



An analysis of capability meta-models for expressing dynamic business transformation

Georgios Koutsopoulos¹ · Martin Henkel¹ · Janis Stirna¹

Received: 7 October 2019 / Revised: 5 November 2020 / Accepted: 11 November 2020 / Published online: 24 December 2020
© The Author(s) 2020

Abstract

Environmental dynamism is gaining ground as a driving force for enterprise transformation. To address the changes, the capabilities of digital enterprises need to adapt. Capability modeling can facilitate this process of transformation. However, a plethora of approaches for capability modeling exist. This study explores how concepts relevant to change have been implemented in the meta-models of these approaches, aiming to visualize relationships among change-related concepts, and identify ways to improve capability modeling toward a more efficient depiction of capability change. The concepts are visualized in concept maps, and a framework is developed to assist the classification of concepts relevant to change functions. Similarities and differences among the existing models are discussed, leading to suggestions toward improvements of capability modeling for capability adaptation.

Keywords Capability · Enterprise modeling · Change · Adaptability · Transformation

1 Introduction

Every modern organization is facing the challenge of adapting to environmental dynamism, which is a significant factor that triggers and drives change processes on a regular basis. The most challenging part of managing changes that originate from the environment of the organization is that they are often unpredictable and sudden, yet they require a response that is not only efficient, but also immediate [1]. In addition, the dependence of organizations on information systems (IS) is on the rise. IS are an innate part of the business. Therefore, any discussion about organizations that need to be highly adaptive also refers to highly adaptive IS, which need to be constantly available and adapting to changing environmental conditions and requirements [2].

As a response to this situation, dynamically adaptive IS emerged exhibiting degrees of variability depending on user requirements and contextual run-time fluctuations. These systems are built with several predefined variation points, and depending on the state of the context, a suitable variant is selected to realize a variation point [2].

It is required for any modern IS to be able to deliver business value in accordance with contextual variations, for example, user preferences, business models of suppliers, local legislations, resource pricing or location [1]. Furthermore, even if adapting single aspects of an organization is important, the ultimate goal of an adaptive organization is to adapt what it is capable of. The concept of capability encompasses a wide spectrum of concepts and associations among different aspects of an organization. Therefore, an organization that desires to be flexible requires capability modeling approaches that address the adaptive and dynamic nature of its capabilities.

A plethora of capability modeling techniques and methods exist that employ different sets of concepts in their meta-models in order to reflect the nature of the capabilities of an organization. However, despite the fact that the concept of capability is used with relative conceptual consistency, different purposes for developing these modeling techniques result in different sets of concepts [3], and hence, significantly different meta-models.

Communicated by Jelena Zdravkovic and Iris Reinhartz-Berger.

✉ Georgios Koutsopoulos
georgios@dsv.su.se

Martin Henkel
martinh@dsv.su.se

Janis Stirna
js@dsv.su.se

¹ Department of Computer and Systems Sciences, Stockholm University, Stockholm, Sweden

The objectives of this paper are (1) *to analyze how the meta-models represent capability adaptability* and (2) *to present findings and suggestions for improving modeling of capability adaptation*. As a part of this work, a framework is developed to facilitate the identification and classification of the elements related to change before exploring their relationships in depth.

This study is a follow-up on our initial study [4] and contains extended analysis of how the existing approaches support modeling of capability change. More precisely, this study extends our previous work by enriching the background and related research and elaborating on a detailed step-wise presentation of methodology, including process models of the research, in order to position this study within its research context, clarify the research process and, in particular the classification of concepts, and address the validity issues of the study. The study has also been extended with the addition of an analysis of the development context related to organizational change of the capability meta-models using their developers' perspectives as stated in the papers that the meta-models were introduced in.

The results are extended by including statistical analysis of the capability meta-models to improve the general understanding of the field, a more detailed presentation of the concept maps that have been used to visualize the concept set derived from the literature review and the analysis of one of the maps to illustrate the value and convenience of visualizing a network of concepts. Additionally, in comparison with [4], the results are extended with an application of the classification method on one of the meta-models used as an illustrative example to depict how the concepts are associated to specific functions of capability change, and the outcome of the classification has been used to create specific concept maps based on the classification. The function-specific concepts maps and concept sets provide deeper insight on how the existing capability meta-models are depicting adaptive attributes of capabilities. Finally, the discussion has been extended to reflect on the extended findings.

The rest of the paper is structured as follows. Section 2 presents an overview of the concepts of interest for this study and related literature. Section 3 provides an overview of the research process and methods employed for the collection and analysis of the meta-models. Section 4 presents the results in terms of concept maps and a description of how the meta-models support functions of capability adaptation. Section 5 discusses the findings and suggests possible improvements. Section 6 presents concluding remarks.

2 Background and related research

The notion of capability has been defined in various ways in the literature. For example, Grabis et al. [5] defined that capability "is the ability and capacity that enables an

enterprise to achieve a business goal in a certain context." It assumes that capability is always defined by certain intention (goal), a defined operational context, and means of achieving the goal. According to the literature search, no studies similar to this one have been conducted before. However, there have been studies that can be considered related, having as main topics capability modeling, capability as used in enterprise architecture, and the dimensions of adaptability.

It should be noted that despite being relevant to change, the term "dynamic capability" [6, 7], which has been widely used, especially in management literature, has been eschewed on purpose in this study, because it can be considered as relatively imprecise [8]. We consider that the term refers to strategic capabilities in contrast to operational capabilities [9]. Both types can be adaptive or not and, in addition, the focus of this study is on exploring how adaptability of capabilities is modeled and not on the differences between strategic and operational management as reflected in capability modeling.

Modeling capability is an ongoing research topic. Koç [10] has conducted a systematic mapping of methods of modeling, designing and developing capabilities, identifying that having a resource-based view and changing environments were the main motivations for using the concept of capability. Development approaches and frameworks were the main solution artifacts from the research. It was also found that methodological support for capability management was scarce and finally, the finding that was considered the most important, was that enterprise models were only exploited to some extent. In another study, Koç et al. [11] performed a systematic literature review on context modeling, being an essential element of capability management, and identified the lack of methodology or language to model context.

To some degree, the concept of capability has found its way into *enterprise architecture*. Zdravkovic et al. [3, 12] analyzed how the concept of capability is employed in various business architecture, enterprise architecture, enterprise modeling and business analysis frameworks. The results state that the concept is used in a similar way, i.e., to represent the ability to achieve a certain result, in all the studied frameworks. There are, however, significant differences in the point of focus around capabilities which is a result of the different purposes of the frameworks, e.g., in [13], the focus is on the strategic viewpoint of capabilities which results in including the concept of goal.

Capability adaptations have been addressed in such terms as business services, business process variants and delivery adjustments. The purpose of capability delivery adjustments is to change capability delivery as a response to the changing context and delivery performance without the need to redesign the capability and underlying IS [14]. Another

study concerning manufacturing systems also emphasized the importance of dynamic reconfiguration of capabilities on run-time [15].

In a study utilizing capability modeling, Loucopoulos et al. [16] have analyzed the main challenges regarding requirements of developing a dynamic system that is aware of its environment. The three main challenges that have been elaborated concern the following:

- The design challenge is associated to the emergent behavior and the dynamics of the system under development and its related environment. The implication of this challenge is that, apart from the goals elicited through traditional requirements engineering processes, a fact that needs to be taken into consideration is whether the system needs to be able to address emergent goals during its lifetime.
- The modeling challenge is associated to not only expecting but also representing properly system behaviors that emerge dynamically. This implies the need for the development of approaches that have the potential to represent, communicate and analyze the system in a way that guarantees that any dynamically emerging requirements will be met.
- The predictability challenge is associated to the impact that the system itself and its behavior have on the environment. This implies the need to take into consideration that the system and its context are in a constant dynamic interaction and affecting each other, and this impact on both directions needs to be predicted.

Finally, what is common among these challenges is that the system's self-awareness during run-time is required, and this can be addressed by implementing transformational capabilities in the system in order to enable the potential of a feedback functionality [16].

A common approach to describe adaptability is by identifying the dimensions and functionalities of adaptive systems. In [17], the concept of adaptability of capabilities is tackled by introducing a framework that includes the main dimensions and interrelated aspects for analyzing and evaluating enterprise adaptability. Their framework consists of three dimensions, namely (1) complexity of the environment, (2) managerial profiling and (3) artifact-integrated components. Similarly, Weyns et al. [18] and Morandini et al. [19] are two studies that include such dimensions. They are described in detail in Sect. 3.4. The distinction between functionalities and knowledge in these two studies was the inspiration for the framework presented in the following section.

3 Research process

This study is part of a project whose aim is to provide methodological and tool support for changing organizations by supporting capability modeling within dynamic contexts. The project is elaborated within the design science research paradigm [20] and follows the guidelines of [21], according to which a method is considered a design artifact. The five activities associated with any design science project according to [21] are:

1. Explicate problem
2. Outline artifact and define requirements
3. Design and develop artifact
4. Demonstrate artifact
5. Evaluate artifact

In particular, this study concerns the first activity included in their design science research method framework, namely "Explicate problem." This step aims to answer the question "What is the problem experienced by some stakeholders of a practice and why is it important?" [21]. The authors define the term problem as the distance between a current state and a desired one [21]. This results in a clarification that this step concerns not only problems but also opportunities that are explicated during the first activity of a design science project. This is also the case in this study. It explores which aspects of change are being addressed by capability modeling methods by analyzing change concepts in their respective meta-models. Therefore, it is the first activity that explicates the opportunity to develop a capability modeling method, optimized for change modeling and analysis. The research process is depicted as a BPMN (Business Process Model and Notation) diagram [22] in Fig. 1. It should be noted that instead of using tasks, the above-mentioned activities have been modeled as sub-processes, in order not only to indicate that they are complex activities that consist of several tasks but also to reduce the complexity of the diagram. Every sub-process has been modeled separately and is described in detail for each research activity in the following sections. The initial task, the collection of the set of capability meta-models was followed by the analysis of the meta-model statistics, the analysis of the meta-models' development context and the creation of the concept dataset, which in return, led to the visualization of the meta-model concepts in a map. In parallel, the development of a classification framework for change-related concepts provided the necessary tool for creating function-specific concept datasets and the respective concepts maps.

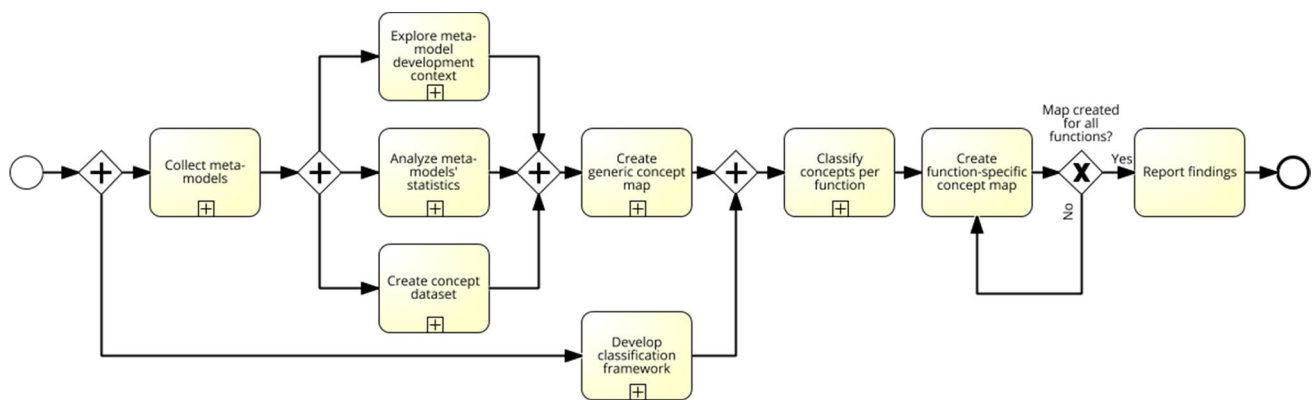


Fig. 1 The research process of this paper, including several sub-processes

3.1 Capability meta-model collection

The process of identifying capability meta-models in the literature was initiated by keyword searches in dblp.org using the terms «capabilit* AND model» and Google Scholar using the terms «“capability OR capabilities AND model” AND “enterprise modelling OR modeling”». The reason behind the difference in the search terms lies in the fact that dblp.org is a database specialized in computer science literature; therefore, the search in Google Scholar had to be narrowed down with additional search terms. This resulted in a set of 672 papers in dblp.org and 169 papers in Scholar.

The process is depicted as a BPMN diagram in Fig. 2.

The inclusion criterion for a study was to contain a conceptual meta-model that contains at least one capability concept. After removing duplicates, a starter set of papers was formed and a snowballing technique was applied using the references in the papers to identify more sources of meta-models until a point of saturation had been reached. Several included meta-models are different versions belonging to the same project; however, they reflect different elaboration levels and include variations in their concept sets, and therefore, they were deemed worth exploring. Finally, several

documentations and specifications of enterprise architecture frameworks [23–27] and modeling languages [1, 28] that include capability viewpoints were included in order to complete the final set of meta-models to be explored. The number of the included meta-models is 64. It is worth mentioning that approaches based on capability mapping, for example [29], despite being valid approaches, have not been included in the study unless a meta-model existed as well.

3.2 Exploration of the meta-models’ development context

Another aspect of the capability meta-model set that was deemed necessary to explore was the development context of the meta-models in terms of the stated desire to address dynamic conditions. This was performed by exploring the perspectives of the meta-model developers as stated in the publications that included the meta-models. The objective is to provide quantified results. In order to achieve this, a deductive content analysis was performed according to the guidelines in [30]. The reason for selecting content analysis is that it is more suitable for the quantification of results [31].

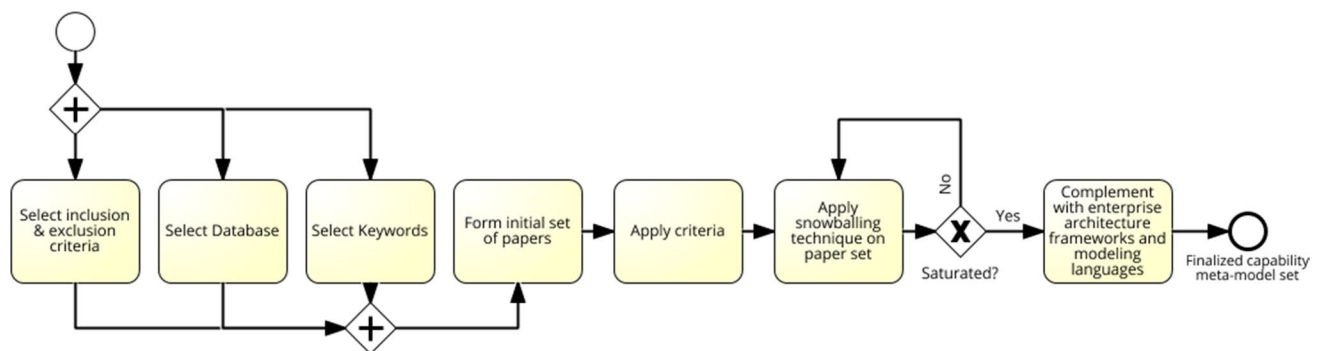


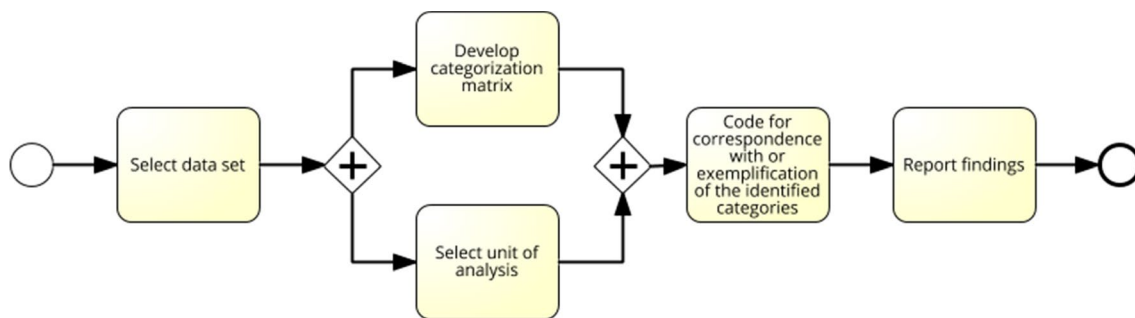
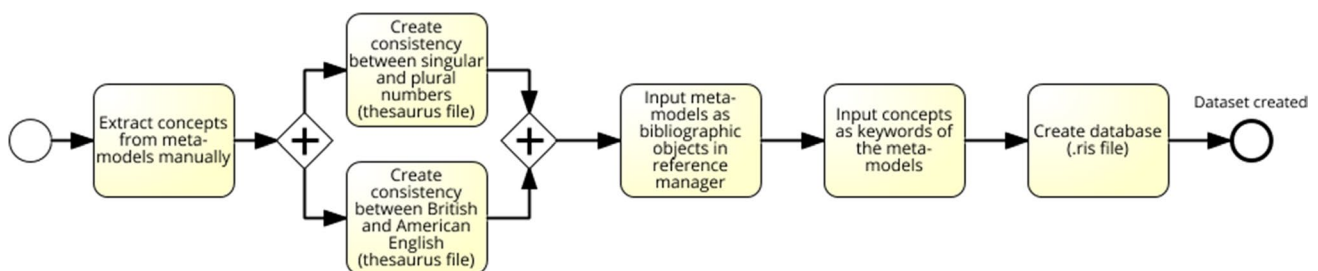
Fig. 2 The meta-model collection sub-process

Table 1 The categorization matrix for the content analysis

	Dynamic	Static	Not specified
<i>What is the development context of capability meta-models?</i>			
Explicit			
Implicit			

Initially, the papers where the 64 capability meta-models were published were used as the dataset. A publication was decided as the unit of analysis to provide insight on the development context of every meta-model individually. In parallel, a categorization matrix was developed to drive the activity of categorizing the papers. The matrix included the “Dynamic,” “Static” and “Not specified” categories to reflect the development context of the meta-models. “Dynamic” reflects change being part of the development context, “Static” when change is excluded from the development context and “Not specified” if there is no specific description in the paper. Another classification concerns whether the context specification is explicit or implicit in the text. The categorization matrix is shown as Table 1.

After the development of the categorization matrix, the data were coded for correspondence with the categories included in the matrix. The final step in the content analysis is the report of the findings. The process is depicted in Fig. 3.

**Fig. 3** The content analysis sub-process**Fig. 4** The concept dataset creation sub-process

3.3 Creation of the concept dataset

This step of the research process included the manual extraction of every class concept from the meta-models in order to create a dataset file valid for the visualization tool.

Certain terms have been converted from plural to singular number, e.g., “resources” to “resource” and British to American English. Every single conversion was registered as a conversion rule in a thesaurus file which is required for the visualization tool, so that any possible future update of the dataset treats identical conversions in the same way to maintain consistency. Afterward, a reference manager was used to create the dataset. The meta-models were created as bibliographic objects, and their concepts were added as keywords to their respective meta-model. The finalized dataset was extracted as an.ris file, which is a valid source for the visualization and analysis tool. The process is also depicted as a BPMN model in Fig. 4.

3.4 Development of classification framework for change-related concepts

In this work, we employ a framework in order to compare existing models used for representing capability adaptation. The framework is based on work done in the area of adaptive IS and contains the basic elements that are needed to express adaptability. The process of developing the framework is depicted in Fig. 5, which shows the steps in the form of a

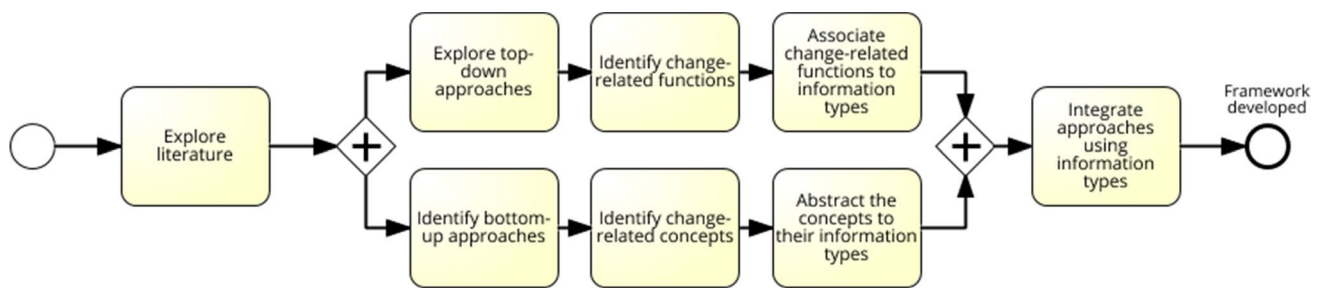


Fig. 5 The framework development sub-process

BPMN diagram, in particular, the classification framework development sub-process of the general research process.

In order to present the framework, we first introduce the work of Weyns et al. [18] and Morandini et al. [19] that the framework is inspired from. These studies were found during the initial task of literature exploration.

Weyns et al. [18] have conducted a study concerning software development; however, their contribution can be applied to adaptive and self-adaptive systems in general, since software, nowadays, is tightly associated to business, being a part of the business or supporting it. Weyns et al. suggested that a model should be capable of describing and reasoning about (1) how the system monitors the environment, referring to context-awareness, (2) how the system monitors itself, referring to self-awareness, (3) how the system adapts itself, and (4) how the system coordinates monitoring and adaptation. In addition, the work suggests that besides addressing a wide spectrum of perspectives, a model should also be extensible for future refinements.

In a similar way, Morandini et al. [19] identify the three core *functionalities* of adaptive and, in particular, self-adaptive systems being (a) sensing the environment in order to recognize “problems,” (b) taking decisions in which behavior to exhibit, and (c) realizing the behavior change by adaptation. They state that, as a result to the above-mentioned functionalities, a system needs *information* about (α) what to monitor and for which symptoms, (β) which alternative behaviors are available, and (γ) the decision criteria for a specific behavior. Thus, they differentiate between needed core functions (a–c) and information (α – γ).

As shown in Fig. 6, the two studies are consistent with each other and enable a conceptual composition that, in return, provides a basis for a capability change concept classification framework. For instance, how the system monitors the environment and itself is associated to the monitoring function and to what to monitor and for which symptom types of information. How the system adapts itself is associated to taking a decision on behavior and realization of the behavior functions and to the available alternative behaviors and decision criteria types of information. The coordination between monitoring and adaptation is associated to the realization of the selected behavior function. This part comprises the top-down approach of the framework elaboration; in other words, associated information elements have been derived from the main change functions using literature sources. This led to the three top-level functions of the framework: observation, decision and delivery (Table 2).

Identifying the concepts that exist in a capability model can also be assisted by a predefined set of adaptability concepts. We here make use of the work by Grabis and Kampars [32] that have suggested such a set of concepts, as shown in Fig. 7. These concepts are dedicated to adaptive capability run-time adjustments consisting of:

- Capability
- KPI
- Goal
- Process
- Adjustment
- Context element

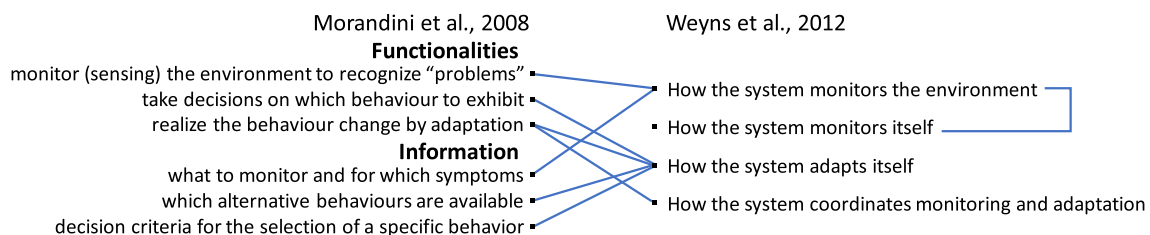


Fig. 6 The association between the two studies

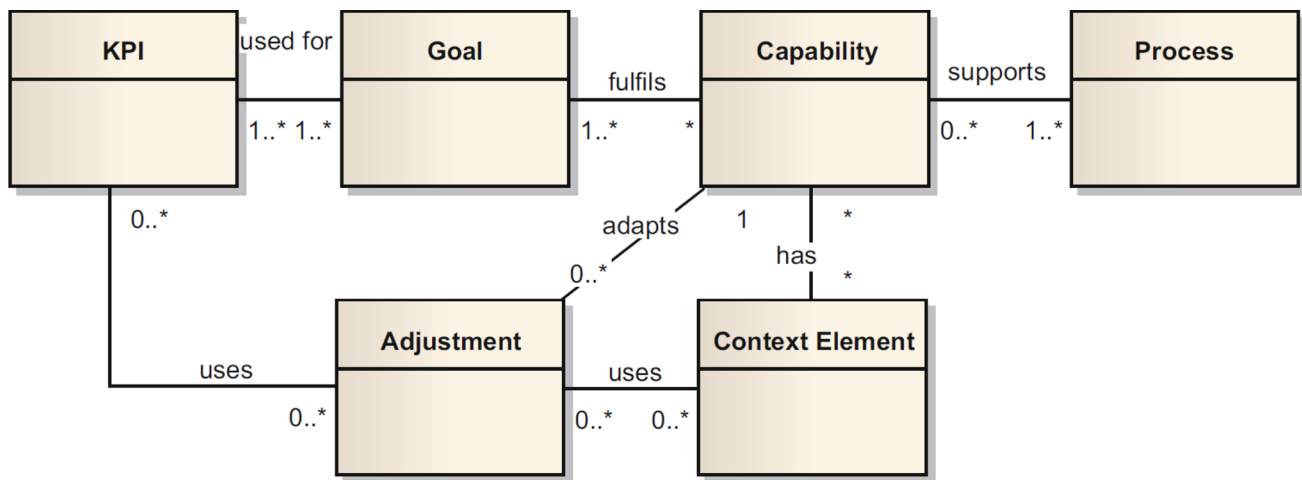


Fig. 7 A fragment of capability meta-model, focusing on adjustments, from [32]

To fit these concepts into the three framework functions, abstraction has been applied on this set of concepts in order to form the second level of the framework. For example, “intention” is derived from abstracting “goal,” “measurement” is derived from abstracting “KPI,” and “realization” is derived from abstracting “process” (Table 2). Combining the result of the abstraction process with the above-mentioned findings from literature resulted in a framework for categorizing the adaptability attributes of capabilities in existing meta-models.

The complete framework is a combination of a top-down approach using the adaptability *functions* and a bottom-up approach using the predefined set of *information* concepts. Overall, the framework describes that change consists of functions which consider specific information types. In particular, change consists of observation, decision and delivery of change. Observation considers the information types measurement, context and the system itself; decision considers intention elements, decision criteria and existing alternatives, as well as delivery considers the realization of the capability, the adjustment and the capability architecture, as shown in Table 2.

The *observation* function refers to monitoring a capability by capturing external and internal data associated to it. As shown in the framework, important information types associated to observation are the *measurement* of a capability’s attributes along with the sources of data, both *contextual* and internal (the *system itself*). On a generic level, observation elements in a meta-model act as a means to depict the sources of data that can be used in order to evaluate a capability’s performance. While the external context data refer to various aspects of the environment that are affecting a capability, like political, technological, economic, etc., the “system itself” refers to all component resources that are relevant to the capability. The observation function thus

Table 2 The classification framework for change-related capability concepts

Functions	Information types	Example concepts
Observation	Measurement	KPI, metric
	Context	Context element, environment
	System itself	Capability, resource
Decision	Intention	Goal, objective
	Alternatives	Variation
	Criteria	Policy
Delivery	Realization	Process, service
	Adjustment	Activity modifier
	Capability architecture	Dependent (capability), specialized (capability)

makes use of measurements, such as specific KPI:s, based on sources of data from the capability’s context and/or from its internal resources.

The *decision* function in the framework refers to all concepts related to the processing of the information captured through observation in order to make a decision on capability adaptation. The analysis based on the factors that are deemed as relevant to the decision is an essential part of this function. The actual decision is also included in this part. Three information types support the decision function. The *intention* driving the adaptation can be represented by for example a desired future state (a goal), or be quantified as objectives. Meta-models may also incorporate the ability to express *alternatives* that fulfill the intention. For example, alternatives may be expressed as several variations of an existing capability design. The information type *decision criteria* captures the information that guides the selection of decision alternatives. For example, general policies may exist stating an organization’s general direction, or goal.

Thus, the decision criteria can be used to complement the intension when deciding among alternatives.

Delivery of capabilities refers to how the decision on adaptation is applied affecting the way a capability change is brought through and how capabilities' interrelationships are affected by run-time adaptations. Important information types are the realization and adjustment of the capability and the capability architecture existing within an organization. *Realization*, being one of the information types considered during delivery, is the capability constituents that may be affected by the change. For example, this may concern processes, services and any other type of behavior element that realizes the capability and delivers value. *Adjustment* concerns the deployment of change to a capability, for example, as an adaptation or transformation of realization elements. The *capability architecture* information type refers to types and roles of capabilities. This helps to depict the impact of a changing capability to its interrelated capabilities. For example, capabilities that depend on other capabilities or a specialization or generalization of the changing capability are affected by the change and how they are affected is reflected in their relation type to the capability. The capability architecture reflects the structure of all the capabilities that are affected through their various relationship types.

The framework addresses the functions of change from a single capability perspective, which means that, even if an organization possesses a plethora of capabilities, change is treated as a single capability phenomenon which potentially affects its interrelated capabilities. In other words, the starting and focal point of a capability change, according to this perspective, is always a single capability. In this way, other capabilities bear relevance only as being part of the system itself, which is observed, and their relevance type with the main capability. Capability architecture becomes relevant only after the change has been delivered, as an information type associated to the impact of a changing capability to its interrelated capabilities. The impact of change on the capability architecture, via delivery, becomes relevant in association with the source change phenomenon. This is the reason behind the classification of the capability and its related internal capabilities as system information type, while any external capabilities are considered part of the capability architecture affected during delivery.

The framework's function elements bear similarities to Boyd's OODA loop [33], from which the functionality names "observation" and "decision" have been inspired. Similarly to the OODA loop, for which Boyd suggests a looping attribute of continuous activity, delivering a capability triggers new internal and external phenomena that need to be observed. That is, as long as the system exists within a dynamic environment, it needs to remain functional to respond to any significant changes to its context, even if these have been generated by the system itself. This is also

in line with the predictability challenge [16], which has been discussed in a previous section, and suggested that a system and its behavior are not only affected by the environment but also affecting the environment, which in return, affects back the system. So, through the environment, a dynamic system is altering itself, and this should be reflected in the framework. This is depicted in Fig. 8, which includes, not only the function and information elements, but also arrows and a circular shape to address functional continuity.

3.5 Meta-models' analysis

The meta-model analysis consists of the statistical analysis and concept classification activities. Both analysis activities begun once the concepts had been extracted from the meta-models. In addition, statistical analysis and concept classification occurred in parallel; however, they are depicted as two sub-processes in order to simplify the visualization of the research process.

3.5.1 Statistical analysis

For the statistical analysis of the meta-models, the 64 capability meta-models that have been identified in the literature have been initially classified based on specific attributes, i.e., their year of publication, their scope and the number of concepts included. While the first two attributes are straightforward to obtain, the latter required a more complex approach. Regarding their scope, the meta-models have been classified in three main categories

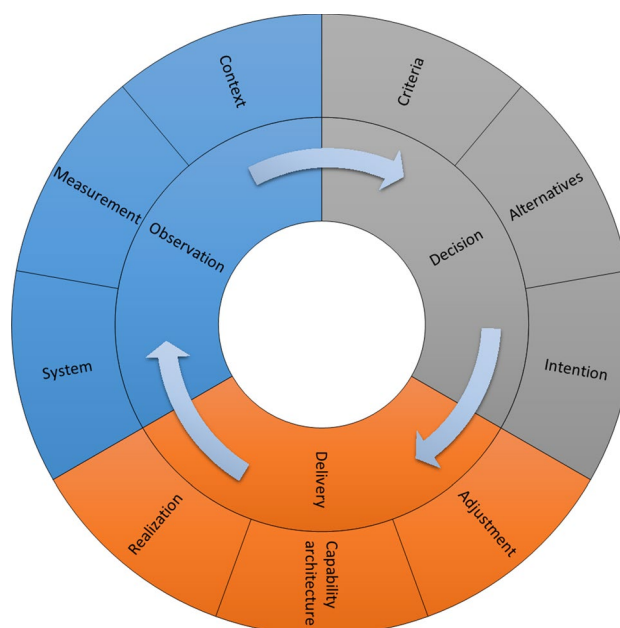


Fig. 8 The classification framework, including the three main change functions and their related information elements

(1) business, (2) IT and (3) both. Every included concept is checked to identify if it is a business or IT term. This results in a business, IT or hybrid scope for the meta-model. The process is depicted in Fig. 9.

3.5.2 Concept classification

The analysis of every meta-model was performed individually; however, a systematic approach was applied. In this section, the analysis method is presented.

The aim of the analysis was to, for each meta-model, classify the concepts in the meta-model according to the framework and map them to information and function elements. This means that each of the concepts found in the models was categorized as belonging to one of the three functional elements (observation, decision, delivery), or as not relevant to capability change.

This part of the process is a thematic analysis [34], which may be inductive, when a research question is derived from the data being analyzed, or theoretical, when the analysis is performed driven by a research question. In a thematic analysis, six steps are involved, in particular, (1) familiarizing with data, (2) generating initial codes, (3) searching for themes, (4) reviewing themes, (5) defining and naming themes and (6) producing the report. In this study, the first two steps of the process have been omitted, since the concept extraction from the meta-models has replaced them. The next three steps have been performed with a theoretical approach, since the analysis is driven by the classification framework. The deviation from a traditional thematic analysis lies in the fact that the themes are pre-existing. The meta-model concepts are used as codes, the information types are used as categories and the framework's function elements are used as the themes. The concepts that are not relevant to change have their own categories as well; however, they are not taken into consideration. Finally, the concept maps are used as the produced report.

3.5.3 The method consists of the following steps

1. Identification of all elements. The usual element types that exist in meta-models are (1) first-class concepts (classes), (2) relationships (of classes), (3) attributes, (4) inheritance and (5) others [35]. In this analysis, the attributes are treated as part of the class elements and inheritance is treated as a special case of relationship, interpreting it as an "is a" association.

In case several capability meta-model fragments exist, as, for example, in viewpoints that are common in enterprise architecture frameworks, they have all been considered. The information types that are being identified and extracted from the meta-model are:

- (a) Class element: The classes are the main source of information in the meta-models. All the concepts comprising the function-specific sets that are presented in the next section have been derived from classes.
- (1) Attribute: Attributes have been rare in the meta-models; however, in some occasions they also provide insight on capability change.
- (b) Association element: The associations consist of several parts that can serve as alternative sources of insight during the analysis. In particular:
 - (1) Classes being associated: The two classes being connected with an association are its most important feature defining the association.
 - (2) Name of the association: The name of the association has been investigated as well. It is a convenient way to identify the existence of implied concepts in the model that are related to capability change.
 - (3) Cardinality: The cardinalities in the meta-models are also worth exploring, especially in certain recursive associations that provide insight on the structural perspective, e.g., capability architecture.

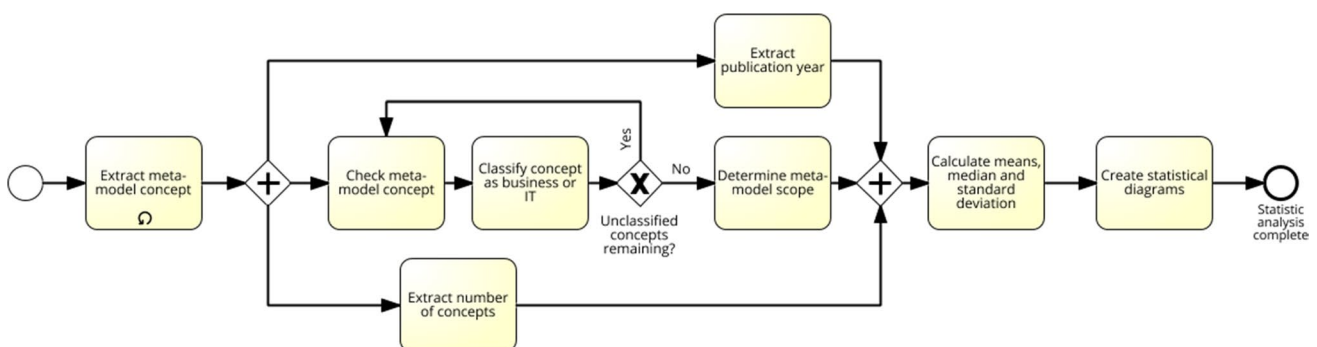


Fig. 9 The statistical analysis sub-process

2. Classification of identified elements.

Initially, the concepts were classified as relevant to capability change or not. After identifying all relevant model elements per type (classes and associations), the associated concepts have been classified according to the framework. This was initially done by first classifying the concepts based on the framework information elements, and, afterward, to the framework functions. Certain concepts have been straightforward to classify. For example, the concept of goal is, by definition, depicting intention; therefore, it is categorized as an intention information element. However, for several other concepts there are unique wordings used as class names tailored for specific capability modeling approaches. In some cases, the concepts were vague and hard to classify. If so, the paper they were extracted from had to be used to provide insight on how the concept has been used in the meta-model and its purpose within the given context, in other words, the paper clarified the concept semantics. It is noteworthy that the vast majority of class concepts were semantically analyzed in their respective papers; however, this did not apply to the association concepts.

In case a concept was associated to different semantics in different papers, which would result in associating it to different functions, the most popular approach would be selected. If popularity could not be applied, for example due to equally popular approaches, the concept would be associated to both functions. The sub-process is shown in Fig. 10, depicted as a BPMN diagram.

3.5.4 An example of meta-model analysis

In this section, an example of analysis of one of the 64 meta-models is presented. The specific meta-model, extracted from [36], has been selected as an example because it has a business scope, includes all three change function type concepts and chronologically is positioned as a relatively recent publication. This means that it resembles the majority of the meta-models; therefore, it is a representative example. In addition, the number of concepts in the model is nine, which results in a small size model, convenient for the example

presentation. Figure 10 depicts the example with the identified elements marked. It concerns classes and associations. No cardinalities have been deemed significant, and no attributes exist in the meta-model.

All the classes have been categorized according to function elements, and this is depicted in Fig. 11 as colored lines depicting their categorization (colored border combined with a descriptive legend in the figure). In particular, blue, gray and orange borders depict observation, decision and delivery, respectively. The light green color depicts the concepts that have been evaluated as irrelevant to capability change. For example, the recursive association “Refined to” depicts information about the structure of a goal and does not associate directly to intention, alternatives or criteria for a decision. It is the “Task goal” that is classified as decision element. The meta-model associations are only implying the existence of a concept in the meta-model; therefore, their association is depicted as a dashed colored border. For example, the association name “achieves” implies the existence of measurement, which in return, relates to observation according to the framework. On its own, it could refer to an intention element too; however, the association is directed toward the intention concept “Mission goal” which is semantically associated to intention information type. The result of classifying the meta-model concepts classification is depicted in Fig. 11. It is interesting to note that traditional modeling concepts do not apply to the classification process. In particular, a generalization would indicate that the two concepts belong to the same type; however, as shown in the example, the two special cases of capabilities are focused on describing concepts pointing to types of capabilities and not capabilities themselves. The specific concepts, “Simple capability” and “Complex capability,” are types that are specifically describing capability attributes relevant to capability interrelations or capability architecture. Therefore, according to the perspective of single capability change, they are associated to the delivery function of a capability change according to the framework, since simple and complex capability are types that become relevant only as far as the delivery of change is concerned, even though their superclass, capability, is referring to the capability itself, which refers to the

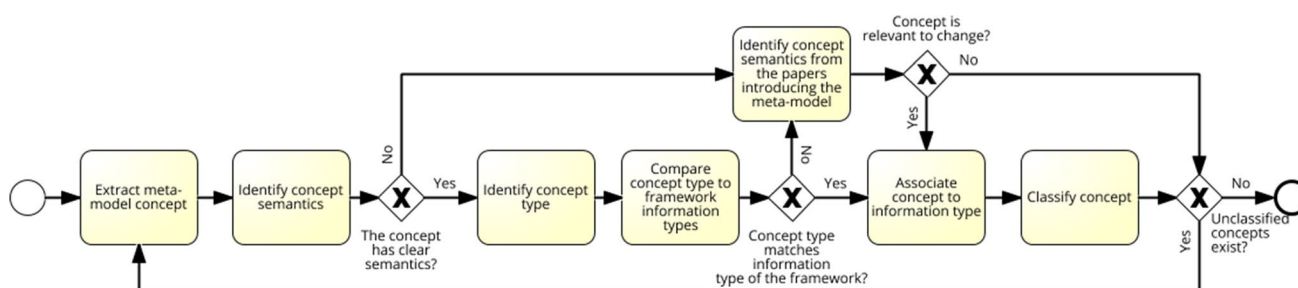


Fig. 10 The concept classification sub-process

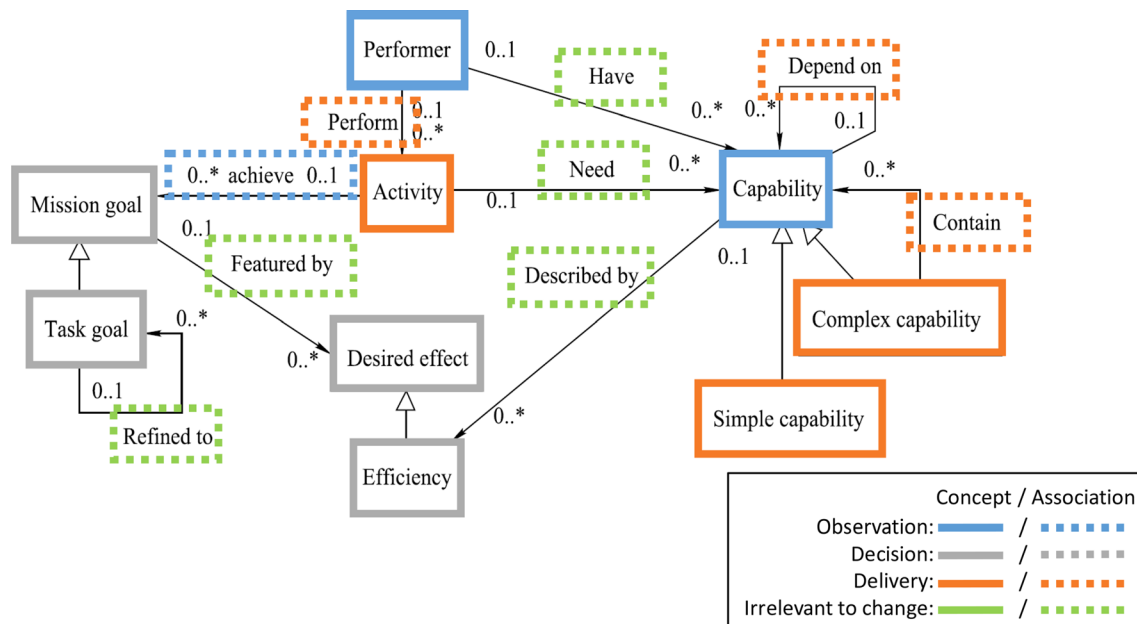


Fig. 11 The result of classifying a meta-model's concepts according to the classification framework, annotated from [36]

system itself, the organizational internal context that is being observed. The “Complex capability” refers to the fact that a capability may contain other capabilities, which directly reflects a capability interrelation type; therefore, it is classified as a delivery concept, in comparison with “Capability.” For this reason, the “Capability” concept is classified as a concept related to the observation function, similar, to the performer, which refers to a human resource, another part of the organization system.

3.6 Development of capability meta-model concept map

The purpose of this part of the work is to provide an overview of the capability meta-model concepts collected in the previous steps. This includes both generic and function-specific maps. The sub-process is depicted as a BPMN diagram in Fig. 12.

Having extracted the concepts and developed a dataset using a reference manager, importing it into the VOSviewer tool [37] was the next step. The tool uses the VOS (visualization of similarities) method [38] and is useful in showing occurrences and co-occurrences of terms in the

meta-models, along with a grouping of concepts based on common co-occurrences. It is commonly used for the automatic visualization of bibliographic networks but can be valuable for visualizing any type of network. In this case, the task was performed manually; however, the rest of the task has been performed using the tool. Initially, a map was selected for creation based on bibliographic data. The analysis and counting method were selected, in particular, an analysis based on co-occurrence of keywords with full counting method, which means that all concepts have the same weight without taking into consideration the number of concepts per meta-model. The previously developed thesaurus file is also imported and the minimum number of concept occurrence is selected. Once the selected concepts are verified, an initial concept map is generated. Selecting the scale and weight, along with the label and line size and normalization method finalizes the concept map. The clustering is performed automatically by the tool.

In this study, we are focusing on the class elements because they can be subjected to thematic analysis [34] resulting in descriptive themes, since their nature is mainly descriptive and explicit. The association elements have also been thematically analyzed; however, the resulting themes

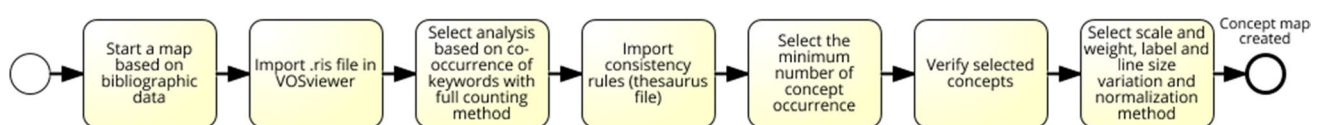


Fig. 12 The concept map creation sub-process

can be considered latent [34], representing underlying meanings, since the associations are naturally bridging first class concepts and cannot exist independently. Descriptive themes may be identified at the manifest level, which means that they are directly observable in the source and are used as a way to categorize information, while the latent themes involve interpretation [39]. The purpose of the development of the concept maps was to provide a descriptive overview of the concepts; therefore, the concepts derived from classes are included, but the association concepts are excluded, in order to maintain the descriptive nature of the maps, and avoid possible confusion between existing and implied concepts.

3.7 Research validity

This study does not fall in the category of systematic literature review or mapping study; however, it fulfills the main purposes of both research types, which, having explored a given domain, synthesize the information and structure the identified information, respectively [40]. Therefore, the reliability and validity of this study can be addressed through the lens of mapping studies' validity evaluation guidelines [41]. According to these guidelines, five types of validity need to be taken into consideration, in particular:

- Descriptive validity

Descriptive validity concerns the accurate communication of the observed data from the primary sources, especially regarding the collection of empirical data. This study has not involved any empirical data collection; therefore, there is no threat for this type of validity.

- Theoretical validity

This is the ability of the study to capture what is intended to be captured. It concerns threats existing in two phases of the research, in particular, while selecting the studies and while extracting data. In the case of this study, both the selection of the studies and the extraction of data have been performed systematically, and concerned selecting studies based on clear criteria and extracting the concepts that were included in meta-models with no exception. Therefore, bias that could pose any threat to the validity of this study has been taken into consideration.

- Generalizability

It concerns the ability of the results to be applicable in other contexts. The results of the study are generalizable for studies with similar goals, especially considering that all vast majority of existing capability meta-models have

been included; however, different goals may require different approaches.

- Interpretive validity

This type of validity refers to possible bias while interpreting the data to produce conclusions. In this study, the interpretation has been performed and cross-checked by all three co-authors to reduce any possible individual bias.

- Repeatability

This type of validity requires detailed reporting of the research process. In this study, the extensive and detailed method section, including process diagrams, aims to reduce any possible threats to validity.

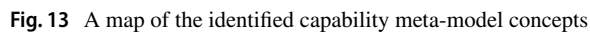
4 Results

This section consists of the findings derived from exploring the capability meta-models as found in the literature. The majority of the meta-models address a generic business or system domain, depending on their scope; however, there were a few exceptions where the meta-model was developed as a tool for domain-specific design and analysis, for example, in the domains of transportation [42], strategic sourcing [43] or the military [44].

4.1 Visualization of capability meta-model concepts

The interconnections among the concepts have been visualized using VOSviewer [37]. A total of 707 unique concepts were identified in the 64 sources. All concepts with at least two occurrences are included to avoid domain-specific terms that have only been used once. This does not exclude domain-specific terms that have been used in different versions of domain-specific meta-models. All 153 concepts with at least two occurrences have been included. Figure 13 depicts the visualized result. The proportional size of the circular elements and their labels reflect their number of occurrences across all models, the distance among the elements reflects their relatedness, and the thickness of the connecting lines reflects the strength of the link between the two elements, in other words, the frequency of co-occurrence in the meta-models [37].

Filtering out the majority of the included concepts by setting the occurrence limit to seven occurrences results in an abstracted version of the visual network that facilitates identifying the most common concepts encountered in capability meta-models. The abstract version is shown in Fig. 14.



The diagram is a network graph with 'capability' as the central hub. It is surrounded by several clusters of related concepts:

- Blue Cluster (Left):** Includes 'context', 'value', 'actor', and 'service'. These are interconnected with each other and with 'capability'.
- Green Cluster (Top Right):** Includes 'goal', 'constraint', 'kpi', 'process', and 'driver'. These are interconnected with each other and with 'capability'.
- Red Cluster (Bottom Right):** Includes 'requirement', 'organization', 'role', and 'activity'. These are interconnected with each other and with 'capability'.
- Purple Cluster (Bottom Left):** Includes 'resource'.

The edges between nodes are colored to match their respective clusters, showing the flow of relationships between the central 'capability' and these various domains.

- Goal
- Resource

- Process
- Service
- Organization
- Role
- Driver

- Activity
- Value
- Actor
- Requirement
- KPI
- Constraint
- Context

It is notable that the majority of the commonly encountered concepts are also associated to the adaptability of capabilities matching the examples in the framework or being associated to the explored information elements. For example, activity is a delivery element and requirement is a criterion element.

Apart from the identification of the most popular concepts related to capabilities in the existing meta-models, the abstracted version of the concept map provides an opportunity to present the value of a concept map and the information it includes. All the concept maps in this paper bear the same characteristics; however, the size of the datasets that motivate the generation of the concept maps is restrictive toward a detailed presentation. For example, while the abstracted concept map of Fig. 14 includes 15 concepts, which results in 105 link strength values, whose presentation is feasible within this paper, the concept map in Fig. 13 consists of 11,628 link strength values, a fact which indicates that the analysis of that map is not feasible within this paper. However, the map can provide certain insight on these concept relationships. Every line connecting a pair of concepts in the map indicates a link between them, in particular, it depicts the fact that these two concepts coexist in at least one meta-model, for example, the concepts of service and context in Fig. 14. As a result, the absence of a line indicates that the two concepts do not co-occur in any meta-model, as for example, the missing link between the concepts activity and KPI. Additionally, the thickness of the line connecting two concepts indicates how many co-occurrences of the pair exist in the dataset. This is named link strength or association strength [37]. A thicker line depicts a higher number of co-occurrences. For example, in Fig. 14, the concept of capability is connected to the concept of resource with a significantly thicker line than its connection to value. The concepts are visualized as circles which are called nodes [45]. Their size indicates the number of occurrences of the concept in the dataset. For example, in the abstracted concept map of Fig. 14, the nodes associated to the concepts of resource or goal are bigger than the ones associated to the concepts of constraint or role, which means that the concepts have been encountered in more meta-models. The colors of the nodes and the lines indicate the cluster they belong too. The colors bear no semantic meaning and are assigned randomly. The clustering is performed automatically by the VOSviewer tool using an implemented clustering method

[46]. For non-bibliometric small scale concept maps like the abstracted version in Fig. 14, the value of the clusters is limited; however, it still provides an indication of possible relevant concepts. For example, in the given concept map, the concepts of capability, resource, requirement, activity, organization and role belong to one cluster, goal, process, constraint, driver and KPI to a second one, and service, actor, value and context to a third one. The concept of capability, being the focal point of this analysis, should belong to all clusters; however, this is not possible in the tool. The first cluster seems to encompass concepts related to the organizational infrastructure aspect of capabilities, while the second one indicates their goal fulfilling aspect. Finally, the last cluster seems to encompass concepts related to a value-delivering aspect of capabilities. In any case, these are mere indications and, in addition, automated clustering results should be subjected to expert validation [46].

Table 3 includes the dataset of all the associations between the concepts, and their occurrences, which motivated the generation of the concept map of Fig. 14.

The same type of analysis can be performed for all the concept maps; however, the size of the respective datasets would render the resulting table impossible to present within this paper.

4.2 Capability meta-model development context

The content analysis facilitated the exploration of the development context of the capability meta-models. The results of the analysis uncover how the included set of meta-models is related to a dynamic context through the perspectives of the meta-model developers, as stated explicitly or implicitly in the publications. The results indicate a strong relation between capability modeling and the need to address a dynamic context. The vast majority of the publications include statements about capabilities existing in changing environments and their meta-models aiming to depict this condition. Statements about dynamic contexts have been more common than about static contexts. 54 out of 64 publications include explicit dynamic context statements, and 9 include implicit dynamic statements. Fourteen of these publications also have explicit statements about static context, which means that 40 publications mentioned dynamic but no static context. In addition, one publication has no explicit statements about either context type; in other words, there were no publications with exclusively static context statements. Table 4 depicts the distribution of the publications using the categorization matrix that drove the content analysis.

It is not feasible to report the analysis in full extend in this article; therefore, a fragment of the analysis follows, including representative examples of the capability meta-model development context, according to the predefined categories.

Table 3 All the associations between the concepts, and their occurrences

Link strength between concepts		Total link strength	Links	Occurrences
Capability		140	14	57
9	Activity	22	10	9
11	2	41	9	12
19	4	67	13	21
21	1	88	14	23
9	1	27	10	9
7	1	35	13	8
8	1	35	12	8
11	1	45	13	12
8	1	32	10	9
8	0	31	8	8
7	0	31	9	8
5	0	26	7	7
10	1	39	12	11
7	0	29	10	7
			Constraint	
			Organization	0
			Context	0
			Value	0
			KPI	0
			Driver	3
			Requirement	2
			Process	2
			Actor	2
			Role	3
			Goal	7
			Resource	5
			Service	6

Table 4 The distribution of meta-models based on the results of the content analysis

	Dynamic (exclusively)	Dynamic or static	Not specified
<i>What is the development context of capability meta-models?</i>			
Explicit	40 (~62.5%)	14 (~22%)	0
Implicit	9 (~14%)	1 (~1.5%)	0

4.2.1 Dynamic context

The dynamic context existing in the publications has been identified via a diverse set of keywords and phrases referring to it. For example, it can be concluded that the meta-model of Azevedo et al. [47] considers a dynamic context from their statement: “This approach is especially valuable to competitive and changing environments, which requires both planning and ability to adapt.”. The codes “change” and “adapt” are derived from this phrase which connect it to a dynamic context and result in the meta-model being categorized as developed for a dynamic context. The same applies in another example, where Loucopoulos and Kavakli [48] also make their perspective clear with the following statement: “...the work presented in this paper is based on the need for the design of services that meet the challenges of alignment, agility and sustainability in relation to dynamically changing enterprise requirements.”. The common codes that were derived from the entire set of publications and explicitly relate to a dynamic context are “change,” “adapt,” “modify,” “transform,” “adjust,” “dynamic environment,” “turbulent environment” and “transition.” In a few cases, the identification of the dynamic context used the meta-model concepts as a complementary source, as, for example, in [49], that includes the concept “Capability Transformation” and [50], which includes drivers, governance and motivation of change. In addition, in two enterprise architecture meta-models, information about the development context was acquired from their websites. In the case of MoDAF, the statement “The Ministry of Defence Architecture Framework (MODAF) is an internationally recognized enterprise architecture framework developed by the Ministry of Defence (MOD) to support defence planning and change management activities.”, exists only on the website [51] and not in the published specification.

The publications where the dynamic context was implicitly existing required a deeper analysis with periphrastic coding. For example, [52] has no explicit statements about a dynamic context; however, the TOGAF [25] standards are adopted and employed, which include an explicitly dynamic context. In another example, Walker [53] states that “Capabilities planning is intended to facilitate trade decisions across capabilities, across components, and between warfighting and enterprise needs,” which implies

an environmental dynamism and capability change without stating it explicitly.

Because of their explicit relation to change, the concepts of the meta-models of this category were included in the concept dataset. The papers that belong to this category are [13, 26, 42–44, 47, 49, 50, 54–85] (explicit) and [52, 53, 86–92] (implicit).

4.2.2 Dynamic or static context

Taking into consideration that there were no publications describing an exclusively static context, this category concerns the ones that describe both types of context. All the statements about static context were included as a means to acknowledge the two types and in many cases, compare them. The following excerpt from [93] is representative of this category: “The specification of a SOA is presented as a UML model and those models are generally considered to be static; however, any of SoaML constructs could just as well be constructed dynamically in response to changing conditions. Capabilities represent an abstraction of the ability to affect change.”. In another representative example from [36] “...it is necessary to incorporate static concepts [...] into dynamic concepts of system execution.”.

In the one publication [94] whose development context was categorized as “Implicit,” there were implications of a dynamic context; however, they were not deemed strong enough to be categorized as explicitly dynamic. In particular, it is stated that “Capability maximum can be modified by changes the numbers of resources, [...] values of attributes of resources and by changing the process.”, which is relevant to change but it is unclear, whether a dynamic context is explicitly considered or it is a simple mention to changing attributes.

Being relevant to change, the publications and meta-models assigned to this category have also been included in the concept dataset. This category’s publications are [1, 23–25, 27, 36, 48, 93, 95–100] (explicit) and [94] (implicit).

4.2.3 Not specified

No publications with no references to their meta-models’ development context were identified, which can be considered well in line with the principles for academic and technical publications in this field.

4.2.4 A fragment of the analysis table

Due to the lack of space, it is not feasible to present the entire analysis table; therefore, Table 5 shows an illustrative sample of the analysis.

4.3 Meta-model statistics

The meta-models that have a business scope have their classes named using exclusively business terms, for example, “value” [67], “opportunity” [99] and “resource” [47]. The ones that have an IT scope consist exclusively of classes named after IT terms, like “attribute value” [61], “binary value” [56] and “service interface” [93]. The third type of meta-model based on their scope is the one that includes both business and IT terms. 48 out of 64 meta-models (75%) have a business scope, 4 of them (6.25%) have an IT scope, and 12 (18.75%) have a hybrid business and IT scope. This distribution is illustrated in a pie chart in Fig. 15.

The publications from which the meta-models have been extracted range from 2005 to 2018. The bar chart of Fig. 16 depicts the number of meta-models per year and per scope.

Regarding the number of concepts per meta-model, it has been calculated using only the number of concepts that existed as classes. There has been significant diversity in the results. In particular, the lowest number of concepts in a meta-model was 5 and the highest was 51. Overall, the average number of concepts in the entire set of meta-models is approximately 17.48, the median is 14, and the standard deviation is approximately 10.23. The concept number range is 5–51.

The vast majority of the meta-models have a business scope, their average number of concepts in is approximately 18.13, the median is 14.5, and the standard deviation is approximately 10.78. In the business and IT scope set of meta-models, the average concept count is approximately 17.33, the median is 15, and the standard deviation is 8.06. The concept number ranges are 6–51 and 5–34, respectively. The results of these two subsets do not differ much from each other. However, when it comes to the minority of the meta-models, the ones with an IT scope, the results are different. The average concept count in this subset is 10.25; the median is 8.5 with a standard deviation of approximately 5.54. The concept number ranges from 5 to 19.

In order to illustrate the distribution of capability meta-models based on the number of included concepts, a box and whiskers graph with inclusive median calculation of quartiles and outlier points has been created to facilitate visual comparison. The result is presented in Fig. 17. The means is marked with an “x” in the boxes, and the line depicts the median.

4.4 Function-specific findings

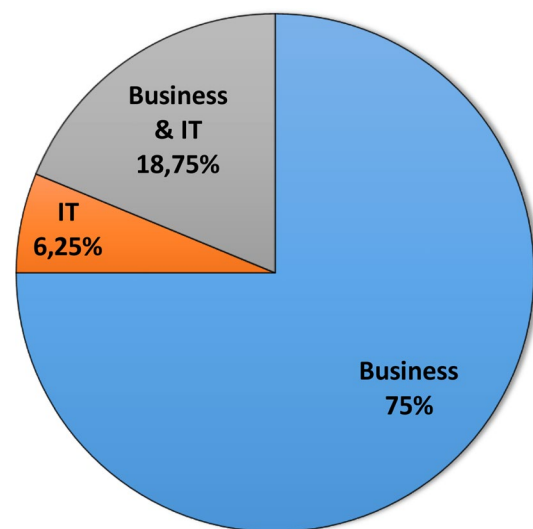
The initial classification of concepts enabled the potential to identify the distribution of function-specific concepts in the meta-models. As a result, a classification of the meta-models was possible, based on which types of change functions they have implemented. In particular, the majority of

Table 5 A fragment of the analysis table developed during the content analysis

Excerpt	Codes	Category
[54] “This study seeks to analyze, design and implement a software prototype which integrates business architecture capabilities and design models to facilitate change impact analysis.”	Change	Dynamic (explicit)
[57] “A capability represents a manageable unit of change...”	Change	Dynamic (explicit)
[71] “...it will enable more effective management of complex and multi-faceted transformations needed to keep up with rapid change...”	Transformation, Change	Dynamic (explicit)
[96] “Capability-oriented enterprise modeling can provide an effective and promising solution to face well-known problems in changing environments.” “...Capability requires Process [...] for modeling the static aspects of the business...”	Change static	Dynamic static (Explicit)
[72] “In order to survive, organization should engage on search operations which involve the evaluation of the current situation and changes to the organizational capabilities, if needed.”	Change	Dynamic (explicit)
[75] “Two themes stand out as reasons for using capabilities connected to other elements: describing a company, and govern changes.”	Change	Dynamic (explicit)
[93] “The specification of a SOA is presented as a UML model and those models are generally considered to be static, however any of SoaML constructs could just as well be constructed dynamically in response to changing conditions. Capabilities represent an abstraction of the ability to affect change...”	Static change	Dynamic static (Explicit)
[90] “Capability descriptions [...] are split into information transformation and state of the world change...” “...information transformation and state of the world changes define the capability...”	Change transformation	Dynamic (implicit)
[80] “Based on a maturity assessment, an organization can target a <i>capability increment</i> to improve its capabilities and their maturities by undergoing a change initiative to increase performance for a particular capability.”	Improvement change	Dynamic (explicit)
[83] “dynamically adjusts role and permission assignments based on contextual information” “the model addresses the issue of flexible delegation of capabilities”	Adjustment flexibility	Dynamic (explicit)

the analyzed capability meta-models included at least one concept of each change function type. More specifically, 42 out of 64 meta-models (65.6%) included concepts associated to all three functions, observation, decision and delivery. Concepts associated only to observation and decision existed in 7 meta-models (10.9%), while decision and delivery only existed in 6 meta-models (9.4%) and observation and delivery in 4 of the meta-models (6.3%). There were also meta-models where only one of the change functions was identified through associated concepts. Regarding meta-models that depicted decision only and delivery only, there were 2 of each type (3.1% each) and 1 (1.6%) that included only observation concepts. Figure 18 depicts a Venn diagram that represents this classification.

In addition, Chi-square tests [101] were conducted to identify whether significant associations existed among the three change functions' concepts' existence in meta-models. In other words, the tests were performed to test the hypothesis that the existence of one type of concepts in a meta-model was significantly associated to the existence of any other type. For example, including decision concepts may have been associated to the inclusion of observation concepts. However, the tests that were conducted to test every

**Fig. 15** The distribution of meta-models based on their scope

possible correlation proved that no significant associations exist.

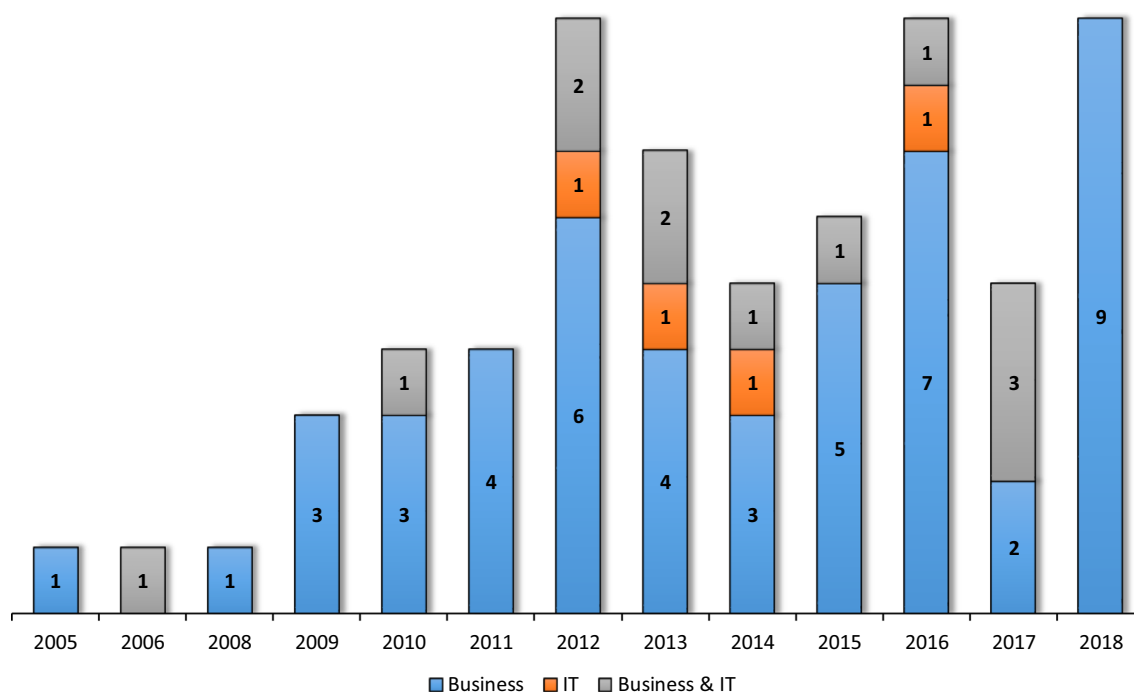


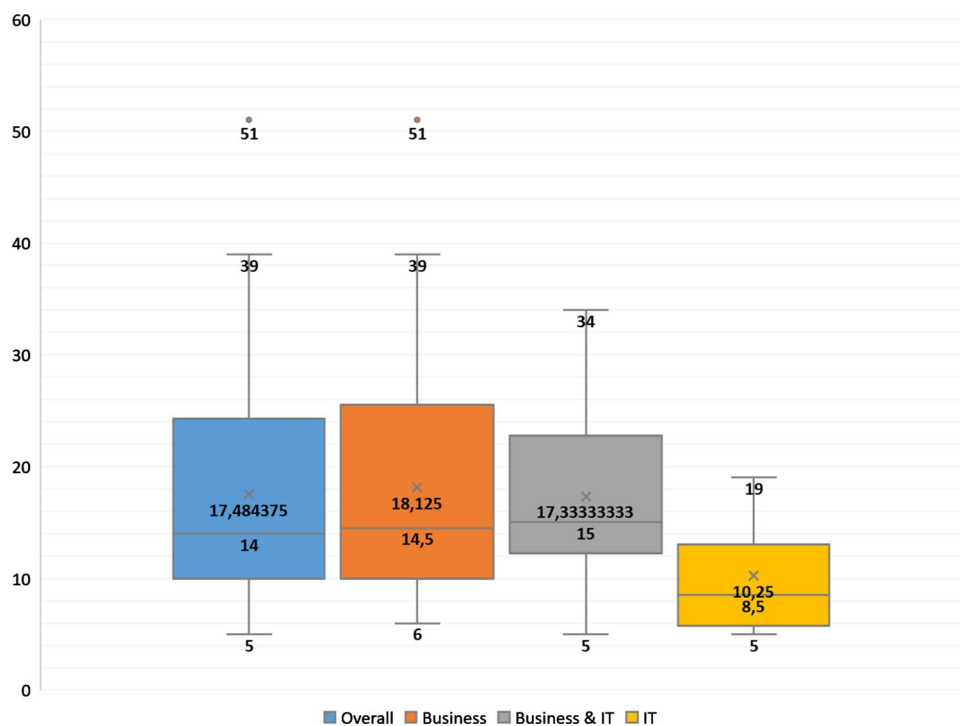
Fig. 16 Capability meta-model distribution per scope and year of publication

4.4.1 Findings for observation

The majority of the meta-models, in particular, 54 out of 64, included at least one concept associated to observation.

There have been several similarities and differences identified among the models during the analysis. Context seems to be one of the more common concepts included in a capability meta-model. However, various terms have been used to describe it, sometimes using similar terms, for

Fig. 17 A box and whiskers graph depicting the distribution of concept count in capability meta-models, overall and per scope



example, context, environment or ecosystem, or by specializing the included model concepts like environmental factor [76], situation [95], location [72], service context [64] or context set [99]. The same trend has been observed concerning measurement and metrics. Measurement [54], calculation [96], computation action [88] and assessment [73] are examples of similar concepts that refer to the activity of measuring a capability. There are also specializations of the concept, e.g., capability calculation [48] and scorecard [65]. An interesting fact concerning measurement is that many of the meta-models have selected to include the concept not as a class item, but as an association between classes. The textual descriptions of the associations are also diverse. A few examples of measurement-related associations are “used to evaluate” [57], “measures” [58], “measured by” [1] and “assessed by” [80]. The metric, as a standard of measurement, also exists in various similar forms. The most common terms that exist in the meta-models in order to address metrics are metric [42, 43, 70, 98] and indicator [1], especially defined as key performance indicator (KPI) [49, 52, 57, 75, 80, 99]. Several similar or specialized forms exist as well, for example, business KPI, asset KPI [49] and assessment metric [50]. Some models use the terms value and property as a way to express metrics, like capability value [56], attribute value [61] capability property [63] and property value [91].

Apart from the examples provided above, the complete set of class concepts associated to observation is listed in Table 6 along with the number of occurrences in the meta-model. Forty-nine concepts have been included in this set. Regarding the details of how they are associated to each other in terms of co-occurrence, relatedness and clustering an observation-specific concept map has been developed as shown in Fig. 19.

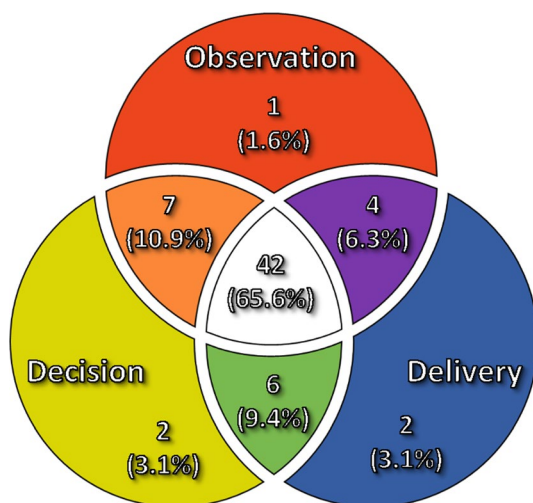


Fig. 18 The distribution of meta-models based on the included function-specific concepts

4.4.2 Findings for decision

Led by the popularity of the concepts depicting intention, this set of concepts concerning decision has been the most popular one. Only seven out of the 64 meta-models have not included any decision concepts.

A capability is meant to fulfill a purpose. Hence, the intention concepts are the most popular theme among the ones explored in this study. There have been several similar concepts to depict intention with the most popular being goal as, for example in [82, 94, 102], followed by driver, as in [54, 55, 57, 74], objective, as in [78] and requirement [42]. Other variations that have been used to depict intention are business goal [87], capability goals [53], task and mission goal [36], mission [79], desired effect [86], need [85] and enterprise vision as in [26]. There also exist specific intention concepts like competitive advantage and superior profit [103]. Finally, there are associations that imply intentionality, e.g., the association “accomplishes” [97].

Capability alternatives have been modeled by various means. The most self-explanatory terms used are current and planned capability [53], current and desired capability [49], and emerging capability [69]. Less intuitively named concepts that, however, imply the existence of capability alternatives are process variant [1], alternative relation [13], capability enabling bundle [47], the attribute “potential” in a capability class [59], along with the concepts configuration [43] and capability configuration [76], which imply the existence of multiple configurations for a capability.

The criteria for decisions on capability adaptations or adjustments have been modeled with a diversity of terms as well. Any concept that guides or affects a decision can belong to this set. A few examples that exist in the meta-models are condition and rule [86], plan [92], capability planning [44], policy [60], course of action [24], guidance [72], capability roadmap [50], capability offer [28] or even more specialized like contextual constraints [83].

Just like the observation-specific concept map that was presented in the previous section, a decision-specific concept map has been developed too, in order to depict all the information regarding the relations among the decision concepts derived from meta-model classes. The concept map is presented in Fig. 20. Complementing the examples discussed above, the complete set of decision concepts identified through the meta-model analysis is included in Table 7. The number of concepts included in this set is 38.

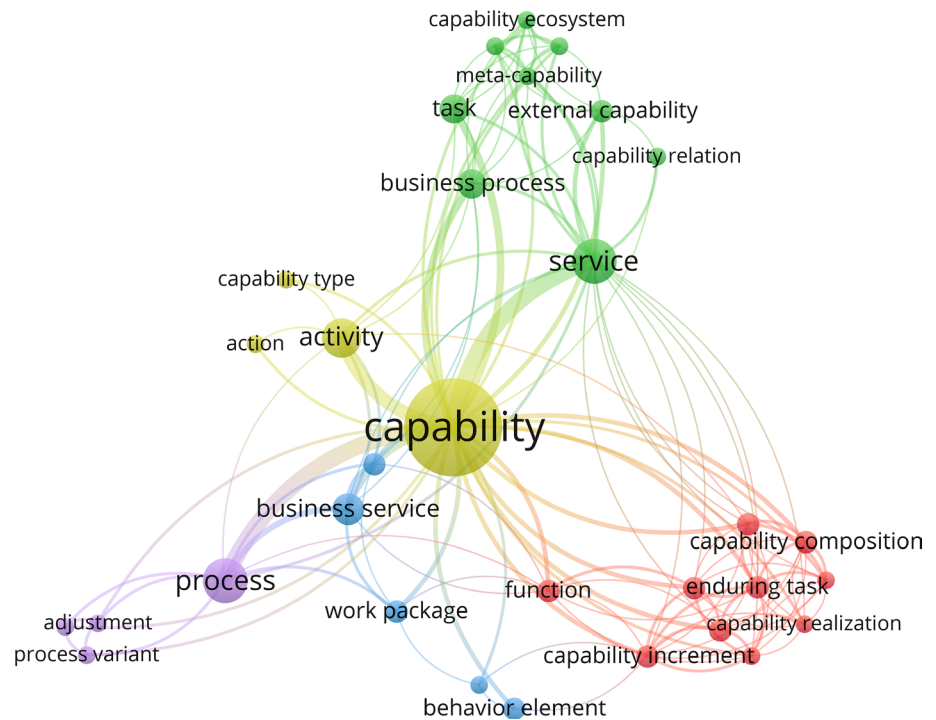
4.4.3 Findings for delivery

This set of concepts has also been widely observed due to the inclusion of realization concepts. Only ten among the 64 selected meta-models did not contain any realization, adjustment or capability architecture concepts.

Table 6 The complete set of observation class concepts identified, along with their associated number of occurrences in the meta-models

Capability	57	Internal capability	3	Enumeration value	2
Resource	21	Measurable property	3	Environment	2
Organization	11	Structure element	3	Location	2
Role	9	Ability	2	Location type	2
Actor	8	Actual organization	2	Operational entity	2
KPI	8	Actual organization resource	2	Organizational capability	2
Context	7	Assessment	2	Organizational unit	2
Measure	6	Business item	2	Owner	2
Metric	5	Calculation	2	People	2
Performer	5	Capability context	2	Person	2
System	5	Capacity	2	Person type	2
Asset	4	Context indicator	2	Range value	2
Business Capability	3	Context type	2	Resource type	2
Context element	3	Dynamic value	2	Situation	2
Context element range	3	Ecosystem	2	Skillset	2
Context set	3	Enterprise	2	Software	2
Indicator	3				

Fig. 19 The observation-specific concept map



In the vast majority of capability definitions, the concept is considered associated to the delivery of value. Therefore, a capability that changes is most likely to affect its delivery as well. Examples of the most common concepts that exist in meta-models and reflect capability realization are the concepts of process, as in [77], business process [55], service [67, 68], business service [25], behavior element [47], capability realization [93], task [89], action [62] and activity [27]. There have also been associations depicting delivery for example “delivers” [80] and “realizes” [84].

Activity modifier [100] and adjustment [24] are interesting concepts, since they refer specifically to the modification of the capability.

Several approaches have been selected as well to address capability architecture in the meta-models. The concept refers to capability interrelationships and interactions. First of all, distinguished classes have been used, for example, capability composition [78], capability dependency [23] or capability relation [13]. In addition, several associations have been used to depict capability relationships, recursive

associations of capabilities being the most common case observed. In particular, “has” [55], “realizes” [47], “interacts with” [59], “specifies” and “extends” [90], aggregation [62], “can be combined with,” “is decomposed into” and “used by” [70], “contains” [26] and “is part of” [87] are only a few examples of recursive associations. Finally, specializations and generalizations have been used to depict capability interrelations. In particular, examples of specializations of capability are external capability [13], simple and complex capability [36], business capability and technological capability [70]. These concepts express relevant capabilities or their types, in association to the main capability change phenomenon; therefore, they cannot be classified as system itself (observation) but are only taken into consideration through the impact on them and their associations, which is generated by the changing capability. Meta-capability connected

to capability with an association “alters/designs” [64] is another way selected to depict capability interrelationships.

A delivery-specific concept map has been developed as well in order to depict how the concepts associated to this function are related to each other in the analyzed meta-models from the literature. It is presented in Fig. 21. The number of included concepts is 31, and they are all presented in Table 8.

5 Discussion

In order to facilitate the development of a modeling method for capability adaptation, we have opted toward the analysis of the concepts comprising a meta-model that have the potential to be included in a future meta-model used in the

Fig. 20 The decision-specific concept map

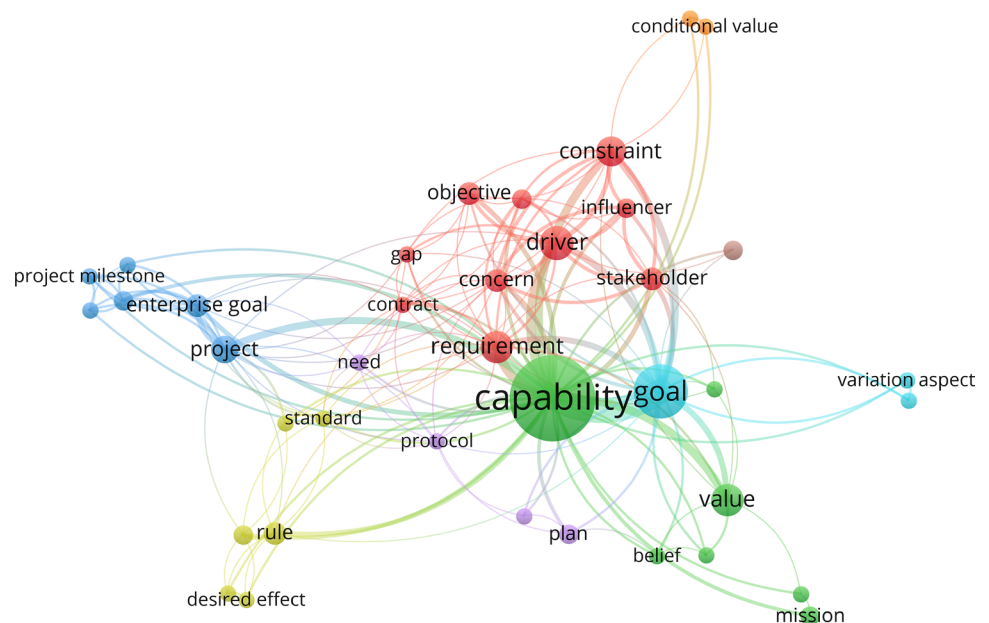


Table 7 The complete set of decision concepts identified, along with their associated number of occurrences in the meta-model classes

Goal	23	Course of action	3	Gap	2
Driver	9	Enterprise vision	3	Need	2
Requirement	8	Influencer	3	Organizational norm	2
Value	8	Mission	3	Policy	2
Constraint	7	Plan	3	Project milestone	2
Project	6	Belief	2	Protocol	2
Concern	4	Competitive advantage	2	Result	2
Enterprise goal	4	Conditional value	2	Standard	2
Objective	4	Configuration no longer used	2	Strategy	2
Rule	4	Constrained value	2	Variation aspect	2
Stakeholder	4	Contract	2	Variation point	2
Business goal	3	Desired effect	2	Vision statement	2
Condition	3	Efficiency	2		

method instead of analyzing the meta-models holistically. The context in which the components of a meta-model have originated from remains significant; therefore, before developing the concept dataset, a content analysis was performed on the papers to identify if the developers of the meta-models considered a dynamic or static context for their artifacts. The vast majority of the meta-models' development context are relevant to a dynamic context. This finding indicates not only the suitability of the meta-model concepts for the change-related analysis and classification, but also a strong association between the concepts of capability and change, as existing in the literature. In addition, the meta-model scope and domain specification attributes have been taken into consideration.

The exploration of the diversity of the existing capability meta-models has provided valuable insights into modeling of capability adaptability. The exploration has also provided the opportunity to identify specific areas of improvement. In this section, we examine the implications of the findings concerning the meta-models based on their diversity and the potential to use them for future improvements when it comes to modeling of capability adaptability.

5.1 Capability meta-model diversity

The diversity of the surveyed meta-models has been addressed using several different factors and a number of findings have been derived, respectively.

5.1.1 Diversity by scope and number of concepts

Regarding the distribution of models based on the number of concepts and their scope, it is notable that the business and business combined with IT scopes are similar. However, the IT scope set of meta-models has a significantly lower number of concepts. This can be explained by the fact that meta-models with IT scope have been extracted from publications whose nature can be considered more technical and they usually aim for a focused solution of specific problems. Therefore, a smaller number of precise concepts of technical nature that are relevant to the problem being addressed is probably considered a more efficient way of modeling capabilities with an IT scope. On the contrary, the business and hybrid meta-models include more concepts related to broader perspectives of the capability and the organization that owns it. This makes sense if the fact that a capability, by definition, encompasses a wide spectrum of core business concepts, is taken into consideration. Our conclusion is that the model detail needs to fit into the domain where it is applied, and currently, the IT design domain apparently is in favor of narrower in scope but more detailed models.

5.1.2 Diversity by change function inclusion

The analysis of the meta-models based on change functions (observation, decision, delivery) resulted in several findings. First of all, the differences of the function-specific sets, in

Fig. 21 The delivery-specific concept map

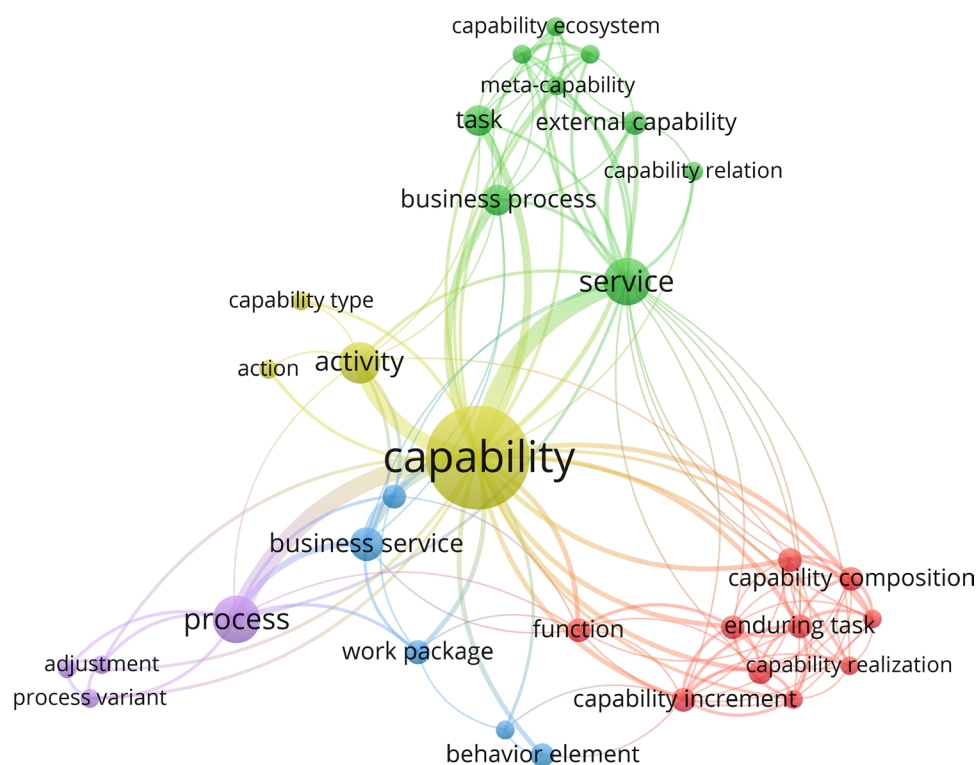


Table 8 The complete set of delivery concepts identified, along with their associated number of occurrences in the meta-models

Process	12	External capability	3	Capability ecosystem	2
Service	12	Function	3	Capability realization	2
Activity	9	Information system service	3	Capability relation	2
Business service	6	Operational activity	3	Capability specialization	2
Business process	5	Resource interaction	3	Capability type	2
Task	5	Work package	3	Meta-capability	2
Behavior element	3	Action	2	Plateau	2
Capability composition	3	Adjustment	2	Process variant	2
Capability dependency	3	Capability configuration	2	Service context	2
Capability increment	3	Capability delivery pattern	2	Service ecosystem	2
Enduring task	3				

terms of quantity of included concepts, were not vast, yet they cannot be ignored. The observation-specific set was the largest and the delivery-specific set was the smallest; this indicates that the developers of capability meta-models to date have expressed a greater interest in modeling the observation function of capability change, somewhat less interest about the decisions concerning the change, and noticeably less about the delivery of the capability and the change. Our conclusion is that what remains to be explored as part of our future work is if the observation function is more demanding in terms of required concepts or if it is a mere coincidence and the functions should be balanced when it comes to the coverage in terms of concepts.

5.1.3 Diversity in domain-specific concepts

As far as the modeling of adaptive capabilities is concerned, there are areas in which improvements are deemed necessary. Despite the differences concerning the scope of the explored meta-models, being business, IT or both, or their domain, being general purpose or domain-specific (e.g., military or transportation), important similarities have been identified in the form of commonly used concepts. Even when comparing domain-specific models, there are major similarities despite the non-overlapping domains. This indicates that the utility of modeling capabilities is not bound to specific domains, and improving capability adaptability may have a major impact in several domains.

A fact that needs to be taken into consideration is that not all capability meta-models address adaptability in detail but this cannot be considered a deficiency. Domain-specific approaches serve their purpose by including the specifics of the given domain. Every model or meta-model is an artifact developed in order to address these specific purposes. However, a conclusion here is that adaptability is becoming the new constant in enterprise transformation, and hence, enterprise modeling methods, which are used to support organizations, should aim to support the transformation regardless of the application domain.

5.1.4 Diversity in concept semantics

During the analysis step of cross-checking the concept semantics throughout the papers in which they were introduced, no contradicting approaches have been identified. For example, a concept being associated to entirely different semantics in two meta-models. On the contrary, even not identical, yet similar, concepts are often associated to the same semantics. In addition, even though there have been numerous variations of similar concepts, the meta-models bear more similarities than differences. From the perspective of our classification framework, the meta-models bear differences on the concept level, but once abstraction is applied to the concepts and they are mapped to the information types, the meta-models all are more similar than one would expect on the outset. This argument is also supported by the fact that on the higher level of the framework, on the change function level, the majority of the meta-models address all three functions. In this regard, our conclusion is that it is reasonable that a meta-model used for expressing capability adaptability covers all of the three functions.

5.2 Toward improving capability modeling for adaptation

Several modeling practices have been deemed useful for improving modeling for adaptive capabilities. These will also serve as guidance for our future work.

5.2.1 Meta-model alignment

When creating a meta-model to support capability adaptability, the initial idea could be to compose a capability change meta-model by combining *all* the examined meta-models. However, this may not lead to an ideal model. The resulting meta-model may simply not possess the combined properties of the combined meta-models. From a logic perspective, the *composition fallacy*, which is based on the mistaken assumption that what is true for the parts

is also true for the whole that is composed of these parts [104], may be a useful principle guiding the process in order to avoid problematic generalizations. This applies also to cases where multiple models are combined as elements to form a single “mega-model” [105]. This approach of combining models has been suggested concerning capability meta-models, but then it was specifically referring to the observation function of the capability’s context [106]. Therefore, instead of concentrating efforts on creating a mega-model, alternative ways to address this need to be employed. In, particular, this study, through the identification and classification of change-related function-specific concepts that have been extracted from a plethora of existing meta-models, has contributed toward the alignment of modeling approaches. The fact that the meta-model concepts that were identified as the most common are significantly overlapping with the ones that address the adaptive nature of capabilities, means that the task of improving capability modeling for adaptations is an achievable task since a solid core set of concepts already exists. Our conclusion is that rather than creating a mega-model, it may be more useful to perform an alignment of meta-models merging all the relevant concepts. An alignment of capability concepts has been suggested before, due to the different levels of design details in the existing meta-models [3]; however, this study contributes by decomposing the meta-models and classifying the concepts as a preparatory step for alignment.

5.2.2 Reducing the level of abstraction

The analysis of the meta-models identified that abstract concepts exist in several of them. The associations between these concepts and the concept of capability are sometimes not entirely clear, which can be a possible source of confusion. For example, the concept Gap in [25, 74] could be an efficient way to identify the need for change adaptation. However, it needs to be decomposed before being valuable for capability adaptation. Our conclusion is that reducing the level of abstraction is one way toward improved support of modeling capability adaptation. For every abstract concept, the relevant concepts associating it to capabilities need to be identified, in order to clarify the relation, its purpose, and to ensure that no significant information is missing.

5.2.3 Including latent and missing concepts

A finding that can assist the development of a capability change modeling method is associated to a phenomenon whose occurrence in the meta-models is quite common—there are concepts that are not included as classes or even associations; however, their existence and relation are implied. This is also relevant to the analysis of association

concepts which, as mentioned earlier, can result in latent concepts with underlying meanings [39]. For example, “capability meets goal” implies a measurement concept. In another example, the association “accomplishes” implies the existence of a goal and associated measurements. Therefore, even if the focus of this study has been on class concepts, the presentation of the association concepts that we consider as a source of latent themes in our results should facilitate the identification of any implied concepts. From a meta-modeling perspective, any association and relationship are instantiations of a relationship class that exists in a meta-meta-model [35] and this needs to be taken into consideration before the development of a capability modeling method. Including any implied concepts is another way to optimize the meta-models toward the adaptation or self-adaptation of capabilities.

Finally, there are probably concepts that can provide valuable assistance in modeling adaptive capabilities that remained unidentified and not implied during the exploration of the existing meta-models; however, they should be included in a method for capability modeling. Identifying these concepts is another future step of this research project as a means to achieve optimization of capability modeling.

6 Conclusions

A wide variety of capability modeling approaches exists, and a variety of concepts related to transformation and adaptation are included as elements of their respective meta-models. The general aim of this article was to assist the development of a modeling method for capability-oriented systems and organizations by exploring the diversity of the existing capability meta-models and identify opportunities for improvements. A framework that facilitates the classification of change-related concepts in capability meta-models has been presented. The framework contains the main functionalities of an adaptive or self-adaptive capability-oriented system. Observation, decision and delivery are the main function elements. Each of these functional elements is further specified in a set of information elements. The framework has been used to analyze 64 capability meta-models. As a result, an overview of the concepts in the meta-models and their association to function elements has been investigated. This analysis contributes toward the identification of the concepts required for an improved modeling method that focuses on the dynamic adaptation of capabilities.

Funding Open access funding provided by Stockholm University.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing,

adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Sandkuhl, K., Stirna, J. (eds.): *Capability Management in Digital Enterprises*. Springer, Cham (2018). <https://doi.org/10.1007/978-3-319-90424-5>
- Morin, B., Barais, O., Jezequel, J.-M., Fleurey, F., Solberg, A.: Models@ run.time to support dynamic adaptation. *Computer* **42**, 44–51 (2009). <https://doi.org/10.1109/MC.2009.327>
- Zdravkovic, J., Stirna, J., Grabis, J.: A comparative analysis of using the capability notion for congruent business and information systems engineering. *Complex Syst. Inform. Model. Q.* (2017). <https://doi.org/10.7250/csinq.2017-10.01>
- Koutsopoulos, G., Henkel, M., Stirna, J.: Dynamic adaptation of capabilities: exploring meta-model diversity. In: Reinhartz-Berger, I., Zdravkovic, J., Gulden, J., Schmidt, R. (eds.) *Enterprise, Business-Process and Information Systems Modeling*, pp. 181–195. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-20618-5_13
- Grabis, J., Zdravkovic, J., Stirna, J.: Overview of capability-driven development methodology. In: Sandkuhl, K., Stirna, J. (eds.) *Capability Management in Digital Enterprises*, pp. 59–84. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-90424-5_4
- Teece, D.J., Pisano, G., Shuen, A.: Dynamic capabilities and strategic management. *Strateg. Manag. J.* **18**, 509–533 (1997). [https://doi.org/10.1002/\(SICI\)1097-0266\(199708\)18:7%3c509::AID-SMJ882%3e3.0.CO;2-Z](https://doi.org/10.1002/(SICI)1097-0266(199708)18:7%3c509::AID-SMJ882%3e3.0.CO;2-Z)
- Helfat, C.E., Finkelstein, S., Mitchel, W., Peteraf, M., Singh, H., Teece, D., Winter, S.G.: *Dynamic Capabilities: Understanding Strategic Change in Organizations*. Blackwell Pub., Malden (2007)
- Koutsopoulos, G.: Modeling organizational potentials using the dynamic nature of capabilities. In: Joint Proceedings of the BIR 2018 Short Papers, Workshops and Doctoral Consortium, pp. 387–398. CEUR-WS.org, Stockholm, Sweden (2018)
- Helfat, C.E., Winter, S.G.: Untangling dynamic and operational capabilities: strategy for the (n)ever-changing world. *Strateg. Manag. J.* **32**, 1243–1250 (2011). <https://doi.org/10.1002/smj.955>
- Koç, H.: Methods in designing and developing capabilities: a systematic mapping study. In: Ralyté, J., España, S., Pastor, Ó. (eds.) *The Practice of Enterprise Modeling*, pp. 209–222. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25897-3_14
- Koç, H., Hennig, E., Jastram, S., Starke, C.: State of the art in context modelling—a systematic literature review. In: Iliadis, L., Papazoglou, M., Pohl, K. (eds.) *Advanced Information Systems Engineering Workshops*, pp. 53–64. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07869-4_5
- Zdravkovic, J., Stirna, J., Grabis, J.: Capability consideration in business and enterprise architecture frameworks. In: Sandkuhl, K., Stirna, J. (eds.) *Capability Management in Digital Enterprises*, pp. 41–56. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-90424-5_3
- Loucopoulos, P., Kavakli, E.: Capability modeling with application on large-scale sports events. In: *AMCIS 2016 Proceedings*, pp. 1–10. Association for Information Systems, San Diego (2016)
- Grabis, J., Kampars, J.: Design of capability delivery adjustments. In: Krogstie, J., Mouratidis, H., Su, J. (eds.) *Advanced Information Systems Engineering Workshops*, pp. 52–62. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39564-7_5
- Caesar, B., Nieke, M., Köcher, A., Hildebrandt, C., Seidl, C., Fay, A., Schaefer, I.: Context-sensitive reconfiguration of collaborative manufacturing systems. In: *MIM 2019 Post-Proceedings*. Elsevier, Berlin (2019)
- Loucopoulos, P., Kavakli, E., Chechina, N.: Requirements engineering for cyber physical production systems. In: Giorgini, P., Weber, B. (eds.) *Advanced Information Systems Engineering*, pp. 276–291. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21290-2_18
- Petrevska Nechkoska, R., Poels, G., Zdravkovic, J.: Enterprise adaptability using a capability-oriented methodology and tool support. In: *Proceedings of the 2nd International Workshop on Practicing Open Enterprise Modelling Within OMILAB (PrOse) Co-located with 11th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modelling (PoEM 2018)*, Vienna, Austria, October 31, 2018, pp. 61–72 (2018)
- Weyns, D., Malek, S., Andersson, J.: FORMS: unifying reference model for formal specification of distributed self-adaptive systems. *ACM Trans. Auton. Adapt. Syst.* **7**, 1–61 (2012). <https://doi.org/10.1145/2168260.2168268>
- Morandini, M., Penserini, L., Perini, A.: Towards goal-oriented development of self-adaptive systems. In: *Proceedings of the 2008 International Workshop on Software Engineering for Adaptive and Self-Managing Systems—SEAMS'08*, p. 9. ACM Press, Leipzig (2008). <https://doi.org/10.1145/1370018.1370021>
- Hevner, A., Chatterjee, S.: *Design Research in Information Systems*. Springer US, Boston (2010). <https://doi.org/10.1007/978-1-4419-5653-8>
- Johannesson, P., Perjons, E.: *An Introduction to Design Science*. Springer, Cham (2014). <https://doi.org/10.1007/978-3-319-10632-8>
- Object Management Group (OMG): *Business Process Model and Notation* (2011)
- NATO: *NATO Architecture Framework v.4* (2018). https://www.nato.int/nato_static_fl2014/assets/pdf/pdf_2018_08/201801_180801-ac322-d_2018_0002_naf_final.pdf
- The Open Group: *Archimate 3.0.1. Specification* (2017). <https://publications.opengroup.org/i162>
- The Open Group: *The TOGAF® Standard, Version 9.2* (2018). <https://publications.opengroup.org/standards/togaf/specifications/c182>
- UK Ministry of Defence: *Ministry of Defence Architecture Framework V1.2.004* (2010). <https://www.gov.uk/guidance/mod-architecture-framework>
- USA Department of Defense: *Department of Defense Architecture Framework 2.02* (2009). <https://dodcio.defense.gov/Library/DoD-Architecture-Framework/>
- Object Management Group (OMG): *Value Delivery Modeling Language* (2015). <https://www.omg.org/spec/VDML/1.0>
- Beimborn, D., Martin, S.F., Homann, U.: Capability-oriented modeling of the firm. Presented at the IPSI Conference, Amalfi, Italy January (2005)

30. Elo, S., Kyngäs, H.: The qualitative content analysis process. *J. Adv. Nurs.* **62**, 107–115 (2008). <https://doi.org/10.1111/j.1365-2648.2007.04569.x>
31. Vaismoradi, M., Turunen, H., Bondas, T.: Content analysis and thematic analysis: implications for conducting a qualitative descriptive study: qualitative descriptive study. *Nurs. Health Sci.* **15**, 398–405 (2013). <https://doi.org/10.1111/nhs.12048>
32. Grabis, J., Kampars, J.: Adjustment of capabilities: how to add dynamics. In: Sandkuhl, K., Stirna, J. (eds.) *Capability Management in Digital Enterprises*, pp. 139–158. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-90424-5_8
33. Boyd, J.R.: The essence of winning and losing. Unpublished Lecture Notes **12**, 123–125 (1996)
34. Braun, V., Clarke, V.: Using thematic analysis in psychology. *Qual. Res. Psychol.* **3**, 77–101 (2006). <https://doi.org/10.1191/1478088706qp063oa>
35. Bork, D., Karagiannis, D., Pittl, B.: A survey of modeling language specification techniques. *Inf. Syst.* **87**, 101425 (2020). <https://doi.org/10.1016/j.is.2019.101425>
36. Qi, Y., Wang, Z., Dong, Q., He, H.: Modeling and verifying SoS performance requirements of C4ISR systems. *J. Syst. Eng. Electron.* **26**, 754–763 (2015)
37. van Eck, N.J., Waltman, L.: Software survey: VOSviewer, a computer program for bibliometric mapping. *Scientometrics* **84**, 523–538 (2010). <https://doi.org/10.1007/s11192-009-0146-3>
38. van Eck, N.J., Waltman, L.: VOS: a new method for visualizing similarities between objects. In: Decker, R., Lenz, H.-J. (eds.) *Advances in Data Analysis*, pp. 299–306. Springer, Berlin (2007). https://doi.org/10.1007/978-3-540-70981-7_34
39. Saldaña, J.: *The Coding Manual for Qualitative Researchers*. Sage, Los Angeles (2009)
40. Petersen, K., Vakkalanka, S., Kuzniarz, L.: Guidelines for conducting systematic mapping studies in software engineering: an update. *Inf. Softw. Technol.* **64**, 1–18 (2015). <https://doi.org/10.1016/j.infsof.2015.03.007>
41. Petersen, K., Gencel, C.: Worldviews, research methods, and their relationship to validity in empirical software engineering research. In: 2013 Joint Conference of the 23rd International Workshop on Software Measurement and the 8th International Conference on Software Process and Product Measurement, pp. 81–89. IEEE, Ankara, Turkey (2013). <https://doi.org/10.1109/IWSM-Mensura.2013.22>
42. Plum, N.: TRAK Enterprise Architecture Metamodel (2018). <https://sourceforge.net/projects/trakmetamodel/>
43. Rafati, L., Roelens, B., Poels, G.: A domain-specific modeling technique for value-driven strategic sourcing. *Enterp. Model. Inf. Syst. Archit.* **13**, 1–29 (2018). <https://doi.org/10.18417/emisa.13.8>
44. Anteroine, J.: The holistic military capability life cycle model. In: 2012 7th International Conference on System of Systems Engineering (SoSE), pp. 167–172. IEEE, Genova (2012). <https://doi.org/10.1109/SYSoSE.2012.6384199>
45. van Eck, N.J., Waltman, L.: Visualizing bibliometric networks. In: Ding, Y., Rousseau, R., Wolfram, D. (eds.) *Measuring Scholarly Impact*, pp. 285–320. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10377-8_13
46. Waltman, L., van Eck, N.J., Noyons, E.C.M.: A unified approach to mapping and clustering of bibliometric networks. *J. Inform.* **4**, 629–635 (2010). <https://doi.org/10.1016/j.joi.2010.07.002>
47. Azevedo, C.L.B., Iacob, M.-E., Almeida, J.P.A., van Sinderen, M., Pires, L.F., Guizzardi, G.: Modeling resources and capabilities in enterprise architecture: a well-founded ontology-based proposal for ArchiMate. *Inf. Syst.* **54**, 235–262 (2015). <https://doi.org/10.1016/j.is.2015.04.008>
48. Loucopoulos, P., Kavakli, E.: Capability oriented enterprise knowledge modeling: the CODEK approach. In: Karagiannis, D., Mayr, H.C., Mylopoulos, J. (eds.) *Domain-Specific Conceptual Modeling*, pp. 197–215. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39417-6_9
49. Loucopoulos, P., Kavakli, E., Anagnostopoulos, D., Dimitrakopoulos, G.: Capability-oriented analysis and design for collaborative systems: an example from the Doha 2022 World Cup Games. In: *Proceedings of the 2018 10th International Conference on Computer and Automation Engineering—ICCAE 2018*, pp. 185–189. ACM Press, Brisbane, Australia (2018). <https://doi.org/10.1145/3192975.3192998>
50. Malik, N.: Enterprise business motivation model: full model documentation v.4.2 (2013)
51. MOD Architecture Framework. <https://www.gov.uk/guidance/mod-architecture-framework>. Last accessed 2020/07/12
52. Antunes, G., Barateiro, J., Becker, C., Borbinha, J., Vieira, R.: Modeling contextual concerns in enterprise architecture. In: 2011 IEEE 15th International Enterprise Distributed Object Computing Conference Workshops, pp. 3–10. IEEE, Helsinki, Finland (2011). <https://doi.org/10.1109/EDOCW.2011.9>
53. Walker, S.K.: *Capabilities-Based Planning—How It Is Intended to Work and Challenges to Its Successful Implementation*. Army War College, Carlisle, Barracks (2005)
54. du Toit, F.A., Tanner, M.: A business architecture capability meta model and tool-set for providing function point estimation for enterprise architecture management. In: *Proceedings of the International MultiConference of Engineers and Computer Scientists*, pp. 482–494. Hong Kong (2015)
55. Barroero, T., Motta, G., Pignatelli, G.: Business capabilities centric enterprise architecture. In: Bernus, P., Doumeingts, G., Fox, M. (eds.) *Enterprise Architecture, Integration and Interoperability*, pp. 32–43. Springer, Berlin (2010). https://doi.org/10.1007/978-3-642-15509-3_4
56. Straube, C., Kranzlmüller, D.: Model-driven resilience assessment of modifications to HPC infrastructures. In: Mey, D., Alexander, M., Bientinesi, P., Cannataro, M., Clauss, C., Costan, A., Kecskemeti, G., Morin, C., Ricci, L., Sahuquillo, J., Schulz, M., Scarano, V., Scott, S.L., Weidendorfer, J. (eds.) *Euro-Par 2013: Parallel Processing Workshops*, pp. 707–716. Springer, Berlin (2014). https://doi.org/10.1007/978-3-642-54420-0_69
57. Becker, C., Antunes, G., Barateiro, J., Vieira, R., Borbinha, J.: Modeling digital preservation capabilities in enterprise architecture. In: *Proceedings of the 12th Annual International Digital Government Research Conference on Digital Government Innovation in Challenging Times - dg.o'11*, p. 84. ACM Press, College Park, MD (2011). <https://doi.org/10.1145/2037556.2037570>
58. Kudryavtsev, D., Grigoriev, L., Bobrikov, S.: Strategy-focused and value-oriented capabilities: methodology for linking capabilities with goals and measures. In: Molnar, W.A., Proper, H.A., Zdravkovic, J., Loucopoulos, P., Pastor, O., de Kinderen, S. (eds.) *Complementary Proceedings of the 8th Workshop on Transformation & Engineering of Enterprises (TEE 2014)*, and the 1st International Workshop on Capability-Oriented Business Informatics (CoBI 2014), pp. 15–26. CEUR-WS.org, Geneva, Switzerland (2014)
59. Klinkmüller, C., Ludwig, A., Franczyk, B., Kluge, R.: Visualising business capabilities in the context of business analysis. In: Abramowicz, W., Tolksdorf, R. (eds.) *Business Information Systems*, pp. 242–253. Springer, Berlin (2010). https://doi.org/10.1007/978-3-642-12814-1_21
60. DeLoach, S.A., Oyenian, W.H., Matson, E.T.: A capabilities-based model for adaptive organizations. *Auton. Agents Multi-Agent Syst.* **16**, 13–56 (2008). <https://doi.org/10.1007/s10458-007-9019-4>
61. Bhiri, S., Derguech, W., Zaremba, M.: Modelling capabilities as attribute-featured entities. In: Cordeiro, J., Krempels, K.-H.

- (eds.) Web Information Systems and Technologies, pp. 70–85. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36608-6_5
62. DeLoach, S.A., Garcia-Ojeda, J.C.: O-MaSE: a customisable approach to designing and building complex, adaptive multi-agent systems. *Int. J. Agent Oriented Softw. Eng.* **4**, 244–280 (2010)
 63. Homann, U., Levy, M., Merrifield, E., Appel, D., Davidson, E., Isaacs, S., Judah, N.: Efficient and Flexible Business Modeling Based Upon Structured Business Capabilities (2006). <http://patents.google.com/patent/US20060116922A1/en>
 64. Danesh, M.H., Loucopoulos, P., Yu, E.: Dynamic capabilities for sustainable enterprise IT—a modeling framework. In: Johansson, P., Lee, M.L., Liddle, S.W., Opdahl, A.L., Pastor López, Ó. (eds.) *Conceptual Modeling*, pp. 358–366. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25264-3_26
 65. Rauffet, P., Cunha, C.D., Bernard, A.: Managing resource learning in distributed organisations with the organisational capability approach. *Int. J. Technol. Manag.* (2016). <https://doi.org/10.1504/IJTM.2016.075902>
 66. Aldea, A., Jacob, M.E., Van Hillegersberg, J., Quartel, D., Franken, H.: Capability-based planning with ArchiMate. In: 17th International Conference on Enterprise Systems (ICEIS), pp. 352–359. SciTePress (2015)
 67. Danesh, M.H.: A socio-technical modeling framework for designing enterprise capabilities. In: Proceedings of the Doctoral Consortium Papers presented at the 11th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modelling (PoEM 2018), pp. 31–46. CEUR-WS.org, Vienna, Austria (2018)
 68. Baccar, S., Rouached, M., Abid, M.: A capabilities driven model for web services description and composition. *Int. J. Bus. Inf. Syst.* **22**, 26 (2016). <https://doi.org/10.1504/IJBIS.2016.075715>
 69. Antunes, G., Vieira, R., Borbinha, J.: Capabilities and Requirements Engineering: Research Challenges. In: INCOSE International Symposium, vol. 23, pp. 590–605 (2013). <https://doi.org/10.1002/j.2334-5837.2013.tb03041.x>
 70. Papazoglou, A.: Capability-based planning with TOGAF® and ArchiMate®, (2014)
 71. Object Management Group (OMG): Value Delivery Modeling Language v.1.1 (2018). <https://www.omg.org/spec/VDML/1.1>
 72. Antunes, G., Borbinha, J.: Capabilities in systems engineering: an overview. In: Falcão e Cunha, J., Snene, M., Nóvoa, H. (eds.) *Exploring Services Science*, pp. 29–42. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36356-6_3
 73. Jacob, M.-E., Quartel, D., Jonkers, H.: Capturing business strategy and value in enterprise architecture to support portfolio valuation. In: 2012 IEEE 16th International Enterprise Distributed Object Computing Conference, pp. 11–20. IEEE, Beijing, China (2012). <https://doi.org/10.1109/EDOC.2012.12>
 74. Mikloš, J.: A meta-model for the spatial capability architecture. *J. Theor. Appl. Inf. Technol.* **43**, 301–305 (2012)
 75. Bergström, S.: Modelling Business Capabilities with Enterprise Architecture: A Case Study at a Swedish Pension Managing Company (2015). <http://www.diva-portal.org.ezp.sub.su.se/smash/get/diva2:860618/FULLTEXT01.pdf>
 76. MODAF ontological data exchange mechanism (MODEM) (2012). https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/63980/20130117_MODAF_MODEM.pdf
 77. Liu, L., Russell, D., Xu, J., Webster, D., Luo, Z., Venters, C., Davies, J.K.: Modelling and simulation of network enabled capability on service-oriented architecture. *Simul. Model. Pract. Theory* **17**, 1430–1442 (2009). <https://doi.org/10.1016/j.simp.2009.06.012>
 78. Li, W., Badr, Y., Biennier, F.: Improving web service composition with user requirement transformation and capability model. In: Meersman, R., Panetto, H., Dillon, T., Eder, J., Bellahsene, Z., Ritter, N., De Leenheer, P., Dou, D. (eds.) *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, pp. 300–307. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41030-7_21
 79. Zhang, W., Wang, Z., Zhao, W., Yang, Y., Xin, X.: Generating executable capability models for requirements validation. *J. Softw.* (2012). <https://doi.org/10.4304/jsw.7.9.2046-2052>
 80. Becker, C., Antunes, G., Barateiro, J., Vieira, R.: A capability model for digital preservation: analysing concerns, drivers, constraints, capabilities and maturities. Presented at the International Conference on Preservation of Digital Objects (iPRES 2011), Singapore November 1 (2011)
 81. Rafati, L., Poels, G.: Capability Sourcing Modeling a High-Level Conceptualization Based on Service-Dominant Logic. Springer, Berlin (2014). <https://doi.org/10.1007/978-3-319-07869-4>
 82. Nunes, I., Faccin, J.G.: Modelling and implementing modularised BDI agents with capability relationships. *Int. J. Agent Oriented Softw. Eng.* **5**, 203 (2016). <https://doi.org/10.1504/IJAOS.2016.080887>
 83. Malamateniou, F., Themistocleous, M., Prentza, A., Papakonstantinou, D., Vassilacopoulos, G.: A context-aware, capability-based, role-centric access control model for IoMT. In: Perego, P., Andreoni, G., Rizzo, G. (eds.) *Wireless Mobile Communication and Healthcare*, pp. 125–131. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58877-3_16
 84. Gongolidis, E., Kavakli, E., Loucopoulos, P., Kalloniatis, C.: Migrating eGovernment services in the Cloud: a capability modelling approach. In: Proceedings of the 20th Pan-Hellenic Conference on Informatics—PCI'16, pp. 1–6. ACM Press, Patras (2016). <https://doi.org/10.1145/3003733.3003783>
 85. OASIS Committee: Reference Architecture Foundation for Service Oriented Architecture Version 1.0 (2012). <http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/cs01/soa-ra-v1.0-cs01.html>
 86. Tingting, Z., Xiaoming, L., Zhixue, W., Qingchao, D.: Capability-oriented architectural analysis method based on fuzzy description logic. *Comput. Sci. Inf. Syst.* **13**, 287–308 (2016). <https://doi.org/10.2298/CSIS15022046T>
 87. Bravos, G., Loucopoulos, P., Dimitrakopoulos, G., Anagnostopoulos, D., Kiouisi, V.A.: A capability—driven modelling approach applied in smart transportation & management systems for large scale events. *EAI Endorsed Trans. Internet Things* **3**, 153051 (2017). <https://doi.org/10.4108/eai.31-8-2017.153051>
 88. Wang, Z., Zhang, W., Dong, Q., He, H., Zhao, W.: A light way of enterprise modeling and simulation for C4ISR system based on xUML. *J. Command Inf. Syst.* **8**, 2829–2838 (2012)
 89. Feltus, C., Petit, M.: Building a responsibility model including accountability, capability and commitment. In: 2009 International Conference on Availability, Reliability and Security, pp. 412–419. IEEE, Fukuoka, Japan (2009). <https://doi.org/10.1109/ARES.2009.45>
 90. Derguech, W., Bhiri, S., Curry, E.: Using ontologies for business capability modelling: describing what services and processes achieve. *Comput. J.* **61**, 1075–1098 (2018). <https://doi.org/10.1093/comjnl/bxy011>
 91. Derguech, W., Bhiri, S., Curry, E.: Designing business capability-aware configurable process models. *Inf. Syst.* **72**, 77–94 (2017). <https://doi.org/10.1016/j.is.2017.10.001>
 92. Kolini, F., Janczewski, L.J.: Cyber defense capability model: a foundation taxonomy. In: CONF-IRM (2015)
 93. Object Management Group (OMG): Service Oriented Architecture Modeling Language (SoaML) Specification Version 1.0.1 (2012). <https://www.omg.org/spec/SoaML/1.0.1/PDF>
 94. Roubtsova, E., Michell, V.: Behaviour models clarify definitions of affordance and capability. In: Proceedings of the 2014 Workshop on Behaviour Modelling-Foundations and

- Applications—BM-FA'14, pp. 1–10. ACM Press, York, United Kingdom (2014). <https://doi.org/10.1145/2630768.2630774>
95. Rauffet, P., Da Cunha, C., Bernard, A.: Conceptual model and IT system for organizational capability management. *Comput. Ind.* **63**, 706–722 (2012). <https://doi.org/10.1016/j.compind.2012.05.004>
 96. Pastor, O., Ruiz, M., Koç, H., Valverde, F.: Capability-based communication analysis for enterprise modelling. *Enterp. Model. Inf. Syst. Archit.* **13**, 1–24 (2018). <https://doi.org/10.18417/emisa.13.4>
 97. Ge, B., Hipel, K.W., Yang, K., Chen, Y.: A data-centric capability-focused approach for system-of-systems architecture modeling and analysis. *Syst. Eng.* **16**, 363–377 (2013). <https://doi.org/10.1002/sys.21253>
 98. Iacobucci, J.V.: Rapid Architecture Alternative Modeling (RAAM): A Framework for Capability-Based Analysis of System of Systems Architectures (2012). <https://smartech.gatech.edu/handle/1853/43697>
 99. Stirna, J., Zdravkovic, J.: Development of a modeling language for capability driven development: experiences from meta-modeling. In: Comyn-Wattiau, I., Tanaka, K., Song, I.-Y., Yamamoto, S., Saeki, M. (eds.) *Conceptual Modeling*, pp. 396–403. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46397-1_30
 100. Radeck, C., Blichmann, G., Meißner, K.: Modeling and calculating capabilities of composite web applications for assisted end user development. In: Monfort, V., Krempels, K.-H., Majchrzak, T.A., Traverso, P. (eds.) *Web Information Systems and Technologies*, pp. 58–82. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66468-2_4
 101. Vogt, W.: *Dictionary of Statistics & Methodology*. SAGE Publications Inc, Thousand Oaks (2005). <https://doi.org/10.4135/9781412983907>
 102. Aldea, A., Jacob, M.-E., Hillegersberg, V.J., Quartel, D., Franken, H.: Serious gaming for the strategic planning process. In: Proper, H.A., Lin, K.-J., Marchand-Maillet, S., Ralyte, J. (eds.) *Proceedings—16th IEEE Conference on Business Informatics, CBI 2014*, pp. 183–190. Institute of Electrical and Electronics Engineers Inc. (2014). <https://doi.org/10.1109/CBI.2014.45>
 103. Rafati, L., Poels, G.: Capability sourcing modeling. In: Iliadis, L., Papazoglou, M., Pohl, K. (eds.) *Advanced Information Systems Engineering Workshops*, pp. 77–87. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07869-4_7
 104. Sinnott-Armstrong, W., Fogelin, R.J.: *Understanding Arguments: An Introduction to Informal Logic*. Cengage Learning, South Melbourne (2015)
 105. Barbero, M., Jouault, F., Bézivin, J.: Model driven management of complex systems: implementing the macroscope's vision. In: *15th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ecbs 2008)*, pp. 277–286. IEEE, Belfast, Northern Ireland (2008). <https://doi.org/10.1109/ECBS.2008.42>
 106. Feltus, C., Grandry, E., Fontaine, F.-X.: Capability-driven design of business service ecosystem to support risk governance in regulatory ecosystems. *CSIMQ* (2017). <https://doi.org/10.7250/csimq.2017-10.05>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Georgios Koutsopoulos is a Ph.D. candidate in the Information Systems unit of the Department of Computer and Systems Sciences (DSV) of Stockholm University, Sweden, from where he also received his Ph.Lic. and M.Sc. degrees in Computer and Systems Sciences. He is also a graduate of the Department of History and Philosophy of Science, National and Kapodistrian University of Athens, Greece. His research focus is on enterprise modeling methods, in particular, modeling of changing business capabilities in organizations. Other research areas include Requirements Engineering and Business Process Modeling.



Martin Henkel works as an Associate Professor at the Department of Computer and Systems Sciences, Stockholm University. His research areas include enterprise capability design and modeling for change analysis. As a consultant and researcher, Martin has performed enterprise modeling and software analysis and design in several domains such as health care, IT systems for energy efficiency, building maintenance and parts purchasing.



Janis Stirna has received a degree of Doctor of Philosophy in Computer and Systems Sciences from the Royal Institute of Technology (KTH), Sweden, in 2001 and was awarded Associate Professor (docent) from Jönköping University, Sweden, in 2008. In 2015, he was promoted to full professor at Department of Computer and Systems Science (DSV), Stockholm University. Stirna's research interests are enterprise modeling methods and tools, organizational patterns, as well as capability and ecosystem

modeling. Stirna is an author or co-author of more than 100 research reports and publications as well as three text books on enterprise modeling. He has participated in 14 EU-financed research projects and has co-developed four enterprise modeling methods, including 4EM—for enterprise modeling and CDD—capability-driven development. Stirna is currently vice-chair of the IFIP Working Group 8.1 Design and Evaluation of Information Systems.