This is a pre-print version of this paper, which was presented at BIS Conference 2018. Please cite it properly:
Timm F., Klohs K., Sandkuhl K. (2018) Application of Inductive Reference Modeling Approaches to Enterprise Architecture Models. In: Abramowicz W., Paschke A. (eds) Business Information Systems. BIS 2018. Lecture Notes in Business Information Processing, vol 320. Springer, Cham

Application of Inductive Reference Modeling Approaches to Enterprise Architecture Models

Felix Timm[1], Katharina Klohs[1], Kurt Sandkuhl[1]

[1] Chair of Business Information Systems, University of Rostock, Germany
{felix.timm, katharina.klohs, kurt.sandkuhl}@uni-rostock.de

Abstract. Enterprise architectures (EA) help organizations to analyze interrelations among their strategy, business processes, application landscape and information structures. Such ambitious endeavors can be supported by using reference models for EA. In recent years, research has increasingly been investigating inductive methods for reference model development. However, the characteristics of EA models have not been considered in this context yet. We therefore aim to adapt existing inductive approaches to the domain of reference enterprise architectures development. Using design science research our work contributes to the reference modeling community with (i) a comparative analysis of inductive reference modeling methods regarding their applicability to EA models, (ii) the refinement of an identified approach to reference EA design and (iii) its application in a use case.

Keywords: Reference Enterprise Architecture, Inductive Reference Modeling, Enterprise Architecture, Literature Review.

1 Introduction

Enterprises need to be aware of the relations among their strategy, processes, applications and infrastructures to be able to rapidly react on changing demands in the market and within their organization. The Enterprise Architecture (EA) research domain contributes to this purpose by providing methods and tools to establish a more holistic perspective on enterprises [1, 2]. This includes to systematically capture and develop an EA using modeling languages like ArchiMate [3]. Such models represent different architectural layers of an enterprise, such as business, application and technology architecture. Since EA projects are highly time- and resource-consuming, organizations would benefit from reference models for EAM. A Reference Enterprise Architecture (R-EA) can be defined as a generic EA for a class of enterprises that is used as foundation in the design and realization of the concrete EA [4].

Many methods exist to develop reference models—both deductive and inductive approaches [5]. Especially, the latter one gains increasing attention by the research community in recent years [6]. Unfortunately, inductive approaches focus primarily on business process model structures [7] and research lacks investigating their applicability towards EA structure like ArchiMate [8]. Nevertheless, we claim that the benefits inductive methods provide for reference process development will also qualify for R-EA development once these methods can be adapted.

In order to fill this research gap, we propose a method for inductive derivation of a R-EA using approaches of existing inductive methods. We therefore use design science research (section 2) in the domain of inductive reference modeling (section 3). We perform a literature review on current inductive methods and assess them regarding their applicability to EA modeling against prior defined requirements (section 4). We further refine one suitable approach to the EA domain (section 5) and demonstrate the new method by dint of a use case (section 6).

2 Research Design

We structure our research report in terms of the design science research (DSR) methodology proposed by [9]. Therein we aim to extend existing methods for developing inductive reference process model to their application to the domain of R-EA development. Peffers et al. define five activities for DSR projects. We performed them as follows [9]:

For the *problem identification,* we conducted a literature study to investigate the existing body of knowledge in the field of inductive reference modeling methods and their existence in the EA domain (see sections 3 and 4). We combined the approaches by Kitchenham [10] and Webster and Watson [11] to thoroughly study existing research. The results verified our stated absence of an inductive approach that addresses EA structures. In the DSR activity to *define the objectives for a solution* we analyzed EA structures and defined requirements they possess towards inductive approaches (see section 4.1). Using these, we performed a comparative analysis and identified suitable approaches and chose the most appropriate one to adapt it for the development of a R-EA, which was the activity to *design and development* in our DSR process (see section 5). The fourth activity is *demonstration* that the artifact actually solves the problem. We therefore applied the resulting artefact to eleven individual EA models and developed a R-EA that fulfilled the prior defined requirements (see section 6). The last DSR activity, *evaluation*, has the intention to observe and measure how well the artifact offers the solution to the given problem. We discuss the benefits and drawbacks of the refined artefact and derive future research activities from these insights (see section 7).

3 Inductive Reference Modeling

In general, reference models are information models developed for a certain problem in a certain application domain. The purpose of their development is to be reused in a concrete application case in this domain. The reuse of a reference model is intended to increase both efficiency and effectivity of an enterprise's information systems and their change management [12]. From a user-oriented perspective, Thomas understands a reference model as a model used to support the construction of another model [13]. From the perspective of reusability, other authors such as vom Brocke argue that reference models are characterized by the concepts of universality and recommendation [12]. The life cycle of such reference models can be distinguished between the phase of construction and the phase of application [5, 14]. By presenting insights in reference enterprise architecture development we contribute to the first phase, i.e. the construction of reference models.

For constructing reference models, research discusses two generic strategies. While the deductive reference modeling derives reference models from generally accepted knowledge, the inductive approach abstracts from individual models to agree on a common understanding within the reference model [15]. Most established reference models have been developed based on deductive approaches [16]. However, inductive reference modeling provides potential because more and more relevant data in terms of logs and concrete information models of organizations are available. Further, inductively developed reference models tend to have a higher degree of detail, are more mature and seem to be more accepted when it comes to reference model application [6].

Fettke provides a seven phase method for inductive reference model construction [17]. The core of the approach is the derivation of a reference model from presorted individual information models. Although Fettke names clustering as a technique to elicit the reference model, it is not made explicit how it is conducted. In recent years, reference modeling related research addresses this topic and approaches were developed. One work applies abstraction techniques from business process mining to the reference model domain [7]. Other approaches utilize for example natural language processing techniques [18], graph theory [19] or clustering methods [20], but always focus on business process model structures.

4 Comparative Analysis of Inductive Reference Modeling Approaches

In this section we systematically investigate inductive methods regarding their applicability to EA model structures. Therefore, section 4.1 defines criteria, which were used to examine, whether an inductive approach is considered applicable to EA model structures. Section 4.2 presents the results of our systematic literature review and thus the identified approaches before we identify suitable approaches in section 4.3.

4.1 Requirements for inductive R-EA development

Through EA models the complex interrelations between an enterprise's organizational and operational structure with used information systems, processed data and realizing technologies are made explicit. Such models consist of layers and elements, which define different perspectives on the enterprise [2]. We use the TOGAF framework [21] as it is widely accepted among practitioners and comes with a modeling language ArchiMate in version 3.0 [3].

In order to derive requirements for the selection of suitable approaches we shortly distinct business process and EA models from each other. Ahlemann et al. state that EA models are more comprehensive than pure business process models since they represent an organization from different perspectives and are not restricted to the business layer

[1]. Greefhorst and Proper define five elements of an EA [22]: *concerns*, which are related to EA stakeholders and group distinct interests on the EA model; architecture *principles* that guide the EA model; an EA *model*, which relates different elements of the EA with each other; *views* that represent a projection of the EA model regarding a certain concern; and, a *framework* that provides a meta model and modeling guidelines. Further, EA models result in more complex model structures. For example, event-driven process chains (EPCs) define six elements and three relation types while ArchiMate's core framework defines 36 elements and 12 relation types. In summary, business process models can be interpreted as a partial model of EA models, since they can be integrated into an EA as Lankhorst et al. demand [2].

REQ1: The approach should be suitable to other modeling languages. The authors of the approach have to mention the applicability to other modeling languages. This might be other process modeling languages or any modeling language. We dismiss the approach if the authors state that it is solely applicable to the utilized model structure. Otherwise, we concluded this by studying the approach in detail.

REQ2: The approach does not solely rely on the control flow of the model structure. Most inductive approaches focus on the control flow characteristic of business process languages like ECP to derive a reference model. Next to such dynamic relationships, EA models use many other relationship types like the assignment (structural), serving (dependency) or specialization relationship. Thus, we dismiss approaches that are not applicable to static model structures.

REQ3: **The approach has to be adjustable regarding concepts of viewpoint and concerns of ArchiMate.** EA models focus on a more aggregated level of detail in contrast to business process models, which represent a much higher level of detail [2]. ArchiMate uses different viewpoints to address different concerns of the model's stakeholders [22]. The approach may use mechanisms to vary the decision what elements to integrate into the reference model depending on a viewpoint's concern. For example, more criteria than a frequency threshold could be used. This also might support the identification of best practices among the individual models.

We consider any approach that fulfills all three requirements suitable for the application of R-EA development. However, we do not directly dismiss approaches fulfilling REQ1 and REQ2 but not REQ3, because it may still be applicable to sub-models of the ArchiMate models.

4.2 Results of the literature review

For a sound systematic literature review (SLR), research questions should be defined that underlie the review process [10]. The aim of our SLR was to identify articles that propose a method or an approach how to derive inductively a reference model from a set of individual models. These have to address the same problem domain in the same modeling language and, thus, can be somehow processed to derive a common practice or a best practice among them. Further, we wanted to analyze these approaches, whether they can be applied to EA models by using the REQs presented in section 4.1.

After several alterations we used the search term `(reference NEAR/4 model*) AND (inductive* OR mining)`. It was performed on the document title, abstracts and keywords and we used a NEAR-operator in order to gather result like "reference process models". Furthermore, we included the term "mining" since related work often use it instead of "inductive" [23]. The query was performed on the databases SCOPUS (296 results), AISeL (20 results) and IEEExplore (94 results) from 2007 until today. This period was chosen since prior work in [6] already conducted a similar search, which we wanted to extend with the most recent approaches. The authors only identified inductive approaches after 2007 in their work. Then, we excluded irrelevant articles by reading all titles and abstracts. This resulted in 16 results, from which we eliminated three duplicates and then reviewed the 13 full documents. We included every article in English and German language that explicitly addressed inductive reference modeling and propose an approach for reference model derivation. We excluded work that offered techniques, e.g. from the process mining domain, but lacked in describing its application to reference model development. Additionally, we used related literature reviews to include relevant articles that we did not capture yet [6, 7, 23, 24]. In the end, we identified 21 approaches for inductive reference modeling. We analyzed them against REQ1, REQ2 and REQ3. The results are presented in Table 1.

Table 1. Identified Approaches and their fulfilment of requirements.
(**"+"** fulfils REQ; **"-"** does not fulfil REQ; **"o"** may be applicable with major adjustments)

APPROACH	REQ1	REQ2	REQ3	DECISION
Rehse et al. [25]	-	-	-	Inapplicable
Scholta [27]	+	+	+	Applicable
Leng, Jiang [28]	+	-	-	Dismissed
Ardalani et al. [16]	+	+	-	Applicable
Rehse, Fettke [31]	+	-	-	Dismissed
Li, et al. [19]	-	**o**	-	Inapplicable
Gottschalk et al. [33]	-	**o**	-	Inapplicable
Rehse et al. [34]	-	**o**	-	Inapplicable
Martens et al. [36]	+	+	-	Applicable
Sonntag et al. [38]	-	+	-	Inapplicable
Martens et al. [40]	+	**o**	-	Dismissed
Li et al. [26]	-	**o**	-	Inapplicable
Ling, Zhang [18]	+	**o**	-	Dismissed
Li et al. [29]	-	**o**	-	Inapplicable
La Rosa et al. [30]	+	**o**	-	Dismissed
Fettke [20]	-	**o**	-	Inapplicable
Yahya et al. [32]	+	**o**	-	Dismissed
Rehse et al. [7]	+	**o**	-	Dismissed
Aier et al. [35]	+	**o**	-	Dismissed
Li et al. [37]	-	**o**	-	Inapplicable
Yahya et al. [39]	+	**o**	-	Dismissed

4.3 Identification of suitable approaches

Only three approaches were identified that are suitable for their application to EA models regarding our requirements, i.e. the approaches from [16, 27, 36]. Although only the approach by Scholta [27] fulfilled all three requirements we deem the approaches by Ardalani et al. [16] and Martens et al. [36] applicable to EA models as well. Nine approaches were assessed as inapplicable. Their authors explicitly state that they were specifically developed for certain model structures like EPCs [34], WS-BPEL [26] or Workflow Nets [33]. Some focus on the analysis of process models' control flow [25, 37] and, thus, cannot be applied to EA model structures. The remaining approaches were dismissed due to major adjustments one would have to make using them for EA model structures. For example, some use graph theory to derive reference models [32], while others use clustering algorithms [30] or transformation matrices for their approach [40]. Although it is possible to represent ArchiMate models in graphs or matrices, the approaches' effort highly increases with the complexity of different relationships ArchiMate defines. We exclude them from the scope of this work, but discuss their potential for future research at the end of this article.

Based on the defined requirements in section 4.1 we chose the approach by Scholta [27] for applying it to ArchiMate models. Scholta's is the only approach that proposes a method for inductively deriving a best practice reference model and therefore defines a number of characteristics that also can be applied to ArchiMate models, which fulfils REQ3. Still, in a prior research paper we already successfully applied the approach by Ardalani et al. [16] to ArchiMate models by manually calculating the minimal graph-edit distances of AchiMate model parts. The same may be done for the approach by Martens et al. [36]. The remainder of this paper depicts our application of Scholta's approach and demonstrates the results by dint of a use case.

5 An Adjusted Approach for Inductive R-EA Development

The approach by Scholta aims to semi-automatically derive reference process models that represent best practices from a set of individual process models [27]. The approach is called "RefPA" and addresses the problem domain of public administration. However, we understand it to be applicable to other problem domains as well, although the author omits to discuss "RefPA's" generalizability. The fact, that the approach, to date, is only conceptually described allows us to modify it on the conceptual level, since the procedure is well documented.

RefPA defines the following objectives: to keep the source models' structure; to identify commonalities of the source models; to consistently group certain segments of the source models; and, to evaluate these groups in order to identify best practices. After collecting all source models in step 0, Scholta defines five steps to develop the reference model. In step 1 all source models are merged together, containing all nodes and edges of any source model. Step 2 subsequently detects all nodes that occur in all source models and, hence, are common elements. In step 3 Scholta uses the concept of the SQL constructs GROUP BY, WHERE and CONTAINS to identify groups in each source model. For instance, process nodes are grouped by a certain document they process (GROUP BY), while they may fulfil certain conditions (WHERE) or a concrete node may have to be in this group (CONTAINS). After groups were identified, they are evaluated and ranked using various criteria. Scholta lists 21 different criteria, such as processed documents, lead-time or personnel costs. Depending on the criteria and the problem domain, the best groups are identified as best practices. Finally, in step 5 all common elements are nodes as well as the best groups are assembled together to a reference models, which contains common as well as best practices, as Scholta claims. In the following, we will discuss these five steps regarding necessary adjustments for ArchiMate models. For more insights into the approach we encourage the reader to study the work by Scholta in [27].

Step 0: Provide Source Models. In contrast to process models, ArchiMate models may be of high complexity. Hence, we define further requirements towards the source models: (i) likewise to [27], we demand every source model to address the same problem domain; (ii) each source model has to comply with a predefined viewpoint structure, which assigns each viewpoint used to a certain EA concern according to [2]—otherwise the source model needs to be aligned to it; and, (iii) in order to guarantee consistent identification of commonalities and best practices, source models have to represent the same level of granularity.

Step 1: Create Merged Model. For each viewpoint from the model structure we conduct one iteration of Step 1 → Step 2 → Step 3 → Step 4. We understand ArchiMate viewpoints as sub models, since they represent different concerns regarding the EA model [2]. The following adjustments address the reference model derivation for one single viewpoint before they are assembled to the resulting reference model in Step 5. A Merged ArchiMate Model is created, using one model repository. Each source model is represented by the corresponding ArchiMate viewpoint. This guarantees a merged model, while the source model's varieties are still documented by single viewpoints. This helps conducting step 2. As a preparation, we create a reference viewpoint for the viewpoint at hand. If the source viewpoints follow a common structure of elements on the lowest level of granularity, it should be integrated in the reference model as well.

Step 2: Identify Common Elements. In line with Scholta, we analyze the source viewpoints for common elements and relations. To date, we assume that elements with the similar semantic are labeled with the similar syntax. In addition, we integrate a frequency threshold as used in [16]. It may vary for each viewpoint and is defined depending on the reference model's purpose. In contrast to Scholta, Ardalani et al. define an element a common practice if it occurs in the majority of source models [16]. A prior defined threshold is applied for each element and relation of the source viewpoint. All elements and relations above the threshold are integrated in the reference viewpoint.

Step 3: Group elements. Although this is essential to derive best practices, it is the vaguest step of Scholta's approach. Again, ArchiMate models are much more complex than process models and, thus, there are a lot different possibilities to group model parts. Still this does not guarantee the identification of best practices. During the case study (see section 6) we identified approaches for grouping the viewpoints' elements. This depends from the EA layer, which the current viewpoint addresses. On the business layer, sub-functions or sub-process may be grouped as well as processes that are assigned to the same business role. On the application layer, data objects may be grouped regarding the phenomenon they relate to (e.g. customer profile). Again, the overall purpose of the reference model has to be considered before grouping elements. For each source viewpoints the same grouping decision has to be applied.

Step 4: Evaluate Groups. We deem all criteria proposed by Scholta to be suitable to ArchiMate models. Next to them, there may exist domain-specific criteria, depending on the problem the reference model intends to address. Consequently, before choosing the criteria one has to understand how the model at hand may help the future model users to solve this problem in the best way. Moreover, it has to be clarified whether the available source models offer the required information. The ranking of element groups may use multiple criteria. For different viewpoints, different criteria may be applied.

Step 5: Assemble Reference Model. After all reference models are developed, the reference viewpoints are integrated into the final reference model. There may exist duplicates, since some elements or relations are defined in various reference viewpoints.

6 Case Study from the Financial Domain

We applied the adjusted RefPA approach to R-EA development in the financial domain. It aims to holistically capture all relevant aspects of the financial organization affected by regulation in order built a reference model that helps financial institutes to effectively and efficiently implement a compliance organization. Therefore, the reference model follows the structure of enterprise architectures and uses. One part of the reference model addresses the prevention of "other criminal acts" (abbr. ssH for German "sonstige strafbare Handlungen"), which is regulated in §25h Abs. 1 KWG and applies for every financial institution [41]. Such crimes may be fraud, corruption or treason.

For this part of the reference model, we conducted and transcribed 11 structured interviews with responsible persons from distinct financial institutes. We developed the structure of the interviews using deductive techniques. Therefore, we consulted compliance experts for a first structure of a ssH prevention system and studied the KWG law as well as guidelines provided by the Federal Financial Supervisory Authority [42]. Afterwards, all interviews were transferred to EA models using the same modeling structure and guidelines, which was done according to Lankhorst [2]. Each individual EA model was structured by eight different ArchiMate viewpoints, which displayed different aspects of the EA models. Each viewpoint was related to a certain purpose. In the following, we concentrate on one specific viewpoint in order to demonstrate the application of Scholta's RefPA approach. As depicted in section 5 the procedure was applied for each viewpoint.

In *Step 0* all requirements were fulfilled: we clarified the model's purpose (i); we defined a thorough viewpoint structure (ii); and, since we used the same structured interviews the same level of granularity of the source models could be assumed (iii). We conducted the subsequent steps for each of the eight ArchiMate viewpoints. This is demonstrated by dint of one viewpoint that addresses how prevention of other criminal acts is conducted on a strategic level. The viewpoint's purpose is to recommend financial institutes what business functions should be implemented on a management level for preventing such crimes and further, what business roles are related to these tasks. For this purpose, we use the Business Function Viewpoint, which is defined by The Open Group [3]. In *Step 1* a merged model was created, which incorporated all business functions and roles as well as their relations among each other. We defined a corresponding reference viewpoint next to the source viewpoints. As a structural framework for the reference viewpoint we used the general management framework we deduced during the interview design—each interviewee verified it. These functions were namely "Building an Organizational Framework", "Corporate Hazard Analysis", "Implementation of a Prevention System" and "Monitoring".

In *Step 2,* we identified common elements and relations in the viewpoint. After a CSV-Export we defined a threshold of 80% and identified 41 common business functions, and five common business roles. We argue that using this threshold still enables to capture commonalities, while we can ignore outliers that may occur due to the interview as a model elicitation method. We further assumed that elements with the same semantic were labeled with the same syntax because we used the same model element library when modeling the source models. During conducting *Step 3,* we decided to group the source viewpoint's models by dint of the four generic business functions identified during Step 1 (see Fig. 1). Group G3 splits into two groups, namely "G3.1 Measures for Internal Hazards" and "G3.2 Measures for External Hazards". The intention was to gather not only the common business function related to the four management functions, but also to accompany them with identified best practices from the source viewpoints. To do this evaluation in *Step 4* we found it most reasonable to use a criterion that identifies the group, which contains most aspects mentioned by the Federal Financial Supervisory Authority in [42]. In the case that multiple groups from different source viewpoints met this criterion, we integrated the group that contained the most business functions. During Step 5 we integrated the groups from the prior steps and assembled the reference viewpoint.

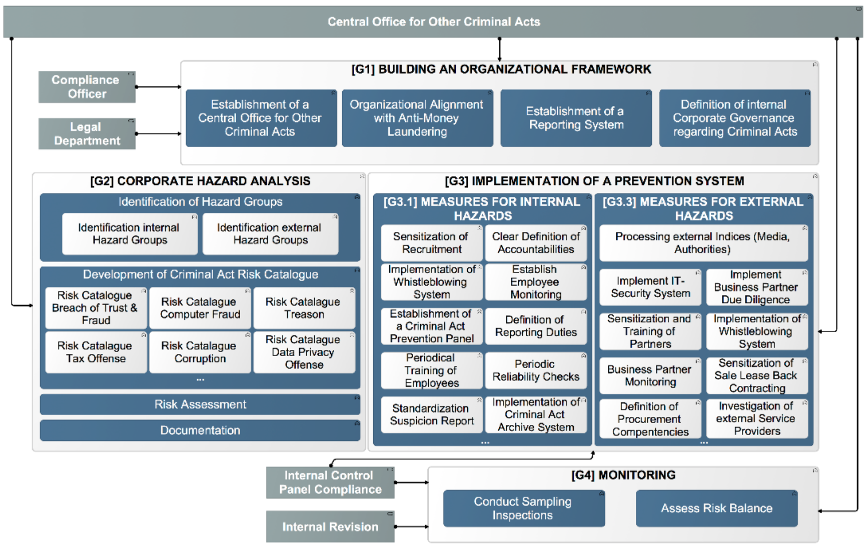

Fig. 1. Reference Viewpoint: Strategic Level of Preventing Other Criminal Acts

Fig. 1 depicts an extract of the developed reference viewpoint for implementing a prevention system for other criminal acts on a strategic level. The complete view has 96 business function elements, which are assigned to eleven distinct business roles via 16 assignment relations. Further, 95 composition relations are included. The Open Group defines a *Business Function* as an internal behavior and groups the behavior based on resources, skills, knowledge, etc. It is performed by a *Business Role* (using the *assignment* relation), which is defined as the responsibility to perform specific behavior and requires certain competencies from individuals to qualify for it. Composition among business functions is represented by dint of the *composition* relation [3]. In **Fig. 1** visualizes this by graphical composition among the business functions.

7 Discussion of Results and Conclusion

In this article we address the research gap that the reference modeling discipline lacks an inductive approach tailored for EA models. By applying a DSR process we therefore define requirements that EA models have towards an inductive derivation of reference enterprise architecture (R-EA) based on multiple individual EA models. We apply these to existing inductive methods that we identified during a systematic literature review, and which almost merely address business process structures. We identify the approach by Sholta [27] to be the most appropriate for our purposes and refine it in order to demonstrate its successful application in a use case for R-EA development. Domain experts validated the resulting R-EA in a first workshop session. Nevertheless, there are several open issues observed for both (i) the developed artefact in particular and (ii) inductive approaches for R-EA development in general we identify for future research.

First, the current approach is conducted manually using ArchiMate modeling tools and spreadsheets. Once the individual EA models meet the input requirements of Step 0, the following steps may be automatized—including the grouping of Step 3. Second, the best practice elicitation of Step 4 needs to be investigated in more detail. Although the experts in the validation phase considered our chosen criteria as plausible, we see flaws in it. Since the individual models were gathered before, may have been certain information we could have inquired, if we would have known what criteria we would use in Step 4 of the approach. Further, the concept of best practice is very vague and should be investigated in much more detail for appropriate criteria selection. Last, when automatizing the approach, it may be interesting to move from a viewpoint-based approach to a model library based approach. This would allow deriving the reference model at once and preventing the effortful model integration.

From a general perspective towards the inductive development of a R-EA, some of the ideas and algorithms of the identified approaches from the literature review should still be considered for further adaptation. First, EA models could be transformed into graph structures in order to automate the merging step; as done in [30, 39]. Second, clustering methods could be applied in order to identify commonalities beyond single EA elements but also sub-models; as done in [18, 39]. Third and last, natural language process could help to analyze syntactical and semantical similarities of EA elements, that are not labeled similarly, but describe the same phenomenon [30, 32].

References

1. Ahlemann, F., Stettiner, E., Messerschmidt, M.: Strategic Enterprise Architecture Management. Challenges, Best Practices, and Future Developments. Springer Berlin (2012)

2. Lankhorst, M.: Enterprise Architecture at Work. Modelling, Communication and Analysis. Springer Berlin; (2017)
3. ArchiMate 3.0 specification. Open Group standard (2016)
4. van der Beek, Wijke ten Harmsen, Trienekens, J., Grefen, P.: The Application of Enterprise Reference Architecture in the Financial Industry. In: Aier, S., Ekstedt, M., Matthes, F., Proper, E., Sanz, J.L. (eds.) 7th Workshop TEAR 2012, Held at The Open Group Conference 2012, Barcelona, Spain, October 23-24, 2012. Proceedings, 131. Springer Berlin Heidelberg
5. Fettke, P., Loos, P.: Referenzmodellierungsforschung. Wirtschaftsinf 46, 331–340 (2004)
6. Rehse, J.-R., Hake, P., Fettke, P., Loos, P.: Inductive Reference Model Development. Recent Results and Current Challenges. In: Mayr, H.C., Pinzger, M. (eds.) INFORMATIK 2016, September 26-30, Klagenfurt, Austria, vol. P-259 GI, Bonn (2016)
7. Rehse, J.-R., Fettke, P., Loos, P.: Eine Untersuchung der Potentiale automatisierter Abstraktionsansätze für Geschäftsprozessmodelle im Hinblick auf die induktive Entwicklung von Referenzprozessmodellen. In: Rainer Alt, Bogdan Franczyk (eds.) WI2013, Leipzig, Germany. (2013)
8. Timm, F., Sandkuhl, K., Fellmann, M.: Towards A Method for Developing Reference Enterprise Architecture. In: Leimeister, J.M., Brenner, W. (eds.) WI2017, St.Gallen (2017)
9. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A design science research methodology for information systems research. Journal of Management Information Systems 24, 45–77 (2007)
10. Kitchenham, B., Pretorius, R., Budgen, D., Pearl Brereton, O., Turner, M., Niazi, M., Linkman, S.: Systematic literature reviews in software engineering – A tertiary study. Information and Software Technology 52, 792–805 (2010)
11. Webster, J., Watson, R.T.: Analyzing the Past to Prepare for the Future. Writing a Literature Review. MIS Q 26, (2002)
12. Vom Brocke, J.: Referenzmodellierung. Gestaltung und Verteilung von Konstruktionsprozessen. Logos, Berlin (2003)
13. Thomas, O.: Understanding the Term Reference Model in Information Systems Research: History, Literature Analysis and Explanation. In: Bussler, C. (ed.) BPM Conference, Nancy, 3812, Springer, Berlin (2006)
14. Thomas, O.: Management von Referenzmodellen. Entwurf und Realisierung eines Informationssystems zur Entwicklung und Anwendung von Referenzmodellen. Logos-Verl., Berlin (2006)
15. Becker, J., Schütte, R.: Referenz-Informationsmodelle für den Handel: Begriff, Nutzen und Empfehlungen für die Gestaltung und unternehmensspezifische Adaption von Referenzmodellen. In: Krallmann, H. (ed.) Wirtschaftsinformatik '97, Physica-Verlag HD, (1997)
16. Ardalani, P., Houy, C., Fettke, P., Loos, P.: Towards a minimal cost of change approach for inductive reference model development. ECIS (2013)
17. Fettke, P.: Eine Methode zur induktiven Entwicklung von Referenzmodellen. In: Tagungsband Multikonferenz Wirtschaftsinformatik 2014, MKWI 2014, pp. 1034–1047 (2014)
18. Ling, J., Zhang, L.: Generating hierarchical reference process model using fragments clustering. In: Asia-Pacific Software Engineering Conference, vol. 2016-May (2016)
19. Li, C., Reichert, M., Wombacher, A.: Discovering reference models by mining process variants using a heuristic approach. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 5701 LNCSvol. , pp. 344–362 (2009)
20. Fettke, P.: Integration von Prozessmodellen im Großen: Konzept, Methode und experimentelle Anwendungen. In: Thomas, O., Teuteberg, F. (eds.) Proceedings der 12. Internationalen Tagung Wirtschaftsinformatik (WI 2015), pp. 453–467. Osnabrück (2015)

21. TOGAF Version 9. Van Haren Publishing, Zaltbommel (2010)
22. Greefhorst, D., Proper, E.: Architecture Principles. The Cornerstones of Enterprise Architecture. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg (2011)
23. Rehse, J.R., Fettke, P.: Towards Situational Reference Model Mining - Main Idea, Procedure Model & Case Study. In: Leimeister, J.M., Brenner, W. (eds.) Proceedings der 13. Internationalen Tagung Wirtschaftsinformatik (WI2017), pp. 271–285. St.Gallen (2017)
24. Schoknecht, A., Thaler, T., Fettke, P., Oberweis, A., Laue, R.: Similarity of Business Process Models—A State-of-the-Art Analysis. ACM Comput. Surv. 50, 1–33 (2017)
25. Rehse, J.R., Fettke, P., Loos, P.: An execution-semantic approach to inductive reference model development. 24th, ECIS 2016
26. Li, C., Reichert, M., Wombacher, A.: Discovering reference process models by mining process variants. In: Proceedings of the IEEE International Conference on Web Services (2008)
27. Scholta, H.: Semi-automatic inductive derivation of reference process models that represent best practices in public administrations. 24th ECIS 2016 (2016)
28. Leng, J., Jiang, P.: Granular computing–based development of service process reference models in social manufacturing contexts. Concurrent Engineering 25, 95–107 (2016)
29. Li, C., Reichert, M., Wombacher, A.: Mining business process variants. Challenges, scenarios, algorithms. Data & Knowledge Engineering 70, 409–434 (2011)
30. La Rosa, M., Dumas, M., Uba, R., Dijkman, R.: Business process model merging. An approach to business process consolidation. ACM Transactions on Software Engineering and Methodology 22 (2013)
31. Rehse, J.R., Fettke, P.: Mining reference process models from large instance data. In: Lecture Notes in Business Information Processing, 281, pp. 11–22 (2017)
32. Yahya, B.N., Bae, H., Bae, J.: Generating valid reference process model using genetic algorithm. International Journal of Innovative Computing, Information and Control 8, (2012)
33. Gottschalk, F., van Der Aalst, W.M.P., Jansen-Vullers, M.H.: Mining reference process models and their configurations. In: LNCS 5333, pp. 263–272 (2008)
34. Rehse, J.R., Fettke, P., Loos, P.: A graph-theoretic method for the inductive development of reference process models. Software and Systems Modeling 16, 833–873 (2017)
35. Aier, S., Fichter, M., Fischer, C.: Referenzprozesse empirisch bestimmen. Von Vielfalt zu Standards. Wirtschaftsinformatik & Management : WuM : die Praktikerzeitschrift für Wirtschaftsinformatiker 3, 14–22 (2011)
36. Martens, A., Fettke, P., Loos, P.: A genetic algorithm for the inductive derivation of reference models using minimal graph-edit distance applied to real-world business process data. In: Tagungsband MKWI 2014, pp. 1613–1626 (2014)
37. Li, C., Reichert, M., Wombacher, A.: THE MINADEPT CLUSTERING APPROACH FOR DISCOVERING REFERENCE PROCESS MODELS OUT OF PROCESS VARIANTS. Int. J. Coop. Info. Syst. 19, 159–203 (2010)
38. Sonntag, A., Fettke, P., Loos, P.: Inductive Reference Modelling Based on Simulated Social Collaboration. In: Leimeister, J.M., Brenner, W. (eds.) WI2017, St.Gallen (2017)
39. Yahya, B.N., Bae, H.: Generating reference business process model using heuristic approach based on activity proximity. In: Smart Innovation, Systems and Technologies, vol. 10 (2011)
40. Martens, A., Fettke, P., Loos, P.: Inductive Development of Reference Process Models Based on Factor Analysis. In: Thomas, O., Teuteberg, F. (eds.) Proceedings der 12. Internationalen Tagung Wirtschaftsinformatik (WI 2015), pp. 438–452. Osnabrück (2015)
41. Gesetz über das Kreditwesen. KWG (2017)
42. Bundesanstalt für Finanzdienstleistungsaufsicht (ed.): Auslegungs- und Anwendungshinweise zu § 25c KWG ("sonstige strafbare Handlungen") (2011)