecture modelling language that describes the relationships between business services, processes and the underlying application and technological infrastructure. To do so, the concepts of ArchiMate are organized according to a framework composed of three layers: business, application and infrastructure [11]. The internal busi-

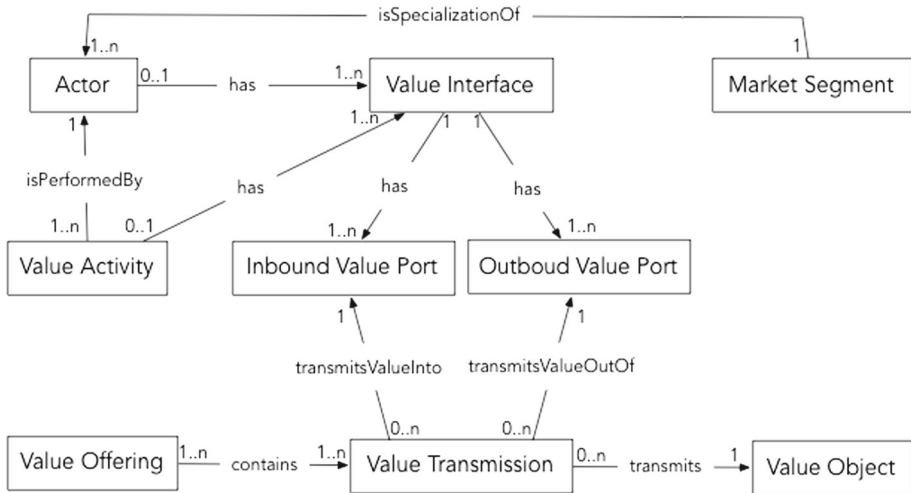


Fig. 3 The e3value meta-model (adapted from [12,35])

Table 2 The concepts of e3value (adapted from [10,12])

Concept	Definition
Value object	A service or a product that is of value to at least one Actor
Actor	An economically independent entity able to exchange Value Objects
Market segment	A group of Actors that evaluate the same Value Object in a similar way
Value port	An abstraction of how an Actor provides or requests Value Objects. It is realized by one or more Value Activities
Value activity	An activity performed by an Actor that directly yields profit or that increases the value of a Value Object
Value offer	A group of Value Ports with the same value flow direction (either inbound or outbound)
Value interface	A group of Value Offers with economic reciprocity. It consists of one or more pairs of Value Offers
Value exchange	The transmission of a Value Object from an outbound to an inbound Value Port
Value transaction	The set of all Value Exchanges associated to the same Value Interface. A Value Transaction is atomic

ness operations interface with the external environment by providing business services and products to the organization’s external clients. Business services are internally realized by business processes that use and transform business objects, which are assigned or executed by business actors. The application layer describes how automated or semi-automated business process are supported by applications, along with the structure, behaviour and dependencies of the applications. The technological infrastructure layer describes how applications are supported by technology in terms of processing, communication and storage. Thus, each layer isolates a domain and provides services that support the operation of the upper adjacent layer. The ArchiMate framework further classifies all of the concepts found in these three layers according to three types: behaviour, active structure and passive structure. These types conform to the same set of rules: behaviour elements are always assigned to active struc-

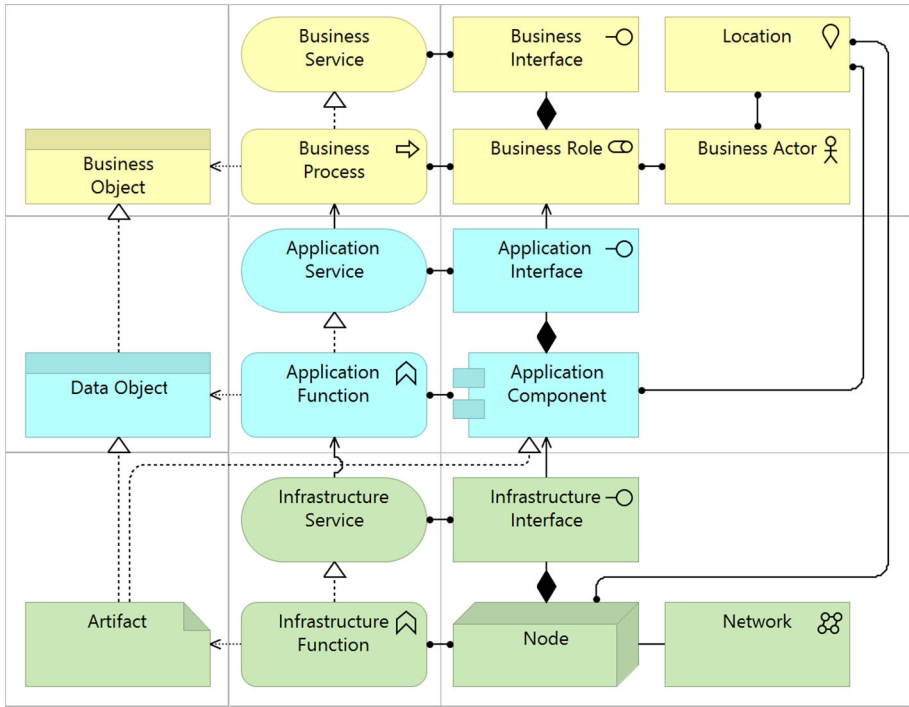


Fig. 4 The ArchiMate meta-model (adapted from [11,21])

Table 3 A subset of concepts from ArchiMate’s business layer (adapted from [11])

Concept	Definition
Value	The relative worth or utility of a Business Service or a Product
Product	A coherent collection of Business Services offered to an internal or external Actor , e.g. an internal department, an external customer
Business service	A behavioural element that fulfils a business need of an Actor
Business interface	A point of access from where a Business Service is made available to the environment
Business process	A behavioural element that sequences activities as a means to create Products or deliver Business Services . A Process is <i>assigned to</i> Actors and <i>uses</i> Business Objects
Business object	A passive structural entity that is relevant from a business perspective to support the operation of the organization
Actor	An active structural entity that is capable of performing behaviour

ture elements, and passive structure elements are only accessed by behaviour elements. This means that active structure elements perform behaviour, while behaviour elements specify how the state of passive structure elements changes. Thus, a passive structure element is only able to change state as the result of behaviour performed on it by some active structure element. Figure 4 shows a simplified view of ArchiMate’s meta-model, and Table 3 summarizes a subset of the concepts found in ArchiMate’s business layer.

2.4 Ontologies

The term *ontology* originates from the Greek *ontos* (being) and *logos* (word). From a philosophical perspective, it is defined as a “systematic explanation of existence”, while in computer science it is often used to convey the notion of a “formal, explicit specification of a shared conceptualisation” [38]. The term *conceptualization* refers to an “abstract, simplified view of the world” that contains “the objects, concepts, and other entities that are assumed to exist in some area of interest” along with “the relationships that hold among them” [39]. The term *explicit* refers to the definition of the “type of concepts used, and the constraints on their use” while *formal* refers to the fact that the conceptualization “should be machine readable”. Finally, *shared* means that the ontology “captures consensual knowledge” [40].

Ontology integration is about identifying shared concepts and relationships between two or more ontologies. Integration relates these elements and uses the resulting knowledge for some purpose, such as merging a group of ontologies into a single ontology [41]. Integration is closely related to ontology *alignment*, *merging* and *mapping*. Ontology *alignment* concerns the discovery of contextual correspondences between two or more ontologies. *Merging* is the creation of a new ontology through the “union of the source ontologies” while capturing “all the knowledge from the original ontologies”. Merging can be challenging since “all correspondences and differences between the ontologies need to be reflected in the merged ontology”. Ontology *mapping* concerns the “representation of the correspondences between ontologies”. Correspondences are typically “stored separately from the ontologies” since they are not “part of the ontologies themselves”. Thus, mapping often represents the results of aligning different ontologies [42].

Reasoning denotes a mechanism that takes facts that are implicit in an ontology and transforms them to explicit facts. Reasoning can be classified as deductive, inductive and abductive [43]. Deductive reasoning derives a logical consequence from a set of assumptions. For example, if all swans are white, and if a bird is a swan, then, by deduction, the bird is white. Induction produces conclusions with a degree of uncertainty: if all swans observed so far are white, then, by induction, all swans are likely to be white, although that fact may not hold universally true. Abduction infers a condition from a consequence: given that swans can fly and that flying is a means of motion, then if a swan has moved, thus, by abduction, the swan has flown.

Description logics (DL) are “a family of logic-based knowledge representation languages suitable for the representation of ontologies”, which can be seen as “a decidable fragment of first-order logic” [44]. DL describe domains in terms of *concepts*, *roles*, and *individuals*. Roles and concepts are related using logical statements named *axioms*. Axioms can be of two kinds: terminological and assertional. *Terminological axioms* describe concepts and properties of the concepts, while *assertional axioms* are statements about the individuals or instances that are compatible with terminological axioms. A T-Box is a finite set of terminological axioms, and a A-Box is a finite set of assertional axioms. Together, the T-Box and A-Box statements define a *knowledge base*.

Several varieties of DL exist with differing degrees of expressiveness. The base DL is the *Attributive Language with Complements (ALC)*, which enables relating the concepts with the following functions: union, intersection, complement, universal restriction and existential restriction. The expressiveness of *ALC* can be augmented with transitive and inverse roles, role hierarchy, cardinality restrictions, qualified cardinality restrictions, concrete domains and enumerated classes.

3 Enterprise model representation and analysis

This section describes fundamentals of the proposed approach, namely schema representation (Sect. 3.1), schema integration (Sect. 3.2) and schema analysis (Sect. 3.3).

3.1 Model representation

A domain model is specified using a conceptual schema that defines its domain-specific entity types and relationship types. Thus, the instances of each entity and relationship are classified according to the types defined in the schema. These conceptual schemas and the underlying information base can be represented as triples (subject, predicate, object) or as ontologies [45, 46].

Enterprise models often involve multiple domains and stakeholders. To manage this fact, ISO 42010 recommends defining viewpoints and generating views according to the viewpoints to promote separation of concerns, i.e. isolating certain aspects of an architecture according to the concerns of its stakeholders [47]. Since ontologies can be analysed with graph-based and semantic queries, a viewpoint can be specified as a set of axioms from which views are derived from. This approach makes possible specifying viewpoints such as “determine all entities of type Business Process that relate with at least two entities of type Business Object via the *use* relationship”.

3.2 Model integration

Enterprise models cross-cut multiple domains, ranging from strategy, goals and business processes, to information systems, performance indicators and technological infrastructure. One approach to handle multiple-domain models is defining a common meta-model that unifies all the relevant concepts found in the multiple domains being addressed [22]. A benefit is that the meta-model can be designed with conceptual consistency as a principle, but at the expense of specificity and design effort. Moreover, such meta-models often need to generalize or specify concepts at a high level of abstraction in order to make concept unification possible. This approach may prove not to be adequate to address the concerns of the stakeholders of the model in terms of the required granularity or level of detail. As a result, some authors argue that this approach is only appropriate to integrate domains that already share similar concepts at a similar level of abstraction [5, 48–50]. An alternative approach is using federated models or domain-specific modelling languages [4, 18–20, 22, 24, 28, 51]. Domain specificity is also acknowledged by the application of situational modelling to enterprise architecture management as it facilitates the selection of a suitable modelling approach according to the constraints and goals of the project at hand [3, 48, 52, 53].

This article proposes a design artefact that consists of a set of domain-specific ontologies (DSO) that are linked together by functions that act as transformation maps [7, 54, 55]. Each DSO is regarded as a domain-specific modelling language that conceptualizes and represents the concerns bounded to a given domain. The domain-specific ontologies can be complemented by an upper ontology (UO). An UO is a schema that defines generic entity and relationship types from where the domain-specific ontologies can be derived from. Using an UO facilitates the integration of domain-specific ontologies and guides the development of new ontologies [56]. Examples of related upper ontologies include the Unified Foundational Ontology [27], the Simple Knowledge Organization System Reference [57], and the General Formal Ontology [58].

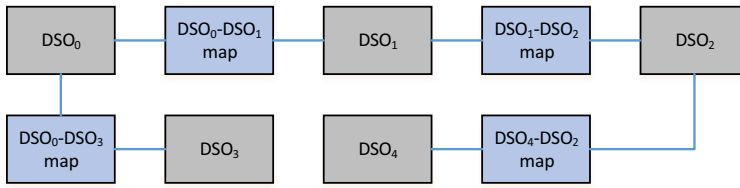


Fig. 5 Using four transformation maps to federate five domain-specific ontologies

The modelling of multiple domains is addressed through the specification of multiple DSO along with transformation maps that link the domain-specific concepts according to the concerns of the stakeholders. Thus, a transformation map states the contextual relationships between a pair of DSO. As a result, a transformation map cannot be regarded as universal as it is designed to translate how concepts relate in a specific context and for a specific purpose.

A *transformation map* is defined as a conceptual schema and specifies the semantics behind the relationships between schemas. The same schema may map to more than one schema, which means that the same concept may be mapped to multiple concepts in different schemas. A transformation map describes the transformation *functions* between the concepts pertaining to a pair of DSO. A function may describe several types of concept relationships, such as isomorphic 1:1 equivalence relationships, 1:n aggregation, concept classification, as well as rule-based or pattern-based transformations [7, 20, 54]. The definition of a function must also consider mapping deficiencies, such as incompleteness, redundancy, excess and overload [59]. There are specific techniques to handle these deficiencies depending on the integration requirements [46, 60].

Figure 5 exemplifies the transformation mapping between five ontological schemas. Each map defines the structural and semantic relationships that are required to match their types.

3.3 Model analysis

Model analysis focuses on the examination of model artefacts, properties and dependencies [5]. Since the conceptualization of multiple domains implies the definition of a landscape of schemas (DSO), the analysis task can be classified as:

- *Intra-schema analysis*, when it targets a single schema S ,
- *Inter-schema analysis*, when it targets a pair of schemas (S_1, S_2) , or
- *Chained analysis*, when it combines intra- and inter-schema analysis over two or more schemas (S_1, \dots, S_n) .

The analysis of schemas can be accomplished through logical inference or graph analysis. Graph analysis is applicable because an ontology intrinsically represents a labelled and directed graph. This means that the corresponding RDF(S) graph of (subject, predicate, object) triples can be analysed using a semantic query language such as SPARQL [61].

Description logics uses inference to provide five types of reasoning: subsumption, instance checking, relation checking, concept consistency, and knowledge base consistency. Reasoning can be used to analyse the entity and relationship types specified in the conceptual schema as well as the individual instances of these types. *Subsumption* and *instance checking* analyse model heterogeneity and involve classifying types and its individuals in order to refactor the model. *Relation checking* determines how entity types relate to each other through a chain of relationship types. Thus, it supports dependency analysis, i.e. checking whether two elements are related to each other and determining the nature of the relationship.

It also supports analysing the coverage of a model, e.g. to determine how the model’s artefacts intersect architecture layers. It also plays a role in interface analysis as it enables assessing cohesion and coupling of an artefact. Finally, *relation checking* aids the computation of metrics, e.g. used for complexity and cost/benefit analysis. *Concept consistency* can be used to verify the existence of contradictions between the definitions of a type. This can be used to determine whether rules or requirements are met by a schema. *Knowledge base consistency* checking can be used to verify whether the definition of an individual entity or relationship complies with the definition of the corresponding type, i.e. whether a model is valid against the specification of a meta-model.

The application of reasoning to analyse dependencies is limited to the relationships between different types since it is not possible to analyse relationships between type properties [50]. Reasoning is also not a good candidate to perform the computation of metrics required for complexity analysis and cost/benefit analysis. In these scenarios, reasoning requires to be complemented with other analytical techniques, such as graph analysis.

4 Federating ArchiMate, e3value, and BMC

This section describes an application of the concepts described earlier to the representation, integration and analysis of the ArchiMate, e3value and BMC modelling languages as depicted in Fig. 6. Integrating these models into a federated model landscape proves to be attractive because it contributes to understanding how the business model of an organization fits into a value network, and how the processes of an organization realize the corresponding

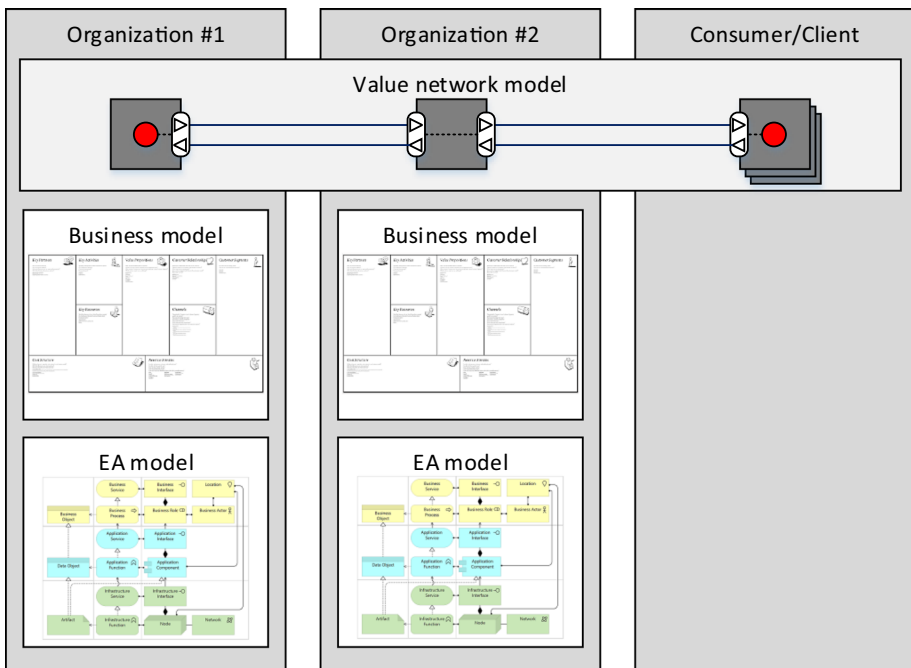


Fig. 6 The federated BMC, e3value and ArchiMate models

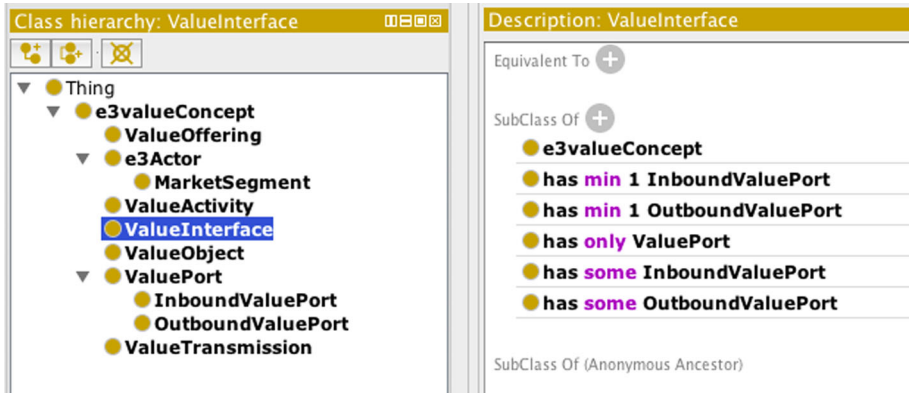


Fig. 7 The T-box OWL schema of e3value's Value Interface

business model [13, 14, 35]. This goal is achieved by performing three steps: (1) representing schemas that specify the ArchiMate, BMC and e3value modelling languages, (2) defining the transformation maps that integrate these schemas for the purpose of alignment checking and (3) analysing the federated models. These steps are described in the following three sections.

4.1 Model representation

Model representation concerns creating conceptual schemas that specify the domain to be analysed. It entails specifying the concepts behind a modelling language as an ontological schema. This section deals with representing the e3value, business model canvas, and ArchiMate meta-models using ontological schemas. Therefore, the e3value concepts are specified as a OWL-DL schema that represents its T-box. OWL object properties define the constraints from the meta-model, such as the restrictions that e3value defines over the concept of Value Interface as depicted in Fig. 7. ArchiMate entities are specified as OWL Classes, and relationships as Object Properties. Restrictions are specified using Inverse Object Properties and Super Object Properties so that ArchiMate's derived relationships can be extracted through the use of reasoners. Dependencies between entities are represented with Super Object Property chains. The depends from and depends to Object Property are inverse and transitive Super Object Property relationships that specify ArchiMate's aggregation, composition, assignment, usage and realization relationships.

4.2 Model integration

Schema federation is accomplished through the specification of *transformation maps* that relate the concepts defined in the domain-specific ontologies. This section describes the integration of the ArchiMate, e3value and BMC schemas using three domain-specific ontologies. But other schema integration configurations are possible, e.g. using ArchiMate as an upper ontology and considering e3value and BMC as two DSO. Other alternatives include using an ontology such as TOVE or the Enterprise Ontology [26] as upper-level ontology and integrating the ArchiMate, e3value and BMC schemas with this UO.

A transformation map defines functions that map the relevant types between a pair of schemas. Table 4 describes the source material that was used to define the transformation

Table 4 Sources used to ground the mapping between BMC, e3value and ArchiMate

Languages	Source
BMC ↔ e3value	Gordijn et al. [13]
BMC → ArchiMate	Jacob et al. [36], Meertens et al. [37], Fritscher and Pigneur [62]
e3value → ArchiMate	de Kinderen et al. [14,63], Pombinho et al. [35], Pombinho and Tribolet [64]

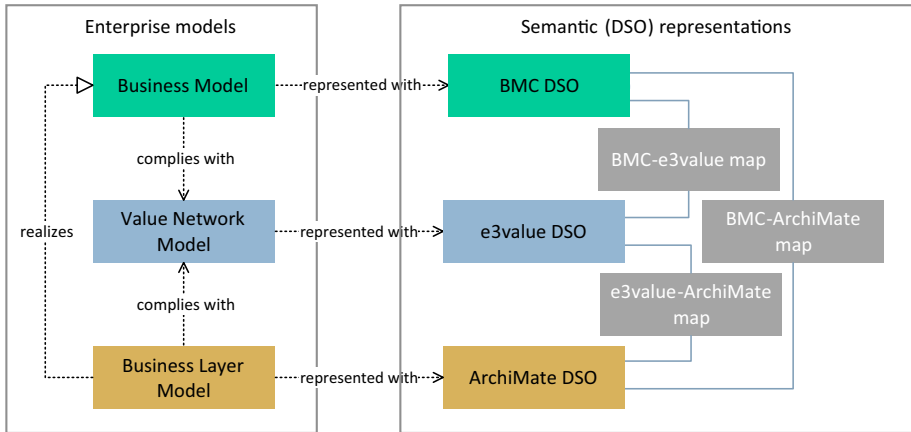


Fig. 8 Relationships between the three enterprise modelling languages from a value exchange perspective and the corresponding DSO representations

functions between e3value, business model canvas and the ArchiMate languages. It is important to remark that is not a goal of this paper to demonstrate the soundness of these mappings but instead to assess whether semantic techniques can be used to represent, integrate and analyse different enterprise models, regardless of the actual mapping functions.

Schema representation and integration are context dependent as the mappings are designed to address specific concerns. For this demonstration, we set as the primary concern the *analysis of business models from a value network perspective*. This means that the e3value schema defines the baseline to analyse how value exchange occurs, while the BMC and ArchiMate schemas are used to understand the strategic and operational perspectives of value exchange. Thus, federating these schemas allows to understand the alignment between these domains. Figure 8 depicts the federation strategy here described, where the value network concepts (e3value) are satisfied by the strategic business model (BMC) and realized by the operational business concepts (ArchiMate). One value network model relates to many BMC and ArchiMate models, as there is one BMC and ArchiMate model for each organization that is part of the value network.

An overview of the mapping functions between the three schemas is shown in Fig. 9: e3value: Key Partners and e3value: Customer Segments have direct correspondence to ArchiMate: Actors and to ArchiMate: Business Actors. A BMC: Revenue Stream corresponds to e3value: Value Transmission. Value transmission corresponds to a ArchiMate: Business Service and gets value from BMC: Key Partners using a certain BMC: Key Activity to transmit a BMC: Value Proposition to a certain customer segment through

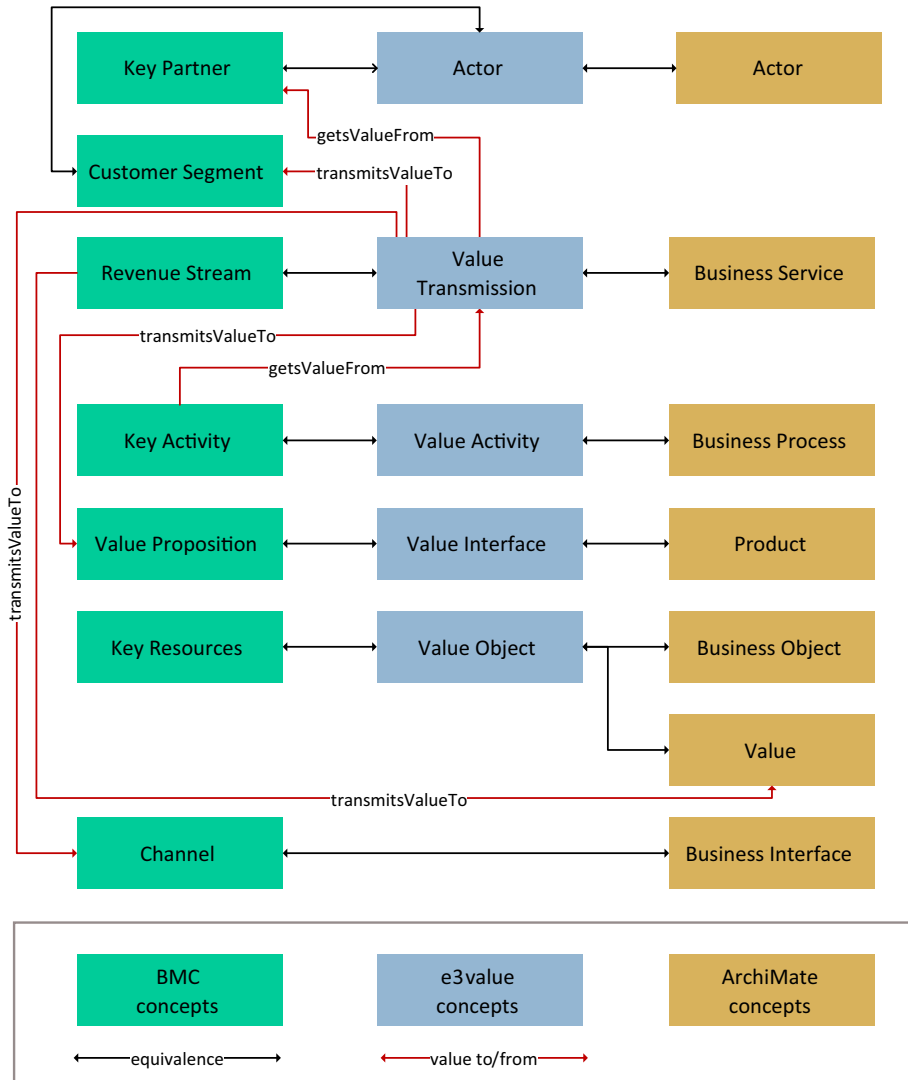


Fig. 9 Overview of the mapping functions between BMC, e3value and ArchiMate

a BMC: Channel. A BMC: Value Proposition corresponds to e3value: Value Interface and ArchiMate: product. A BMC: Key Activity corresponds to e3value: Value Activity and to ArchiMate: Business Process. The BMC: Key Resources correspond to e3value: Value Object that map to tangible ArchiMate: Business Objects. Finally, BMC: Channel corresponds to ArchiMate: Business Interface.

The set of mapping functions between each pair of schemas is implemented as a separate ontology so that the relationships do not become coupled to a specific meta-model. Figure 10 shows part of the OWL-DL mapping functions that specify e3value’s Value Transmission concept. The next sections detail the rationale of the mapping between the e3value, BMC and ArchiMate languages.

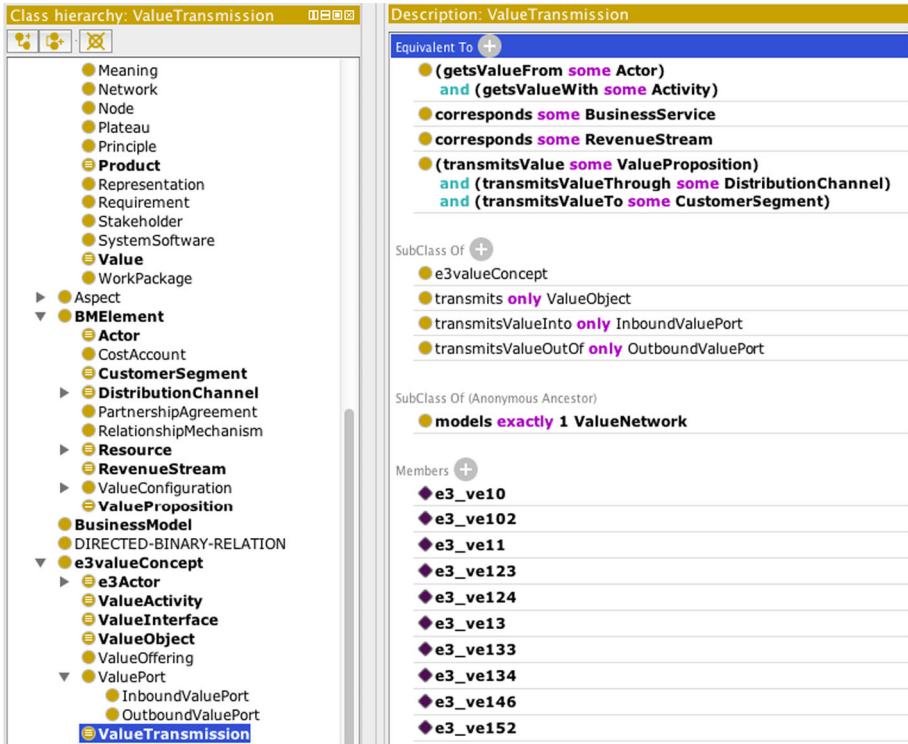


Fig. 10 Partial OWL-DL specification of e3value’s value transmission

4.2.1 Mapping BMC and e3value

Although modelling an organization from different perspectives, the business model canvas and e3value share a number of similarities. Gordijn et al. [13] compares and maps the concepts between these two languages. Table 5 summarizes how the five core concepts of e3value relate to BMC. Note that the BMC classifies actors as either clients or partners, while that distinction is not evident in e3value. As a result, the BMC defines customer segments, customer relationships, channels, value proposition and revenue streams as client-side building blocks, whereas key partners, key activities, key resources and cost structure are considered partner-side building blocks.

4.2.2 Mapping BMC to ArchiMate

The mapping functions described in Table 6 result from prior work on attempting to align the concepts found in the BMC and ArchiMate languages [36,37,62]. Not all transformations are considered here. In particular, the relationships between internal Cost Structure and Value were disregarded since only the external business services are visible from a value network perspective.

4.2.3 Mapping e3value to ArchiMate

The integration of e3value and ArchiMate models is constrained by the semantic gap that derives from the different abstractions used to represent the same economic transactions [14].

Table 5 Mappings between BMC and e3value

BMC	e3value	Relationship	Rationale
Customer segment	Actor	Equivalence (1:1)	Customer Segments are groups of economic entities (Actors) that generate profit
Customer segment	Market Segment	Equivalence (1:1)	Market Segment is a kind of collective Actor
Key partner	Actor	Equivalence (1:1)	Key Partners are external Actors that collaborate with an organization
Channel	Value Transmission	Aggregation (1:n)	Value Transmission delivers Value to Customers through one or more Channels
Key activity	Value Activity	Equivalence (1:n)	Key Activities are Value Activities that specify how the business model is executed
Revenue stream	Value Transmission	Aggregation (1:n)	Revenue Stream corresponds to a set of Value Exchanges
Key resources	Value Object	Equivalence (1:1)	Key Resources are the objects that contribute to the generation of Value
Value proposition	Value Interface	Equivalence (1:1)	Value Proposition corresponds to the Value Objects that flow through the ports of a Value Interface

Table 6 Mapping from BMC to ArchiMate's business layer

BMC	ArchiMate	Relationship	Rationale
Customer segment	Business Actor	Equivalence (1:1)	Customer Segments are external groups of Actors
Key partner	Business Actor	Equivalence (1:1)	Key Partners are external Actors that support the operation of the business model
Channel	Business Interface	Equivalence (1:1)	Channel interfaces with Customer segments to deliver Value Propositions
Revenue stream	Value	Equivalence (1:1)	Value depicts the net result of a Revenue Stream
Value proposition	Service, Product	Aggregation (1:n)	Service or Product represent the externalization to a customer of a Value Proposition
Value proposition	Value	Aggregation (1:n)	Value depicts the relative worth of a Service or Product from the perspective of an Actor
Key activity	Process, Function	Equivalence (1:n)	Key Activity is realized as a Business process or Business Function

Table 7 Mappings from e3value's concepts to ArchiMate's business layer

e3value	ArchiMate	Relationship	Rationale
Actor	Business Actor	Equivalence (1:1)	Actor is a Business Actor that generates profit or increases the utility of a Product or Service
Market segment	Business Actor	Equivalence (1:1)	Market Segment is a kind of Actor
Value interface	Product	Equivalence (1:1)	Value Interface groups Value Objects that are provided by an Actor. The Value Objects are realized as Business Services and aggregated as Products
Value transmission	Business Service	Equivalence (1:1)	Value is transmitted through the usage and completion of a Business Service
Value activity	Business Process	Equivalence (1:n)	Value Activity is a Business process that supports the delivery of Value Objects
Value object	Business Object	Equivalence (1:n)	Value Object is a tangible or intangible Business Object
Value object	Value	Equivalence (1:n)	Value is the perceived worth of a Business service or Product, which is transmitted as a Value Object

One approach to address the abstraction gap is using an intermediate language as described by Pombinho et al. [35], Pombinho and Tribolet [64] and de Kinderen et al. [63], who use DEMO's transaction model [65,66] as a bridge between e3value and ArchiMate. The main advantage of using DEMO is exploiting PSI theory's distinction axiom. This axiom states that the acts performed by an organization are classified either as ontological, infological and datalogical. These classes relate with the three human abilities: ontological acts relate to *performa* (deciding, judging, etc.), infological acts relate to *informa* (deducing, reasoning, computing, etc.), and datalogical acts relate to *forma* (storing, transmitting, etc.). DEMO also separates the teleology of an enterprise from its ontology, and focuses on the later while abstracting from the former. The combination of these two features acts as a filter that reduces complexity and improves model conciseness [66]. Based on these approaches, we use the mapping functions described in Table 7 to integrate e3value's concepts with ArchiMate's business layer.

4.3 Model analysis

Analysis generates information from a single model or from a federated set of models. It may focus on (1) the specification of a schema of an enterprise modelling language, (2) an instantiation of a schema of a specific modelling scenario and (3) a case of a modelling scenario. Let us consider the ArchiMate enterprise modelling language to provide an example. The first case of analysis would target ArchiMate's meta-model, i.e. the representation as an ontological schema of its meta-model. This enables to explore its concepts, direct relationships and derived relationships. Examples of analysis include determining which concepts can be related through ArchiMate's assignment relationship, or explaining how an actor is involved in the realization of a business service. The second case would target an instance

of an ArchiMate model. Consider the model of a business process P containing activities P_1, P_1, P_3 and actors A_1, A_2 , along with control flow relationships between activities, and assignment relationships between activities and actors. The analysis of this scenario entails looking at the instances of ArchiMate's meta-model. For instance, one can determine which actor instance is assigned to activity P_1 , or which activities are sequenced after activity P_2 . Another example is checking the conformance of the instantiated schema with regard to ArchiMate's meta-model, i.e. checking whether the instantiated entities and relationships conform with the meta-model. If an activity is assigned to another activity, this would violate the meta-model and would be detected as an inconsistency. Finally, the third case of analysis targets the cases or instances of process P . An example of analysis is determining the actual actors assigned to P_1 .

Graph analysis is performed with the SPARQL language [61]. A simple example is using a SPARQL query to traverse the graph and find all business services in an ArchiMate model. SPARQL queries may also make use of nesting and operators such as union, filters, value aggregation and path expressions. SPARQL queries can also be used to analyse multiple schemas using inter-schema analysis or chained queries. An example of a chained query is finding the relationships between BMC's customer segments and ArchiMate's business services. As depicted in Fig. 9, a business service has a derived relationship to BMC's customer segment via e3value's value transmission. This chained query translates to the following in SPARQL.

Listing 1 SPARQL query relating BMC's Customer Segment to ArchiMate's Business Service.

```
SELECT ?cs ?bs WHERE {
  ?cs rdfs:type bmc:CustomerSegment.
  ?vt rdfs:type e3v:ValueTransmission.
  ?bs rdfs:type archi:BusinessService.
  ?vt bmc-e3v:transmitsValueTo ?cs.
  ?vt e3v-archi:correspondsTo ?bs.
}
```

The mapping functions presented in Tables 5, 6 and 7 were translated to further SPARQL queries.

5 Case study

This section evaluates the artefact using competency questions. Competency questions are a means to assess the completeness and consistency of an ontological schema [67–69]. In this paper, the questions are designed to check whether a federated enterprise modelling scenario is consistent with regard to the set of mapping functions described in the previous section.

The scenario describes the business model of ePharmacare, an electronic health initiative. ePharmacare provides health services to chronic medicated patients through an electronic service hub. The services include registering treatment plans, supporting the prescription, ordering and delivery of drugs and other pharmaceutical products, following up the treatments prescribed by doctors to patients, and enabling chronic patients to receive notifications as well as to log drug intake and the effects of medication. ePharmacare is therefore a virtual network of organizations and includes doctors and chronic patients as well as pharmacies, pharmaceutical companies, the national health insurance system, as well as healthcare providers. From a business perspective, the services of ePharmacare are considered to be value-adding to its stakeholders since they are designed to provide economic savings while increasing the traceability of treatments and the ability to perform cause-effect analysis. The enterprise

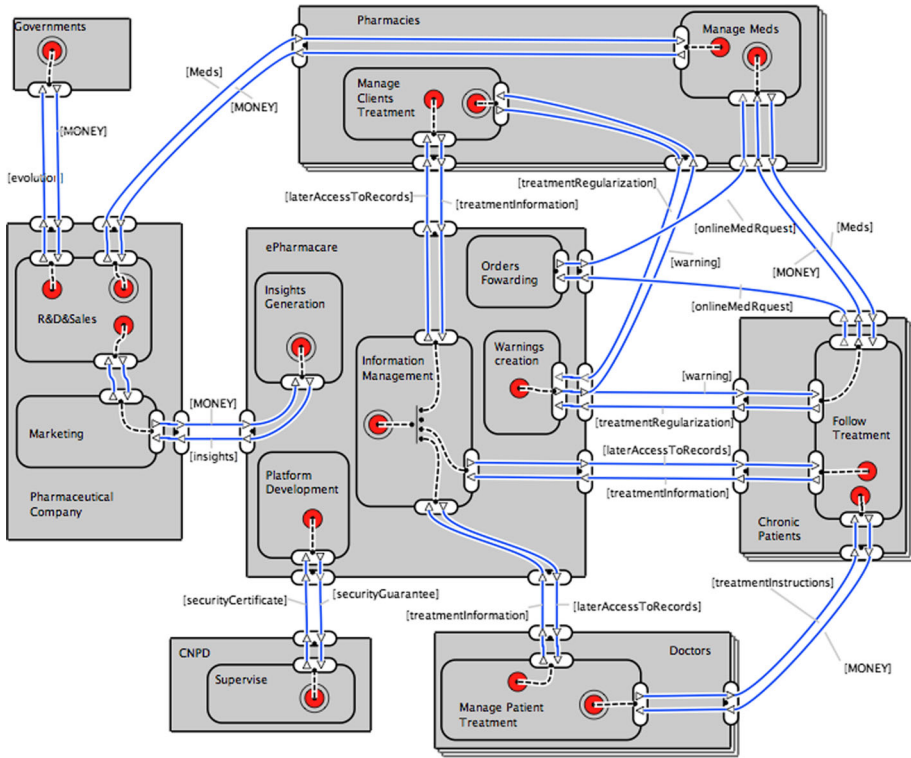


Fig. 11 The e3value model of ePharmcare

models of ePharmcare used in this case study result from the analysis of existing project documentation and business process models, and from workshops and semi-structured interviews with the project’s stakeholders [70–72]. The subset of models included in this paper is the following:

- the e3value value network model of ePharmcare that links together the individual business partners (Fig. 11),
- the BMC of ePharmcare (Fig. 16),
- the BMC of a subset of ePharmcare’s partners (Fig. 17),
- the ArchiMate models that describe the business services and processes of ePharmcare (Fig. 14), as well as those of a subset of its business partners (Fig. 15).

These models serve different purposes. The value network model describes the value flow between the partners. Each partner has its own business model canvas describing its individual business model, along with ArchiMate models describing the underlying business services and processes.

5.1 Schema evaluation

The integrated schemas are evaluated by testing the ability to assess the 18 statements described in Table 8. These competency questions are designed so that each of the transformation functions is tested by at least one competency question, thus covering all of the mapping functions.

Table 8 List of competency questions

1	Customer Segments or Key Partners (BMC) that are not represented in e3value as an Actor
2	Actors (e3value) not represented in a BMC model
3	Actors (e3value) not represented in a ArchiMate model
4	Business Actors (ArchiMate) not represented in a e3value model
5	Revenue Streams not represented in a e3value model
6	Key Activities not represented in e3value or ArchiMate models
7	Value Activities not represented in BMC or ArchiMate models
8	Business Processes not represented in BMC or e3value models
9	Value Transmission without a relationship to Business Service
10	Business Service without a relationship to Value Transmission
11	Value Transmission not related to the delivery of Value Proposition or not related to a Key Activity
12	(Key Activity, Key Partner) pair that has no mapping to Value Transmission
13	(Value Proposition, Customer Segment, Distribution Channel) triple that has no mapping to Value Transmission
14	Value Objects without a corresponding Key Resource
15	Value Objects not represented as a Value or a Business Object
16	Value or Business Objects not represented as a Value Objects
17	Channels without related concepts on a ArchiMate model
18	Business Interface without related concepts on a BMC model

Each question was translated to a SPARQL query. The following query implements question 1 by subtracting the Customer Segments that relate to an e3value Actor from the set of all BMC Customer Segments. The same subtraction is applied to Key Partners. The result is a set of BMC Customer Segment and Key Partner individuals that do not correspond to any Actor individual in the e3value model and thus do not comply with this constraint.

Listing 2 SPARQL query for question 1.

```

SELECT ?cs ?kp WHERE {
  ?cs rdf:type bmc:CustomerSegment.
  MINUS {
    ?cs rdf:type bmc:CustomerSegment.
    ?actor rdf:type e3v:e3vActor.
    ?cs bmc-e3v:correspondsTo ?actor.
  }
  UNION {
    ?kp rdf:type bmc:Actor.
    MINUS {
      ?kp rdf:type bmc:KeyPartner.
      ?actor rdf:type e3value:e3vActor.
      ?kp bmc-e3v:correspondsTo ?actor.
    }
  }
}

```

value	businessObject	valueObject
integrated:AR3_ReceiveMoney		integrated:e3_MONEY
integrated:AR2_ReceiveMoney		integrated:e3_MONEY
integrated:AR_Receive_money		integrated:e3_MONEY
	integrated:AR_Patient_Information	integrated:e3_treatmentInformation
	integrated:AR_Patient_Information	integrated:e3_laterAccessToRecords
	integrated:AR4_Warning	integrated:e3_warning
	integrated:AR3_Warning	integrated:e3_warning
	integrated:AR3_PatientInformation	integrated:e3_treatmentInformation
	integrated:AR2_Meds	integrated:e3_meds
	integrated:AR_Aggregated_insights	integrated:e3_insights
	integrated:AR3_Meds	integrated:e3_meds
	integrated:AR4_Meds	integrated:e3_MONEY
	integrated:AR4_MedsDeliveryConfirmation	integrated:e3_treatmentRegularization
	integrated:AR2_AggregatedInsights	integrated:e3_insights
	integrated:AR_Security_Certificate	integrated:e3_securityCertificate
	integrated:AR5_TreatmentInformation	integrated:e3_treatmentInformation
	integrated:AR3_OnlineMedRequest	integrated:e3_onlineMedRequest
	integrated:AR1_SecurityCertificate	integrated:e3_securityCertificate
	integrated:AR_Warning	integrated:e3_warning
	integrated:AR4_OnlineMedRequest	integrated:e3_onlineMedRequest
	integrated:AR4_TreatmentInformation	integrated:e3_treatmentInstructions
	integrated:AR4_TreatmentInformation	integrated:e3_treatmentInformation
	integrated:AR3_TreatmentRegularization	integrated:e3_treatmentRegularization
	integrated:AR_Med_Request	integrated:e3_onlineMedRequest

Fig. 12 Results of the SPARQL query showing the relationships between value objects and business objects

The remaining questions were implemented in a similar fashion. Figures 18 and 19 in the Appendix illustrate the answers to the first nine questions. Each answer that is a nonempty set of individuals is an inconsistency at instance level because an answer contains the individuals that do not comply with the given constraint. Thus, if all questions produce empty answers, then the scenario is deemed consistent with regard to that set of constraints.

SPARQL queries can be used to analyse individual schemas as well as the integrated schemas. The following query iterates through all Business Object and discovers all Value Objects associated with it. The results of this query are depicted in Fig. 12.

Listing 3 SPARQL query that discovers the value objects associated with business objects.

```

SELECT ?value ?businessObject ?valueObject WHERE {
  ?value rdfs:type archi:Value.
  ?valueObject rdfs:type e3v:ValueObject.
  ?valueObject e3v-archi:corresponds ?value.
  UNION {
    ?businessObject rdfs:type archi:BusinessObject.
    ?valueObject rdfs:type e3v:ValueObject.
    ?valueObject e3v-archi:corresponds ?businessObject.
  }
}
    
```

The models can also be analysed using a logical reasoner instead of SPARQL. In this case, the queries are specified with OWL-DL and a reasoner is used to infer facts from a model. The six statements shown in listing 4 were processed with the Hermit reasoner using the Protégé tool. Figure 13 shows the results of reasoning.

Query results 1	Query results 2	Query results 3
Sub classes (0)	Sub classes (0)	Sub classes (0)
Instances (13)	Instances (18)	Instances (28)
<ul style="list-style-type: none"> ◆ e3_Marketing ◆ e3_Platform_Development ◆ e3_Warnings_Creation ◆ e3_Supervise ◆ e3_Provision_Meds ◆ e3_treatmentRegularization ◆ e3_Insights_Creation ◆ e3_R&D&Sales ◆ e3_Manage_Patient_Treatment ◆ e3_Orders_Fowarding ◆ e3_Information_Management ◆ e3_Follow_Treatment ◆ e3_Manage_Clients_Treatment 	<ul style="list-style-type: none"> ◆ e3_vp40 ◆ e3_vp373 ◆ e3_vp374 ◆ e3_vp51 ◆ e3_vp64 ◆ e3_vp192 ◆ e3_vp193 ◆ e3_vp65 ◆ e3_vp385 ◆ e3_vp50 ◆ e3_vp8 ◆ e3_vp396 ◆ e3_vp384 	<ul style="list-style-type: none"> ◆ e3_vi181 ◆ e3_vi238 ◆ e3_vi119 ◆ e3_vi456 ◆ e3_vi187 ◆ e3_vi373 ◆ e3_vi79 ◆ e3_vi111 ◆ e3_vi411 ◆ e3_vi434 ◆ e3_vi135 ◆ e3_vi310 ◆ e3_vi12
Query results 4	Query results 5	Query results 6
Sub classes (0)	Sub classes (0)	Sub classes (0)
Instances (1)	Instances (1)	Instances (1)
<ul style="list-style-type: none"> ◆ EP_Customer_Loyalty 	<ul style="list-style-type: none"> ◆ e3_Information_Management 	<ul style="list-style-type: none"> ◆ e3_ve11

Fig. 13 Results of OWL-DL reasoning, queries 1–6

Listing 4 Example of model analysis using OWL-DL reasoning.

```

# 1. All Value Activities
ValueActivity
# 2. Inbound Value Ports for the Actor "ePharmacare"
InboundValuePort and (inverse has some (inverse has value ePharmacare))
# 3. Value Interfaces of the Value Activities?
ValueInterface and (inverse has some ValueActivity)
# 4. Value Propositions that have Value "ReceiveMoney"
ValueProposition and has some Value and has value ReceiveMoney
# 5. Value Activities that have Actor ePharmacare
ValueActivity and isPerformedBy some e3Actor and has value ePharmacare
# 6. Value Transmission that transmit value into the inbound Value Port vp12
ValueTransmission and transmitsValueTo some InboundValuePort and has value vp12s

```

Note that the proposed artefact is not able to resolve the detected inconsistencies. Indeed, some inconsistencies may be simple syntactical mismatches that can be corrected semi-automatically. But contradictory interpretations of the same business concept at semantic level need to be addressed by the stakeholders using techniques such as concept elicitation or workshops. Therefore, the approach described in this paper is not meant to automate the process of resolving the inconsistencies but to automatically detect the inconsistencies with regard to a specific set of constraints.

This section described the ePharmacare scenario to demonstrate the ability of using semantic techniques to (1) represent the meta-models of three enterprise modelling languages, BMC, ArchiMate and e3value, as well as instances of models specified in these languages, (2) federate the models using transformation functions and (3) analyse the integrated models. The results of the integrated model analysis show, on the one hand, the ability to use SPARQL and OWL-DL to analyse individual schemas as well as integrated schemas, and, on the other, the ability to assess the consistency of a model regarding a set of constraints.

6 Conclusions

Enterprise models cross-cut different domains that can be observed from multiple viewpoints. These models enable stakeholders to communicate, understand, analyse, design and transform an enterprise system. This paper explored the application of semantic techniques to integrate and jointly analyse different enterprise modelling languages.

The design artefact presented in this paper is a federated set of schemas that represent domain-specific languages (conceptual schemas), and the instances of those schemas (information bases). Each domain is represented as a domain-specific schema (DSO) that specifies entity and relationship types. A federated model landscape is generated by integrating the schemas using transformation maps that relate the relevant types across the different domains. The specification of a transformation map is situational, i.e. the mapping functions depend on the context and purpose of the integration.

The schemas are formalized with RDFS and the OWL-DL languages, which makes possible analysing the models with graph analysis or with reasoners that perform logical inference. The analysis may target a single domain or a combination of integrated domains. Different types of analysis are possible, including assessing the conformance of a model against a set of constraints.

The utility of the artefact is demonstrated through the integration of three languages that describe an organization from different perspectives, namely the business model canvas, e3value and ArchiMate. This involved specifying the conceptual schemas behind each of these modelling languages, along with the transformation maps between the schemas. A case study was used to define the information base and to assess the ability of semantic techniques to integrate and analyse the different enterprise modelling languages.

The main contribution of this paper is accounting how semantic techniques, in particular ontology-based techniques, can be used to computationally represent, integrate and analyse enterprise models. As such, a fundamental challenge resides in defining mapping functions between the schemas that fit the context and purpose of the integration. Defining such functions may prove to be a complex or even unfeasible task depending on the semantic gap between the languages being mapped. Nevertheless, the application of semantic techniques to enterprise modelling brings value to the enterprise engineering community of practice as it facilitates the integration and analysis of diverse modelling domains.

Acknowledgements This work was partially funded by FCT, Fundação para a Ciência e a Tecnologia, through projects UID/CEC 50021/2013 and DataStorm EXCL/EEI-ESS 0257/2012, and by the European Commission under the Seventh Framework Programme for research and technological development and demonstration activities under Grant Agreement No. 269940 (TIMBUS project, <http://timbusproject.net/>).

Appendix

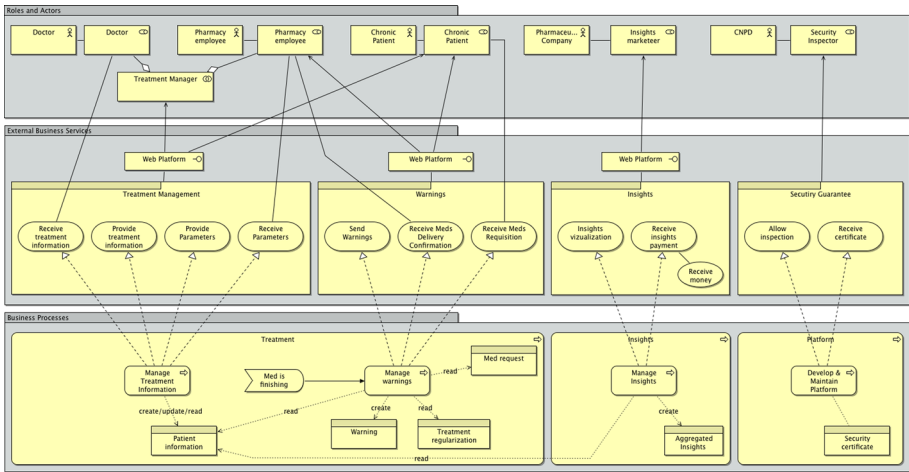


Fig. 14 The ArchiMate business layer model of ePharmacare

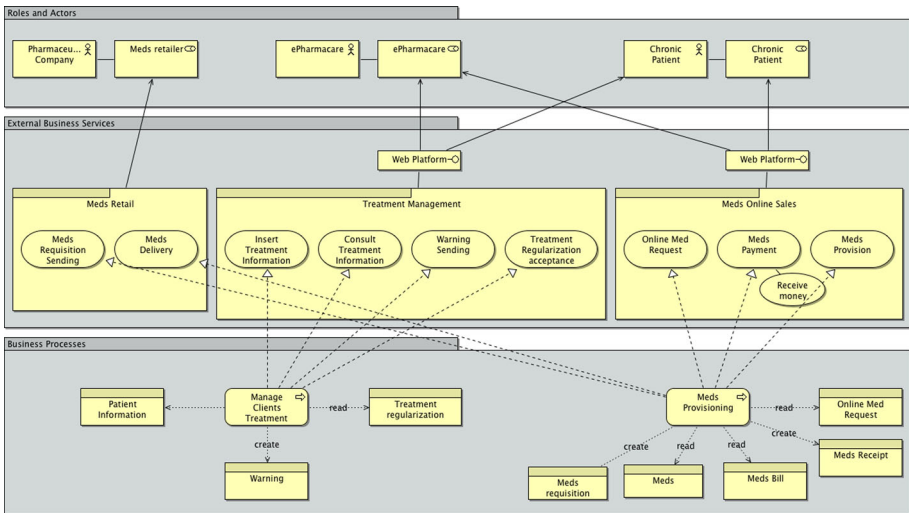


Fig. 15 The ArchiMate business layer model of a pharmacy within the ePharmacare network

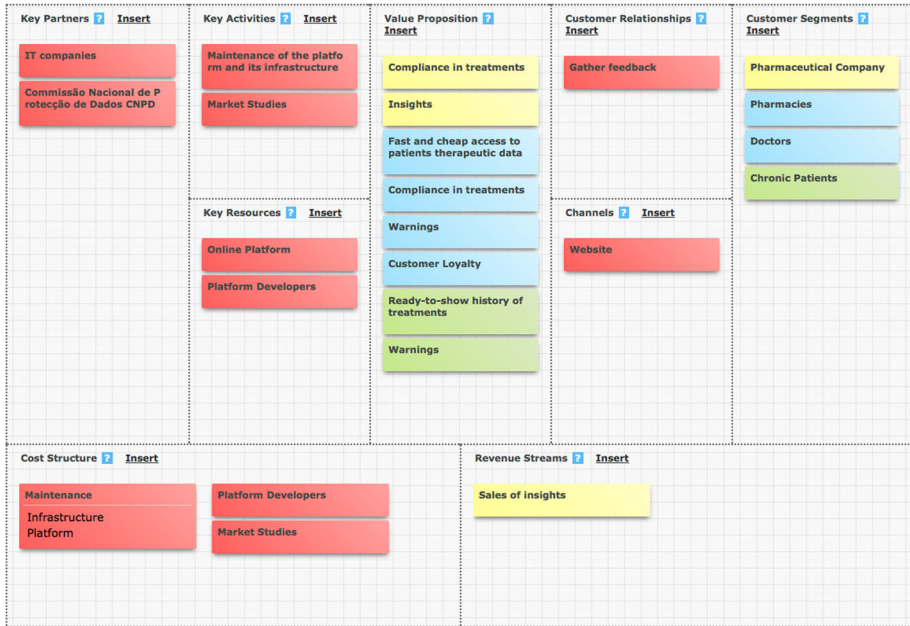


Fig. 16 The business model canvas of ePharmacare

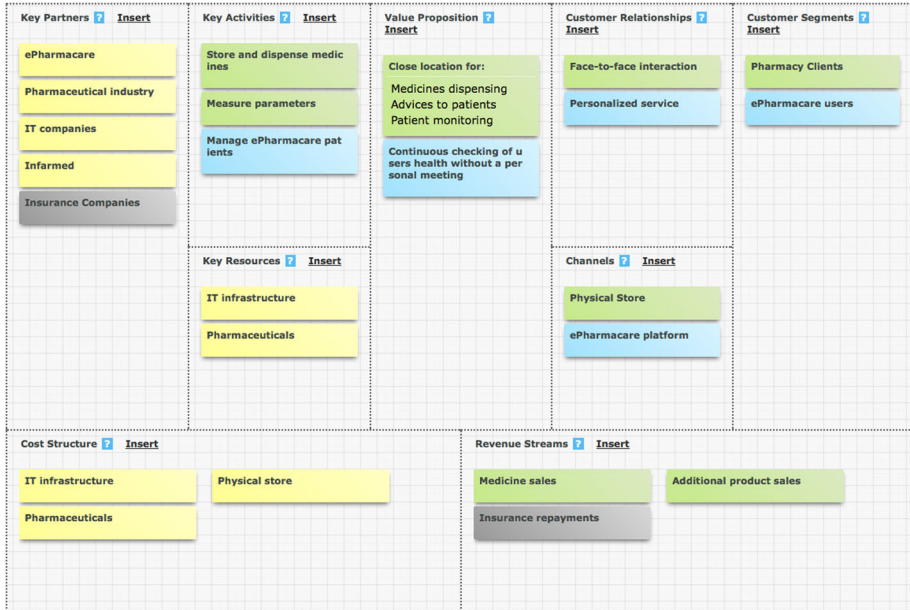


Fig. 17 The business model canvas of a pharmacy within the ePharmacare network

[QUESTION 1]

cs	kp
integrated:CP_IT_companies	
integrated:CP_Insurance_companies	
integrated:CP_Infarmed	
integrated:EP_IT_companies	

[QUESTION 2]

e3actor

[QUESTION 3]

e3actor
integrated:e3_Governments

[QUESTION 4]

businessActor

[QUESTION 5]

revenueStream
integrated:CP_Insurance_payments
integrated:CP_Additional_product_sales

[QUESTION 6]

kaNoe3value	kaNoarchimate
integrated:CP_Inform_the_community	
integrated:EP_Market_studies	
	integrated:CP_Manage_ePharmacare_patients
	integrated:PC_Medicines_production
	integrated:EP_Maintenance_of_the_platform
	integrated:CP_Store_and_dispense_medicines
	integrated:CP_Measure_parameters
	integrated:PC_Marketing

Fig. 18 Answers to competency questions 1–6

[QUESTION 7]	
<u>valueActivityN0bmc</u>	<u>valueActivityN0arch</u>
integrated:e3_Insights_Creation	
integrated:e3_Follow_Treatment	
integrated:e3_Supervise	
integrated:e3_Manage_Patient_Treatment	
integrated:e3_Information_Management	
integrated:e3_Warnings_Creation	
integrated:e3_Orders_Fowarding	integrated:e3_Orders_Fowarding

[QUESTION 8]	
<u>businessProcessN0bmc</u>	<u>businessProcessN0e3value</u>
integrated:AR_Treatment	
integrated:AR_Insights_Process	
integrated:AR_Manage_Treatment_Information	
integrated:AR5_ManagePatientTreatment	
integrated:AR1_Supervise	
integrated:AR4_FollowTreatment	
integrated:AR_Manage_Insights	
integrated:AR1_Platform	
integrated:AR_Platform	
integrated:AR_Manage_Warnings	
	integrated:AR_Treatment
	integrated:AR_Insights_Process
	integrated:AR1_Platform
	integrated:AR_Platform

[QUESTION 9]	
<u>valueTransmission</u>	
integrated:e3_ve460	
integrated:e3_ve146	
integrated:e3_ve235	
integrated:e3_ve236	
integrated:e3_ve319	
integrated:e3_ve164	
integrated:e3_ve354	
integrated:e3_ve134	
...	
...	
...	

Fig. 19 Answers to competency questions 7–9

References

- Lankhorst M (2004) Enterprise architecture modelling—the issue of integration. *Adv Eng Inform* 18(4):205–216
- Lankhorst M (2013) Enterprise architecture at work: modeling, communication, and analysis, 3rd edn. Springer, New York
- Aleatrati Khosroshahi P, Aier S, Hauder M, Roth S, Matthes F, Winter R (2015) Success factors for federated enterprise architecture model management. In: Persson A, Stirna J (eds), CAiSE 2015 Workshops, LNBP 215, Springer, Berlin, pp 413–425
- Zdravkovic J, Stirna J, Kirikova M, Karagiannis D, Winter R (2015) Advanced enterprise modeling. *Bus Inform Syst Eng* 57(1):1–2. doi:10.1007/s12599-014-0367-8 ISSN 2363-7005

5. Bucher T, Fisher R, Kurpjuweit S, Winter R (2006) Enterprise architecture analysis and application. An exploratory study. In: Proceedings of the 1st workshop on trends in enterprise architecture research (TEAR)
6. Johnson P, Ekstedt M (2007) Enterprise architecture: models and analyses for information systems decision making. Lightning source incorporated
7. Antunes G, Barateiro J, Caetano A, Borbinha J (2015) Analysis of federated enterprise architecture models. In: 23rd European conference on information systems (ECIS)
8. Narman P, Johnson P, Nordstrom L (2007) Enterprise architecture: a framework supporting system quality analysis. In: 11th IEEE international enterprise distributed object computing conference (EDOC)
9. Osterwalder A, Pigneur Y (2010) Business model generation: a handbook for visionaries, game changers, and challengers. Wiley, Hoboken
10. Gordijn J (2004) Value creation from E-business models, chapter e-business value modelling using the e3-value ontology. Elsevier, Butterworth-Heinemann
11. The Open Group (2013) ArchiMate 2.1 specification. The open group
12. Gordijn J, Akkermans JM (2003) Value-based requirements engineering: exploring innovative e-commerce ideas. *Requir Eng* 8(2):114–134
13. Gordijn J, Osterwalder A, Pigneur Y (2005) Comparing two business model ontologies for designing e-business models and value constellations. In: Proceedings of the 18th BLED eConference, pp 6–8. <http://aisel.aisnet.org/bled2005/15>
14. de Kinderen S, Gaaloul K, Proper HE (2012a) Integrating value modelling into archimate. In: Exploring services science, Lecture Notes in business information processing, Springer, Berlin, pp 125–139
15. Pombinho J (2015) Value-oriented enterprise transformation—design and engineering of value networks. Ph.D. thesis, Universidade de Lisboa—Instituto Superior Técnico
16. Sousa P, Caetano A, Vasconcelos A, Pereira C, Tribolet J (2006) Enterprise architecture modeling with the uml 2.0. In: Rittgen P (ed), Enterprise modelling and computing with UML, Idea Group Inc, pp 67–94
17. Caetano A, Silva A, Tribolet J (2009) A role-based enterprise architecture framework. In: Proceedings of the ACM symposium on applied computing, pp 253–258. ACM
18. Frank U (2014) Multi-perspective enterprise modeling: foundational concepts, prospects and future research challenges. *Softw Syst Model* 13(3):941–962
19. Bork D (2015) Using conceptual modeling for designing multi-view modeling tools. In: Americas Conference on information systems (AMCIS) 2015, 21st Americas conference on information systems (AMCIS). <http://eprints.cs.univie.ac.at/4315/>
20. Zivkovic S, Kuhn H, Karagiannis D (2007) Facilitate modelling using method integration: an approach using mappings and integration rules. In: European conference on information systems (ECIS), pp 2038–2049
21. Iacob ME, Meertens LO, Jonkers H, Quartel DAC, Nieuwenhuis LJM, Van Sinderen MJ (2013) From enterprise architecture to business models and back. *Softw Syst Model* 13(3):1059–1083
22. Fischer R, Aier S, Winter R (2007) A federated approach to enterprise architecture model maintenance. *Enterp Model Inform Syst Archit* 2:14–22
23. Bjeković M, Proper HA, Sottet J-S (2012) Towards a coherent enterprise modelling landscape. In: Emerging topics in the practice of enterprise modeling : 5th IFIP WG8.1 working conference on the practice of enterprise modeling, pp 1–12
24. Bjeković M, Proper HA (2013) Challenges of modelling landscapes: pragmatics swept under the carpet? In: Proceedings of third international symposium business modeling and software design, Lecture Notes in business information processing (Book 173), Springer, pp 11–22
25. Proper HA, Lankhorst MM (2014) Enterprise architecture—towards essential sensemaking. *Enterp Model Inform Syst Archit* J 01(01):5–21
26. Uschold M, Gruninger M (1996) Ontologies: principles, methods and applications. *Knowl Eng Rev* 11(02):93–136
27. Guizzardi G, Wagner G (2010) Using the unified foundational ontology (ufo) as a foundation for general conceptual modeling languages. In: Theory and applications of ontology: Computer applications, Springer, pp 175–196
28. Prackwieser C, Buchmann RA, Grossmann W, Karagiannis D (2014) Overcoming heterogeneity in business process modeling with rule-based semantic mappings. *Int J Softw Eng Knowl Eng* 24(8):1131–1159. <http://eprints.cs.univie.ac.at/4243/>
29. Gordijn J, Akkermans H (2001) Designing and evaluating e-business models. *IEEE Intell Syst* 16(4):11–17
30. Offermann P, Blom S, Schönherr M, Bub U (2010) Artifact types in information systems design science—a literature review. In: Winter R, Zhao JL, Aier S (eds), Global perspectives on design science research,

- volume 6105 of Lecture Notes in computer science, pages, Springer, Berlin, pp 77–92. ISBN 978-3-642-13334-3. doi:[10.1007/978-3-642-13335-0_6](https://doi.org/10.1007/978-3-642-13335-0_6)
31. von Alan RH, March ST, Park J, Ram S (2004) Design science in information systems research. *MIS Q* 28(1):75–105
 32. de Simoni G (2015) Innovation insight: leveraging ontology to exploit business value hidden in information. *Gartner innovation*. <https://www.gartner.com/doc/2990119/innovation-insight-leveraging-ontology-exploit>
 33. Chesbrough H (2007) Business model innovation: it's not just about technology anymore. *Strategy Leadersh* 35(6):12–17
 34. Osterwalder A (2004) The business model ontology: a proposition in a design science approach. Ph.D. thesis, University of Lausanne, Ecole des Hautes Etudes Commerciales HEC, Lausanne, Switzerland
 35. Pombinho J, Aveiro D, Tribolet J (2014) Linking value chains—combining e3value and demo for specifying value networks. In: Fourth enterprise engineering working conference (EEWC), Funchal, Madeira
 36. Iacob ME, Quartel D, Jonkers H (2012) Capturing business strategy and value in enterprise architecture to support portfolio valuation. In: 16th IEEE international enterprise distributed object computing conference (EDOC)
 37. Meertens LO, Iacob ME, Jonkers H, Quartel D, Nieuwenhuis LJM, van Sinderen MJ (2012) Mapping the business model canvas to archimate. In: Proceedings of the 27th annual ACM symposium on applied computing, pp 1694–1701. ACM. <http://doc.utwente.nl/82858/>
 38. Gruber TR (1993) A translation approach to portable ontology specifications. *Knowl Acquis* 5(2):199–220
 39. Guarino N, Oberle D, Staab S (2009) Handbook on Ontologies, chapter what is an Ontology?, Springer, Berlin, pp 1–17
 40. Studer R, Benjamins R, Fensel D (1998) Knowledge engineering: principles and methods. *Data Knowl Eng* 25:161–198
 41. Pinto HS, Gómez-Pérez A, Martins JP (1999) Some issues on ontology integration. In: Proceedings of IJCAI99's workshop on ontologies and problem solving methods: lessons learned and future trends, vol 18, pp 1–12
 42. Davies J, Studer R, Warren P (2006) Semantic web technologies: trends and research in ontology-based systems. Wiley, Hoboken
 43. Lenzerini D, Milano M, Poggi A (2004) Ontology representation and reasoning. Technical report, University di Roma La Sapienza
 44. Vaculin R (2009) Process mediation framework for semantic web services. Ph.D. thesis, Department of Theoretical Computer Science and Mathematical Logic, Faculty of Mathematics and Physics, Charles University
 45. Olivé A (2007) Conceptual modeling of information systems. Springer, Berlin
 46. Breitman K, Casanova MA, Truszkowski W (2007) Semantic web: concepts, technologies and applications. Springer, Berlin
 47. ISO/IEC/IEEE (2011) ISO/IEC/IEEE 42010:2011 Systems and software engineering—architecture description. ISO/IEC/IEEE
 48. Saat J, Franke U, Lagerstrom R, Ekstedt M (2010) Enterprise architecture meta models for it/business alignment situations. In: 14th IEEE international enterprise distributed object computing conference (EDOC)
 49. Buckl S, Buschle M, Johnson P, Matthes F, Schweda CM (2011) A meta-language for enterprise architecture analysis. In: 16th international conference on exploring modeling methods for systems analysis and design (EMMSAD)
 50. Niemann KD (2005) From enterprise architecture to IT governance. Vieweg /& Sohn Verlag, Friedr
 51. Kurpjuweit S, Winter R (2007) Viewpoint-based meta model engineering. In: 2nd international workshop on enterprise modelling and information systems architectures (EMISA)
 52. Buckl S, Schweda CM, Matthes F (2010) A design theory nexus for situational enterprise architecture management. In: 14th IEEE international enterprise distributed object computing conference workshops (EDOCW)
 53. Caetano A, Pereira C, Sousa P (2011) Generating multiple consistent views from business process models. In Lecture Notes in business information processing, research and practical issues of enterprise information systems. Springer, Berlin
 54. Antunes G, Bakhshandeh M, Mayer R, Borbinha J, Caetano A (2014) Using ontologies for enterprise architecture integration and analysis. *Complex Syst Inform Model Q* 1:1–23
 55. Mayer R, Antunes G, Caetano A, Bakhshandeh M, Rauber A, Borbinha J (2015) Using ontologies to capture the semantics of a business process for digital preservation. *Int J Digit Libr* 15(2–4):129–152. doi:[10.1007/s00799-015-0141-7](https://doi.org/10.1007/s00799-015-0141-7)

56. Granitzer M, Sabol V, Onn KW, Lukose D, Tochtermann K (2010) Ontology alignment—a survey with focus on visually supported semi-automatic techniques. *Future Internet* 2:238–258
57. Miles A, Bechhofer S (2009) Skos: simple knowledge organization system reference, w3c recommendation. Technical report, W3C
58. Herre B, Heller H, Burek P, Hoehndorf R, Loebe F, Michalek H (2007) General formal ontology (gfo): a foundational ontology integrating objects and processes. part i: Basic principles. Technical report, Research Group Ontologies in Medicine (Onto-Med), University of Leipzig
59. Wand Y, Weber R (1993) On the ontological expressiveness of information systems analysis and design grammars. *Inform Syst J* 3(4):217–237
60. Fettke P, Loos P (2003) Ontological evaluation of reference models using the bunge-wand-weber model. In: *Proceedings of the AMCIS 2003*
61. Harris S, Seaborne A (2013) SPARQL 1.1 Query language. W3C. <http://www.w3.org/TR/sparql11-query/>
62. Fritscher B, Pigneur Y (2011) Business it alignment from business model to enterprise architecture. In: *Advanced information systems engineering workshops*, Springer, Berlin, pp 4–15
63. de Kinderen S, Gaaloul K, Proper HA (2012b) Bridging value modelling to archimate via transaction modelling. *Softw Syst Model* 13(3):1043–1057
64. Pombinho J, Tribolet J (2012) Modelling the value of a system's production—matching demo and e3value. In: *6th international workshop on value modelling and business ontology*, Vienna, Austria
65. Dietz JLG (2006) *Enterprise ontology: theory and methodology*. Springer, Berlin
66. Dietz JLG, Hoogervorst JAP, Albani A, Aveiro D, Babkin E, Barjis J, Caetano A, Huysmans P, Iijima J, van Kervel S et al (2013) The discipline of enterprise engineering. *Int J Organ Des Eng* 3(1):86–114
67. Fox MS, Gruninger M (1998) Enterprise modeling. *AI Mag* 19(3):109
68. Grüninger M, Fox MS (1995) Methodology for the design and evaluation of ontologies. In: *Proceedings of the IJCAI-95 workshop on basic ontological issues in knowledge sharing*
69. Bezerra C, Freitas F, Santana F (2013) Evaluating ontologies with competency questions. In: *Web intelligence (WI) and intelligent agent technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, vol 3, pp 284–285. doi:10.1109/WI-IAT.2013.199
70. Lapão LV, Gregorio J, Ferreira T, Cavaco A, Lovis C, Russo G, da Silva MM (2013) ehealth services for enhanced pharmaceutical care provision: From counseling to patient education. In: *IEEE 2nd international conference on serious games and applications for health (SeGAH)*, pp 1–7
71. Gregorio J, Ferreira TL, Cavaco A, da Silva MM, Lovis C, Lapão LV (2013a) Community pharmacies and ehealth services: barriers and opportunities for real primary healthcare integration. In: *IEEE 26th international symposium on computer-based medical systems (CBMS)*, pp 393–396
72. Gregorio J, Cavaco A, da Silva MM, Lovis C, Lapão LV (2013b) Challenges of information technologies adoption to enable community pharmacies ehealth services. *Int J Clin Pharm* 35(3):490–498



Artur Caetano is tenured assistant professor of information systems at the Department of Computer Science and Engineering, Instituto Superior Técnico (IST), University of Lisbon, and researcher at the Information and Decision Support Systems (IDSS) group, INESC-ID. His areas of research intersect enterprise modelling, business process management, conceptual modelling and semantic web technologies, and he is involved in the development of computational techniques to aid the representation, analysis and (re)design of enterprise models. Artur regularly contributes to European framework programme research and development projects related to these topics and works with the industry in knowledge transfer and enterprise transformation projects as a consultant, and also as an ArchiMate and BPMN trainer. He is member of the ACM and IEEE.



Gonçalo Antunes is a postdoctoral researcher at IDSS, INESC-ID. He holds a Ph.D. in computer science and engineering from IST, University of Lisbon. His areas of interest include enterprise model analysis and enterprise architecture. He has been involved in several national and European framework research projects, and published in peer-reviewed journals and in international conferences in topics related to enterprise model analysis, information life-cycle management and semantic web technologies.



João Pombinho holds a Ph.D. in computer science and engineering from IST, University of Lisbon, concerning enterprise engineering and value-oriented systems design. His academic interests also include value modelling, enterprise architecture and enterprise transformation. João is currently invited assistant professor of computer science and engineering at IST, and researcher at INESC-ID. João began his professional career in 2005 in the Portugal Telecom group as IT Planning and Change Manager. He then became responsible for Business Accounts and Requirements Management at ZON Multimedia. He is currently IT Demand Manager at NOS Comunicações. He is a regular speaker at industry events about IT demand management, enterprise architecture and value-oriented systems.



Marzieh Bakhshandeh is Ph.D. student at IST, University of Lisbon, and a researcher at the IDSS group, INESC-ID. Her research interests are semantic web technologies, enterprise architecture model analysis, in particular process model similarity matching.



José Granjo is software engineer at Screen Interaction AB, Sweden, where he leads the front-end product development and user experience design for monitoring global telecom networks. He has a M.Sc. in Computer Engineering and Information Systems from IST, University of Lisbon and was a researcher at the Information and Decision Support Systems group at INESC-ID, where he applied ontologies to analyse and represent enterprise models.



José Borbinha is associate professor of computer science and engineering at IST, University of Lisbon, and coordinator of the IDSS group at INESC-ID. His main research interests are systems engineering, conceptual modelling, requirements engineering, information management (especially on records management and digital preservation), enterprise architecture and information systems. He was CIO of the Portuguese National Library (1998–2005). He is member of the ACM and IEEE. He is a founding member and elected chair (2008–2010) of the IEEE Technical Committee on Digital Libraries, and member of the Advisory Board of the Dublin Core Metadata Initiative. He is chair of the Steering Committee of the conference Theory and Practice of Digital Libraries (TPDL) and Portugal's representative at ISO's TC46/SC11 WG14 "Records requirements in enterprise Architecture".



Miguel Mira da Silva is associate professor of information systems at the University of Lisbon, and coordinator of the Digital Transformation group at the INESC-INOV research institute. Miguel graduated in 1989 and received his master's degree in 1993 in electrical and computer engineering from the University of Lisbon, received his Ph.D. in 1997 in computing science from the University of Glasgow, and a "Sloan Fellowship" master in management in 2005 from the London Business School. Miguel has supervised 5 Ph.D. and over 100 M.Sc. projects, and published over 150 research papers in international conferences and journals, as well as four teaching books.