Can an enterprise model help in mapping capabilities?

Ilia Bider[1,2] and Erik Perjons[1]

[1] *Stockholm University, Borgarfjordsgatan 12, Kista, Stockholm, 164 55, Sweden*

[2] *University of Tartu, Ülikooli 18, 50090 Tartu, Estonia*

Abstract

The paper investigates the potential of enterprise models for identifying organizational capabilities, including the hidden ones - those that fall under the radar, i.e., unknown to the management. The hidden capabilities might be critical in an emergency or for expanding the business. After presenting a working definition of capability, the study outlines a set of requirements on enterprise modeling techniques based on the definition. Fulfilling these requirements makes a modeling technique potentially useful for the task of identifying capabilities, especially the ones that are hidden somewhere inside the organization. These requirements emphasize having capabilities that are easily distinguishable in the model, depicting the processes that manage the resources employed, and having means to expand the model based on the already built parts. The requirements are also tested against some of the enterprise modeling techniques.

1. Introduction

We start with an example presented in [1], in which a company whose primary business was in decline desperately needed a solution to avoid going out of business. They hired a management consultant to find out what they could do. The consultant investigated the business but struggled to find a solution. However, in the end, he found a room on the top floor where three people worked. It turned out that the management did not know what exactly they were doing. After investigation, it was discovered that they did some work not only for the company itself but also for external organizations. They had a stock of external orders that they could not complete themselves because they were only three people. The work done for external organizations tended to be a way out for the struggling company. The company changed its business and survived.

What the consultant found in the end was a capability that was under the management's radar. At this particular time, it was the only capability that could save the company. The question that can be derived from the example is how to find and map all capabilities of an organization, even the ones that the management does not know about or consider to be not significant for mapping. Nevertheless, these are capabilities that can save a company from going out of business or help expand to areas substantially different from the current business. In other words, these capabilities are the basis for so-called Industry (business) model innovation - which amounts to changing the position in the value chain, entering new markets, and/or other types of radical changes, see [2].

The questions that we want to raise in this paper are:

1. Can an enterprise model help find all capabilities of an organization, even those that the management is unaware of (such as a local initiative)?
2. Can the enterprise model help evaluate the quality/uniqueness of all the capabilities found?

3. What kind of modeling technique should be employed to have the model that is useful for the above?

In this paper, we will limit our investigation to visual models, the ones that represent an organization as a graph that consists of nodes and edges (arrows). This is a kind of models that could be understood and investigated by non-technical stakeholders. For an enterprise model to be useful in capability mapping, it should include fragments, i.e., groups of modeling elements, that represent capabilities. If we have such a model, identifying capabilities will be relatively simple. Another useful feature is if the modeling technique can help expand the initial model, i.e., help find additional fragments representing capabilities that remain under the radar of the management. The last important feature is that the model depicts all elements engaged in completing actions related to capabilities. These three features will allow us to evaluate each identified capability, e.g., whether it can serve as a way out of a crisis or can be used to expand the company's offerings.

This paper aims to suggest a set of requirements on a modeling technique regarding its appropriateness for finding and evaluating, if not all, the major part of an organization's capabilities. This set of requirements is also tested on three techniques, namely IDEF0 [3], Fractal Enterprise Model (FEM) [4], [5], and ArchiMate [6]. The choice is based on (1) all these techniques allow us to build a model that includes recurring activities in an enterprise; (2) we have experience in using them in practice and/or educational contexts. As far as FEM is concerned, this modeling technique is our invention. Furthermore, we will be using a simple example of an assembly line to illustrate our line of thought. The example originated from the book [7], which we consider one of the best books on ArchiMate. The book presents an ArchiMate model of the case, which we use, more or less, as-is. We also created similar models for the two other techniques.

The rest of the paper is structured in the following way. In section 2, we discuss some theoretical issues related to our topic. In Section 3, we suggest a set of requirements on a modeling technique for it to be appropriate for the task of finding and evaluating capabilities. In Section 4, we match our requirements against three enterprise modeling techniques. Section 5 contains concluding remarks and plans for the future.

2. Background

2.1. What is a capability?

As there are too many approaches to defining a capability, in this section, we will present a working definition of the term that we use in this paper. In [8], the author has made an extensive analysis of the term, both from the point of view of what a capability is, and what it is not. The suggested definition in [8] more or less satisfies our needs, but we want to give it a slightly different form, namely:

A capability is something that an organization can do based on:

1. *It already did the same thing in the past on a recurrent basis*
2. *Using resources under its own control, and*
3. *Achieving success in some cases but may failing in other cases, and*
4. *Resources used in the past to complete actions are still available for deployment.*

Though the capability, as defined above, represents a possibility to complete an action, it is based on the fact that this action was completed with some success in the past using some resources and that these resources are still available. By resources, we consider anything engaged in completing actions, like trained people, equipment, spare parts, manuals describing the procedures, etc. As soon as some of these resources are depleted, the capability ceases to exist. The latter means that when

evaluating a capability, we need to look not only at the actions completed in the past but also at how the organization manages the resources engaged in these actions.

Though the definition above is more or less consistent with the OMG definition: "Ability to perform a particular kind of work and deliver desired value" [9], our definition is specifically adjusted to search for capabilities in an enterprise model. It is narrower than the general definition, as it is based on repetitive behavior and may not cover other types of capabilities. The latter includes capabilities acquired by training in the military, training for emergencies, or so-called dynamic capabilities, which enable an organization to evolve and adapt in response to changes in the environment [10]. However, we believe that the suggested definition is enough for our purpose, as an enterprise model normally shows what the company actually does; it is quite doubtful that we can identify other types of capabilities analyzing such a model.

2.2. Research approach

As our goal is to create a set of requirements on a modeling technique, the natural method for doing it is Design Science (DS). DS is a research paradigm [11,12,13] that focuses on looking for generic solutions for known and unknown problems. The result of a DS research project is a solution for a problem in the terminology of [13] or an artifact in the terminology of [11]; alternatively, the result can be in the form of "negative knowledge" stating that a certain approach is not appropriate for solving problems of certain kinds [13]. Though the purpose of this paper is just to create a set of requirements on a modeling technique and to test them on a limited number of techniques, the overall goal is to have a method of evaluating a modeling technique for the purpose of discovering capabilities, which will include the requirements as an integral part.

The method chosen for creating requirements consists of a logical analysis of the definition of capability and deriving what a model and modeling technique need to provide, i.e., requirements.

3. Requirements on modeling technique

3.1. Requirements on expressive power

Expressive power is a property of the language that Wikipedia defines as "the breadth of ideas that can be represented and communicated in that language" [14]. We will define the requirements on expressive power by logically analyzing the definition given in Section 2.1. Bullet 1 in this definition means that capability is based on the actions an organization completes regularly. Thus, the first requirement on the technique is:

(1) *The technique should be able to represent the recurrent actions.* They can be called differently, e.g., process, service, function, etc. Preferably, actions should be represented as nodes in the graph, making them easier to identify.

The second bullet in the definition means that the capability is using resources under the organization's control when carrying out recurrent actions. Thus, the second requirement on the technique is:

(2) *The technique should be able to represent resources used in the recurrent action and show their attachment to the action.* Again, the resources can have different names, like data, people, etc. It is better if the resources are also represented as nodes, but this is not mandatory.

The third bullet in our definitions requires that at least part of the actions have achieved some success. We will not translate this bullet into a requirement, considering that an as-is enterprise model of an organization depicts what the company regularly does. If it is a normal company, we consider that it does not repeat actions that do not lead to success in many cases. This is a preliminary decision,

which can be reverted later. Note also that a "to-be" enterprise model might have actions that have never been repeatedly completed by the organization.

The fourth bullet in the definitions means that resources used in action are available for future iterations of actions. It means that the organization should manage these resources so that they are not in a depleted state. This leads us to the third requirement:

(3) *The technique should be able to represent actions aimed at having resources used in the activities related to a capability in good shape.* The meaning of these actions is different and depends on the resource. For example, it can be hiring new members of staff to substitute for those who left or are about to leave, or servicing the equipment on a regular basis.

3.2. Requirements on discovery power

The term *discovery power* is not an accepted term, but we believe it to be important for our goal. We can give a working definition of the term *discovery power* of a modeling language:

"Degree of help provided by the structure of enterprise modeling language to expand a partly built model or fill gaps in it".

The question arises of why we need help from a language structure to expand a partly built enterprise model or fill gaps in it. Would not having enough expressive power be sufficient? After all, there are other methods for finding how an enterprise functions, like modeling patterns [15], which can be used. There are two reasons to include a requirement on the discovery power in our set:

1. Our goal is to find all capabilities of an organization, even the hidden ones – unknown to the management. This means that we need to expand our model to half-hidden corners of the organization; therefore, we might need any help we can get. Having a technique that guides finding new activities is a plus.
2. The search for unknown capabilities usually starts in the situation of a crisis (see the example in the introduction). In such a situation, we need to build an enterprise model that helps us to find all capabilities as quickly as possible. If the employed modeling technique helps us in this work, there is no need to employ more heavy means not connected to the language, which may require more time.

Based on the deliberation above, we can add the fourth requirement on the modeling technique:

(4) *The technique should possess enough discovery power to ensure that the enterprise model covers all organization's activities and can be built (relatively) quickly.* This is an optional requirement. Note that if an organization has substantial experience in building models using some special technique, or it already has an extensive enterprise model that satisfies the first three requirements, it might not need the discovery power in the modeling technique.

4. Analyzing modeling techniques

To test our set of requirements, in this section, we will analyze three enterprise modeling techniques, namely IDEF0 [3], Fractal Enterprise Model (FEM) [4], [5], and ArchiMate [6].

4.1. IDEF0

A model in IDEF0 [3] consists of functional blocs of the type presented in Figure 1, which represents an assembly process for robots. In this figure, a rectangle represents a function (repetitive behavior). Arrows coming from the left represent the inputs of the function, and arrows coming out on the right represent the outputs of the function. Arrows coming to the bottom represent mechanisms used to

transform inputs into outputs, which can be people, equipment, etc. Arrows coming to the top represent control mechanisms that steer the function execution; there are none in Figure 1.

The connections between two functions are presented as an output from one function used as an input, mechanism, or control for another function. Some inputs can come outside the given organization, and some outputs can go outside the given organization. An example of connections is presented in Figure 1, right side, which shows that inputs to *Assembly* are produced by other functions of the same company. Note that outputs do not need to come as inputs; they can come to the top or to the bottom of another functional module, which means that output can become a control or mechanism. For example, a hiring function of HR can produce a mechanism for the *Assembly* function (*Operators*).
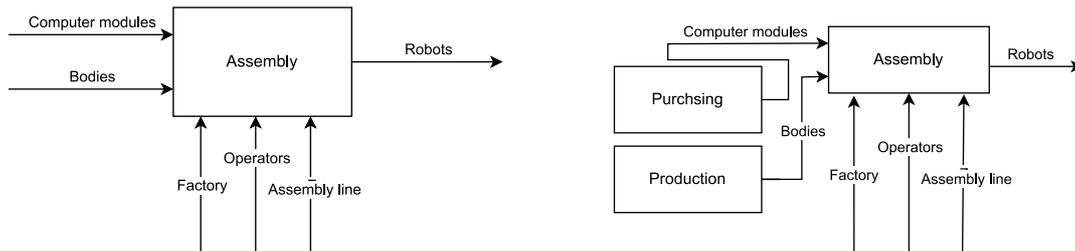

Figure 1: IFDEF0. Left side – a typical functional module; right side – connection between functional modules

In Table 1, we present our analysis of IDEF0 against the four requirements introduced in Section 3. From this table, it is relatively clear that IDEF0 is a good candidate for building an enterprise model that can be used to find capabilities. The only drawback here is that it is difficult to see how the activities aimed at maintaining resources can be found. These activities can be important; see, for example, a case considered in [16].

Table 1
Analysis of IDEF0

Requirement	Fulfillment	Explanation
#1	Full	Recurrent activities are represented as functional nodes.
#2	Full	Resources are shown as arrows coming into a functional node from the left, to the top, or to the bottom.
#3	Partial	It can represent functional modules that add new elements to resources, but it is not clear how to represent maintenance, for example, service or reparation of equipment in a total model.
#4	Full	As soon as a functional module is added to the model and resources that it uses are shown as arrows coming into the node, and the outputs are determined, a bunch of questions arises naturally, e.g., "From where are inputs coming, and to where are the outputs going?".

4.2. Fractal enterprise model

A building block of a Fractal Enterprise Model (FEM) [4] is a process with assets attached to it. In Figure 2, left side, we present a FEM fragment that corresponds to the IDEF0 fragment in Figure 1, left side. In the FEM model, an oval represents a process – that is, a repetitive behavior; a rectangle represents an asset – that is, a set of things or actors that can be used in or managed by a process. An arrow with a solid line represents the *used-in* relation between an asset and a process; such an arrow is provided with one or several labels from the standardized set of 8 labels. An arrow with a dashed

line represents the *managed-by* relation between the asset and the process; such an arrow is provided with one or several labels from the standardized set of three labels – *Acquire, Maintain*, and *Retire*. The first label means that new elements are added to the asset, the second label means that some elements are changed to restore their working conditions, and the third label means that the elements are removed from the set.
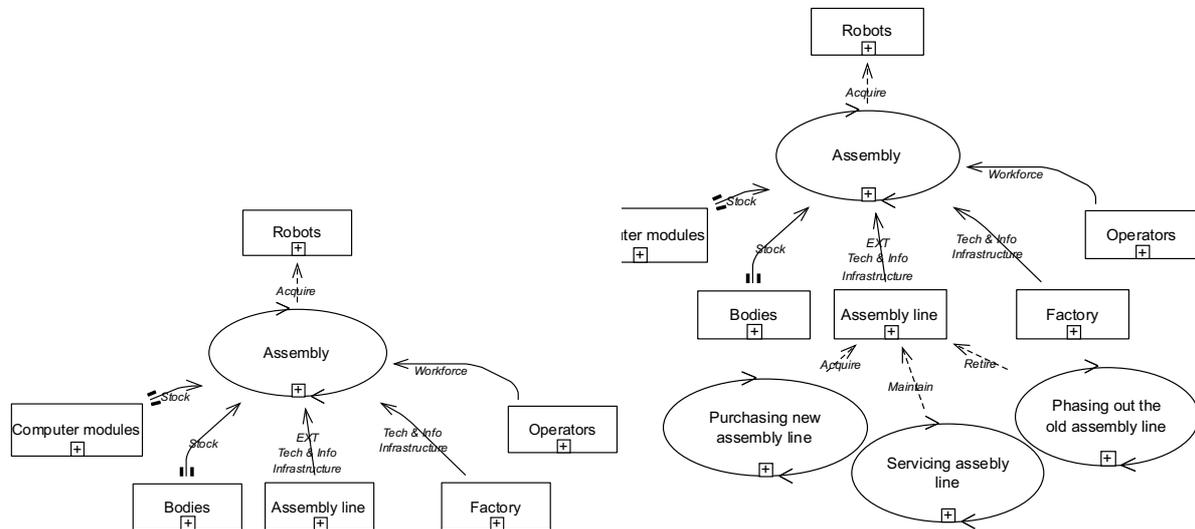

Figure 2: FEM. Left side – a typical FEM fragment; right side – connection between fragments

The label stock has a special meaning, showing that some of the elements of the asset are consumed in each successful process run (a run is one cycle of repetitive behavior represented by a process). The special meaning of stock is also represented by a tale of the corresponding arrow. The Stock label can be assigned to an arrow alone or can be accompanied by some other label. When alone, it represents some consumables – parts of the assembly, reserve parts, etc. This label roughly corresponds to the input concept of IDEF0 (see Fig. 1). FEM also includes elements to represent the business context of a company, but we will not discuss them in this paper.

Figure 2, right, shows how the fragment on the left side can be extended by adding management processes to one of the assets, namely, the assembly line. After the management processes have been added, each of them can be expanded by adding assets used in them. The expansion of a FEM model is done using two kinds of archetypes (patterns): process-assets archetypes and asset-processes archetypes. A process-assets archetype represents the kinds of assets that can be used in a given category of processes. There are several archetypes of this kind. The upper part of Figure 3 presents a so-called generic process-assets archetype, which can be applied to any process. There are also specific process-assets archetypes. An asset-processes archetype shows the kinds of processes that are aimed at changing the given category of assets. There is only one such archetype, which is presented in the bottom part of Figure 3.

In Table 2, we present our analysis of FEM against the four requirements introduced in Section 3. From this table, it is relatively clear that FEM is a good candidate to be used for building an enterprise model that can be used for finding capabilities.

4.3. ArchiMate

The thinking behind ArchiMate [6] is represented in Figure 4. In the middle, there is an action node; in this case, it is a business process. To the right, there is an actor assigned to run the process, and to the left, there is a passive object that the process accesses. The central node represents an action, and the other two – represent some of the resources engaged in the action. In ArchiMate, the non-action nodes represent active or passive objects. Examples of active objects are an actor or a role; examples

of passive objects are a contract or data store. An active object is assigned to complete the action, which is represented by an arrow with a rounded tale, see Figure 4. A passive object can be accessed by the action, which is represented by a dashed arrow. Dependent on the direction of the arrow, access can be read/write or both; the latter is represented by having the arrow head on both ends of the connector, as in Figure 4. There are several types of nodes that represent actions, e.g., a process (as in Figure 4), a service, etc. Also, there are several layers, i.e., the business, application, and technology layers, and each of them has action nodes, active nodes, and passive nodes. The elements of different layers can be connected by special kinds of relations.
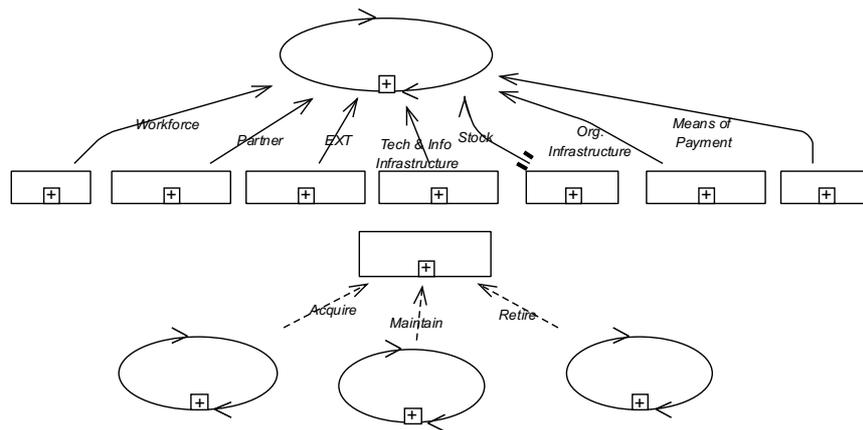

Figure 3: Upper part – a process-asset archetype; bottom part – the asset-processes archetype

Table 2
Analysis of FEM

Requirement	Fulfillment	Explanation
#1	Full	Recurrent activities are presented as processes.
#2	Full	Assets represent resources used in and produced by processes. They are connected to a process by arrows; a label on the arrow shows the nature of the resource and how it is used.
#3	Full	Management process can be attached to any resource (asset) to show how new elements are acquired and existing elements are maintained and retired.
#4	Full	As soon as an asset is introduced in the model, one can use the archetype from Figure 3 to start looking for management processes. As soon as a process is introduced in the model, one can use one of the process-assets archetypes to start looking for assets used in it.

An example of an ArchiMate diagram that depicts the same case as for IDEF0 in Figure 1, left part, and for FEM in Figure 2, left part is presented in Figure 5. The example was adapted from [7]. In this diagram, the green elements represent the technological layer, and the yellow elements represent the business layer. We have only one such element – *Operator,* which is a role played by some person(s). As we can see from this diagram, the model includes resources used in the assembly action, but not all of them are connected directly to the node that represents the action. The nodes depicting *Factory* and *Operator* are connected only indirectly. In addition, the same action can be presented by more than one node in the same or different layers. This is shown in Figure 6, in which *Business service* is realized by *Business Process* and *Business object* is connected only to *Business process*.
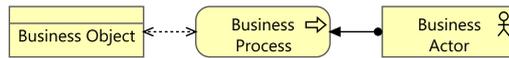
Figure 4: Thinking behind the ArchiMate

Note that ArchiMate has a special concept for representing capabilities. However, the elements that express this concept need to be connected to lower-level elements that realize the capability. Our goal is to identify capabilities by analyzing the model that depicts recurrent activities in the enterprise. When a capability is identified in an ArchiMate model, it can be depicted as an element of the model and connected to the elements that realize it. This, however, is outside our consideration, as we are only interested in how to identify capabilities in an enterprise model that shows recurrent activities inside the enterprise.

In Table 3, we present our analysis of ArchiMate against the four requirements introduced in Section 3. From this table, it is relatively clear that ArchiMate is not a particularly good candidate for building an enterprise model that can be used to find capabilities. This said, if an organization already has a detailed enterprise model expressed in ArchiMate, it can be used for finding capabilities, though this can be quite cumbersome; see explanations in Table 3.
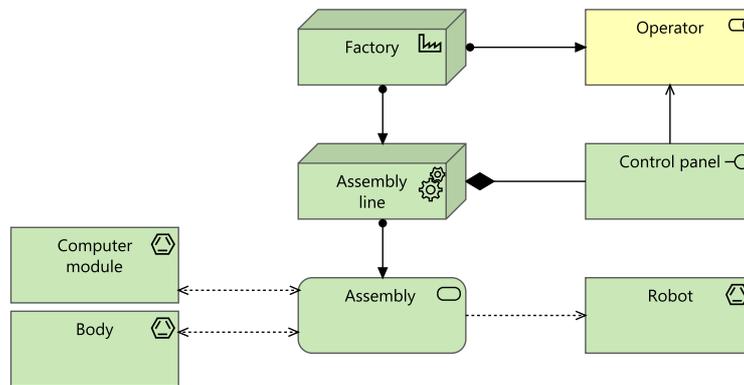

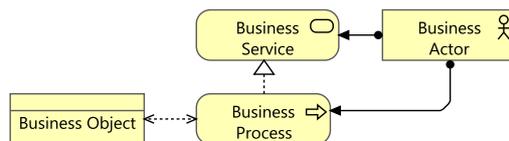**Figure 5:** Assembly example in ArchiMate

Figure 6: One capability is split into two different concepts: Business Service and Business Process

Table 3
Analysis of ArchiMate

Requirement	Fulfillment	Explanation
#1	Full (but cumbersome)	Recurrent activities are presented as action nodes. There is a possibility that one action from the capabilities point of view is presented by several nodes where one node *realizes* or *serves* the other node. In this situation, there might be a need to merge the nodes; see the next row.
#2	Full (but cumbersome)	Active and passive nodes represent resources used in and produced by an action node. They are connected to action nodes by arrows, and the type of arrow shows in which way they are used in the

Requirement	Fulfillment	Explanation
		action. An arrow can be supplied with a label that gives an additional explanation. The drawback here is that resources may not be directly connected to the action node, see Figure 5; they can also be split between different action nodes that represent the same action from the capabilities point of view, see Figure 6.
#3	Partial (and cumbersome)	There are difficulties in **attaching management processes** to the active nodes. This is because an active node in one action is not allowed to become passive in another action. A workaround is to create a passive copy of an active node and connect it to the active node by association, which makes a model difficult to understand. This is a **known** drawback of ArchiMate.
#4	Partial	ArchiMate has substantial discovery power to find out what IT applications are used in an action, and who is participating in the action. It has much less discovery power to find other things that are engaged in action and to find management processes.

5. Concluding remarks and plans for the future

The summary of the work reported in this paper is as follows. The question that we have posed is whether having or building an enterprise model can help to find and map all capabilities that exist in an organization, even those that might not be known to the management. The latter can be critical in the case of a crisis. We have assumed that the answer depends on the modeling technique used for building the model. We proceeded to create a set of requirements on the modeling technique used for building the model. This was done based on the analysis of the working definition of the term capability, which was adapted from [8]. The result was a set of four requirements, three of which are related to the expressive power of the modeling technique, and one - to the discovery power of it. The requirements were used to analyze three different enterprise modeling techniques.

The results achieved so far are preliminary but encouraging. Though the requirements are defined in broad terms, they could be matched against the existing techniques and produce some recommendations on the appropriateness of the analyzed techniques for capabilities mapping. So far, there is no detailed matching procedure, and the matching was done in an ad-hoc fashion based on the authors' experience in enterprise modeling. To proceed, we need to get acceptance of the results from other experts in the field. Presenting our ideas at the workshop devoted to capabilities is a step in this direction.

To make our ideas into an artifact that can be disseminated:

- The requirements definition should be extended and detailed
- A procedure for matching should be designed and adequately documented
- Expert opinions should be obtained and incorporated into the above
- Documentation with good examples should be made available for the wider audience

The authors intend to start working on this list in the near future.

Other directions that span from this paper are as follows:

- How to represent capabilities in the enterprise model? As was discussed in Section 4.3, ArchiMate already has a concept of capability and a way of connecting it with other concepts. Neither IDEF0 nor FEM includes such a concept. In FEM, one can use the subclassing feature that allows one to define subclasses for any concept, and depict elements that belong to a subclass with the help of background or border color [5]. In practical terms, it means that a

capability is defined as a subclass of processes, and some background color is assigned to it. Then, all occurrences of this class of processes will get this background color to show where in the company this capability is realized.[1] How to define capabilities in IDEF0 in a formal way is not clear.

- How to assess whether a capability is substantial enough, so that a new business can be started based on it? This includes setting some parameters that describe the "solidness" of the capability.
- How to build a model of a new business activity based on a promising capability, and how to define/depict the transformation that needs to be performed? This should be investigated separately for each modeling technique. As far as our technique – FEM – is concerned, these issues are discussed in a number of published and unpublished works, such as: [17] [16] [18].

Acknowledgments

References

[1] Hoverstadt, P.: The Fractal Organization: Creating Sustainable Oragnizations with the Viable System Model. John Wiley & Son (2008)

[2] Giesen, E., Berman, S. J., Bell, R., Blitz, A.: Three ways to successfully innovate your business model. Strategy & Leadership, Vol. 35 Issue: 6, pp. 35(6), 27-33 (2007)

[3] NIST: Integration definition for function modeling (IDEF0), Draft Federal Information Processing Standards, Publication 183, 1993. In: IDEF. (Accessed 1993) Available at: https://www.idef.com/idefo-function_modeling_method/

[4] Bider, I., Perjons, E., Elias, M., Johannesson, P.: A fractal enterprise model and its application for business development. SoSyM 16(3), 663–689 (2017)

[5] Bider, I., Perjons, E., Klyukina, V.: Tool Support for Fractal Enterprise Modeling. In : Domain-Specific Conceptual Modeling. Springer (2022) 205–229

[6] The open group: ArchiMate® 3.2 Specification. In: The Open Group. (Accessed 2022) Available at: https://publications.opengroup.org/standards/archimate/specifications/c226

[7] Wierda, G.: Mastering Archimate Edition 3.1. P&A (2022)

[8] Tell, A. W.: What Capability Is Not. In Johansson, B., Andersson, B., Holmberg, N., eds. : Perspectives in Business Informatics Research. BIR 2014, LNBIP 194. Springer (2014) 128–142

[9] OMG: Value delivery modeling language (VDML). (Accessed October 2023) Available at: https://www.omg.org/spec/VDML/

[10] Teece, D., Pisano, G., Shuen, A.: Dynamic capabilities and strategic management. Strategic Management Journal 18(7), 509–533 (1997)

[11] Hevner, A., March, S. T., and, P.: Design Science in Information Systems Research. MIS Quarterly 28(1), 75-105 (2004)

[12] Gregor, S., Hevner, A. R.: Positioning and Presenting Design Science Research for Maximum Impact. MIS Quarterly 37(2), 339-335 (2013)

---

[1] In the current version of the FEM toolkit [5], using these means will not allow the same process to realize two different capabilities, but this is a technical limitation, which can be overcome in the future.

[13] Bider, I., Johannesson, P., Perjons, E.: Design science research as movement between individual and generic situation-problem-solution spaces. In : Organizational Systems. An Interdisciplinary Discourse. Springer (2013) 35-61

[14] Wikipedia: Expressive power (computer science). Available at: https://en.wikipedia.org/wiki/Expressive_power_(computer_science)

[15] Fowler, M.: Analysis Patterns: Reusable Object Models. Addison-Wesley Professional (2015)

[16] Bider, I., Lodhi, A.: Moving from Manufacturing to Software Business: A Business Model Transformation Pattern. In : Enterprise Information Systems. ICEIS 2019, LNBIP 378. Springer (2020) 514-530

[17] Bider, I., Perjons, E.: Defining Transformational Patterns for Business Model Innovation. In : Perspectives in Business Informatics Research: 17th International Conference, BIR 2018, Stockholm, Sweden, LNBIP, vol. 330, pp.81-95 (2018)

[18] Fractalmodel.org: Workshop: Business Model Innovation (BMI) with FEM. In: Fractal Enterprise Model. Available at: https://www.fractalmodel.org/fem-workshop/