ArchiMEO: A Standardized Enterprise Ontology based on the ArchiMate Conceptual Model

Knut Hinkelmann[a], Emanuele Laurenzi[b], Andreas Martin[c], Devid Montecchiari[d], Maja Spahic[e] and Barbara Thönssen[f]

School of Business, FHNW University of Applied Sciences and Arts Northwestern Switzerland, 4600 Olten, Switzerland

Keywords: Enterprise Ontology, Enterprise Architecture, ArchiMate, Enterprise Modeling.

Abstract: Many enterprises face the increasing challenge of sharing and exchanging data from multiple heterogeneous sources. Enterprise Ontologies can be used to effectively address such challenge. In this paper, we present an Enterprise Ontology called ArchiMEO, which is based on an ontological representation of the ArchiMate standard for modeling Enterprise Architectures. ArchiMEO has been extended to cover various application domains such as supply risk management, experience management, workplace learning and business process as a service. Such extensions have successfully proven that our Enterprise Ontology is beneficial for enterprise applications integration purposes.

1 INTRODUCTION

The advent of digitalization has led to the generation of an increasing amount of data in enterprises. Data that originates from multiple heterogeneous sources might share the same meaning but have a different structure. Data elements might have different names and can be processed by different applications. For example, different software applications for Supply Chain Management (SCM), Enterprise Resource Planning (ERP) or Customer Relationship Management (CRM) offer different functionalities, but they all deal with data about suppliers, products or customers. Different data models require a quite high engineering effort to be integrated. The integration is commonly done by designing integration adapters, which also include the development of several unit test cases (Ritter and Holzleitner, 2015). If the data presents a shared conceptualization, the integration effort can be avoided and also lead to additional benefits. Data can be correctly and uniformly interpreted so to enable building intelligent information systems (Emmenegger et al., 2012, 2017). Moreover, analytic power can be exploited to obtain valuable insights. For example, Business Intelligence applications can use data from different applications to support decision-making and prediction.

Ontologies have been successfully used to effectively combine data from multiple sources (Wache et al., 2001). They address the issue of semantic heterogeneity of data from different sources. Ontologies are formal models with explicitly defined concepts and named relationships. Enterprise Ontologies (Dietz, 2006) contain definitions of business concepts and relationships among them. Although various Enterprise Ontologies (Fox et al., 1996; Uschold et al., 1998; Dietz, 2006; Leppänen, 2005; Bertolazzi et al., 2001) exist, they have not yet achieved a standardization.

The objective of this paper is to present an Enterprise Ontology that can be used as a standardized reference. The ontology serves as a basis for cases where a shared conceptualization and interchangeability of data is required. The ontology is called ArchiMEO and can be extended to address a specific application domain.

The remainder of the paper is structured as follows: section 2 provides the relevant information concerning Enterprise Architecture Descriptions (EAD) and Enterprise Ontologies. Next, section 3 describes the ArchiMEO development approach. An overview of the ArchiMEO structure and the description of the chosen ontology representation language is given in

[a] https://orcid.org/0000-0002-1746-6945
[b] https://orcid.org/0000-0001-9142-7488
[c] https://orcid.org/0000-0002-7909-7663
[d] https://orcid.org/0000-0002-8969-1973
[e] https://orcid.org/0000-0003-1625-0162
[f] https://orcid.org/0000-0002-1825-5234

Hinkelmann, K., Laurenzi, E., Martin, A., Montecchiari, D., Spahic, M. and Thönssen, B.
ArchiMEO: A Standardized Enterprise Ontology based on the ArchiMate Conceptual Model.
DOI: 10.5220/0009000204170424
In *Proceedings of the 8th International Conference on Model-Driven Engineering and Software Development (MODELSWARD 2020)*, pages 417-424
ISBN: 978-989-758-400-8

section 4. Finally, section 5 presents the manner ArchiMEO was used and extended to address specific application domains.

2 STATE OF THE ART ON CONCEPTUALIZATION OF ENTERPRISE KNOWLEDGE

Enterprise Ontologies contain a shared conceptualization of enterprise aspects. There exist several Enterprise Ontologies such as the Toronto Virtual Enterprise (TOVE) (Fox et al., 1996); The Enterprise Ontology (Uschold et al., 1998); Context-Based Enterprise Ontology (Leppänen, 2005); Core Enterprise Ontology (CEO) (Bertolazzi et al., 2001); and Resource-Event-Agent (REA) (Geerts and McCarthy, 2000, 2002).

Enterprise Architectures also describe concepts and relations of an enterprise. A definition of Enterprise Architecture that is in line with the ISO/IEC/IEEE 42010 standard[1] defines an enterprise architecture as "fundamental concepts or properties of an enterprise in its environment embodied in its elements, relationships, and in the principles of its design and evolution." This means that the definition of concepts and relations for describing an enterprise architecture can be regarded as an enterprise ontology.

Enterprise Architecture modeling (and description) and ontology modeling originally stem from two different application domains and recently started to be merged. According to Dietz and Hoogervorst (2008) "the terms 'Enterprise Ontology' and 'Enterprise Architecture' [now] belong to the standard vocabulary of those professionals who are concerned with re-designing and re-engineering enterprises". The term 'ontology' emerged in the context of Artificial Intelligence and the Word Wide Web, particularly of the Semantic Web (Dietz, 2006). The term 'Enterprise Architecture' became generally known as a management topic in the end of the 1980ies, for example through the Zachman Enterprise Architecture Framework (Zachman, 1987). Zachman (2009) has renamed his Enterprise Architecture Framework as an "Enterprise Ontology".

There are two different perceptions of Enterprise Architecture. One perception is as a high level abstraction (of reality) with the purpose of reducing complexity and increasing stakeholder's understanding and communication (amongst others by Chen et al. (2008) and Dietz (2006). According to Dietz (2006) the most dominant problem, stated in scientific as well as in popular science on enterprise management, is complexity and how it can be managed. He claims that because of the complexity of enterprises a conceptual model is needed that "only shows the essence of the operation of an enterprise" and therefore "the model abstracts from all realization and implementation". Dietz (2006) also stated that it is enough to have a conceptual model of Enterprise Architecture, which is independent from any ICT implementation. The other more recent perception of Enterprise Architecture focuses on integrating the graphical models with ontologies (Woitsch et al., 2009; Hinkelmann et al., 2016a; Valtonen et al., 2011) so to provide machine interpretation to models.

There is a huge variety of Enterprise Architecture frameworks like The Open Group Architecture Framework TOGAF (The Open Group, 2018), the Zachmann Framework (Zachman, 2009), the Architecture for Information Systems ARIS (Scheer, 2012), the Best Practice Enterprise Architecture (Hanschke, 2009) and the Enterprise Architecture framework developed in the Plug-IT project (Wache et al., 2010). In his compendium Matthes (2008) gives a detailed description of 34 Enterprise Architecture frameworks, based on clearly structured and well defined criteria.

Schelp and Winter (2009) provide an overview of research on languages for Enterprise Architecture Descriptions. Amongst others they mention ArchiMate (The Open Group, 2017). ArchiMate is a Enterprise Architecture modeling language that was developed by a broad consortium of companies and knowledge institutes (Lankhorst, 2017) and later adopted as a standard by The Open Group. Ettema and Dietz (2009) show ArchiMate is semantically vaguely defined; they demonstrate the benefits of their own ontology for representing Enterprise Architecture Descriptions.

3 ONTOLOGY DEVELOPMENT

To develop our standardized Enterprise Ontology, we applied the widely used development approach described in Noy et al. (2001).

The requirements of business partners were derived first. As suggested in Noy et al. (2001) and Gruninger and Fox (1995), a list of competency questions was created as benchmark. That is, the "ontology is necessary and sufficient to represent the tasks specified by the competency questions and their solution" (Fox et al., 1996). Competency questions are formulated in natural language to determine the

[1] The ISO/IEC/IEEE 42010 (DSCI, 2016) is an International Standard entitled, Systems and software engineering — Architecture description.

scope and evaluate the appropriateness of an ontology. From the questions and answers the required concepts, properties and axioms of the ontology are extracted. Then, competency questions are written formally using first-order logic to specify terminology and axioms (Gruninger and Fox, 1995). The complete list of competency questions is described in (Thönssen, 2013).

The extracted concepts and properties were further refined by interviewing practitioners from different companies. The interviewees proposed to provide a holistic Enterprise Ontology which is able to cover the needed enterprise aspects. The ontology should include different standards and architectures on a general level. The interviewees additionally suggested to consider existing work, from which a basic structure can be derived. The basic structure can then be extended for each company to create domain-specific concepts and relations.

As a second step of the methodology Noy et al. (2001), the interviewees recommendations were addressed as detailed below.

All the Enteprise Ontologies mentioned in Section 2 were analysed. The analysis led to consider them as best practice, guidelines and templates for the development of our standardized Enterprise Ontologies. However, they are proprietary ontologies and there is no evidence for their practical use. The lack of practical usage hinders their wide acceptance, which makes these ontologies unsuitable to be used as standards. Therefore, they were not considered as core of our Enterprise Ontology.

Enterprise Architecture modeling was identified as an additional source for the reuse of already existing concepts. The architecture frameworks listed in Section 2 were examined to identify the main constituents to be represented in an Enterprise Ontology. We identified the concepts and relations that these frameworks provide in order to describe the enterprise architectures. It turned out that the Enterprise Architecture descriptions are not well defined. This is in line with Chen et al. (2008), who were among the first to stress the lack of sound scientific principles for developing Enterprise Architecture descriptions. Given its broad acceptance in practice, the ArchiMate (The Open Group, 2017) standard was finally chosen as the most suitable Enterprise Architecture framework for our standardized Enterprise Ontology.

The ontological representation of ArchiMate constitutes an Enterprise Upper Ontology. However, ArchiMate does not provide general concepts, like location or time. Therefore, we introduced a Top-Level Ontology following (Bertolazzi et al., 2001). Together, the Top-Level Ontology and the ontological representation of ArchiMate build the ArchiMEO ontology (see Figure 1).

As to follow the remaining steps of Noy et al. (2001)'s methdology, (i) the important terms of the ontology were enumerated; (ii) the classes and the class hierarchy were defined; (iii) the properties of classes—slots were defined; (iv) the facets of the slots were defined and finally (v) the instances were created.

4 STRUCTURE OF ArchiMEO

The name ArchiMEO is chosen to indicate its foundation in ArchiMate ("Archi") plus its adaptation and enhancements to serve as a metamodel ("MEO": Meta Enterprise Ontology).

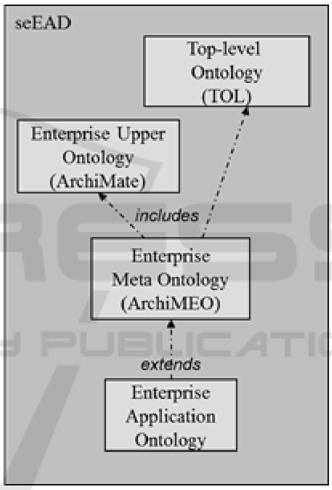

Figure 1: ArchiMEO structure (Thönssen, 2013).

As shown in Figure 1, our semantically enriched Enterprise Architecture Description (seEAD) is logically structured into four parts:

- a Top-level Ontology comprising generic concepts of the world like time, location and event;
- an Enterprise Upper Ontology comprising the ArchiMate concepts represented in an ontology language;
- a meta Enterprise Ontology (ArchiMEO) adapting and enhancing the ArchiMate standard by additional concepts and relations. For example, business processes and business representations such as documents;
- an application-specific ontology, comprising specific concepts of a certain enterprise or domain.

ArchiMEO includes the Top-Level Ontology and the Enterprise Upper Ontology; it can be extended by the Enterprise Application Ontology.

The concepts and relations of seEAD reside in the metamodel (Bézivin, 2004). This is conform to the Meta Object Facility specification (Object Management Group, 2016). Domain-specific enhancements targeting an enterprise can therefore be performed. This allows modelling an enterprise-specific semantically-enriched Enterprise Architecture.

With this approach, an Enterprise Ontology of high quality can be developed and completeness can be ensured. Furthermore, it provides the basis for creating specific views on an Enterprise Architecture Description: for specific viewpoints, model kinds and architecture description languages.

Breaking down the holistic view into top-level concepts leads to structure the semantically-enriched Enterprise Architecture Description. The top-level concepts reflect (a) multiple Enterprise Architecture Frameworks with different aspects and perspectives, (b) different modelling languages that represent one or more of these dimensions and (c) general concepts in enterprise environments such as “Time” or “Location”. ArchiMate concepts are not considered in the top-level hierarchy.

The top-level concepts are detailed as follows:

- The Location and Time concepts consist of spatial and temporal information, respectively.
- The Event concept describes events that are external to a business context (for example environmental disasters that may affect obligations to deliver in a supply chain).
- The ModelType concept consist of notations or modelling languages, which may represent parts of an Enterprise Architecture, e.g. BPMN process models.
- The Perspective concept contains viewpoints according to which the enterprise objects or models can be categorized. For example, the perspectives of the Zachman Framework.
- The Modelling Construct concept depicts the syntax of a modelling language and used to create models. For example, in BPMN a Swimlane is a modelling construct, which may refer to a business entity or role. Thus, a Swimlane represents a modelling construct rather than a distinct concept of an enterprise. It then may be used to represent an enterprise object in a model.
- NCO describes a collection of constructs, which cannot reasonably be assigned to other top-level concepts (e.g. Languages or Specification Standards). NCO stands for ‘non-categorized objects’.
- The Aspect concept categorizes an enterprise object having a certain perspective. In this sense, the Aspect concepts supplements the Perspective concepts. The relationships of enterprise objects with Aspect concepts allow identifying the relevant aspects for certain stakeholders or for the organization.
- The core of ArchiMEO is subordinated to the EnterpriseObject concept. The latter contains all types of objects that occur in an organization. As shown in Figure 2, EnterpriseObject consists of the ArchiMate core concepts along with the relations.

All ArchiMate concepts and relations are subconcepts of the ArchiMEO concept EnterpriseObject. For example, Figure 2 contains the BehaviourElement as a sub-class of EnterpriseObject. Further on, the BusinessBehaviourElement is a specification of the BehaviourElement related to the ArchiMate business perspective and behaviour aspect.

5 THE USE AND EXTENSION OF ArchiMEO

Over the years, the core ontology of ArchiMEO described in Chapter 4 was extended and validated to address specific application domains. An ArchiMEO-based prototype was created in each of the following described use cases.

5.1 Contract Management

Within the DokLife project, a prototype for automatic meta-data creation for contracts was developed (Thönssen and Lutz, 2012). The project goal was to automate and improve a contract’s lifecycle management. Information like contract begin, contract end, contract parties, obligations were automatically extracted using automated metadata generation (Thönssen, 2010). The project used an application-specific extension of the ArchiMEO ontology. This extension defined classes related to contract lifecycle management. For example, specific business events like *bankruptcy* were defined that can trigger actions for contract management.

ArchiMEO was directly linked to external sources as the easyMonitoring business database[2]. Contracted partners are monitored by easyMonitoring and in case

[2]The company Easymonitoring AG offers services for monitoring business partners concerning financial incidents. URL: https://www.easymonitoring.ch/ (retrieved: 19.05.2019)

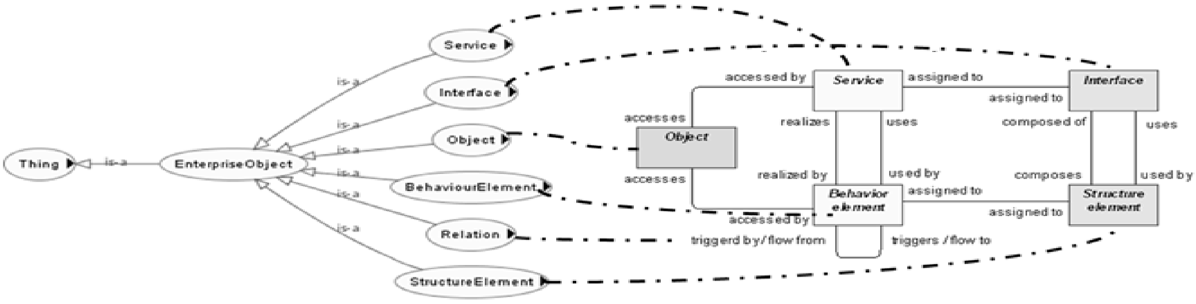

Figure 2: Enterprise Objects Related to the ArchiMate Representation (Thönssen, 2013).

a contract-relevant event happens (e.g. bankruptcy of a partner), automatically an alert is sent and affected obligations and the contract documents in which they are represented are identified based on ArchiMEO reasoning.

5.2 Supply-risk Management

In (Emmenegger et al., 2012), an Early Warning System was developed integrating semantic technologies for the assessment of procurement risks.

ArchiMEO was extended with a Semantic Risk Model. For this, the methodology described in (Gruninger and Fox, 1995) was adopted. Thus, the new concepts and relations were elicited from competency questions. The core of the Semantic Risk Model is based on ontology concepts like *RiskEvent*, *RiskIndicator*, *CrisisPhase*, *WarningSignal* and *Top10ProcurementRisk*.

As risks evolve from external events, concepts from the Top-Level Ontology were integrated to ArchiMEO, i.e., time, event, and location as well as concepts of general interest.

Additionally, semantic rules were developed to be fired against the extended ArchiMEO concepts with the ultimate purpose of inferencing the final level of each top ten procurement risk. Hence, a risk assessment procedure was created embracing a bottom-up approach with values aggregations starting from the detection of atomic risk indicators, e.g. a single-source supplier that is about to go bankrupt or suppliers located in places affected by natural disasters or by political turmoils.

From a technical perspective, the automation of the procurement risk calculation was implemented by integrating an inference engine in the final prototype of the Early Warning System.

5.3 Experience Management

The accessibility of experiential knowledge obtained from previous cases or projects is an important and crucial matter in enterprises and organisations. Experiential knowledge is usually and can be captured in cases, which can be managed and referred to as experience management. Case-based reasoning (CBR) has its roots in cognitive science, machine learning and knowledge-based systems (Martin and Hinkelmann, 2018) and can be used for manage experiences in the form of cases, by supporting the main four R's of CBR retrieve, reuse, revise and retain as main phases to manage cases. To manage enterprise-specific cases with CBR already available generic or specific enterprise knowledge is beneficial.

Martin (2016) developed a new ontology-based case-based reasoning (OBCBR) approach called ICEBERG, which stands for interlinked case-based reasoning, that use the Enterprise Ontology ArchiMEO. The use of ArchiMEO improves the ICEBERG CBR system through the systematic inclusion of enterprise-specific knowledge. Further, the information need about experiential knowledge can be different from one person to another depending on the different roles someone has. To support the different information needs of different stakeholders, ICEBERG was built in such a way that different views, viewpoints, concerns and stakeholders can be considered, which has been derived from the ISO/IEC/IEEE 42010[1] standard likewise as ArchiMEO itself.

The OBCBR foundation of the ICEBERG approach has been laid in the applied research project [sic!] (Martin et al., 2013; Witschel et al., 2015; Martin et al., 2016), which stands for software integration using ontology-based case-based reasoning[3]

The ICEBERG approach has been introduced and described by Martin (2016), and finally been evaluated using a further application scenario (Martin, 2016; Martin and Hinkelmann, 2018). The application scenario was the admission process for master's students at the FHNW School of Business. The triangulated evaluation provided evidence that the

[3]The [sic!] project was funded by the Swiss Commission for Technology and Innovation (CTI).

ontology-based CBR approach support knowledge workers.

5.4 Workplace Learning

ArchiMEO has been used to develop an ontology-based workplace learning approach. The approach aims to support inexperienced employees in public administrations by suggesting historical cases and providing recommendations of experts and learning resources. For this, users' workplace environment is taken into account such as learning preferences as well as required and acquired competencies. Competency questions were created from which both a Domain Ontology (DOMO) and Case-based Reasoning (CBR) Ontology were developed as extensions of ArchiMEO. For instance, the following triples are an excerpt of the new concepts and relations necessary to infer the suitable learning materials: *CompetencyProfile-isAcquiredBy-Worker*; *CompetencyProfile-containsAcquired-Competency*; *Competency-has-CompetencyLevel*; *Worker-hasPreferred-LearningStyle*.

The approach was implemented in the form of a recommendation system, which was integrated to fit the overall architecture of the LearnPAd system platform (De Angelis et al., 2016). The latter comprises mainly the modelling environments, the transformation component, the learning platform's Wiki front-end and the ontology recommender component (including the CBR component). In a first step, enterprise models are transformed into ontology instances conformed to extended ArchiMEO classes and incorporated with the latter. This transformation is called semantic lifting (Kappel et al., 2006). Next, semantic rules are fired against the ontology for recommendation purposes.

5.5 Business Process as a Service (BPaaS)

The Business Process as a Service (BPaaS) (Woitsch and Utz, 2015) use case consists of hybrid usage of models and ontology aiming to align business requirements with cloud offerings. The ultimate goal of BPaaS is to support entrepreneurs in identifying the most appropriate cloud solutions for their business by expressing requirements in a business language rather than an IT language. For this, semantic technologies are used to enable the smart Business-IT alignment. Business requirements are expressed in the form of a business process while Cloud IT-specifications in the form of deployable workflows bundles. Both are modelled employing the BPMN standard and Service Description Model (SDM). The latter extends BPMN to further specify both functional and non-functional (1) business process requirements and (2) IT-specifications.

The existing ArchiMEO concepts representing the BPMN standards were re-used as well as extended to model the class structure of the Service Description Model. The extension work is described in (Hinkelmann et al., 2016b). As an example, non-functional requirements in business language would be *NumberOfProcessExecutionPerYear* and the *FileType* a cloud service should support. From the IT perspective, the values of these two requirements can be used to calculate the minimum amount of *AvailableDataStorage* expected to be offered by a cloud service, i.e. *NumberOfProcessExecutionPerYear * size* of the chosen *FileType*. The new concepts and relations conceived the BPaaS Ontology as an extension of ArchiMEO. In a second phase of the project, two additional sets of ontology representing the functional requirements were created and included in the BPaaS Ontology, i.e. APQC Ontology (reflecting the APQC Process Classification Framework) and the FBPD Ontology (containing the combination between 'verbs' and 'nouns', e.g. Send Invoice).

In a third phase of the project, the BPaaS Ontology was re-used and further extended into the so-called Questionnaire Ontology to create an ontology-based context-adaptive questionnaire (Kritikos et al., 2017). The main advantage of it consists of the significantly reduced time in specifying business requirements, compared to the aforementioned model-based approach. Hence, the questionnaire allows identifying the needed cloud services by answering the least number of questions. For this, a prioritization algorithm was developed, which incorporates the entropy calculation and includes the user preferences, i.e. *Data Security*, *Payment*, *Performance*, *Service support* and or *Target Market*.

6 CONCLUSION

Many companies have to deal with highly complex enterprise-wide IT systems, and lots of interconnected systems need to be managed (Lindström et al., 2006). We developed the ArchiMEO enterprise ontology to deal with the challenge of integrating applications and sharing data in a heterogeneous IT landscape.

Additionally, the representation of the ArchiMate metamodel as an ontology allows for semantic lifting of ArchiMate models (Kappel et al., 2006). The ArchiMEO Enterprise Ontology defines the machineinterpretable semantics of the modeling languages. Because of this 1:1 correspondence of language constructs and ontology concepts, an architecture model can be transformed into an ontology. This so-called semantic lifting Kappel et al. (2006) allows for reuse of knowledge represented in the models for automated reasoning. It has been implemented for workplace learning and BPaaS (see sections 5.4 and 5.5).

Ontology-based metamodeling (Hinkelmann et al., 2018) is an advancement of semantic lifting that seamlessly integrates modeling languages and ontologies. It has been recently implemented in the agile modeling approach described in (Laurenzi et al., 2018), which allows the ontology to evolve over time and to be easily re-used and adapted for new application domains.

In future projects, more model types, resp. languages for models, will be developed and ArchiMEO will be further extended gradually for several other application domains. In parallel the ArchiMEO prototype will be advanced with respect to implementing the link between the ontology and an application database.

REFERENCES

Bertolazzi, P., Krusich, C., and Missikoff, M. (2001). An approach to the definition of a core enterprise ontology: Ceo. In *OES-SEO 2001, International Workshop on Open Enterprise Solutions: Systems, Experiences, and Organizations*, pages 14–15.

Bézivin, J. (2004). In search of a basic principle for model driven engineering. *Novatica Journal, Special Issue*, 5(2):21–24.

Chen, D., Doumeingts, G., and Vernadat, F. (2008). Architectures for enterprise integration and interoperability: Past, present and future. *Computers in industry*, 59(7):647–659.

De Angelis, G., Pierantonio, A., Polini, A., Re, B., Thönssen, B., and Woitsch, R. (2016). Modeling for Learning in Public Administrations - The LearnPAd Approach. In *Domain-Specific Conceptual Modeling*, pages 575–594. Springer International Publishing, Cham.

Dietz, J. L. (2006). *What is Enterprise Ontology?* Springer.

Dietz, J. L. and Hoogervorst, J. A. (2008). Enterprise ontology in enterprise engineering. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 572–579. ACM.

DSCI (2016). Iso/iec/ieee 42010. Website. http://www.iso-architecture.org/42010/; Accessed April 23, 2019.

Emmenegger, S., Hinkelmann, K., Laurenzi, E., Martin, A., Thönssen, B., Witschel, H. F., and Zhang, C. (2017). An Ontology-Based and Case-Based Reasoning Supported Workplace Learning Approach. pages 333–354. Springer.

Emmenegger, S., Hinkelmann, K., Laurenzi, E., and Thönssen, B. (2012). Towards a procedure for assessing supply chain risks using semantic technologies. In *International Joint Conference on Knowledge Discovery, Knowledge Engineering, and Knowledge Management*, pages 393–409. Springer.

Ettema, R. and Dietz, J. L. (2009). Archimate and demo–mates to date? In *Advances in Enterprise Engineering III*, pages 172–186. Springer.

Fox, M. S., Barbuceanu, M., and Gruninger, M. (1996). An organisation ontology for enterprise modeling: Preliminary concepts for linking structure and behaviour. *Computers in industry*, 29(1-2):123–134.

Geerts, G. L. and McCarthy, W. E. (2000). The ontological foundation of rea enterprise information systems. In *Annual Meeting of the American Accounting Association, Philadelphia, PA*, volume 362, pages 127–150.

Geerts, G. L. and McCarthy, W. E. (2002). An ontological analysis of the economic primitives of the extended-rea enterprise information architecture. *International Journal of Accounting Information Systems*, 3(1):1–16.

Gruninger, M. and Fox, M. S. (1995). Methodology for the design and evaluation of ontologies.

Hanschke, I. (2009). *Strategic IT management: a toolkit for enterprise architecture management*. Springer Science & Business Media.

Hinkelmann, K., Gerber, A., Karagiannis, D., Thoenssen, B., Van der Merwe, A., and Woitsch, R. (2016a). A new paradigm for the continuous alignment of business and it: Combining enterprise architecture modelling and enterprise ontology. *Computers in Industry*, 79:77–86.

Hinkelmann, K., Laurenzi, E., Lammel, B., Kurjakovic, S., and Woitsch, R. (2016b). A Semantically-Enhanced Modelling Environment for Business Process as a Service. In *2016 4th International Conference on Enterprise Systems (ES)*, pages 143–152. IEEE.

Hinkelmann, K., Laurenzi, E., Martin, A., and Thönssen, B. (2018). Ontology-based metamodeling. In Dornberger, R., editor, *Business Information Systems and Technology 4.0: New Trends in the Age of Digital Change*, pages 177–194. Springer International Publishing.

Kappel, G., Kapsammer, E., Kargl, H., Kramler, G., Reiter, T., Retschitzegger, W., Schwinger, W., and Wimmer, M. (2006). Lifting Metamodels to Ontologies: A Step to the Semantic Integration of Modeling Languages. In Nierstrasz, O., Whittle, J., Harel, D., and Reggio, G., editors, *Model Driven Engineering Languages and Systems, Proceedings of the 9th International Conference, MoDELS 2006*, pages 528–542. Springer-Verlag, Genova, Italy, lncs 4199 edition.

Kritikos, K., Laurenzi, E., and Hinkelmann, K. (2017). Towards business-to-it alignment in the cloud. In *European Conference on Service-Oriented and Cloud Computing*, pages 35–52. Springer.

Lankhorst, M. (2017). *Enterprise Architecture at Work*. The Enterprise Engineering Series. Springer Berlin Heidelberg, Berlin, Heidelberg.

Laurenzi, E., Hinkelmann, K., and Van der Merwe, A. (2018). An agile and ontology-aided modeling environment. In *IFIP Working Conference on The Practice of Enterprise Modeling*, pages 221–237. Springer.

Leppänen, M. (2005). A Context-Based Enterprise Ontology. In Guizzardi, G. and Wagner, G., editors, *Proceedings of the EDOC International Workshop on Vocabularies, Ontologies and Rules for the Enterprise (VORTE'05)*, pages 17–24, Enschede. Springer Berlin.

Lindström, Å., Johnson, P., Johansson, E., Ekstedt, M., and Simonsson, M. (2006). A survey on cio concerns-do enterprise architecture frameworks support them? *Information Systems Frontiers*, 8(2):81–90.

Martin, A. (2016). *A combined Case-Based Reasoning and Process Execution Approach for Knowledge-Intensive Work*. Phd thesis, University of South Africa.

Martin, A., Emmenegger, S., Hinkelmann, K., and Thönssen, B. (2016). A viewpoint-based case-based reasoning approach utilising an enterprise architecture ontology for experience management. *Enterprise Information Systems*, pages 1–25.

Martin, A., Emmenegger, S., and Wilke, G. (2013). Integrating an enterprise architecture ontology in a case-based reasoning approach for project knowledge. In *Proceedings of the First International Conference on Enterprise Systems: ES 2013*, pages 1–12, Cape Town. IEEE.

Martin, A. and Hinkelmann, K. (2018). Case-Based Reasoning for Process Experience. In Dornberger, R., editor, *Business Information Systems and Technology 4.0: New Trends in the Age of Digital Change*, pages 47–63. Springer International Publishing, Cham.

Matthes, F. (2008). Enterprise Architecture Management Tool Survey 2008. *Information Systems*, pages 1–21.

Noy, N. F., McGuinness, D. L., et al. (2001). Ontology development 101: A guide to creating your first ontology.

Object Management Group (2016). OMG Meta Object Facility (MOF) Core Specification. Website. https://www.omg.org/spec/MOF/2.5.1/PDF; Accessed April 23, 2019.

Ritter, D. and Holzleitner, M. (2015). Integration adapter modeling. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9097, pages 468–482. Springer Verlag.

Scheer, A.-W. (2012). *ARIS—business process modeling*. Springer Science & Business Media.

Schelp, J. and Winter, R. (2009). Language communities in enterprise architecture research. *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology - DESRIST '09*, page 1.

The Open Group (2017). Archimate® 3.0.1 specification. Website. http://pubs.opengroup.org/architecture/archimate3-doc/; Accessed April 23, 2019.

The Open Group (2018). TOGAF 9.2. Website. http://pubs.opengroup.org/architecture/togaf9-doc/arch/index.html; Accessed April 23, 2019.

Thönssen, B. (2010). An enterprise ontology building the bases for automatic metadata generation. In *Research Conference on Metadata and Semantic Research*, pages 195–210. Springer.

Thönssen, B. (2013). *Automatic, Format-independent Generation of Metadata for Documents Based on Semantically Enriched Context Information*. Dissertation, Università degli Studi di Camerino.

Thönssen, B. and Lutz, J. (2012). Semantically enriched obligation management: An approach for improving the handling of obligations represented in contracts. In *International Joint Conference on Knowledge Discovery, Knowledge Engineering, and Knowledge Management*, pages 337–349. Springer.

Uschold, M., King, M., Moralee, S., and Zorgios, Y. (1998). The enterprise ontology the knowledge engineering review, vol. 13. *Special Issue on Putting Ontologies to Use*.

Valtonen, K., Mäntynen, S., Leppänen, M., and Pulkkinen, M. (2011). Enterprise architecture descriptions for enhancing local government transformation and coherency management: case study. In *2011 IEEE 15th International Enterprise Distributed Object Computing Conference Workshops*, pages 360–369. IEEE.

Wache, H., Bergmayr, A., Feldkamp, D., Kondylakis, H., Nikles, S., Plexousakis, D., et al. (2010). plugit. *D3. 2 Specification of Next Generation Modelling Framework. WP3 Specification of the Next Generation Modelling Framework*.

Wache, H., Voegele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H., and Hübner, S. (2001). Ontology-based integration of information-a survey of existing approaches. In Gomez-Perez, A., Gruninger, M., Stuckenschmidt, H., and Uschold, M., editors, *Proceedings of the IJCAI-01 Workshop on Ontologies and Information Sharing (OIS@IJCAI), CEUR Workshop Proceedings 47, CEUR-WS.org*.

Witschel, H.-F., Martin, A., Emmenegger, S., and Lutz, J. (2015). A new Retrieval Function for Ontology-Based Complex Case Descriptions. In *International Workshop Case-Based Reasoning CBR-MD 2015*, Hamburg. ibai-publishing.

Woitsch, R., Karagiannis, D., Plexousakis, D., and Hinkelmann, K. (2009). Business and it alignment: the it-socket. *e & i Elektrotechnik und Informationstechnik*, 126(7-8):308–321.

Woitsch, R. and Utz, W. (2015). Business Processes as a Service (BPaaS): A Model-Based Approach to align Business with Cloud offerings.

Zachman, J. A. (1987). A framework for information systems architecture. *IBM systems journal*, 26(3):276–292.

Zachman, J. A. (2009). John zachman's concise definition of the zachman framework. In *The SIM Guide to Enterprise Architecture*, pages 106–110. CRC Press.