Enterprise Modelling and Information Systems Architectures
Vol. 14, No. 3 (2019). DOI:10.18417/emisa.14.3

A Procedure Model for Situational Reference Model Mining

Jana-Rebecca Rehse[*,a], Peter Fettke[a]

[a] Institute for Information Systems (IWi) at the German Research Center for Artificial Intelligence (DFKI GmbH) and Saarland University, Campus D3.2, 66123 Saarbrücken, Germany

Abstract. *This contribution introduces the concept of Situational Reference Model Mining, i. e., the idea that automatically derived reference models, although derived from identical input data, are intended for different purposes and therefore have to meet different requirements. These requirements determine the reference model character and thus the technique that is best suited for mining it. Situational Reference Model Mining is based on well-known design principles for reference modeling, such as configuration, aggregation, specialization, instantiation, and analogy. We present a procedure model for Situational Reference Model Mining and demonstrate its usefulness by means of a case study. Existing techniques for Reference Model Mining are examined and mapped to their underlying design principles. Our approach provides reference model designers with first guidelines regarding their choice of mining technique and points out research gaps for the development of new approaches to reference model mining.*

Keywords. Reference Model Mining • Reference Model Design • Reference Model Design Principles • Reference Model Construction • Inductive Reference Model Development

Communicated by A. Koschmider. Received 2018-07-04. Accepted after 3 revisions on 2019-04-01.

1 Introduction

1.1 Reuse-oriented Reference Modeling

Reference models can be considered as special conceptual models that serve to be reused for the design of other conceptual models (vom Brocke 2002, 2007). They provide a template for process models in a certain industry and thus facilitate a resource-efficient implementation of the respective process and its adaption to the individual needs of an organization. This way, companies may have access best practices and industry-specific experience. They are able to benefit from the advantages that are associated with Business Process Management (BPM) without investing a lot of resources. These advantages include a higher quality of processes and process models, as it simplifies internal communications by introducing a common terminology (Fettke and Loos 2007).

Under a reuse-oriented conceptualization of reference models, their main purpose is to serve as an orientation in the design of new business process models. In this context, we decipher two general design processes (vom Brocke 2007; vom Brocke and Fettke 2018), as outlined in Fig. 1. Deriving an individual model from a reference model is known as "Design With Reuse". An existing model is used as a blueprint offering guidance to the process model designer by giving suggestions for both content and design of the individual model. Conversely, "Design For Reuse" (DFR) describes the process of constructing a (reference)

---

[*] Corresponding author.
E-mail. Jana-Rebecca.Rehse@iwi.dfki.de
The research in this paper was partly supported by a grant from the German Research Foundation (DFG), project name: "Konzeptionelle, methodische und technische Grundlagen zur induktiven Erstellung von Referenzmodellen (Reference Model Mining)", support code GZ LO 752/5–1. The authors would like to thank the reviewers and the editor-in-chief for their valuable comments, which helped to improve this paper.
Note: This work is based on a previously published conference paper (Rehse and Fettke 2017) and has been considerably adapted and extended.

model for the purpose of being reused, i. e., composing model parts and domain knowledge, such that they achieve a certain degree of universality.

Reusing a reference model entails adopting the model content as well as adapting and extending it to fit the specific application context. In a typical model construction process, the model designer creates a model according to the user's requirements, employing specific methods. The construction process is influenced by both the model quality (effectiveness) and the required time and cost (efficiency). Reference models can be understood as tools that foster both the effectiveness and the efficiency of model construction. They include contents that are relevant for different application contexts (i. e., for DFR) and may serve as the basis for several construction processes (i. e., for DWR). As the model contents do not have to be newly constructed and have already been applied, both effectiveness and efficiency are increased.

1.2 The Reference Modeler's Dilemma

In order to leverage these benefits, organizations need to have access to an applicable, high-quality reference model. Ideally, such a model already exists and is widely established. If not, organizations can design their own company-wide reference model. This is particularly advisable for larger organizations, who execute similar processes at multiple locations, such as a manufacturing company with multiple production sites or a multinational company with multiple subsidiaries. Smaller organizations may motivate and support their respective professional association in activities to design an industry-wide reference model. SMEs might prefer this strategy, such that they are able to standardize their support processes and focus on their core business.

When challenged with the creation of a new reference model, designers face the so-called "reference modeler's dilemma": The less adaptations are necessary to apply a reference model in a company-specific context, the more valuable it is perceived by its users, but the less potential users it has. A more universally applicable model will have more potential users, but each individual application requires more adaptation effort, reducing the value for all users (Becker et al. 2002; vom Brocke 2002). Each designer has to decide individually how to resolve this dilemma for a reference model at hand, as it depends on the intended application context. Therefore, they need a construction method that allows them to directly influence the reference model design. This is the main advantage of manual construction methods. However, these methods typically require a lot of resources (time, cost, personnel), are error-prone, and include subjective decisions that are hard to reproduce. Automated methods are usually resource-efficient, objective, and easily reproducible, but because the method is not adapted to the application context, the resulting reference model is agnostic to its purpose. Hence, automated design decisions make it impossible for designers to address the dilemma in the best possible way.

The latter problem appears in Reference Model Mining (RMM), the research field concerned with (semi-)automatic reference model construction (Rehse et al. 2017). Given a set of input models describing similar processes in different organizations, a RMM technique will abstract from organization-specific features and compute a reference model that comprises the input model's commonalities. In recent years, researchers have developed many new RMM techniques, examining the potentials of different methods for determining model similarities and merging them into a single model. As a result, different mining techniques yield different reference models, even when applied to the same set of input models. Because techniques do not provide further instructions on how to use these models, reference model designers cannot determine the best technique for their specific use case.

In this contribution, our goal is to combine the best of both worlds, i. e., the efficiency of automated methods with the precision of manual ones. We develop a method that requires designers to invest some manual resources in order to achieve

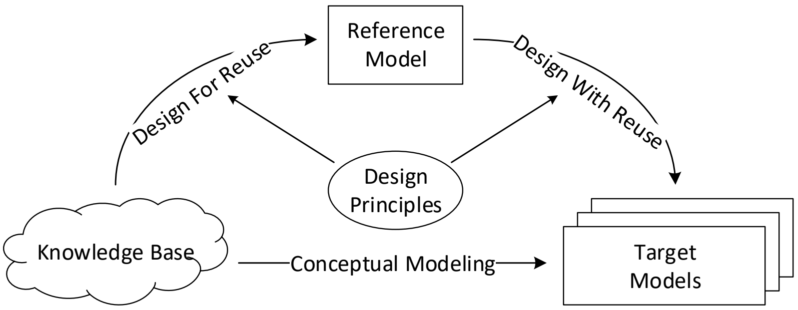

Figure 1: Reuse-Oriented Reference Modeling (cf. vom Brocke 2007; vom Brocke and Fettke 2018)

a reference model that is better suited to their situational purpose. By matching their application context with our characterization of RMM techniques, designers can choose the technique that will yield the best reference model for their purpose. We call this concept “Situational Reference Model Mining” (S-RMM).

1.3 On Situational Reference Model Mining

S-RMM extends RMM towards consciously considering the situational context when designing and using a reference model. This idea is described in Fig. 2, based on the reuse-oriented reference model design process from Fig. 1. RMM techniques automatically derive a reference model from a set of input models (DFR). This reference model is then used for constructing target models in a certain application context (DWR). Enterprise modeling research knows several techniques for constructing a conceptual model from another one. These so-called design principles describe how the original model is adopted, adapted, and extended to create a new model. In the reference modeling context, the five design principles configuration, instantiation, specialization, aggregation, and analogy have been examined in detail (vom Brocke 2007). In S-RMM, the principles apply in both the DFR process, where individual models are analyzed and merged into a reference model, and the DWR process, where the target model is constructed based on the reference model.

The quintessential idea is that mining technique is ultimately determined by the situational context of the reference model application. The goal of reference model design is a model that can be reused to create valuable target models. A model’s value depends on the purpose for which it is used. So, to choose the best RMM technique, designers first have to examine the reference model application context and its requirements to the target models. These requirements determine the design principle that is best suited for constructing the target models, which in turn imposes restrictions and requirements on the reference model. In order for the reference model to fulfill these requirements, the right design principle has to be used in the mining process. This combination of design principles will guide designers in selecting and applying the most appropriate RMM technique.

Let us illustrate this idea with a simple example: A multinational company wants to design a reference model for its administrative processes to better align its national subsidiaries. Some processes, such as taxation, are subject to national law and can therefore not be standardized by the parent company. The reference model should contain “placeholder elements”, such that the subsidiaries can fill in their national taxation processes during target model construction, using the instantiation principle. For other processes, such as compliance documentation, the subsidiaries must strictly adhere to the parent company’s regulations. The reference processes are therefore designed on a

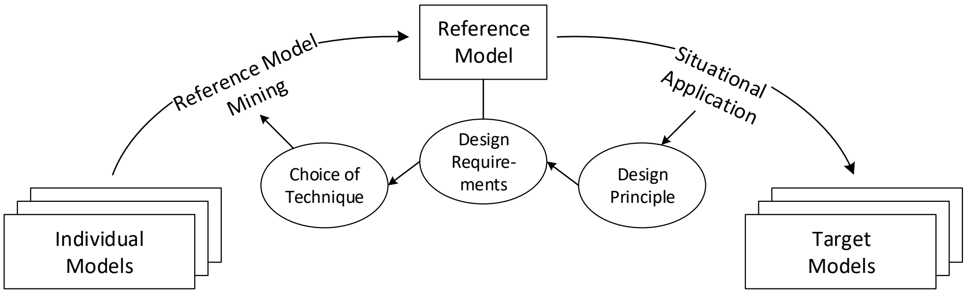

Figure 2: Main Idea behind Situational Reference Model Mining

high level of detail and the subsidiaries design their target models by means of the analogy principle. Both reference models, although designed within the same organization, target a different context with different requirements, and likely require different design methods.

1.4 Research Questions and Outline

To examine the concept of S-RMM, this article is guided by the following research questions:

1. How can RMM be used to design a reference model for a given application case, considering situational contextual factors?
2. How can RMM techniques be matched with situational contexts to produce applicable reference models?
3. In which situational contexts can existing RMM technique be applied? Where are emerging research gaps?

To answer these questions, we follow a design science research approach (Hevner et al. 2004; Peffers et al. 2007). Our research is motivated by the objective to elaborate how existing concepts in reuse-oriented reference modeling can be applied to the relatively new field of RMM. The designed artifact is a procedure model for the situational reference model design, along with a set of guidelines to match a situational context with a design principle and, subsequently, a RMM technique.

The contribution has the following structure. To allow for a better understanding of our contributions, we first clarify our chosen research approach in Sect. 2. The spectrum of related work is presented in Sect. 3. Sect. 4 introduces the concept of S-RMM by defining, explaining, and discussing a ten-stage procedure model for S-RMM. Its application in terms of the five design principles is discussed in Sect. 5, where we analyze the respective requirements and list matching mining techniques regarding their application in a situational context, in order to give concrete guidelines to reference model designers. In Sect. 6, the procedure model and accompanying guidelines are applied and demonstrated in two different scenarios. Case study results and remaining challenges are discussed in Sect. 7, before the paper is concluded with an outlook on future work in Sect. 8.

2 Research Approach and Contributions

In this section, we explain and justify our research approach, motivating its adequacy to answer the research questions described above. Holistically speaking, this article is set out to make a contribution to the ever-evolving research stream of Business Engineering (BE), illustrated in Fig. 3. BE denotes the construction of organizations with engineering methods, considering the organizational strategy, the business processes necessary for its realization, and the supporting software systems (Fettke 2008; Winter 2003, 2008). The strategy level defines the value creation by means of customer processes, which are realized and implemented on the process level. The software or

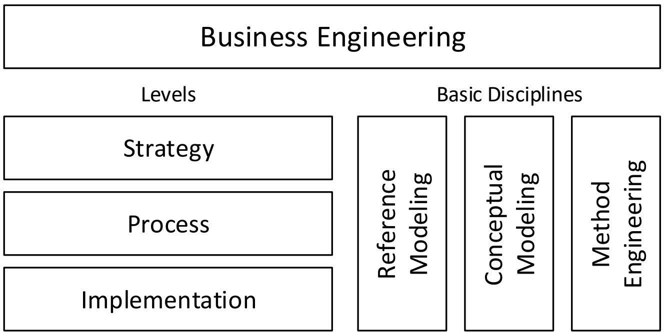

Figure 3: Business Engineering: Levels and Basic Disciplines

implementation level is concerned with supporting the processes with IT solutions.

Across all levels, several disciplines form the basis of BE research and practice. Reference modeling, which is the main discipline addressed in this article, is concerned with designing blueprints and templates to be used as guidelines in the design process. Conceptual modeling describes the graphical representation of tangible and intangible aspects in organizational design. Method engineering transfers the construction concepts towards method design. Each of these discipline entails syntactic aspects concerning the objects and rules necessary for realizing a design, semantic aspects concerning the meaning of these objects, and pragmatic aspects concerning the contextual aspects that influence how these objects are used.

BE is a broad research discipline, covering multiple different areas that share business design as a common interest. The spectrum of applied research methods, as shown in Fig. 4, reflects this breadth. It ranges from very formal, mathematical approaches to very informal, textual ones. Whereas the former focus on defining and proving mathematical properties with formal methods, the latter rely on natural language as their main means of communication. For example, a process is often depicted as a process model, which is essentially a mathematical object (a directed graph) but bears its meaning from natural language in form of the attached textual labels. So, both viewpoints can be justified. In fact, most researchers do not strictly adhere to one approach or the other, but instead position themselves somewhere on the spectrum between the two poles, adjusting the research approach to their individual research object (method engineering). Accordingly, research methods range from formal languages and mathematical proofs over graphical, but still machine-executable process models and human-readable, semi-formal conceptual models to arguments in plain text. As a rule of thumb, the more pragmatic aspects are considered, the less formal a research method will be, as these aspects are typically very difficult or even impossible to formalize.

One of the main ideas in this article is to transfer ideas from method engineering (Henderson-Sellers et al. 2014) into reference modeling, such that reference modeling methods can be designed situatively according to pragmatic requirements to both the model and the method. Our article employs a rather pragmatic research approach, relying much more on prose descriptions than on mathematical definitions. This is due to the following reasons, which make strictly formal approaches impractical or even inapplicable in reference modeling.

Definitions are (still) controversial. After decades of reference modeling research, the definition of a reference model (in plain text) is not yet conclusively established (please refer to Fettke and Loos (2007) and Fettke and vom Brocke (2018) for

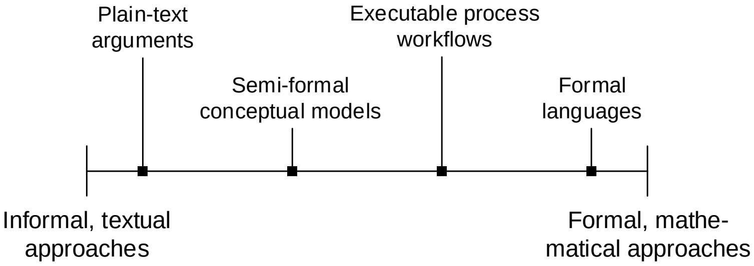

Figure 4: Spectrum of research methods in BE

a summary of the discussion). There appears to be a consensus on the definitional properties "best practice", "universal applicability", and "reusability", which are difficult to define in prose and practically impossible to define in a formal way. However, this imprecision does not influence the relevance of reference models in both research and practice. Any kind of formal definition would not only fall short in expressing all the discussion aspects, it would also unnecessarily restrict the applicability of our research.

Reference modeling is directed towards modeling practitioners. For many researchers, including us, the characterizing property of a reference model is its reusability, i. e., its ability to be used as a conceptual framework in a variety of information system projects (Fettke and Loos 2007, p. 4). This reusability property is fulfilled, if the reference model is adopted into practice. Hence, reference models are designed by researchers or practitioners with the objective to support modelers in designing higher-quality models with less resources. Since reference models are not meant to be implemented in their as-is form, but should act as guidelines in business process (re-)design and implementation projects, precise semantics and formal correctness are secondary. The objective of a reference model is to be understood by humans rather than machines.

None of today's well-established and widely used reference models are formally defined. Models like the IT Infrastructure Library (ITIL) for IT service management (ITSM), the Supply Chain Operations reference model (SCOR) for supply chain management, the enhanced Telecom Operations Map (eTOM) as a reference model for the telecommunications industry, or the SAP reference model as a basis for their ERP system, are ubiquitously used and generally accepted in practice today. However, none of them is defined by means of formal modeling semantics. The ITIL documentation, for example, does not even use a graphical modeling language, but relies solely on textual descriptions of relevant processes, focusing instead on elementary process steps, stakeholders, interfaces, and a common terminology, which are far more important when designing ITSM processes.

This article addresses RMM, which we define as "(semi-)automatically deriving a reference model from a set of input models". One could argue that an automated derivation of reference models is impossible without a high degree of formalization. In fact, drawing from the closely related discipline of process mining, most existing RMM approaches (as listed in Tab. 2) are highly automated and therefore highly formalized. However, the concept of S-RMM, which is introduced in this article, is not just another RMM approach. It is motivated by the observation that different RMM techniques applied to the same input data yield different reference models. Assuming that each approach is valid in itself, how can we select the best technique for designing a reference model?

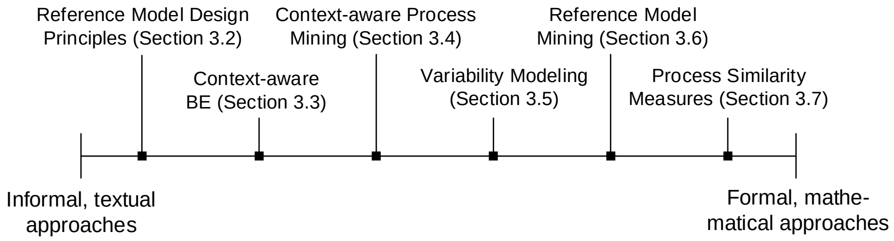

Figure 5: Related Work on S-RMM positioned on the spectrum of BE research methods

The answer to that question depends on the reference modeling context, i. e., the tangible or intangible requirements, stakeholders, and purposes that influence a reference model in a specific situation. Not all of these properties will be represented in RMM input data, but they all play an important role in the design process. Hence, reference model designers need to be equipped with a method to consider these "soft" factors, even when using RMM techniques. Designing such a method is the main objective of this article, which makes the following contributions:

- We introduce the rationale of S-RMM as a method to support model designers in their application of RMM techniques.
- We provide a ten-stage procedure model, built around existing reference model design principles, to guide the design process.
- We analyze existing RMM techniques regarding their instantiation of design principles, providing a concrete guideline, which techniques to use for which principle.
- We identify emerging research gaps regarding (1) a lack of RMM techniques for certain principles and (2) further steps towards establishing S-RMM in practice.

3 Related Work

3.1 Spectrum of Related Work

Even though our contribution employs a pragmatic research approach, it is related to a number of other fields of BE research, spread across all levels of formalization. Fig. 5 summarizes the areas of related work related to S-RMM. As we elaborate above, reference modeling research is inherently informal. Hence, the five reference model design principles, which form the foundation of our contribution, are explained in plain text and supported by conceptual models. On the other end of the spectrum, process model similarity measures, which are needed to identify commonalities between input models, are typically fully formalized and can be automatically computed and expressed in numerical terms. In between, there is a number of other areas related to S-RMM. Developing a reference model in a context-dependent way requires discussing the meaning of "context" in BE and the ways in which they are applied in approaches for context-aware process mining. Variability modeling is closely related to RMM, as they are both concerned with integrating families of individual process models. Existing RMM techniques are the basis for S-RMM. Typically, they are fully automated, but need manual adaptions to increase the reference model quality. Most of these techniques rely on a matching between input models, which may be computed using a state-of-the-art process matching algorithm. In the following, we report on relevant areas of related work, arranging them according to their degree of formalization, from informal to formal.

3.2 Reference Model Design Principles

A design principle is a rule that describes how the content of one model is used in the construction process of another. This includes adopting, adapting, extending, or deleting the model content. In his conceptualization of reuse-oriented reference model design, vom Brocke (2007) identifies configuration, instantiation, specialization, aggregation, and analogy as particularly relevant design principles. As we base our work on this contribution, these are the principles we examine here. Other principles (such as modification suggested by Delfmann (2006)) are further elaborated in the discussion section. In the following, each principle is textually described and illustrated with a value chain diagram (Fettke et al. 2006).

Configuration is the most well-known design principle, hence also the best documented one. It entails adopting or deleting parts from the original model according to the context of the process model domain. Individual model parts are either deleted if they are not necessary in the given context, or selected and derived from a configurable component, which the modeling language may provide. In this case, the model is deliberately configurable, e. g., by industry or size. In our illustration in Fig. 6, the model is configurated by deleting parts P1a and P3b, because they are not necessary in the given configuration context.

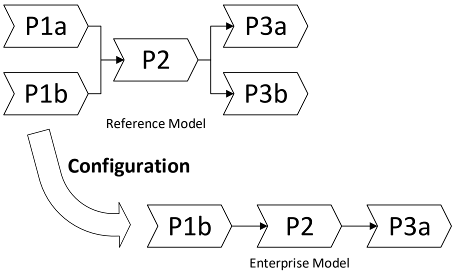

Figure 6: The configuration principle

Instantiation describes the concretion of generic model element by means of a new model. Therefore, the reference model must contain generic placeholders such as process interfaces, which represent general domain aspects. They are designed as a framework for plugging in model parts, considering the requirements of the application domain. Both the framework and the individual requirements can be reused. This principle is illustrated in Fig. 7. The higher-level process interface P2 is refined by a concrete process R, which is adapted to the individual context of the process and provides more details and individual steps.

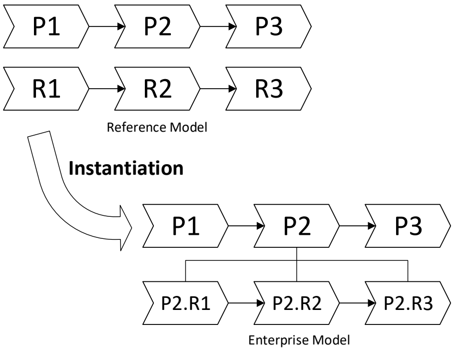

Figure 7: The instantiation principle

Specialization refers to deriving a concrete model from a rather generic one. Entire contents of the latter are adopted into a specific new model, allowing individual modification and extension. The resulting model contains all content of the generic model, potentially on a higher level of detail. This allows adapting the generic model to specific demands of a certain context. This principle is illustrated in Fig. 8. The generic process is specialized by refining element P2 into two sub-elements P2a and P2b, thus increasing the level of detail.

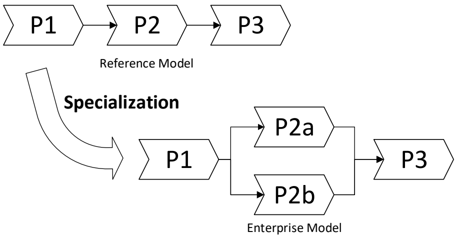

Figure 8: The specialization principle

Aggregation denotes combining two or more individual models in order to form a new one. Content delivered by various input models is adopted into the new model, composed and extended as necessary. The resulting model is composed of the individual model parts. This principle allows adopting model parts from different contexts into a new context, replenishing and integrating them as needed. This way, models can be seen as a reservoir for individual building blocks instead of being monolithic. This is illustrated in Fig. 9. Two individual models are linked to form a new model, with a broader scope.

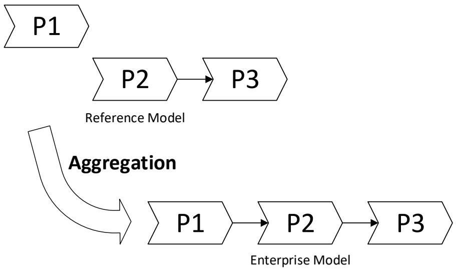

Figure 9: The aggregation principle

Analogy describes a situation, where an individual model serves as an orientation in the design of a new model, such that they are perceived to be coinciding in certain aspects. Seemingly similar solutions are employed in a creative way to tackle new problems. This principle allows for a high degree of freedom regarding the model design. Model parts may or may not be adopted and modified as the context suggests. However, the model quality may decrease in the design process, as the analogy principle does not pose any formal requirements. The principle is illustrated in Fig. 10, where the process elements are reordered aiming to define an new process, which may be applied in a new context.

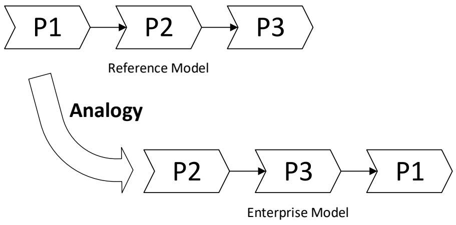

Figure 10: The analogy principle

3.3 Context-aware Business Engineering

The term “context” can generally be defined as “all information intrinsic to an entity” (Chihani et al. 2012) respectively “any information that can be used to characterize the situation of an entity” (Dey 2001). The idea of adapting an artifact, such as a method or a model, to fit the specific context in which it is used, is not new. It originated in the software engineering domain, where both researchers and practitioners realized that a rigid one-size-fits-all methodology for developing a software project is not only unattainable but also inefficient (Henderson-Sellers et al. 2014, p. 5). *Situational Method Engineering*, i. e., the idea to create, use, and adapt a software development method for a particular situation, was developed based on this insight (Henderson-Sellers et al. 2014, p. 3). Other examples stem from the field of Ubiquitous Computing (Chihani et al. 2012). Researchers from these areas provide numerous explanations and formalizations of the terms “situation” and “context”, which could also be adapted into the BPM domain (Kornyshova et al. 2010).

Regarding business processes, the context describes “the environment in which a business process artefact is used” (Born et al. 2009), formalized in terms of a “minimum set of variables containing all relevant information that impact the design and execution of a business process” (Rosemann and Recker 2006). Context variables can be distinguished according to their proximity to the process itself (Rosemann et al. 2006):

1. Immediate Context: Elements beyond the control flow, but essential to understanding and executing the process (e. g., data, resources).
2. Internal Context: Information on the internal organizational environment that impacts the process (e. g., strategy, stakeholders).
3. External Context: Elements beyond the organizational control sphere, but within its business network (e. g., industry-specific practices).
4. Environmental Context: Factors outside the business network, but still relevant to the process (e. g., weather, laws).

The UN/CEFACT defines a "business context" as "the formal description of a specific business circumstance as identified by the values of a set of context categories, allowing different business circumstances to be uniquely distinguished" (UN/CEFACT 2009, p. 58) and differentiates eight different categories of context with regard to their topic (Saidani et al. 2015):

1. Business Process: Activities and goals of the process
2. Product Classification: Goods and services involved in the process
3. Industry Classification: Business sector, addressed markets, and business partners involved in the process
4. Geopolitical Context: Geographical, political, or cultural influences on the process
5. Official Constraints: Requirements impended by laws, regulations, conventions or other legal or governmental restrictions
6. Business Process Role: Actors directly involved in the process
7. Supporting Role: Actors indirectly involved in the process
8. System Capabilities: Limitations of the surrounding systems and standards

There are approaches to provide a formalization of "context" in process modeling to make it concretely usable with modeling languages (Saidani and Nurcan 2007, 2009) as well as a generic model to provide a better understanding of the factors, which a process context might entail (Saidani et al. 2015).

3.4 Context-aware Process Mining

S-RMM is related to process pining in the sense that process mining approaches can generally be used to mine reference models, but not every process mining result can be considered or used as a reference model. Postulating a reuse-oriented understanding of reference modeling, every process model has the potential to be used as a reference model. However, process discovery techniques require a type of different input data (i. e., event logs) and usually measure the resulting models' quality in a different way. "Generalization" as a measure in process mining refers to including additional (unobserved) process behavior in the model (van Dongen et al. 2016), which may not be consistent with the objective of a reference model to provide a more generic process pattern.

Besides conventional discovery techniques, there are other process mining approaches which take situational context into account and combine informal context descriptions with formal process definitions. Van der Aalst and Dustdar (2012) argue that process mining should consider a broader spectrum of contextual factors, as they might influence the process execution. They suggest a division into four context categories:

1. Instance Context: Properties directly related to the individual process instance, such as the order size or type of customer.
2. Process Context: Properties relevant to the process itself, such as available or allocated resources or the number of currently running cases.
3. Social Context: Properties of the executing organization, such as the prioritization of the process, the ability of people to work together in social collaborations, their current stress levels, and internal competition.
4. External Context: Properties beyond the organizational control, such as the weather, economic climate, the current season, and applicable laws.

As in S-RMM, considering the context of a process mining project might lead to better and more specific answers. Some authors have therefore published approaches on how to consider context in process mining. For example, J. Li et al. (2011) mine a process log according to context-dependent process patterns, which the user selected. Bose and van der Aalst (2009) cluster process traces based on their proximity to discover better-fitted process models. Folino et al. (2012) use a similar approach to make more accurate predictions on process performance measures. Mounira and

Mahmoud (2010) suggest a context-aware process mining framework, with the goal to increase process flexibility. In a domain-specific approach, Becker and Intoyoad (2017) apply context-aware process mining in the logistics industry. A comprehensive overview over the different dimensions, properties, and existing approaches to context-aware process mining is given by Koschmider et al. (2019), who conduct an extensive literature review in order to derive a framework for context-dependent mapping of events and activities.

Like discovery approaches, there are other techniques that could generally be applied for RMM, although that is not their primary use case. For example, process model merging is primarily intended for process consolidation, but a consolidated model can also be interpreted as a reference model (La Rosa et al. 2013). The same applies to process model integration, especially in a hierarchical way (Fettke 2015). If the reference model development is targeted towards certain quality aspects, it might make sense to choose it accordingly from process model configurations (Schunselaar et al. 2014).

3.5 Variability Modeling

Variability modeling addresses the issue that companies typically maintain multiple variants of the same process and require techniques to manage them efficiently. It is closely related to reference modeling in general, and RMM in particular, as they both follow the overall objective to integrate a family of similar processes into one model. La Rosa et al. (2017) provide an overview on state-of-the-art variability modeling mechanisms, separating them into four groups based on how they handle variability in the models.

The first group centers on element-based configuration. Those models contain configurable elements (i. e., functions, events, or gateways) that have to be adapted to the target domain when deriving a model variant. For example, a configurable OR-connector can turn into an AND-connector or an XOR-connector in the target model. The element configuration principle is mainly realized by extending existing modeling languages with configurable elements, such as Configurable integrated EPCs (C-iEPCs) (La Rosa et al. 2011; Rosemann and van der Aalst 2007) or Configurable Workflows (C-YAWL) (Gottschalk et al. 2008b; van der Aalst et al. 2006b). Application-based Domain Modeling (ADOM) (Reinhartz-Berger et al. 2009, 2010) works with cardinalities instead of concrete modeling elements and is therefore independent from the modeling language.

The second group focuses on configuring process variants. The model domain is characterized by one or multiple attributes, which can take on different values. For each value or value combination, a complete process variant is specified, such that the model contains multiple process variants at once. When deriving a target model, the best applicable process variant is chosen according to the attribute values for the target domain. Mechanisms that apply this principle are for example Configurative Process Modeling (Becker et al. 2002), Superimposed variants (Czarnecki and Antkiewicz 2005; Czarnecki et al. 2005), or Aggregated EPCs (aEPCs) (Reijers et al. 2009).

Modeling mechanisms from the third group are centered around activity specialization. Typically, they define multiple variants of a generic activity (or “variation point”), one or several of which are then adapted into the target model. The PESOA mechanism (Process Family Engineering in Service-Oriented Applications) by Puhlmann et al. (2005) has applied this to BPMN and UML, although it is directed towards software development, not process modeling. Business Process Family Model (BPFM) (Moon et al. 2008) follows a similar fashion, but only for UML. In Feature Model Composition (Acher et al. 2009, 2010), activities can only be specialized in terms of their input and output.

The fourth and final group of mechanisms works by customizing fragments into new model variants. This is realized by applying change operations to model parts in order to design a target model. The Provop mechanism (Process variants by options) by Hallerbach et al. (2010) defines the reference model as a “base model” with so-called adjustment points where model fragments can be deleted, inserted, moved or modified. Template and Rules (Kumar and Yao 2012) also gives a reference model as a base and defines rules, which determine how this model is adapted in different target model contexts.

3.6 Reference Model Mining

Reference Model Mining (RMM) describes the (semi-)automated derivation of a reference model from a set of individual models by identifying commonalities in a set of input models and constructing a new model on that basis (as illustrated in Fig. 11). This entails automatically detecting similarities between the input models and determining the reference model content. Simultaneously, individual model- or company-specific features are abstracted to ensure that the reference model has a certain degree of universality. The absence of a specific process context enables its re-use. Since process models are typically based on graph structures with textual labels, RMM combines both formal and informal aspects, e. g., finding isomorphic substructures or selecting an appropriate yet representative set of input models.

Due to the complexity of finding, abstracting, and integrating process model commonalities, RMM is related to many other fields of research, including process analytics, natural language processing, process matching, process mining, and graph theory. In this regard, it makes sense to rely on results from closely related disciplines to ensure that RMM approaches reach their full potential. So, as is discussed in Rehse et al. (2017), inductive reference modeling can be separated into two major subproblems, addressing different research questions.

Question 1: How to identify analogies between the input models? As we will see from the RMM approaches below, similarities between process models can exist in different ways, e. g., based on model structure or execution semantics (Becker and Laue 2012). However, each similarity measure is based on correspondences between nodes in the compared models. Such correspondences, called matches, express equivalence or at least a high degree of similarity between node pairs from different models. Finding matchings between process models is the objective of a number of state-of-the-art process matching algorithms, which combine syntactic, semantic, and structural information aiming to identify non-trivial correspondences between model elements (Antunes et al. 2015; Cayoglu et al. 2013). In order to leverage these results, many RMM techniques rely on a provided matching between the input models as an additional input. The decision whether to construct the matching manually or use an existing matching technique is left to the user, ensuring maximum quality and flexibility.

Question 2: How to integrate input model similarities into a reference model? While the first research question is left to process matching, this is the main question addressed by RMM. Based on a set of input models and a matching between them, how can we identify similarities between the input models and integrate them into a reference model, while abstracting from the individual properties of each model? A matching is a first and often necessary, but usually not sufficient step in identifying those model features that constitute the final reference model content. As shown in the next section, this question can be answered in a number of different ways, resulting in a variety of choices for the right RMM technique.

This separation corresponds to the three-stage similarity calculation framework, presented in Schoknecht et al. (2017). Computing the similarity between process models can be separated into a matching phase, where the element-based matches are determined, and a similarity, which is concerned with finding the similarity between the process models, based on the identified matches. Analogous to RMM, some similarity measures require a matching, while others do not. In contrast, similarity measures aim at numerically expressing the similarity between process models, whereas RMM goes a step further by identifying the common features and integrating them into a new model.

A large body of research describes concrete techniques and approaches for RMM. However, they do not rely on a methodological foundation

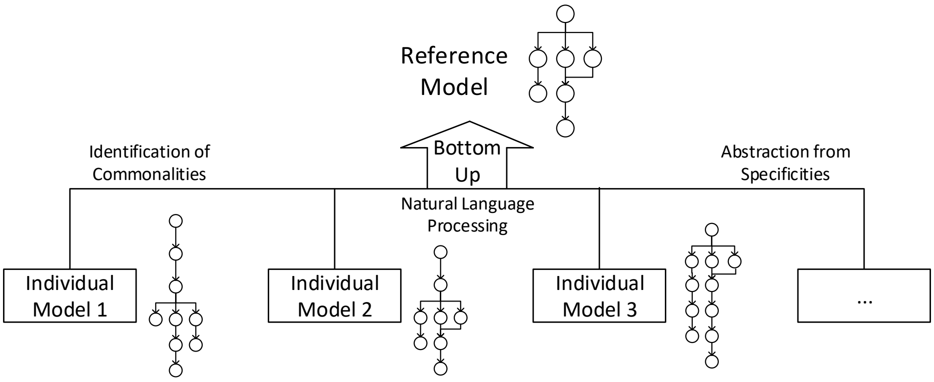

Figure 11: Reference Model Mining

or reflect on the ways of model construction and the requirements of specific use cases. Process variants may either be mined in relation to an existing reference model or without one (C. Li et al. 2011). Different similarity measures, such as frequents common substructures (Rehse et al. 2017) or heuristic approximations of the graph edit distance (Ardalani et al. 2013) are used to determine input model commonalities. Other approaches employ configurable process models (Gottschalk et al. 2008a), genetic algorithms (Martens et al. 2014; Yahya et al. 2012), or process model abstraction (Rehse et al. 2013). While all of these authors support the general purpose of RMM, none of them acknowledge the differences between existing approaches or indicate in which context their suggested technique would be especially useful.

Different mining techniques employ different similarity measures (e. g., structural (Ardalani et al. 2013) or semantic (Rehse et al. 2016)) and construction methods (e. g., deterministic (Rehse et al. 2017) or heuristic (Yahya et al. 2012)), resulting in differences between the mined reference models. In addition, due to restrictions on the input models, not every technique may be applied to every set of input models. Some contributions describe the influence of a single parameter, or a combination of parameters, on the resulting reference model. For example, a frequency threshold will determine the model size and thus the character. The higher the threshold, the smaller and the more generic the resulting reference model. This influences the underlying design principle (e. g., a larger size corresponds to the configuration principle), but not explicitly mentioned as such.

Some authors apply S-RMM by inductively developing reference models for a particular use case in a certain domain, without explicitly considering a generic procedure model or specific design principles (Aier et al. 2011; Gröger and Schumann 2014; Karow et al. 2008). Others have developed techniques that are specific to one domain, such as public administration (Scholta 2016), interorganizational services (Leng and Jiang 2017), or focus on other models, such as enterprise architectures (Timm et al. 2017).

Our intention here is to extend the existing concept of RMM to consider the situational context, i. e., the intended target models, when choosing and executing a mining technique. Therefore, we aim to create unified guidelines for S-RMM, which reference model designers can use for an easier and better application of RMM. Depending on the characteristics they intend for their reference model and target models, designers should be able to make informed choices on their design principles and suitable mining techniques.

3.7 Process Similarity Measures

Identifying similarities between process models is one of the cores in inductive reference modeling.

As discussed above, RMM techniques approach similarity from different angles. They draw from results in process similarity analysis, which summarizes formalized approaches to expressing the degree of analogy between two process models in form of a number between 0 (completely different) and 1 (identical) (van der Aalst et al. 2006a). Regarding formal methods, both Dijkman et al. (2011) and Schoknecht et al. (2017) differentiate three basic dimensions for measuring process similarity, namely the natural language dimension, the structural dimension, and the behavioral dimension.

The natural language dimension focuses on comparing the activity labels within the process. On a syntactical level, two labels are assessed regarding the minimum number of change operations (insertion, deletion, substitution) required to transform one into the other. These operations can be performed on characters (known as string edit distance) or analogously on words (Schoknecht et al. 2017). To address the problem of synonyms and homonyms in labels, some approaches consider the semantic level, i. e., the meaning of the words. This requires additional external resources such as the WordNet ontology, linguistic databases like Wiktionary, or NLP-specific techniques such as parsing and part of speech tagging (Schoknecht et al. 2017). Label correspondences are the basis for several process similarity measures, such as the similarity score based on common activity labels (Akkiraju and Ivan 2010) or label matching similarity (Dijkman et al. 2011).

The second similarity dimension focuses more on the formal definition of process models as labeled graphs, defining similarity based on the differing model structures. The graph edit distance, which transfers the idea of the edit distance to graph structures, is a very popular measure for graph similarity. Due to its exponential complexity, process similarity analysis typically uses heuristic approximations (Dijkman et al. 2009, 2011). Other structural similarity measures include the percentage of common nodes and edges (Minor et al. 2007) and the feature-based similarity estimation (Yan et al. 2010).

Finally, the third dimension tries to take the process model character into account, considering its behavior in terms of the executable process instances. This is beneficial in the sense that due to ambiguities in most process modeling languages structurally different models can still be behaviorally similar or even equivalent. A first notion to measure the degree of process equivalence based on observed behavior is presented by van der Aalst et al. (2006a). Other approaches that require the generation of actual process instances are the longest common subsequence (lcs) of traces (Gerke et al. 2009) or the similarity of causal footprints (van Dongen et al. 2008). Behavioral profiles (Weidlich 2011) are matrices representing the relations and dependencies between the nodes of a process model. They can be derived directly from the model, allowing for a more efficient similarity measurement.

4 Procedure Model for Situational Reference Model Mining

4.1 Overview

In this section, we present a general procedure model for S-RMM (shown in Fig. 12) and describe each of its ten stages in detail. The procedure model is based on the conceptualization of S-RMM in Fig. 2. It describes a generic execution process of an S-RMM application. Each of the ten steps belongs to one of the two generic design processes. DWR is concerned with the target model construction (i. e., the reference model application), while DFR focuses on the reference model construction (i. e., the actual mining). In the generic S-RMM process, DFR (seven steps) is executed before DWR (three steps). In the following, each of the ten stages is explained in detail.

4.2 Determine Situational Context

As a basis for any S-RMM application, the situational context has to be analyzed by determining the intended use for the target models and its necessary characteristics. To analyze the situational

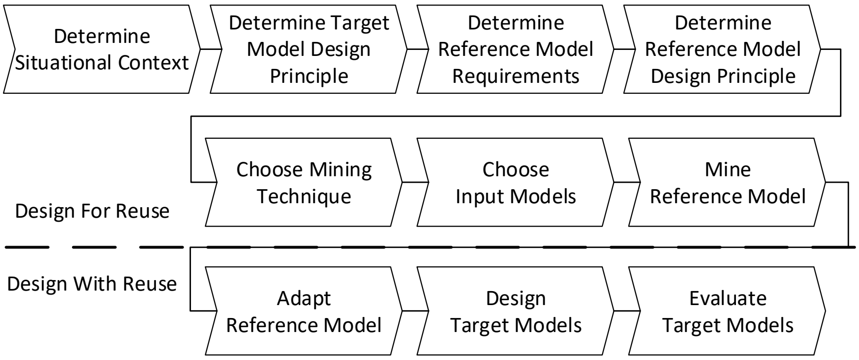

Figure 12: Procedure Model for Situational Reference Model Mining

context of a target process, one can follow different context categorizations, as we have discussed above. Tab. 1 lists the eight context categories specified by UN/CEFACT (2009) and gives an example how each of them might determine the context of a to-be-defined process. The situational context is also influenced by the available input models, as their characteristics determine which design principles may be reasonably applied. The result of this step is a complete description, which allows designers to fully understand the given context.

4.3 Determine Target Model Design Principle

Depending on the situational context and the inferred requirements, the target model design principle is chosen, based on the assessment which of the five principles satisfies the requirements best. Since the design principles are not formally defined, the choice might not be definite, since the context might allow for the application of different principles. In such a case, the choice can be further influenced by other factors, such as the availability of input models or the number and maturity of applicable mining techniques. The chosen principle determines the DWR part of the process.

4.4 Determine Reference Model Requirements

Based on the design principle, several requirements to the reference model design can be determined. These mainly follow from the principle itself, instead of the situational context. However, if certain requirements are not given in a context, some principles might not be applicable. For example, applying configuration might require a configurable reference model, while instantiation calls for generic process interfaces, i. e., placeholder elements in a reference model, which can be individually specified in the target model. The requirements for each principle are elaborated in Sect. 5. The result of this step is a concise list of reference model requirements.

4.5 Determine Reference Model Design Principle

Depending on the required design of the reference model, the process designer has to choose the design principle that should be applied in the mining process in order to fulfill these requirements. In general, there is no restriction on the combination of principles, however, not every combination is supported by an actual mining technique (cf. Tab. 2). Hence, this step is limited not only by the situational context, but also by the availability of mining techniques.

Table 1: Examples for Context Factors according to the eight Categories (UN/CEFACT 2009)

Context Category	Context Factor Example
Business Process	Goals, actions (e. g., digitalizing incoming invoices in a payment process)
Product Classification	Tools (e. g., production machines needed for cutting out metal)
Industry Classification	Customers (e. g., all technology users as customers of an internal IT department)
Geopolitical Context	Cultural idiosyncrasies (e. g., additional checks by a supervisor in more hierarchically structured societies)
Official Constraints	Taxation laws, regulations (e. g., documentation obligation in the pharmaceutical industry)
Business Process Role	Active participants (e. g., sales employee involved in closing a deal)
Supporting Role	Indirect participants (e. g., internal IT staff involved in setting up a CRM system)
System Capabilities	Potentials and limitations (e. g., maximum number of daily work pieces in a production process)

4.6 Choose Mining Technique

Choosing an applicable and appropriate RMM technique is influenced by the chosen target and reference model design principles, but also the situational context that was previously analyzed, as the reference model has to fulfill a number of constraints. Some mining techniques qualify for multiple combinations of target and reference model design principles. In those cases, several measures can assist in choosing an appropriate technique.

- Analyze input models: Some mining techniques pose requirements to the input models, such as block-structuredness or the absence of loops. Although the final input models are determined in the next step, a technique is inapproriate if it cannot be applied to most of the pre-selected models. Especially when handling large or complex models, this step may also benefit from using automated process analysis techniques such as process metrics (Mendling 2008) and process similarity measures (Becker and Laue 2012; Dijkman et al. 2011).
- Compare mining techniques: The design principles only provide a rough distinction between mining techniques. They may also differ in many other aspects, which are more or less favorable in the given context. These include similarity measure and model construction technique, but also parametrization, the availability of external resources, or the processing order. If, for example, some input models are more important than others, a technique that allows for an input model weighting is more applicable than one that treats all input models equally.
- Determine external or practical aspects: The situational context also has an influence on the practical aspects of reference model mining. Therefore, non-functional aspects such as the necessary amount of manual pre-processing or the computational complexity of the chosen technique may also have an impact.
- Compare resulting reference models: Finally, if none of the other steps allows to select one technique over the others, it might be necessary to compare the resulting reference models to determine the best fit for the situational context.

4.7 Choose Input Models

Usually, a set of input models is selected prior to beginning the mining process, as they determine the situational context. However, due to possible restrictions and requirements, the final set of input models can only be selected after the mining technique is chosen. For example, some mining techniques require their input models to be free of loops or duplicate activities. Others are limited to a certain modeling language. In addition to choosing the input models, they might have to be pre-processed (e. g., by making sure certain conventions are fulfilled) and additional input such as a matching between the input models might have to be generated. As the result of this step, all input should be gathered that is necessary to automatically mine the reference model in the next step.

4.8 Mine Reference Model

The reference model is obtained by applying the mining technique to the chosen set of input models. As they depend on both the input data and the situational context, potential parameter configurations have to be individually determined to yield the best-fitting reference model. This might require several iterations to fully determine the influence of the parameter on the resulting reference model. The choice of parameters often determines the target model design principle, so it has to be handled with care. For example, when using aggregation as the reference model design principle, the value of a frequency threshold makes the difference between designing the target models by means of configuration, specialization, or analogy, as seen in Tab. 2. The result of this step is a raw reference model that is finalized in the following step.

4.9 Adapt Reference Model

As RMM techniques are usually fully automated, the resulting model may not fulfill all the requirements derived from the situational context. Hence, it may have to be manually adapted. Adaptation methods include adding, deleting, or renaming nodes, complementing the reference model with deductively developed model parts, changing the modeling or labeling conventions, transforming the model into another (modeling) language, or any other measure that simplifies the design of the target models in the following step. As a result of this step, a designer should obtain a finalized reference model adapted to the given situational context.

4.10 Design Target Models

After the reference model is finalized, it can be used for the target model construction. Therefore, the design principle determined in step 2 is now applied to the reference model. Each target model undergoes a separate construction process, where the individual model requirements are addressed in the best possible way. Due to the previous construction steps, the reference model fulfills the requirements of the target model design principle, as described in Sect. 5. For example, if the target models are to be designed by means of instantiation, the reference model will contain generic process interfaces that can be substantiated by the target model context. Depending on the differences between the individual target context and the chosen design principle, the designed models may or may not differ substantially from each other. The objective of this step is the generation of a set of individualized target models for the given situational context.

4.11 Evaluate Target Models

The goal of applying the S-RMM procedure model is to design a reusable and thus useful reference model and use this as a foundation for high-quality target models. Hence, in the last step, the target models are evaluated against the requirements derived from their intended use case. This step may lead to individual adaptations of the target models, but may also serve to enhance the reference model for further reusing. In addition, this step allows process designers to reflect on the S-RMM process as a whole, pointing out potential improvements.

5 Analysis of Design Principles and Existing Mining Techniques

5.1 Overview

In this section, we examine each design principle regarding its applicability for RMM. In order to provide a guideline for applying the S-RMM procedure model, we analyze existing mining techniques regarding their underlying principles and requirements, as summarized in Tab. 2. For each target model design principle, we suggest corresponding reference model design principles and, for each pair, suitable mining techniques. The analysis is restricted to those combinations of target and reference model design principle described in Sect. 3.2 that are supported by a mining technique.

Tab. 2 suggests both techniques that are explicitly meant for RMM (or inductive reference model development), such as process variant clustering (C. Li et al. 2011), and techniques that are originally intended for another use case, but can be employed accordingly, such as process model merging (La Rosa et al. 2013). A technique was selected if (a) it takes a set of models as input, (b) it outputs a single model that is in some way based on the input models, (c) it is fully automated, and (d) it describes a domain-independent method that can be applied to any set of input models. This excludes methodical frameworks such as Fettke (2014), partially manual approaches such as Gröger and Schumann (2014), or empirical, domain-specific reports such as Karow et al. (2008).

The table can be seen as a complementary guideline for using our procedure model, as it makes concrete suggestions for realizing stage five (Choose mining technique). Each design principle, with its characteristics, requirements, and consequences is analyzed in more detail in the following sections.

5.2 Configuration

Employing configuration as a target model design principle has the objective to derive a target model by selectively adopting parts of the reference model according to the restrictions and specifications of the target model context. In order to allow a selection of model parts in different target contexts, the reference model must cover all aspects that are relevant for a domain. This means that a configurable reference model must be *inclusive*, subsuming all (or most) variations of certain domain aspects to increase its applicability. On the other hand, configuration does not include adding or changing model parts, so the reference model must be characterized by an applicable set of activities and an appropriate level of granularity.

The requirements to a configurable reference model have implications for the mining process in terms of the mining technique and the choice of input models. These models have to consider the variation in the covered domain aspects, as they provide the main reference model content. While some additional content can be added during a manual post-processing step, the majority of the model content should be added during the mining phase, ensuring an inductive model construction. This means that the input models have to be selected in a way that they belong to the same domain, but describe different instantiations of the process within that domain. For example, the models may describe the same process, but vary in terms of sub-domain (e. g., warehousing or wholesaling in the retail domain), organization size (e. g., SME or MNC), local jurisdiction (e. g., administration in different states or countries), or organizational structure (e. g., functional or divisional).

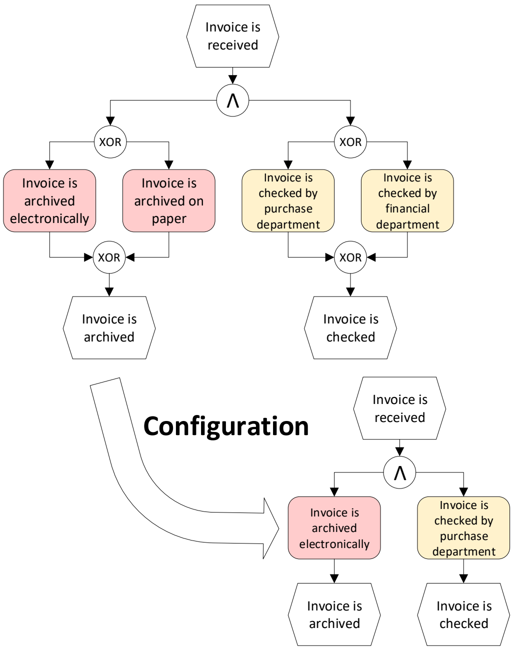

Figure 13: Using configuration as a target model design principle

Table 2: Analysis of Matching Principles and According Mining Techniques

Target Model Design Principle	Reference Model Design Principle	RMM Techniques
Configuration	Aggregation	Ardalani et al. (2013), Fettke (2015), Gottschalk et al. (2008a), La Rosa et al. (2013), C. Li et al. (2011), and Rehse et al. (2016, 2017)
	Analogy	C. Li et al. (2011), Martens et al. (2014), and Yahya et al. (2012)
Instantiation	Aggregation	Rehse et al. (2013)
Specialization	Aggregation	Ardalani et al. (2013), La Rosa et al. (2013), C. Li et al. (2011), and Rehse et al. (2016, 2017)
	Analogy	C. Li et al. (2011), Martens et al. (2014), and Yahya et al. (2012)
Aggregation	–	–
Analogy	Configuration	Gottschalk et al. (2008a) and Schunselaar et al. (2014)
	Aggregation	Ardalani et al. (2013), Fettke (2015), C. Li et al. (2011), and Rehse et al. (2016, 2017)
	Analogy	Martens et al. (2014) and Yahya et al. (2012)

When applying a mining technique to such a set of models, it is essential to include all input models features into the reference model, such that it is maximally configurable. On the other hand, potential commonalities should be identified and merged, such that the reference model is generically applicable. This means that the reference model must be designed by employing either aggregation or analogy. Aggregation is applicable, if all the input models are to be equally considered in the design process and their individual features should be merged to construct a new model. Selected mining techniques must either include entire process models (such as Fettke (2015) and La Rosa et al. (2013)) or offer a threshold for the minimum frequency of model parts (such as Ardalani et al. (2013) and Rehse et al. (2016)). If this threshold is low, the mined reference model is inclusive and thus configurable. Analogy is applicable, if one input model is used as the basis for the reference model and is enriched by additional aspects in the design process.

Fig. 13 shows an exemplary application of configuration as a target model design principle. The reference model describes an auditing process consisting of two subprocesses, archiving the invoice and checking the invoice. The model provides two options for performing each subprocess. It is evident from the model that the XOR-connectors between the process options are considered as design-time decisions instead of the process flow being routed at runtime. An organization intending to design their process based on this reference model needs to configure the model such that it is compliant to the invoice archiving process. If the specific context is not represented in the reference model, configuration cannot be (solely) applied for designing the reference model.

To conclude, *configuration* is an applicable design principle, if

- the reference model should subsume different characteristic values of some domain aspects,
- the input models describe the same process in different contexts, and
- the target model context is covered by one or several input models.

5.3 Instantiation

Employing instantiation as a target model design principle has the objective to derive a target model by embedding concrete process building blocks into places where the reference model provides a generic placeholder. The reference model does not have the objective to specify and define all aspects of its domain. Instead, it deliberately leaves their design to the target model designer and thus allows them to be determined by the target model context. The reference model itself aims to provide a general structure and to describe only generally applicable domain aspects.

The requirements to an instantiable reference model have implications for the mining process in terms of the mining technique and the choice of input models. Necessity, placement, and design of the generic placeholders have to be derived from the input models. The placeholders are either are included in the input models, in which case they have to be chosen accordingly, or that they have to be derived from the content the input models provide. Both cases require aggregation as the necessary reference model design principle. Since the input models are typically taken directly from an application context, they usually do not contain generic placeholders as required for a reference model. This is why the more complicated placeholder construction is also more likely to happen in practice.

Several aspects contribute to making placeholder construction so complicated. First of all, it is hard to automatically determine which domain aspects should be left for target context instantiation, as they depend on the reference model designer's objectives and preferences. Second, even if the content to be abstracted is previously specified, finding a generic placeholder for a set of process model elements requires abstracting them on a structural and semantic level. Structurally, the position of the placeholder may differ among the input models, making it unclear where to insert it in the reference model. Semantically, the placeholder (e. g., the process interface) needs a label that describes the abstracted content, which requires external knowledge in language processing. The only applicable mining technique for the instantiation principle is Rehse et al. (2013), which uses process model abstraction techniques in order to find commonalities among input models. However, even this approach requires a lot of manual pre- and post-processing and is far from a fully automated RMM approach.

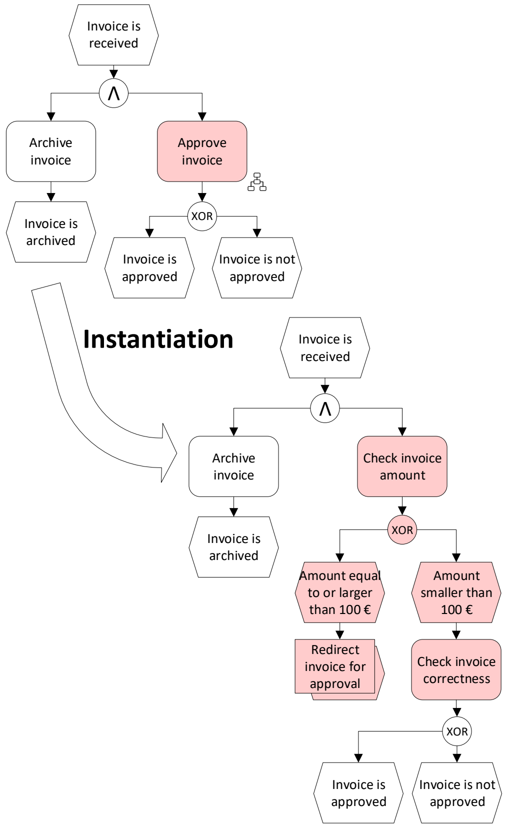

Figure 14: Using instantiation as a target model design principle

Fig. 14 shows an exemplary application of instantiation as a target model design principle. The reference model describes an auditing process, which contains a generic placeholder in terms of a refinement symbol. This means that the activity "Approve invoice" can be specified to describe the actual process in this company. When applying the reference model, the placeholder can be replaced by this detailed process in order to construct a more complete process model.

To conclude, *instantiation* is an applicable design principle, if

- the reference model is a general framework that can or should not specify domain aspects,
- the input models contents are abstractable, and
- the target model context allows for a concretization.

5.4 Specialization

Employing specialization as a target model design principle has the objective to derive a target model by revising the core solution that is provided by the reference model. The reference model unites all universally valid domain aspects with the right degree of granularity and allows target model designers to extend and modify, but not delete, them for their purpose. The entire content of the reference model should be included in the target model, but can be enriched as deemed necessary by the target model context.

The requirements to a specializable reference model have implications for the mining process in terms of the mining technique and the choice of input models. If the reference model should provide generic fragments as universally valid domain aspects, these fragments have to be included in the input models and be frequent enough to be considered universally valid. In addition, the reference model must be applicable in every target model context. It can therefore only consist of universally applicable model parts. This means that the reference model can only contain those model fragments that are present in all the provided input models. On the one side, this requires the input models to be chosen to have enough commonalities such that a meaningful reference model can be derived. On the other side, the employed mining technique has to be able to identify these commonalities and use them to construct a reference model.

When applying a mining technique, the specialization principle is the opposite of the configuration principle, in terms that configuration requires all specificities of the individual models to be included in the reference model. Specialization allows none of the individual model features to be part of the reference model. This means that mining a specializable reference model also requires the mining technique to have a frequency threshold for reference model parts. Here, as opposed to configuration, this threshold should be set as high as possible, such that only the most frequent model parts will be part of the reference model. The mining techniques by Ardalani et al. (2013), La Rosa et al. (2013), C. Li et al. (2011), and Rehse et al. (2016, 2017) fulfill this requirement. They construct a reference model by means of aggregation. It is also possible to mine a specializable reference model by means of analogy, as done by the techniques of Martens et al. (2014) and Yahya et al. (2012).

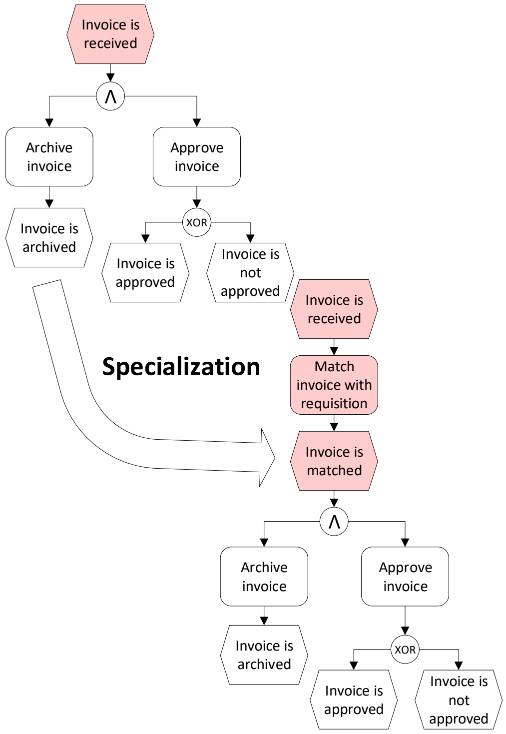

Figure 15: Using specialization as a target model design principle

Fig. 15 shows an exemplary application of specialization as a target model design principle. The reference model describes an auditing process, which consists of two activities that are generally necessary when receiving an invoice, i. e., archiving and approving it. In the target model, these universal steps are specified by a preceding step matching the invoice with the requisition order before further processing. This might be necessary for invoices received in a purchasing unit with connection to an ERP system.

To conclude, *specialization* is an applicable design principle, if

- the reference model should provide a core solution that covers all universal domain aspects,
- the input models share identical fragments describing universally valid domain aspects, and
- the target model context allows for a specific extension and modification.

5.5 Aggregation

Employing aggregation as a target model design principle has the objective to derive a target model by combining the contributions of several reference models each covering another aspect of the target model context. This means that the target model context cannot be covered by a single reference model, but must stem from multiple ones. This might be the case, if no specific reference model exists for a specific process in a certain industry. The target model is then constructed by adopting parts from each reference model and combining and replenishing them in order to design a new model.

Since designing a reference model by means of aggregation means to mine several reference models, there is no applicable mining technique. All of the mining techniques analyzed in the context of this paper have the goal of uniting input models to form a reference model instead of explicitly separating them by a certain criterion. However, this idea, which stems from process variant management, is worth to be considered in more depth. In order to mine a meaningful reference model, the input models should be fairly similar to each other, with large intersections depending on the design principle. In turn, if the provided set of input models is too large and too disjoint, it might not make sense to use them as input models for one monolithic reference model. Instead, the input models can be separated into smaller, but more similar input sets able to generate a meaningful reference model. Clustering techniques based on model characteritics can be used for that purpose. This idea is not completely new, as clustering input models has already been described for example by Fettke (2015), however, this approach still has the goal to integrate all input models into a single reference model. Deliberately separating input models in order to mine different reference models and recombining them to obtain target models by means of aggregation has so far not been considered and remains a topic for future research.

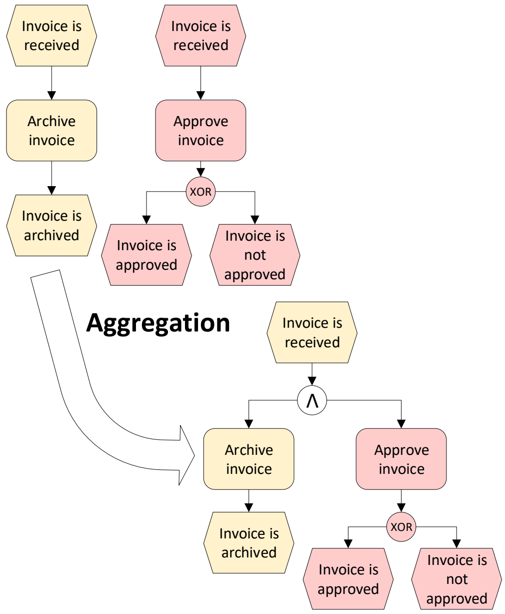

Figure 16: Using aggregation as a target model design principle

Fig. 16 shows an exemplary application of aggregation as a target model design principle. The reference models cover different aspects of the auditing process, i. e., invoice archiving and invoice approval. They are combined into a single model in order to form a target model fulfilling the requirements of its context, where invoices have to be both archived and approved in order to be handled properly.

To conclude, *aggregation* is an applicable design principle, if

- the target model context cannot be described by a single reference model and
- a separate reference model exists for all relevant target models aspects.

5.6 Analogy

Employing analogy as a target model design principle has the objective to derive a target model by transferring a process design from one application context to another. The reference model contains a number of patterns or building blocks that are applicable to its domain, however, they might have to be adapted or modified in order to fit into the target model context. Analogy is the application principle that gives designers the highest degree of freedom in modeling. It is allowed to change, modify, delete, or replenish the reference model according to the target model requirements. However, this also means that constructing a model by means of analogy cannot be described by a set of rules and thus, no guarantees can be given for the quality of the target model.

Because of this lack of clearly defined rules or requirements for the analogy principle, it is difficult to asses its influence on the choice of input models and mining techniques. It can be stated that the reference model should be directly applicable to the target model context in terms of content and degree of abstraction, which means that not a lot of changes should be made to the input models. This allows designing the reference model by means of either configuration, aggregation, or analogy. Deriving a reference model by configuration entails sufficiently substantiating it from a generic set of input models. This is done by both Gottschalk et al. (2008a) and Schunselaar et al. (2014). The techniques by Ardalani et al. (2013), Fettke (2015), C. Li et al. (2011), and Rehse et al. (2016, 2017) use their abstraction parameter to determine the reference model content. The analogy principle lies in between the restrictive specialization principle and the inclusive configuration principle, such that the frequency threshold should be somewhere in the middle in order to mine a reference model that can be used for an analogous design. Finally, the techniques by Martens et al. (2014) and Yahya et al. (2012) base their reference model design on selecting one model as the basis, which should yield a well applicable reference model.

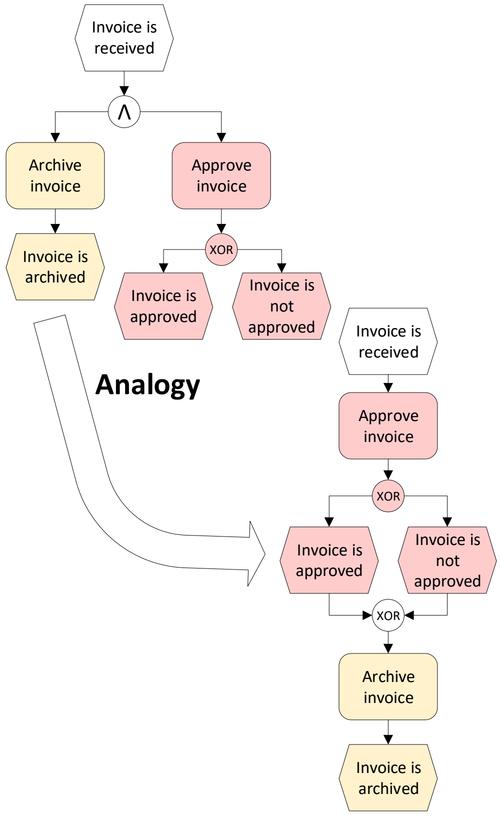

Figure 17: Using analogy as a target model design principle

Fig. 17 shows an exemplary application of analogy as a target model design principle. The reference model describes an auditing process that is directly applicable in the target model context, regarding its scope and level of abstraction. The target model adopts all parts of the reference model, but reorders them to be executed in sequence rather than in parallel.

To conclude, *analogy* is an applicable design principle, if

- the reference model should provide a concrete guideline or example for target model design,
- the input models comprise process variants, and
- the target model context allows for a direct application of the reference model.

6 Case Study

6.1 Objective

In order to illustrate the general feasibility of our procedure model, we use it to design two different reference models by means of S-RMM. For each, we set a situational context for which a reference model should be designed and then execute the ten steps of the procedure model. The case study intends to demonstrate that using the S-RMM procedure model leads to reference models that are more adapted to the context than those designed with RMM alone, while saving resources in the design process. To show that it is necessary to consciously select the right RMM technique for a certain design context, both case study scenarios are based on the same set of input models. The resulting reference models differ considerably from one another in terms of size, structure, and level of detail, meaning that they each fulfill their intended purpose, but could not be interchanged.

Our real-life input processes describe how German universities administer third-party funded projects (Gröger et al. 2014; Gröger and Schumann 2014). Concretely, we selected three process models describing how three different universities retrieve already-granted funds. They are suited for our case study for three reasons. One, they each depict the same process, implemented in different organizational environments, but targeted towards the same goal. Two, they are real-world process models, so the case study supports the practical applicability of S-RMM. Three, the models are publicly available, so our study can be reproduced.

The first property is quite common in public administration processes, making them an ideal domain for inductive reference model development (Scholta 2016). Typically, different public authorities are responsible for performing the same process in different areas (e. g., counties or states). Since these processes by design reach the same goals under similar circumstances, but potentially in a different way, reference model designers can compare them and combine their similarities into a reference model. Another organization can then use this reference model to implement the process, benefiting from the experiences that were already made.

Given the context of university administration, Tab. 3 exemplifies a potential target model context for each design principle. This explains the rationale behind S-RMM and illustrates its widespread applicability. Once the target model design principle is determined, the reference model design principle is chosen based on the analysis from Tab. 2, existing RMM techniques, and the available input data.

Our case study is set out to develop two reference models in different scenarios:

1. A fourth university wants to establish a new department for handling third-party funded projects and therefore needs to (re-)engineer its internal processes. Because these administrative processes should be efficient in supporting the research departments and communicating with the funding agencies, process designers can draw from the three universities' experiences. If a singular model is used as a blueprint for the design, it is by definition a reference model, but it will be specific to the organizational context of one university. Process designers should take all three models into account and adopt the commonalities, which they consider to be industry standards. Especially in large models, these similarities are difficult to identify and integrate manually. Instead, process designers can use RMM to derive a reference model that integrates these similarities and assures a certain quality. This model can be adapted to

Table 3: Target Model Design Principles and Possible Application Contexts for the Case Study Models

Design Principle	Context Description
Configuration	Applicable, if the target model context is unknown at reference model design time, e. g., when developing a configurable piece of standard software to support the third-party funding processes, which is to be sold to as many universities as possible.
Instantiation	Applicable, if the target model context (or parts thereof) is irrelevant to the reference model design, e. g., when describing a general practice for universities on how to handle third-party funding, while deliberately leaving the concrete implementation open.
Specialization	Applicable, if the target model context requires very specific and unique adaptations, e. g., when a university receives sponsoring from an agency that has different reporting requirements.
Aggregation	Applicable, if the target model context comprises several reference model contexts, e. g., when a university intends to implement a process that handles third-party funded and industry-sponsored research projects.
Analogy	Applicable, if the target model context is comparable to the reference model context, e. g., when a university opens a new research administration department where the existing processes have to be implemented in the same way as they already exist at the first department.

fit the individual needs of his university. To find the RMM technique that is most suited for this purpose, they use S-RMM. This scenario is described in Sect. 6.3.

2. A department of education wants to publish a general guideline for its universities on how to handle third-party funding. The objective is to provide them with administrative processes that are accepted in the industry and guarantee a certain quality. Universities with little experience in the field can use them to establish a new department for handling these projects without investing too much administrative effort. Such a guideline should be a reference model that describes a generic process, but leaves enough room for universities to adapt it to their own organizational context. The guideline designer can use RMM to inspect existing processes from universities that successfully carried out third-party funding projects in the past and subsume only their commonalities. S-RMM can be used to identify the right RMM technique. This scenario is described in Sect. 6.4.

6.2 Input Data

The three models depicting third-party fund retrieval at three different universities are shown in Fig. 18. They were originally modeled in BPMN notation and converted into EPCs using the following rules (adapted from Tscheschner 2006).

- A BPMN event is converted into an EPC event.
- A BPMN activity is converted into an EPC function.
- An exclusive gateway is converted into an XOR-connector. The label of such a gateway is depicted as a function preceding the connector. The labels of the outgoing edges are converted into events that succeed the connector.
- An inclusive gateway is converted into an OR-connector.
- A parallel gateway is converted into an AND-connector.
- If a BPMN activity joins multiple control flows, an XOR-connector is inserted preceding the according EPC function.
- A BPMN temporal event is converted into a function, if it represents an action that requires a certain time period (e. g., “wait until”). If it represents a specific point in time, it is converted into an event.

- A BPMN message event without a label is converted into a function representing sending or receiving a message.

Swimlanes, data objects and additional text annotations are not depicted in the EPC. The models already contained harmonized labels, which were manually translated from German into English. The harmonized labels represent equal granularity levels for all models. As we can observe, their activity scope is similar, but they differ in terms of order and message flow, i. e., the sender and recipient of notifications. This is because in each university, the shown process is executed by a different organizational unit. At the first university, shown on the left, the process is initiated by the funds-receiving research department and then sent to the financial department, which handles communication with the sponsor. At the second university, shown in the middle, the process is completely handled by the financial department. At the third university, the receiving research department again initiates the process, but then sends it to the research department, which then handles the booking and communicates with the sponsor and the financial department.

6.3 Process Improvement Scenario

6.3.1 Determine Situational Context

Our individual models describe the same process (retrieving funds from an already granted third-party funded research project) in three different organizations of the same field, i. e., universities. The situational context for this case study is that an organization of the same type, i. e., a fourth university, intends to re-engineer its own third-party funding processes with the help of S-RMM. They intend to build on previous experiences and incorporate them in order to implement efficient and effective administration processes. Public administrations, such as universities, differ from companies, because they do not need to protect their processes as a business secret. Instead, the process models can be shared for the benefit of others.

We can conclude from the analysis of the models that the difference is not in the process, but in its organizational context. The executed process steps are more or less the same, but it is important, which organizational units have to be involved, who initiates the process, and who is responsible for each part. Besides the organizational structure of the university, this is also determined by the administrative and legal conditions at the respective university.

6.3.2 Determine Target Model Design Principle

As we have determined above, the three input models describe a differing organizational process context, which impacts the process itself. The choice of the target model design principle thus depends on the organizational context of the fourth university, intending to use these models for its own purposes. In general, there are several choices for this principle. If, as in this case, the target model context is relevant to the reference model design, analogy, aggregation, or configuration are applicable principles. Instantiation and specialization require the reference model to contain no context-specific elements, they will be addressed in the second case study below.

If the organizational context of the target model university is known, then analogy will be the appropriate principle. Designers could choose the input model with the most similar organizational context and use that as a blueprint for their own process model. Since the context is similar, the model will mostly be directly applicable, with only a small number of changes. However, since in this case study, we know nothing about the fourth university, analogy is also not applicable without making further assumptions.

Applying aggregation as target model design principle requires several reference models with different context aspects, which are included in the target model. In our context, this would be the case if the university intended to implement a joint process for retrieving funds from both third-party funded and industry-sponsored research projects and there existed separate reference models for

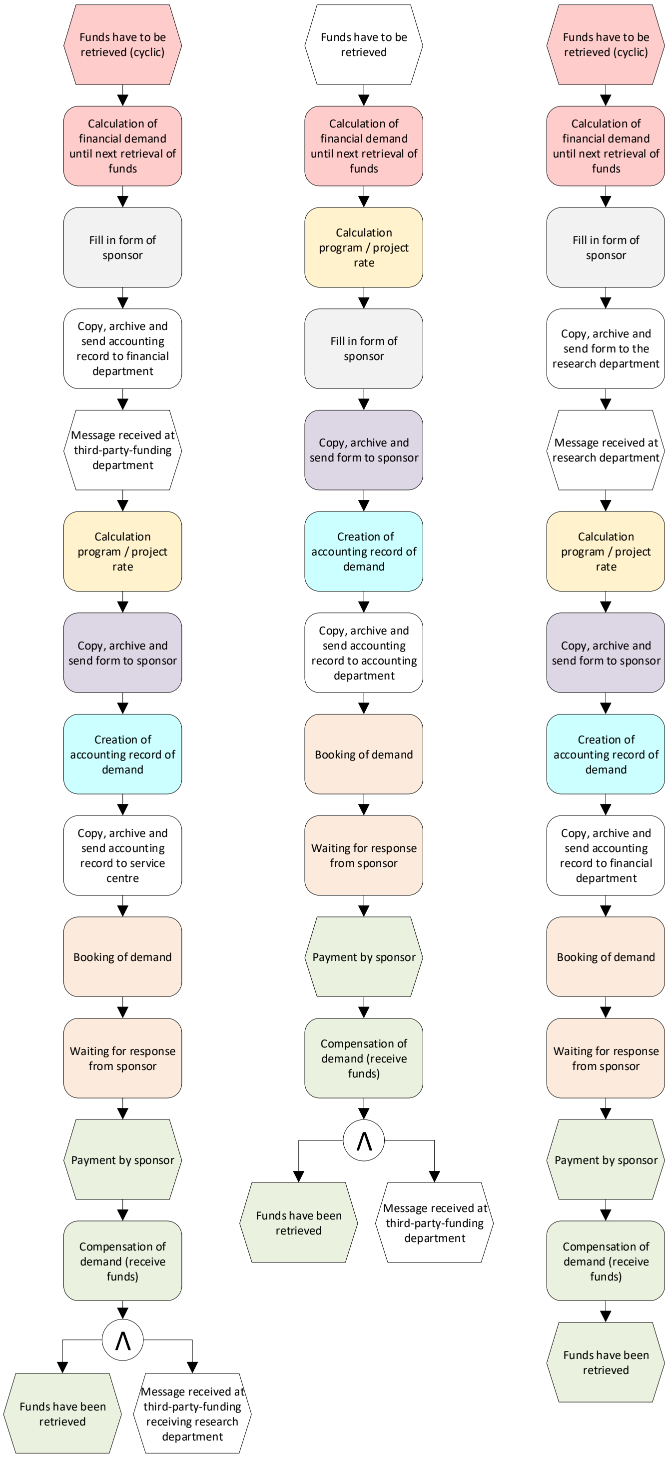

Figure 18: Three models depicting the process for retrieving third-party-funds at three different universities

each. Since our input models focus on one aspect only, aggregation is not applicable.

Without assumptions on the target model context, as many aspects as possible should be included into the reference model. This makes it applicable to as many contexts as possible. This is why configuration is the most appropriate design principle in this case. It allows the inclusion of all three organizational contexts in the reference model, which can then be adapted accordingly. This also allows for the target context to deviate from the input models, as the model parts can be configured independently.

6.3.3 Determine Reference Model Requirements

Once we have chosen configuration as the target model design principle, we are able to list the reference model requirements. As stated above, these are mainly determined by the principle itself instead of the specific context. In this case, we intend to design an inclusive reference model, i. e., a model that includes rather than abstracts all the specificities of the individual models, but streamlines them by stressing the commonalities among them. This means that the scope of the reference model corresponds to the union of the individual models' scopes. Parts of the reference model may cover similar aspects, but from different perspectives and thus be irrelevant for a certain application context. Thus, the model has to be configurative, in a sense that it describes all the possible process executions as seen in the input models in a mostly general way.

6.3.4 Determine Reference Model Design Principle

As we have seen in Tab. 2, the aggregation principle is most often combined with configuration. This application is no exception. Aggregation entails adopting and composing parts of several individual models, such that the resulting model subsumes the input models. Since we intend to design a reference model subsuming all the individual organizational context, we intend to consider each input model equally, hence making aggregation an evident choice.

6.3.5 Choose Mining Technique

For the combination of configuration as target model design principle and aggregation as reference model design principle, several mining techniques are generally applicable. However, all of these techniques may also apply to specialization and analogy as target model design principles. The executed principles depends on the choice of parameters and the weighing of the individual models.

All these techniques have an abstraction threshold parameter that determines the relative frequency of a model element in order to be adopted into the reference model. To mine an inclusive reference model, this abstraction threshold should be set as low as possible, such that all individual model features are included in the reference model. Some of the techniques allow to assign a different weight to the input models, making some input models more important to the reference model design than others. In our case study, all models should be considered equally, so no weight assignment is required.

Since the input models are similar in terms of activity, but differ considerably regarding the activity structure, choosing a mining technique that relies on a structural similarity measure (such as Ardalani et al. (2013), C. Li et al. (2011), and Rehse et al. (2017)) is not recommended. Other approaches are not applicable, as Gottschalk et al. (2008a) use log data as input and Fettke (2015) requires a large set of input models to be clustered into distinctive groups. This leaves process model merging (La Rosa et al. 2013) and the RMM-2 approach (Rehse et al. 2016) as applicable approaches.

To determine which technique better suits our application, we compare the two techniques. The first technique is not defined on a set, but a pair of models and requires users to determine the order in which the input models are merged. While this is generally not an issue, different merging orders may lead to different results, thus contradicting our objective to consider all input models equally. Hence, we chose the second technique, namely the RMM-2 approach as the appropriate mining technique for our case study.

6.3.6 Choose Input Models

The RMM-2 approach is based on a semantic similarity measure instead of a structural one. Therefore, it poses less assumptions on its input models, but it requires a uniform level of granularity, the absence of duplicate nodes and it is unable to handle OR-connectors. It also requires a matching between the input models as additional input in order to determine analogies between the nodes (Rehse et al. 2016). Since all our individual models are free of duplicate nodes and OR-connectors, they could all be used as input models for the mining process. A mapping was defined such that activities and events with equal labels were matched onto one another. This also indicated a uniform level of granularity.

6.3.7 Mine Reference Model

In order to perform the mining, a parameter configuration that represents the chosen design principle has to be found. The RMM-2 approach has only two parameters: an abstraction threshold to determine the minimum frequency of an activity and a noise threshold to determine the minimum frequency of an edge. The latter is meant to avoid unstructured, spaghetti-like models and has to be determined individually for each setting.

As the reference model subsumes parts from all three input models, the abstraction threshold is set below $\frac{1}{3}$ (0.3). The noise threshold was determined experimentally. We tested multiple values and selected a value (0.4) that yielded a structured and fully connected model. The mining process is conducted using an implementation of the approach in our research prototype RefMod-Miner[1] .

6.3.8 Adapt Reference Model

After the mining process is completed, the resulting reference model has to be adapted to serve its intended purpose. In our case, this entailed some remodeling of incorrectly designed model

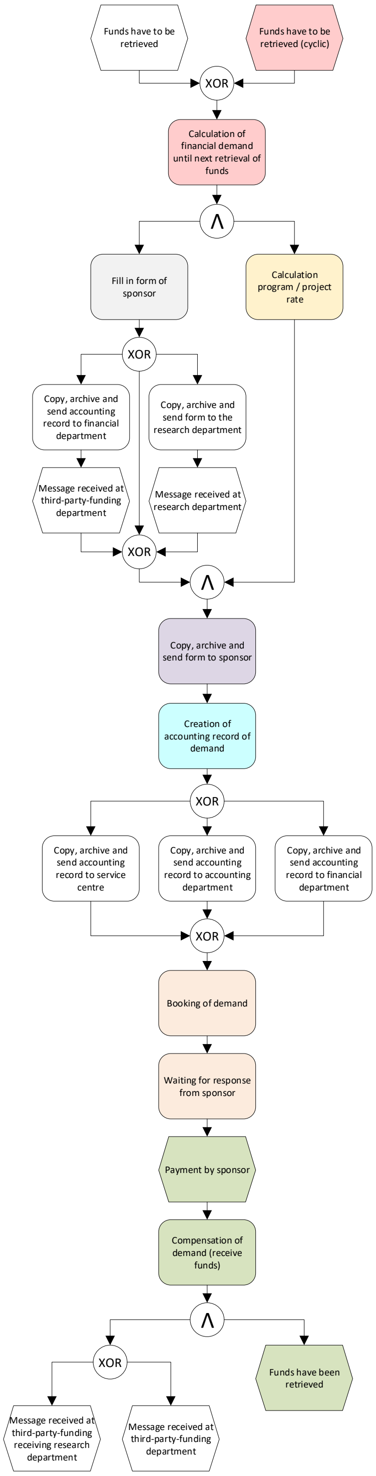

Figure 19: The reference model mined for the process improvement scenario

---

[1] http://refmod-miner.dfki.de

parts, such as merging duplicate connectors. Also, missing connectors were inserted and corresponding edges were adapted accordingly. These errors are caused by the mining technique, which is not focused on syntactical model correctness. In addition, the reference model was newly layouted, as shown in Fig. 19.

6.3.9 Design Target Models

We can see that the finalized reference model intends to generalize the different execution order from the input models, to introduce a large degree of freedom for configuration purposes. The project rate can be calculated either after or before filling out and sending the forms, depending on which organizational unit is responsible for this step. Notifications can be sent to all potentially involved parties. Note that the XOR-connectors in this model should be understood as configurable connectors. Within a certain organizational context, the process will always take one of the potential options, while the other is only applicable in a different organizational context.

Hence, when using this reference model to design a new process for retrieving third-party funds, the target model should be configured at design time. This means analyzing the given organizational context and adopting, deleting, and arranging the individual model parts, such that the process is applicable in the given context.

6.3.10 Evaluate Target Models

In this case study, our main goal was to demonstrate the general usability of the newly designed procedure model for S-RMM, as shown in Sect. 4. We wanted to apply the model in a realistic case study setting, such that we could explicate and discuss the assumptions and conclusions that follow from a certain situational context. In our case, this context was the organizational context of universities, which causes differences within the process design. The resulting configurable reference model allows for the configuration of the target model according to the organizational context at another university. The model is bounded by the context information provided by the individual models. If the organizational target context differs significantly, for example by introducing a new and unknown organizational unit, the reference model will provide less support for the target model design.

6.4 Guideline Process Scenario

6.4.1 Determine Situational Context

Our second scenario relies on the same input models, but assumes a different reference modeling context in order to illustrate the main idea behind S-RMM. We will see that depending on the chosen design principles and mining techniques, we yield a completely different reference model that can be used for a different purpose. In this case, we assume that there is a department of education aiming to facilitate third-party funded research projects for its universities. In order to do that, it analyzes existing administrative processes to give the universities an idea of what such a process entails and how much effort it might take to implement it. The department is not interested in the specific implementation of a process, instead S-RMM should be used to derive a generic process model suitable for getting a rough idea of the underlying process.

We can conclude that, in this context, the individual input model contexts are not of any relevance and should hence not be included in the reference model. It is necessary to abstract as much as possible from the concrete organizational context represented in the input models and instead focus on the commonalities between the processes.

6.4.2 Determine Target Model Design Principle

As in the first scenario, the concrete target model context is unknown. However, in this case, there might not even be a target model context and, if there is, the reference model is not intended to cover it to the full extent. In fact, the reference model design is set out not to cover any specific implementational or organizational details, but to report only on the generic nature of the process.

If the organizational context is not relevant for the target model design, instantiation will be an appropriate principle, as it allows a complete abstraction from the organizational units. Whenever designing the target process, the selected placeholder can be replaced with the specific organizational context of the given university. The input models here contain some context-specific steps (such as receiving and sending notifications), but also some context-independent steps (such as booking a demand). However, instantiation as a target model design principle requires the reference model to be on a higher abstraction level, such that the target model designers can plug-in a company-specific context. In our case study, the reference model should be rather specific. This allows implementing universities to plan its concrete execution. In consequence, instantiation is not an ideal principle.

When the reference model is supposed to contain universally valid domain aspects, specialization is the most applicable principle. Process steps that appear in all (or the majority of) models should be adopted into the reference model, which can then be enhanced with further steps required for the process to fit into the specific organizational context. The reference model provides process model building blocks. These building blocks are the foundation for the target model design.

6.4.3 Determine Reference Model Requirements

When choosing specialization as the target model design principle, the reference model should only contain universally valid domain aspects, i. e., those process steps and other elements that are required in the process independent from the target model context. Opposed to the scenario above, this scenario requires an exclusive scenario where only those steps that are included in every input model should be included in the reference model. The reference model can thus be considered as the intersection of the input models. Of course, we cannot determine from the input models alone that these steps are really necessary in all contexts or that these are the only necessary steps to take. These questions have to be answered when adapting the mined reference model. However, inspecting the provided input models gives a first estimate of potential reference model content.

6.4.4 Determine Reference Model Design Principle

We can see from Tab. 2 that there are two possible reference model design principles for the target model design principle specialization. As explained above, aggregation is suitable if we intend to combine aspects from different input models, e. g., if the necessary steps (in the opinion of the reference model designer) appear in different input models. If we intend to design the reference model following the general design of one or several input models, analogy is the applicable principle. In our case, all the input models contain the necessary steps, so there is no need for aggregation. In consequence, the analogy principle is chosen.

6.4.5 Choose Mining Technique and Input Models

The difference between analogy and aggregation as reference model design principles can be illustrated by the construction process. Aggregation techniques usually analyze the common structures in the input models and integrate them all into a newly constructed process model. Analogy techniques gradually adapt an existing model, such that it becomes more similar to all other input models. The latter are typically heuristic techniques, adapting the models by means of change operations.

Referring to Tab. 2, the mining techniques presented by C. Li et al. (2011), Martens et al. (2014), and Yahya et al. (2012) are applicable to the principle combination specialization / analogy. The first contribution contains two different mining techniques, a heuristic technique based on an existing reference model and a clustering technique to merge existing process variants. The heuristic technique employs analogy as design principle, the clustering technique aggregation. In this section, we refer solely to the heuristic approach, which is based on an approximation of the graph-edit distance as a similarity measure. The other two contributions (Martens et al. 2014; Yahya et al. 2012) both present mining techniques based on genetic algorithms.

In general, all three techniques would be applicable to our use case. However, the technique by Yahya et al. (2012) is rather limited, as it does not consider process model semantics (i. e., different types of connectors). Of the two remaining techniques, we chose the heuristic mining technique by C. Li et al. (2011) due to its supposedly lower runtime and better accountability.

The chosen mining technique requires input models to be block-structured and does not allow them to contain OR-connectors. All input models fulfill these criteria, so they are all used as input models. To initiate the process, one of the models is randomly selected as the initial reference model.

6.4.6 Mine and Adapt Reference Model

Besides the input models, the heuristic mining technique requires some optional parameters, i. e., the weighting of the input models and a maximum number of iteration steps. Since there is no input model that we consider especially relevant to the reference model design, we weight them all identically. The models are small enough that the computationally complex technique should be able to handle them. The number of iterations is therefore not restricted.

The mining process was again conducted using an implementation of the approach in our research prototype RefMod-Miner. We performed several iterations of the mining process, each time picking another input model as the initial model. Finally, the reference model computed based on the second input model (shown in the middle of Fig. 18) was the best fit for the described context, as it contains all universal process steps in the right order, but does not contain any context-specific messaging function or corresponding event. Since the mined result was already syntactically correct and semantically expedient, no further adaptations were required. The final reference model is shown in Fig. 20.

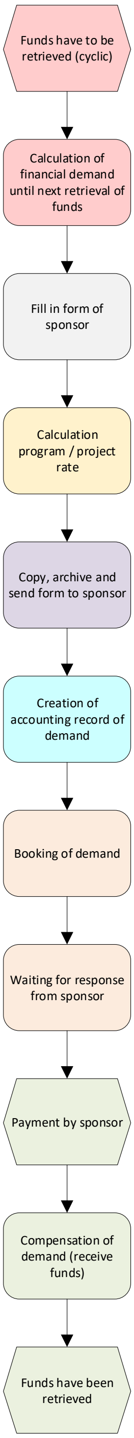

Figure 20: The reference model mined for the guideline process scenario

6.4.7 Design and Evaluate Target Models

The primary purpose of the reference model in this scenario is not the design of specific target models, but a generic description of existing processes. Since the model contains all those process steps that are present in all input models and connects them into a reasonable process model, it fulfills this purpose and can serve as the basis for an in-depth process description. It can also be used as a starting point for potential target models. In this case, it is important for reference model designers to note which aspects from the input models are not included in the reference model, but should be considered to implement a meaningful process. The reference model documentation should therefore point towards the additional information required for target model design. In our case study, this would include the department that initiates and executes the process, the departments that have to be included or notified, and the department that is in charge of communicating with external process participants (e. g., the sponsor). Target model designers should include these factors into the process by means of specialization.

7 Discussion

The purpose of the case study in Sect. 6 was to demonstrate the utility and general feasibility of the concept of S-RMM. By deliberately selecting two very different, but generally applicable RMM use cases, we show that successfully applying RMM requires more than efficient algorithms. However, even though our procedure model provides reference model designers with a basic guideline, additional aspects must be considered before practitioners will be able to leverage the full potential of RMM.

The scenarios that we consider in our case study are rather classical use cases for reference modeling. For both process (re-)engineering and guideline documentation, the multitude of reference modeling benefits (including increased process quality, industry alignment, standardized terminology) depend on the availability of a pertinent reference model. If no such model exists, many organizations are unwilling to invest the time, cost, and personnel resources that are necessary to design a new one. This is understandable from both an economical and a business perspective, as such a design project is not always a worthwhile cause. If the reference model has no prospect of reuse either within the own organization or in others (e. g., in case of an industry standard), the required efforts cannot be economically justified. The same is true if the concomitant benefits mentioned above are not required or even counterproductive to the target model design process. This might be the case if a process is to become a competitive advantage and therefore should be designed in a new and unique way.

This problem is inherent to reference modeling and cannot be completely solved by S-RMM. Designing a reference model will always require a substantial amount of work from its stakeholders, such that it becomes a valuable asset. Automation is set out to address this issue, but current approaches are not mature enough to be used on their own. In this context, we position S-RMM as a new paradigm to reference modeling. If (a) the target model design could benefit from using a reference model, (b) the reference model development is economically justified, and (c) the company is willing to design a new reference model, but unsure whether it can muster the required resources, S-RMM as a hybrid automated-manual method might be the right solution.

One could also question why we use the term “situation” to describe the concept of S-RMM. In a way, a situationally designed model is the antipode of a reference model, given that the latter implies universality, which contradicts situational and therefore individual adaptation. However, we have chosen this ostensible oxymoron on purpose, because a reference model does not apply equally well in every context. Instead, it makes sense to situationally design multiple reference models for the same domain, if they serve different purposes. This is also why a reference model can be designed situationally, even if the designers don’t yet know the details of the target model design. They can include those application details that are already known and use design principles such as instantiation to give users the necessary freedom of decision, as we show in the second case study scenario.

In this regard, we should also examine how S-RMM relates to SME, given the analogous terminology. SME makes the case for situationally adapting a (software development) method to the circumstances of a project or an organization. The artifact in question is the to-be-designed method, which can be directly influenced. In S-RMM, the main focus is not the method, but the reference model, i. e., the model's result, that is to be situationally adapted. For this purpose, the best mining method is selected and executed. This means that the designer's influence on the situationally adapted reference model is more indirect than on the situationally adapted software development method and that S-RMM comprises the situational design of not one, but two artifacts. Despite these differences, we have deliberately chosen this terminology, as the underlying idea of S-RMM is the same as in SME: An artifact is much more valuable, when it's consciously adapted to the circumstances in which it is used.

Although they stem from a process documentation project at real-world universities, we chose the set of input models mainly for demonstrative purposes. Given that they represent the same process at different organizations, they adequately demonstrate the influence different of the organizational context on the process design. However, since we did not have access to additional documentations or process execution data, we could only use the process models to draw conclusions. Even if such data would have been available, the context assessment would still require a lot of manual work, since none of the existing mining techniques are capable of handling such additional information. For a demonstrative case study, we also had to make a number of assumptions regarding the target model context, with a few aspects remaining open. While both our scenarios are reasonable, yet slightly artificial, use case for reference modeling, there exists a plethora of other use cases, where the feasibility of S-RMM remains to be elaborated.

As we have seen in the first case study scenario, parametrization may or may not be a decisive factor in reference model construction. The influence of parameters on the reference model contents and design has to be determined individually for each technique. For example, the order in process model merging should not influence the resulting model, while a frequency threshold (as for example in Ardalani et al. 2013; Rehse et al. 2016) determines whether a reference model is the intersection or the union of the input models. In this case, the parameter value also determines the design principle. Depending on the relative frequency of the reference model elements, configuration, analogy, or specialization are applicable. This is why multiple mining techniques apply to several combinations of target and reference model design principle, as seen in Tab. 2.

While the choice of design principle provides a first guideline towards a better-fitting reference model, it is not sufficient to choose the right mining technique for a use case. In the case study scenarios, after selecting both the target model and the reference model design principles and deducing appropriate requirements, several mining techniques were applicable. Their assets and drawbacks, and thus the final choice of technique, could only be determined after either analyzing the technique in more detail (as in scenario 2) or applying it to the input models (as in scenario 1). This means that the technique might not fulfill all requirements from the previous step. As shown in Tab. 2, the finally chosen approaches are only one of several applicable techniques for the combination of aggregation and configuration, respective analogy and specialization. Applying another technique might not yield a reference model as the union or intersection of input models, but it might still be a meaningful reference model in a number of different use cases.

The case study also demonstrates a problem in RMM that is imminent to inductive techniques. The final result depends on the content of the input data. This has two implications. First, the reference model cannot contain any elements that were not present in any of the input models. Second, the mined reference model might be misleading in terms of its generality, as a mining technique draws conclusions from a small and finite set of input models to the whole process domain (problem of induction). Both problems cannot be fixed algorithmically. Addressing them is the reference model designer's responsibility. If the model needs to be changed to serve its purpose, they either have to adapt the set of input models or the model itself. Since the available input models are usually limited, this involves more manual effort, which should have been avoided by using an automated approach in the first place.

Choosing aggregation and configuration as the design principles in the first scenario was a conscious decision. Our analysis of existing mining techniques in Sect. 5 shows that the aggregation principle is predominant in reference model construction, while configuration is the mainly followed principle in reference model application. Due to the nature of RMM, this is not surprising. When deriving a reference model with a certain degree of universal applicability from a set of input models, aggregating their common features is an obvious approach, but it is not the only one that achieves a meaningful model. Constructing the reference model by adapting one input model to reduce the overall difference to the other input models realizes the analogy principle, as we have demonstrated in the second case study scenario using the technique by C. Li et al. (2011).

On the other hand, a reference model that aggregates aspects from different sub-domains has to be configured in order to obtain a context-specific target model. A reference model that contains the most common fragments requires specialization or instantiation as appropriate target model design principles. To conclude, although aggregation/configuration is prominent, this principle combination is not automatically applicable, but depends on the characteristics of the mining technique. This is also why we chose two different principles in the second scenario, demonstrating the necessity for all the principles.

The analysis in Sect. 5 also shows that the instantiation principle is underrepresented in RMM. This is supported by the fact that it is the only principle not considered in the case study. Most existing RMM approaches are not capable of handling input models with varying degrees of abstraction, but instead require the same level of detail across all input models. This level is then reproduced in the reference model. The generic placeholder elements, necessary for deriving target models by means of instantiation, cannot be derived this way. They would require approaches capable of abstracting input models with varying degrees of specificity.

Our analysis also reveals that current approaches are not able to construct the reference model by means of aggregation. One reason might be that aggregation draws on several conceptual models covering different aspects of the situational context to be composed in the target model. None of the existing mining techniques is explicitly aimed to mine several different reference models covering different aspects of the defined domain. However, such a scenario is realistic, for example when the reference model is supposed to cover a large application domain, but only input models from smaller subdomains are available.

8 Conclusion

In this contribution, we introduce the concept of Situational Reference Model Mining. The reference modeler's dilemma states that it is necessary to balance a reference model's generality against its value for each customer. Manual reference modeling methods give their users full control over how they address the dilemma for their individual use case, but are resource-intensive and error-prone. Automated methods are resource-efficient, but situation-agnostic. S-RMM provides a middle ground, investing some resources to achieve a better-fitting and therefore more valuable reference model. We intend to give organizations better access to high-quality reference models by

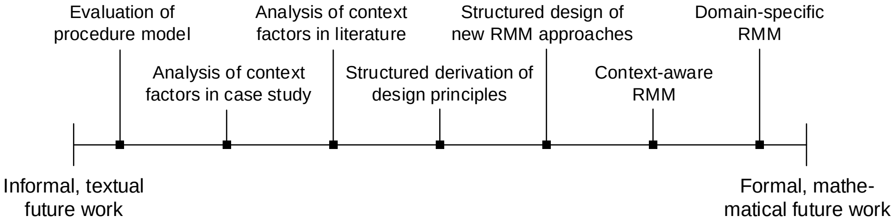

Figure 21: Future research questions positioned on the spectrum of BE research methods

providing them with a method to use automated reference modeling to their advantage. As this idea has not yet been elaborated in the respective literature, our procedure model is intended as a first recommendation for a concrete approach. This way, we intend to increase the practical applicability of RMM and make its benefits available to a wider range of users.

As evidenced by this text-heavy contribution, the idea behind S-RMM is a rather informal one, relying on textual descriptions and conceptual models. This is mainly because many relevant process context aspects (such as company strategy or employee relations) cannot be easily formalized, much less expressed in a form suitable for automation. On the spectrum of BE research methods, our contribution is located further on the left side. However, as shown by the related work and the analysis of existing mining techniques, S-RMM also encompasses more formal aspects. Hence, the open research questions are spread across the whole spectrum, as summarized in Fig. 21. In the following, we shortly outline each future research question, in increasing order of formalization.

How can the procedure model be validated? The procedure model, in combination with the analysis of existing techniques, is supposed to be a guideline for both reference modeling researchers and practitioners. However, it has not yet been evaluated by being applied in a large-scale context. Elaborating it in more detail, evaluating it by means of design science, and gaining more experience in practical applications of existing RMM techniques remains one of the major objectives of further reference modeling research. Our underlying assumptions should also be critically assessed. For example, in some cases it could make sense to develop situationally adequate target models instead of choosing an appropriate mining technique.

Which context factors are really relevant for RMM? Since we have not yet applied our findings in a more realistic case study with practitioners, we cannot finally say which contextual factors actually have influence on RMM and whether they can be appropriately represented by the framework we suggest. As we explain above, the design principles provide some guidance, but are too broadly phrased in order to provide concrete instructions. This is by design, because a strict guideline would be limited to specific use cases and thus too narrow for a universal application. An analysis of deductive approaches to reference modeling as well as existing RMM applications and case studies (such as Aier et al. 2011; Gröger and Schumann 2014; Karow et al. 2008) should be a good starting point to analyze relevant context factors. The goal should be to express these factors in a unified conceptual model and assess each for opportunities of formalization and data collection.

How can new RMM techniques fill the identified research gaps? Our analysis of existing mining techniques in Tab. 2 also acts as a gap analysis, identifying further research potentials and objectives and allowing for a more structural design of new mining techniques. The main motivation for this contribution is to increase the practical applicability of RMM. Currently, there exists a number of publications that focus on technical and methodical aspects, as well as a few implementations, but few concrete suggestions for their application. By coining the term “Situational Reference Model Mining”, we emphasize that the choice of technique is relevant, i. e., they cannot always be used interchangeably. The gaps outlined in the discussion should thus be successively filled by new techniques, allowing for a better applicability of S-RMM.

Which additional design principles are beneficial and how can they be derived? In this contribution we draw on the five principles configuration, instantiation, specialization, aggregation and analogy, as defined in vom Brocke (2007). However, these are not the only principles to be considered for reference model design. In future work, it would be beneficial to develop a conceptual framework for reference model design and use this to derive new design principles in a structured way. As examples, we suggest to consider the following principles:

- Modification, as suggested by Delfmann (2006) allows all changes to the reference model that do not result in erroneous or inconsistent models.
- Elimination allows designers to delete unnecessary elements from a reference model.
- Union allows to merge several models, without aggregating their contents.

How to design context-aware and domain-specific RMM techniques? The concrete assessment of relevant context factors is also impeded by the fact that there are no techniques for context-aware RMM. As our analysis reveals, none of the existing approaches is able to take context factors into account. The only additional input is the mapping between the input models, which may provide some context. One explanation might be that it is very difficult, if not impossible, to consider context factors before knowing the concrete reference modeling use case. This is why generic RMM techniques, like process mining techniques, either only factor in context already present in the models (such as similarities between input models) or rely on the user to (implicitly) provide it, for example by selecting the right input models. Also considering additional process data, such as process documentations, organizational models, or execution logs, could generally be helpful in designing a more powerful mining technique. Whether or not there will be developments towards context-dependent mining techniques, remains to be seen. Developing domain-specific RMM techniques that are restricted to a certain industry (e. g., healthcare or public administration) might be interesting, as such approaches are capable of intrinsically considering relevant context factors. A priori knowledge of relevant context data could be the final step towards a strictly formal and thus fully automated RMM technique.

References

Acher M., Collet P., Lahire P., France R. (2009) Composing feature models. In: van den Brand M., Gašević D., Gray J. (eds.) Software Language Engineering. Springer, pp. 62–81

Acher M., Collet P., Lahire P., France R. (2010) Managing Variability in Workflow with Feature Model Composition Operators. In: Baudry B., Wohlstadter E. (eds.) Software Composition. Springer, pp. 17–33

Aier S., Fichter M., Fischer C. (2011) Referenzprozesse empirisch bestimmen – Von Vielfalt zu Standards. In: Wirtschaftsinformatik & Management 3(3), pp. 14–22

Akkiraju R., Ivan A. (2010) Discovering Business Process Similarities: An Empirical Study with SAP Best Practice Business Processes. In: Maglio P., Weske M., Yang J., Fantinato M. (eds.) Service-Oriented Computing. Springer, pp. 515–526

Antunes G., Bakhshandelh M., Borbinha J., Cardoso J., Dadashnia S., et al. (2015) The Process Model Matching Contest 2015. In: Kolb J., Leopold H., Mendling J. (eds.) Proceedings of the 6th International Workshop on Enterprise Modelling and Information Systems Architectures. Köllen, pp. 127–155

Ardalani P., Houy C., Fettke P., Loos P. (2013) Towards a Minimal Cost of Change Approach for Inductive Reference Model Development. In: Proceedings of the 21st European Conference on Information Systems Vol. Research Papers, 127. AIS

Becker J., Delfmann P., Knackstedt R., Kuropka D. (2002) Konfigurative Referenzmodellierung. In: Becker J., Knackstedt R. (eds.) Wissensmanagement mit Referenzmodellen: Konzepte für die Anwendungssystem- und Organisationsgestaltung. Springer, pp. 25–144

Becker M., Laue R. (2012) A comparative survey of business process similarity measures. In: Computers in Industry 63(2), pp. 148–167

Becker T., Intoyoad W. (2017) Context Aware Process Mining in Logistics. In: Proceedings of the 50th CIRP Conference on Manufacturing Systems. Elsevier, pp. 557–562

Born M., Kirchner J., Müller J. (2009) Context-driven business process modeling. In: International Workshop on Managing Data with Mobile Devices, pp. 6–10

Bose R., van der Aalst W. (2009) Context Aware Trace Clustering: Towards Improving Process Mining Results. In: International Conference on Data Mining. SIAM, pp. 401–412

Cayoglu U., Dijkman R., Dumas M., Fettke P., Garcia-Banuelos L., et al. (2013) The Process Model Matching Contest 2013. In: Lohmann N., Song M., Wohed P. (eds.) Business Process Management Workshops. Springer, pp. 442–463

Chihani B., Bertin E., Suprapto I. S., Zimmermann J., Crespi N. (2012) Enhancing existing communication services with context awareness. In: Journal of Computer Networks and Communications 2012 Article ID 493261

Czarnecki K., Antkiewicz M. (2005) Mapping Features to Models: A Template Approach Based on Superimposed Variants. In: Glück R., Lowry M. (eds.) Generative Programming and Component Engineering. Springer, pp. 422–437

Czarnecki K., Helsen S., Eisenecker U. (2005) Formalizing cardinality-based feature models and their specialization. In: Software process: Improvement and practice 10(1), pp. 7–29

Delfmann P. (2006) Adaptive Referenzmodellierung. Methodische Konzepte zur Konstruktion und Anwendung wiederverwendungsorientierter Informationsmodelle. Logos

Dey A. (2001) Understanding and using context. In: Personal and ubiquitous computing 5(1), pp. 4–7

Dijkman R., Dumas M., García-Bañuelos L. (2009) Graph Matching Algorithms for Business Process Model Similarity Search. In: Dayal U., Eder J., Koehler J., Reijers H. A. (eds.) Business Process Management. Speinger, pp. 48–63

Dijkman R., Dumas M., Van Dongen B., Käärik R., Mendling J. (2011) Similarity of business process models: Metrics and evaluation. In: Information Systems 36(2), pp. 498–516

Fettke P. (2008) Empirisches Business Engineering. Grundlegung und ausgewählte Ergebnisse. Habilitationsschrift, Universität des Saarlandes

Fettke P. (2014) Eine Methode zur induktiven Entwicklung von Referenzmodellen. In: Suhl L., Kundisch D., Beckmann L. (eds.) Tagungsband Multikonferenz Wirtschaftsinformatik. Universität Paderborn, pp. 1034–1047

Fettke P. (2015) Integration von Prozessmodellen im Großen: Konzept, Methode und experimentelle Anwendungen. In: Thomas O., Teuteberg F. (eds.) Internationale Tagung Wirtschaftsinformatik. Universität Osnabrück, pp. 453–467

Fettke P., Loos P. (2007) Perspectives on Reference Modeling In: Reference Modeling for Business Systems Analysis Fettke P., Loos P. (eds.) Idea Group Publishing, pp. 1–20

Fettke P., Loos P., Zwicker J. (2006) Business Process Reference Models: Survey and Classification. In: Bussler C., Haller A. (eds.) Business Process Management Workshops. Springer, pp. 469–483

Fettke P., vom Brocke J. (2018) Referenzmodell. In: Gronau N., Becker J., Leimeister J. M., Overhage S., Suhl L. (eds.) Enzyklopädie der Wirtschaftsinformatik - Online-Lexikon 10th ed. http://www.enzyklopaedie-der-wirtschaftsinformatik.de/. GITO

Folino F., Guarascio M., Pontieri L. (2012) Discovering Context-Aware Models for Predicting Business Process Performances. In: Meersman R., Panetto H., Dillon T. e. a. (eds.) OTM Confederated International Conferences. Springer, pp. 287–304

Gerke K., Cardoso J., Claus A. (2009) Measuring the compliance of processes with reference models. In: Meersman R., Dillon T., Herrero P. (eds.) OTM Confederated International Conferences. Springer, pp. 76–93

Gottschalk F., van der Aalst W., Jansen-Vullers M. (2008a) Mining Reference Process Models and Their Configurations. In: Meersman R., Tari Z., Herrero P. (eds.) OTM Workshops. Springer, pp. 263–272

Gottschalk F., van Der Aalst W., Jansen-Vullers M., La Rosa M. (2008b) Configurable workflow models. In: International Journal of Cooperative Information Systems 17(02), pp. 177–221

Gröger S., Decker J., Schumann M. (2014) Do Universities Get the Hang of Working Efficiently? – A Survey of the Influencing Factors on the Adoption of Electronic Document and Workflow Management in German-speaking Countries. In: Americas Conference on Information Systems. AIS, pp. 576–590

Gröger S., Schumann M. (2014) Entwicklung eines Referenzmodells für die Gestaltung des Drittmittel-Prozesses einer Hochschule und Ableitung von Einsatzgebeiten für Dokumenten- und Workflow-Management-Systeme.. Georg-August-Universität. webdoc.sub.gwdg.de/ebook/serien/lm/arbeitsberichte_anwebus/2014_01.pdf

Hallerbach A., Bauer T., Reichert M. (2010) Capturing variability in business process models: the Provop approach. In: Journal of Software Maintenance and Evolution: Research and Practice 22(6-7), pp. 519–546

Henderson-Sellers B., Ralyté J., Ågerfalk P. J., Rossi M. (2014) Situational method engineering. Springer

Hevner A., March S., Park J., Ram S. (2004) Design science in information systems research. In: MIS quarterly 28(1), pp. 75–105

Karow M., Pfeiffer D., Räckers M. (2008) Empirical-Based Construction of Reference Models in Public Administrations.. In: Bichler M., Hess T., Krcmar H., Lechner U., Matthes F., Picot A., Speitkamp B., Wolf P. (eds.) Tagungsband Multikonferenz Wirtschaftsinformatik. GITO, pp. 1613–1624

Kornyshova E., Deneckère R., Claudepierre B. (2010) Contextualization of method components. In: International Conference on Research Challenges in Information Science. IEEE, pp. 235–246

Koschmider A., Mannhardt F., Heuser T. (2019) On the Contextualization of Event-Activity Mappings. In: Daniel F., Sheng Q. Z., Motahari H. (eds.) Business Process Management Workshops. Springer, pp. 445–457

Kumar A., Yao W. (2012) Design and management of flexible process variants using templates and rules. In: Computers in Industry 63(2), pp. 112–130

La Rosa M., Dumas M., Ter Hofstede A., Mendling J. (2011) Configurable multi-perspective business process models. In: Information Systems 36(2), pp. 313–340

La Rosa M., Dumas M., Uba R., Dijkman R. (2013) Business Process Model Merging: An Approach to Business Process Consolidation. In: ACM Transactions on Software Engineering and Methodology 22(2), 11:1–11:42

La Rosa M., van Der Aalst W., Dumas M., Milani F. (2017) Business Process Variability Modeling: A Survey. In: ACM Computing Surveys 50(1), 2:1–2:45

Leng J., Jiang P. (2017) Granular computing–based development of service process reference models in social manufacturing contexts. In: Concurrent Engineering 25(2), pp. 95–107

Li C., Reichert M., Wombacher A. (2011) Mining business process variants: Challenges, scenarios, algorithm. In: Data & Knowledge Engineering 70(5), pp. 409–434

Li J., Bose R., van der Aalst W. (2011) Mining Context-Dependent and Interactive Business Process Maps Using Execution Patterns. In: zur Muehlen M., Su J. (eds.) Business Process Management Workshops. Springer, pp. 109–121

Martens A., Fettke P., Loos P. (2014) A Genetic Algorithm for the Inductive Derivation of Reference Models Using Minimal Graph-Edit Distance Applied to Real-World Business Process Data. In: Kundisch D., Suhl L., Beckmann L. (eds.) Tagungsband Multikonferenz Wirtschaftsinformatik. Universität Paderborn, pp. 1613–1626

Mendling J. (2008) Metrics for Business Process Models. In: Metrics for Process Models - Empirical Foundations of Verifiction, Error Prediction, and Guidlines for Correctness. Springer, pp. 103–133

Minor M., Tartakovski A., Bergmann R. (2007) Representation and Structure-Based Similarity Assessment for Agile Workflows. In: Weber R., Richter M. (eds.) International Conference on Case-Based Reasoning. Springer, pp. 224–238

Moon M., Hong M., Yeom K. (2008) Two-Level Variability Analysis for Business Process with Reusability and Extensibility. In: International Computer Software and Applications Conference. IEEE, pp. 263–270

Mounira Z., Mahmoud B. (2010) Context-aware Process Mining Framework for Business Process Flexibility. In: International Conference on Information Integration and Web-based Applications & Services. ACM, pp. 421–426

Peffers K., Tuunanen T., Rothenberger M., Chatterjee S. (2007) A design science research methodology for Information Systems Research. In: Journal of Management Information Systems 24(3), pp. 45–77

Puhlmann F., Schnieders A., Weiland J., Weske M. (2005) Variability mechanisms for process models. 17. Hasso-Plattner-Institut

Rehse J.-R., Fettke P. (2017) Towards Situational Reference Model Mining - Main Idea, Procedure Model & Case Study. In: Leimeister J. M., Brenner W. (eds.) Internationale Tagung Wirtschaftsinformatik. AIS, pp. 271–285

Rehse J.-R., Fettke P., Loos P. (2013) Eine Untersuchung der Potentiale automatisierter Abstraktionsansätze für Geschäftsprozessmodelle im Hinblick auf die induktive Entwicklung von Referenzprozessmodellen. In: Alt R., Franczyk B. (eds.) Internationale Tagung Wirtschaftsinformatik. Universität Leipzig, pp. 1277–1291

Rehse J.-R., Fettke P., Loos P. (2016) An Execution-Semantic Approach to Inductive Reference Models Development. In: European Conference for Information Systems Vol. Research Papers, 80. AIS

Rehse J.-R., Fettke P., Loos P. (2017) A graph-theoretic method for the inductive development of reference process models. In: Software & Systems Modeling 16(3), pp. 833–873

Reijers H., Mans R., van der Toorn R. (2009) Improved model management with aggregated business process models. In: Data & Knowledge Engineering 68(2), pp. 221–243

Reinhartz-Berger I., Soffer P., Sturm A. (2009) Organisational reference models: Supporting an adequate design of local business processes. In: International Journal of Business Process Integration and Management 4(2), pp. 134–149

Reinhartz-Berger I., Soffer P., Sturm A. (2010) Extending the adaptability of reference models. In: IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans 40(5), pp. 1045–1056

Rosemann M., Recker J. (2006) Context-aware process design: Exploring the extrinsic drivers for process flexibility. In: Latour T., Petit M. (eds.) International Conference on Advanced Information Systems Engineering. Namur University Press, pp. 149–158

Rosemann M., Recker J., Flender C., Ansell P. (2006) Understanding context-awareness in business process design. In: Spencer S., Jenkins A. (eds.) Australasian Conference on Information Systems. AI

Rosemann M., van der Aalst W. (2007) A Configurable Reference Modelling Language. In: Information Systems 32(1), pp. 1–23

Saidani O., Nurcan S. (2007) Towards context aware business process modelling. In: Workshop on Business Process Modeling, Development, and Support

Saidani O., Nurcan S. (2009) Context-awareness for adequate business process modelling. In: International Conference on Research Challenges in Information Science. IEEE, pp. 177–186

Saidani O., Rolland C., Nurcan S. (2015) Towards a generic context model for BPM. In: Hawaii International Conference on System Sciences. IEEE, pp. 4120–4129

Schoknecht A., Thaler T., Fettke P., Oberweis A., Laue R. (2017) Similarity of Business Process Models – A State-of-the-Art Analysis. In: ACM Computing Surveys 50(4), 52:1–52:33

Scholta H. (2016) Semi-Automatic Inductive Derivation of Reference Process Models that Represent Best Practices in Public Administrations. In: European Conference for Information Systems Vol. Research-in-progress papers, 53. AIS

Schunselaar D., Verbeek E., Reijers H., van der Aalst W. (2014) Using Monotonicity to Find Optimal Process Configurations Faster. In: Accorsi R., Ceravolo P., Russo B. (eds.) International Symposium on Datadriven Process Discovery and Analysis. Citeseer, pp. 123–137

Timm F., Sandkuhl K., Fellmann M. (2017) Towards A Method for Developing Reference Enterprise Architectures. In: Leimeister J. M., Brenner W. (eds.) Internationale Tagung Wirtschaftsinformatik. AIS, pp. 331–345

Tscheschner W. (2006) Transformation from EPC to BPMN. In: Business Process Technology 1(3), pp. 7–21

United Nations Centre for Trade Facilitation and Electronic Business (2009) Core Components Technical Specification Version 3.0.. United Nations

van der Aalst W., Alves de Medeiros A., Weijters A. (2006a) Process Equivalence: Comparing Two Process Models Based on Observed Behavior. In: Dustdar S., Fiadeiro J., Sheth A. (eds.) Business Process Management. Springer, pp. 129–144

van der Aalst W., Dreiling A., Gottschalk F., Rosemann M., Jansen-Vullers M. (2006b) Configurable Process Models as a Basis for Reference Modeling. In: Bussler C., Haller A. (eds.) Business Process Management Workshops. 3812. Springer, pp. 512–518

Van der Aalst W., Dustdar S. (2012) Process Mining Put into Context. In: IEEE Internet Computing 16(1), pp. 82–86

van Dongen B., Carmona J., Chatain T. (2016) A Unified Approach for Measuring Precision and Generalization Based on Anti-alignments. In: La Rosa M., Loos P., Pastor O. (eds.) Business Process Management. Springer, pp. 39–56 van Dongen B., Dijkman R., Mendling J. (2008) Measuring Similarity between Business Process Models. In: Bellahsene Z., Leonard M. (eds.) International Conference on Advanced Information Systems Engineering. Springer, pp. 450–464

vom Brocke J. (2002) Referenzmodellierung: Gestaltung und Verteilung von Konstruktionsprozessen. Logos

vom Brocke J. (2007) Design Principles for Reference Modeling: Reusing Information Models by Means of Aggregation, Specialisation, Instantiation, and Analogy. In: Fettke P., Loos P. (eds.) Reference Modeling for Business Systems Analysis. Idea Group Publishing, pp. 47–75

vom Brocke J., Fettke P. (2018) Referenzmodellierung. In: Gronau N., Becker J., Leimeister J. M., Overhage S., Suhl L. (eds.) Enzyklopädie der Wirtschaftsinformatik – Online-Lexikon 10th ed. GITO

Weidlich M. (2011) Behavioural Profiles – A Relational Approach to Behaviour Consistency. Doctoral Thesis, University of Potsdam

Winter R. (2003) Modelle, Techniken und Werkzeuge im Business Engineering In: Business Engineering: Auf dem Weg zum Unternehmen des Informationszeitalters Österle H., Winter R. (eds.) Springer, pp. 87–118

Winter R. (2008) Design science research in Europe. In: European Journal of Information Systems 17(5), pp. 470–475

Yahya B., Bae H., Bae J., Kim D. (2012) Generating valid reference business process model using genetic algorithm. In: International Journal of Innovative Computing, Information and Control 8(2), pp. 1463–1477

Yan Z., Dijkman R., Grefen P. (2010) Fast Business Process Similarity Search with Feature-Based Similarity Estimation. In: Meersman R., Dillon T., Herrero P. (eds.) OTM Confederated International Conferences. Springer, pp. 60–77