

Strategies for Capability Modelling: Analysis Based on Initial Experiences

Sergio España¹, Jānis Grabis², Martin Henkel^{3(✉)}, Hasan Koç⁴,
Kurt Sandkuhl⁴, Janis Stirna³, and Jelena Zdravkovic³

¹ Universitat Politècnica de València, Valencia, Spain
sergio.espana@pros.upv.es

² Riga Technical University, Riga, Latvia
grabis@rtu.lv

³ Stockholm University, Stockholm, Sweden
{martinh, js, jelenaz}@dsv.su.se

⁴ University of Rostock, Rostock, Germany
{hasan.koc, kurt.sandkuhl}@uni.rostock.de

Abstract. Competitiveness and growth on an international market is for many businesses tightly coupled to their ability of quickly implementing new company strategies, business services and products or market entries. Capability management is among the approaches proposed to tackle these challenges. A feature is capturing the context of capability delivery and providing mechanisms for configuring the delivery. Among the work on capability management is the capability-driven design and delivery (CDD) approach that has been proposed by the EU-FP7 project CaaS. The aim of this paper is to contribute to CDD by (i) introducing different strategies for capability modelling, (ii) elaborating on the differences between these strategies, and (iii) contributing to an understanding of what strategy should be used under what preconditions. The paper addresses these aspects by describing the strategies and initial experiences gathered with them.

Keywords: Capability modelling · Capability management · Goal-first strategy · Process-first strategy · Concept-first strategy

1 Introduction

Competitiveness and growth on an international market is for many businesses tightly coupled to their ability to quickly implement new company strategies, business services and products or market entries. Businesses that offering products and services based on information technology (IT) require a way to efficiently adapt their services together with the IT infrastructure for delivering them. Capability management is among the approaches that have been proposed to tackle these challenges. One of the key features is capturing the delivery context of customer services and providing mechanisms for configuring or generating its delivery.

Authors are ordered alphabetically to reflect equal contributions to this work.

© Springer International Publishing Switzerland 2015

A. Persson and J. Stirna (Eds.): CAiSE 2015 Workshops, LNBIP 215, pp. 40–52, 2015.

DOI: 10.1007/978-3-319-19243-7_4

Among the approaches to capability management the capability-driven design and delivery (CDD) approach has been proposed by the EU-FP7 project CaaS [14, 15]. Recently, work on CDD has focused on development strategies for capability management that so far have been designed and applied independently of each other. There is not enough knowledge about the differences among the strategies and their suitability for different enterprise contexts.

The contributions of the paper are: (i) description of different strategies for capability modelling, (ii) discussion of the differences of the strategies and (iii) analysis of the preconditions of each strategy and its fit for different organisational situations. The initial experiences of using the strategies are also presented.

The rest of the paper is structured as follows. Section 2 summarises the background for the work from the area of capability management and presents the main aspects of the CaaS approach to capability design and delivery. Section 3 discusses the strategies for capability modelling and initial experiences. Section 4 presents a comparative analysis of the approaches. Section 5 summarises the work and draws conclusions.

2 Background

This section briefly summarises the existing work in the area of capability management and the CDD approach developed in the CaaS project.

2.1 Capability Management

The term capability is used in different areas of business information systems. In the literature there seems to be an agreement about the characteristics of the term capability, but still there is no wide acceptance of the term. The definitions mainly put focus on “combination of resources” [2], “capacity to execute an activity” [1], “perform better than competitors” [4] and “possessed ability [8]”.

The capabilities are enablers of competitive advantage; they should help companies to continuously deliver a certain business value in dynamically changing circumstances [5]. They can be perceived from different organisational levels and thus utilised for different purposes. According to [6] the performance of an enterprise is the best, when the enterprise maps its capabilities to IT applications. Capabilities are directly related to business processes that are affected from the changes in context, e.g., regulations, customer preferences and system performance. As companies in rapidly changing environments need to foresee the variations and respond to them [3], the affected processes and services need to be adjusted quickly. I.e., adaptations to changes in context can be realised promptly if the required variations to the standard processes have been anticipated and defined in advance and can be instantiated.

In the CaaS project capability is defined as *the ability and capacity that enable an enterprise to achieve a business goal in a certain context* [15]. Ability refers to the level of available competence, where competence is understood as talent intelligence and disposition, of a subject or enterprise to accomplish a goal; capacity means availability of resources, e.g. money, time, personnel, tools. This definition uses the

notion of context, thus stresses the need to take variations of the standard processes into consideration. To summarise, capabilities are considered as specific business services delivered to the enterprises in an application context to reach a business goal. To facilitate capability management, we propose business service design explicitly considering delivery context by an approach that supports modelling both the service as such and the application context.

2.2 The CaaS Approach to Capability Design and Delivery

The main goal of the CaaS project is to facilitate a shift from the service-oriented paradigm to a capability delivery paradigm. This paradigm shift requires development a new methodical framework supporting capability-driven design and development in general and it also requires changes in the development processes and engineering methods used. The CDD approach is supported by a set of method components.

The CaaS methodology for capability-driven design and development consists of the following top-level method components; Capability design, Enterprise modelling (EM), Context Modelling, support for Reuse of capability design, and Run-time delivery adjustment. The *Capability design* method component contains three approaches to start with capability design (goal, process and concepts). The *EM* component is required for identifying business goals related to the capability under consideration. EM includes for example business process modelling and goal modelling. The *Context modelling* component is used to specify the potential context situations of the capability. The component for supporting *Reuse* specifies how the concept of patterns can be used to elicit, elaborate and store enterprise models. The component for *Run-time capability adjustment* concerns the specification of adjustments needed in order to cater to changes in capability context.

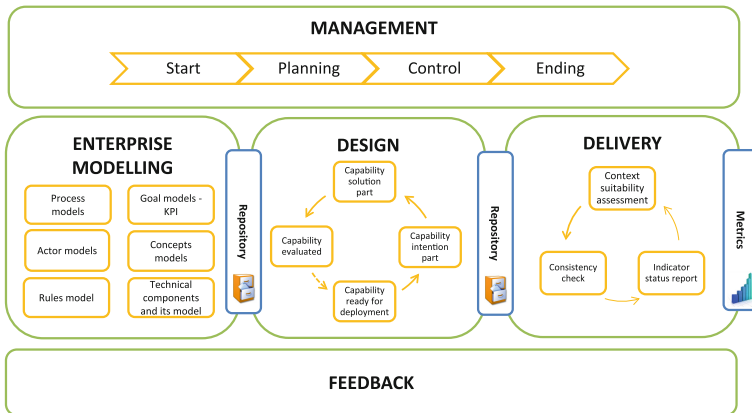


Fig. 1. CDD methodology overview

The CDD methodology is structured according to three phases (Fig. 1). The *EM phase* represents the traditional approach to business design and development of information systems. Existing EM approaches (e.g. EKD [7, 8] and 4EM [9]) will be

applied to give input for the capability design. Other EM methods are also potentially applicable. The capability *design phase* concerns the design of capabilities (see next section.) The *delivery* phase entails the execution and monitoring of the capability.

3 Strategies for Capability Modelling

Different strategies for capability modelling and design have been explored and so far three strategies have been elaborated for use by the industrial partners. The strategies consist of three steps. As the different strategies are basically proposing different ways to identify and design capabilities, only step 1 is different for the strategies whereas steps 2 and 3 are the same for all strategies:

Step 1, capability design: there are three alternative pathways of proceeding with capability design – starting with goals, starting with business service processes, or starting with business concepts. Each pathway is described in the Sects. 3.1 to 3.3.

Step 2, capability evaluation: Capability evaluation checks whether the result of capability design is feasible from the business and technical perspective before committing to capability implementation. The capability feasibility can be assessed using simulation and cost/benefit analysis. A failed evaluation may trigger a new cycle of the capability design phase.

Step 3, development of capability delivery application: The capability development activity prepares the capability for the deployment. The indicators for monitoring and algorithms for runtime adjustments are packaged as a runtime-support application. The capability design also serves as a basis for modifying or implementing the IT components used for capability delivery.

Furthermore, all three strategies are based on the CaaS meta-model [15]. Relevant terms are shown in Table 1. They will be used in Sects. 3.1 to 3.3.

Table 1. Concepts used in capability design strategies

Concept/term	Explanation
Capability	Capability is the ability and capacity that enable an enterprise to achieve a business goal in a certain context
KPI	Key Performance Indicators (KPIs) are measurable properties that can be seen as targets for achievement of Goals
Context Element	A Context Element is representing any information that can be used to characterise the situation of an entity
Context Set	Context Set describes the set of Context Elements that are relevant for design and delivery of a specific Capability
Context Element Range	Context Element Range is used to specify boundaries of permitted values for a specific Context Element and for a specific Context Set
Goal	Goal is a desired state of affairs that needs to be attained. Goals can be refined into sub-goals and expressed in measurable terms such as KPIs
Process	Process is series of actions that are performed in order to achieve particular result. A Process supports Goals and has input and produces output in terms of information and/or material
Process Variant	Process variant is a part of the Process, which uses the same input and delivers the same outcome as the process in a different way
Pattern	Patterns are reusable solutions for reaching business Goals under specific contexts. The context defined for the Capability (Context Set) should match the context in which the Pattern is applicable

3.1 Goals-First Capability Design

Goals are used to model the intentional perspective of organisational design. Hence, capability design can start with analysing the existing goal hierarchy and/or setting new goals and then shifting to analysing how they should be reached in terms of capabilities, business processes and which context properties should be considered. The modelling process includes steps as outlined below. It is also worth mentioning that the modelling process is not entirely sequential. Instead it is rather iterative and incremental, i.e. the modeller has to develop all parts of the capability design (e.g. goals, capability, context, process) in balance and consistent with each other.

Activity G1: Analyse the overall business vision and goals. The existing business goals need to be analysed with respect to identifying which goals are required to be supported by capabilities. Goals are typically organised in a goal hierarchy with more strategic goals on the top and more operational goals below. The goals that require capability support typically are on a more operational level because capabilities are concerned with explicit business designs and concrete actions. But in principle, some top-level goals could also be supported by capabilities. The goals that are the likely candidates for capability design should have elaborated KPIs for monitoring. If the existing goals do not have an explicit design that connects the KPIs and goals, then this needs to be established. A goals model is central for this activity.

Activity G2: Identify specific capabilities required by goals. The goals are analysed, relevant capabilities are defined and relationship between the capabilities and the goals established to indicate how the capabilities support the goals. The contribution of a capability to a goal should also be analysed with respect to the whole goal hierarchy, e.g. if a capability is deemed to support several sub-goals in the same goal hierarchy, then it might be more appropriate to associate it with their top-goal.

Activity G3: Analyse the existing business processes. The relationship to existing business processes is to be established. If the business processes documentation exists and processes are linked to goals, then the capability should be associated with the same business processes that its goals are. This is sufficient at this stage because the process variants will be designed later, once the application context is modelled.

Activity G4: Identify and model the context affecting the identified capabilities. For each capability the context set in which it is applicable should be defined, including the relevant context element and the ranges within which the capability is applicable.

Activity G5: Analyse and define process variants. A capability is realised by a set of business processes. In many cases there are some adjustments to capability delivery according to changes in context. To produce a more complete capability design, the overall business process is analysed and potential context changes are assessed to identify variations in capability delivery. This leads to defining process variants.

Activity G6: Model delivery adjustments. The needed capability delivery adjustments and links them to the capability design are specified. The adjustments are defined by analysing the context changes and associated process variants.

Activity G7: Review and/or incorporate relevant patterns. Capability design is based on the existing best practices. Hence we foresee that at any stage of this process the

capability designer should be able to review the existing patterns that present relevant best practices in the form of process variants, concept models, algorithms and include them in the capability design.

Our initial experiences with the goal-first strategy to capability design come from two cases within the CaaS project. Both cases are in a public organisation offering on-line services for citizens. The gathered experience concerns the *starting point for goal modelling*, and reflections on the way *goals, KPI and capabilities are interlinked*.

Regarding the first experience, *the starting point for goal modelling*, it was clear from both cases that the threshold for starting the goal modelling was low. This was due to two factors: either the organisation had top-level goals clearly defined or the organisation had a sense of what changes that were needed (this was the case for the two cases at the public organisation). E.g., in one case the public organisation had existing problems with their SOA platform and therefore defined several goals intended to improve the platform. Thus, the starting point could be either goals/visions or existing issues.

While working with the goal-first capability design strategy it was clear that the ways that the *goals were interlinked with KPIs and capabilities* was non-trivial. E.g., in the public organisation it was difficult to identify KPIs that could be used to monitor the capability delivery. While it is natural to define KPIs that are related to high-level business goals, such as the goal “To provide incident resolution”, it was more difficult to express KPIs to provide valuable information during monitoring of capability delivery. A conclusion here is that a set of guidelines for KPI definition based on capability goals are needed. E.g., asking questions such as “What would be a lead indicator to show when there is a high risk that my capabilities will not fulfil the goals?” can guide this part of the goal-first strategy.

3.2 Process-First Capability Design

The process-first capability modelling pathway proposes that the starting point of the capability design is a process underlying a business service. The business service is further refined and extended by adding context awareness and adaptability, so as to establish a capability that can deliver this service in varying circumstances.

Many organisations at this level have already defined and modelled business processes that are implemented to offer business services. Hence, the process-first capability design assumes that services are modelled and implemented as business process models. The modelling process includes the following activities:

Activity P1: Define scope. The organisation offers services based on business processes that are already modelled. To design the capabilities by means of business processes the capability designer first selects the service and sets the scope of the capability design depending on various factors, such as optimising the services with high process costs or managing services that frequently change and require the adjustment of business processes.

Activity P2: Define level of granularity. This step defines the abstraction level, at which the processes supporting the business service to be improved are identified and

analysed. An option to describe different levels of granularity could be applying the decomposition method proposed by [12]. This method differentiates between a *main process*, which does not belong to a larger process and is decomposed into sub-processes. Regarding the business goals and offered capabilities needed to reach them, the method user most probably models at *main process* level. Nevertheless, the main processes might be refined by sub-processes in Activity P3.

Activity P3: Identify processes/activities/tasks. This identifies the processes modelled and used by the enterprise that are relevant for capability delivery. For this purpose 5-policies approach proposed by [11] can be applied, which describes a general strategy for identifying processes. It should be emphasised that capturing possible variations of the processes are not included in these activities.

Activity P4: Analyse existing BPM and refine if necessary. The activity assures that selected business process models are up-to-date and applies changes if required.

Activity P5: Identify or name the capability to be delivered. The capability designer has a view on selected business service and supported processes. The information that the capability designer acquired during the execution of prior steps can be used as an input to establish a capability definition. The level of granularity is to be considered when identifying capabilities since overly refined capabilities could lead to complex models. If the capabilities are unclearly defined or cannot be established at this level yet, then the modeller should execute the upcoming activities to identify the context elements via observing the variations. In that case the capability is refined or established in Activity P9, where the method user has an enhanced view on the goals, processes and context elements as of in Activity P5.

Activity P6: Update goals model and KPIs. The capability designed should be aligned with the goals that an enterprise aims to achieve. To check if business goals are satisfied during the capability delivery, KPIs are used to measure the achievement of goals. This steps analyses and updates the goal models as well as KPIs, if any exist. If no goals model is available, then the designer continues to activity P7.

Activity P7: Develop goals model and KPIs. Goals define the requirements to be fulfilled during the delivery of a capability. If the enterprise has no goals model, then it is created in alignment with the scope of the business service. To check if business goals are satisfied during the capability delivery, KPIs are used to measure the achievement of goals. The modeller develops KPIs required by the goals developed in this activity or updated in the aforementioned activity.

Activity P8: Relate goals, capabilities and processes. This activity establishes the connection between the developed/identified/analysed components. The behaviour of the components under varying situations should be studied in the following step.

Activity P9: Identify and model context. A capability is defined by specific business services, a defined application context for these business services and goals of the enterprise to be reached. The context of the capability delivery, is modeled i.e. the potential application context where the offering is supposed to be deployed. Four method components are used, namely, find variations, capture context element, design context, and prepare for operational use.

Activity P10: Model delivery adjustments. This activity outlines the components needed to adjust capability adjustment at runtime and refers to the method component.

Activity P11: Link components. This activity finalises the capability model with interlinking the outputs that have been previously developed.

Our initial experiences with the process-first strategy to capability design are based on two cases of the CaaS project. The first case is from a software vendor in the utilities industry, which owns a business service provider (BSP) executing a complete business process for a business function outside of an organisation. The second case is from a public organisation providing electronic services to municipalities, which are then used by citizens. The gathered experience concerns the eliciting of context elements, modelling goals and process variants, and identification of patterns.

In both use cases the business processes required for capability delivery were modelled. Regarding the first experience, *context element eliciting*, we became aware of the need to differentiate between process variables and context elements, since both types of parameters originate during the execution of the process but have different design and runtime implications. For this purpose, we provided guidelines on what constitutes a context element and which parameters should be treated as process variables. Moreover, a framework was proposed to classify the parameters causing variability in the business processes. The parameters were analysed in detail to decide whether they qualify as context elements or process variables. Concerning the *modelling of process variants*, the use cases have also proven that the values of such parameters have an influence on the decision logic and it is sometimes confusing to distinguish variability from business processes decisions. To solve this problem the term variation point and an initial set of guidelines were introduced. Particularly in the second use case we identified services that share the same parts of the business process models, which only to some extent differ from one another. Here we proposed a new primitive “variability refinement” to eliminate redundancy.

The process-first capability design strategy requires an analysis of business process models. This allowed us to investigate whether there were processes representing the best practices in the company, which can be captured as *patterns*. In both use cases *goals model* did not exist and had to be developed from the scratch. Here, the methods and guidelines in [12] were applied for goals modelling. Moreover the involvement of the domain experts and product owners to modelling sessions was required.

3.3 Concept-First Capability Design

Concepts are used for modelling the static aspects of the business, such as product structures, organisational structures, customer profiles, material, as well as information used and produced by the business processes. The concept models can be seen as knowledge models of the organisation. Capabilities may be designed by starting with analysing the existing knowledge structures and their relationships with the application context. The following activities illustrate such a way of working:

Activity C1: Analyse the existing concepts. This step aims to identify concepts describing relevant products and/or services that are realised in the company. They may

be modelled as a whole or as aggregate concepts; in some cases the concepts associated with the supporting information structure is also modelled.

Activity C2: Elicit candidate capabilities. The purpose of this step is to identify products or services (modelled as concepts) that need to be realised by capabilities taking into account the findings from analysing the dependencies in the previous step.

Activity C3: Analyse dependencies between the identified capabilities and existing business processes and business goals. This activity aims to identify which business goals are relevant and what are the KPIs that monitor their achievement, as well as what business processes are used to realise the concepts, e.g. products have development, production, sales and support processes. In some cases these processes might not be fully defined and/or modelled and if so, the necessary models may need to be defined later if they are influenced by the context and subjects to variability. New or additional business goals and KPIs might also be defined at this step.

Activity C4: Identify the context affecting the identified capabilities. For each capability the context set in which it is applicable should be defined. Note that this activity is the same as G4 in the goal-first strategy.

Activity C5: Analyse and define process variants. A capability is realised by a set of business processes, there might be different process variants needed for different product versions, or some variations of the manufacturing process need to be introduced because of different material or customer requirements. This activity corresponds to G5 in the goal-first strategy.

Activity C6: Model delivery adjustments. This step specifies the needed capability delivery adjustments and links them to the overall capability design. This activity corresponds to G6 in the goal-first strategy.

Activity C7: Review and/or incorporate relevant patterns. This activity corresponds to G7 in the goal-first strategy.

Our experiences with the concepts-first approach come from one case in the CaaS project. The case is within an organisation operating in the maritime business. The scope of the case was to apply the CDD approach to analyse the capabilities of the organisation, with a focus on its compliance with maritime regulations. The organisation already had a well-defined organisational structure, with well-defined areas of responsibilities, thus the concept-first strategy was deemed suitable. Our initial experiences with the concept-first strategy concerns the *drivers for capability definition* and the *complexity of mapping capabilities to goals and processes*.

The *drivers for capability definition* are slightly more complex when using a concept-first strategy, compared to a process or goal-first strategy. The reason is that process and goal models often have an inherent hierarchical structure, and it is thus possible to identify a certain level that correspond to, or easily maps to, capabilities. For the concept-first approach it is instead necessary to look at groups of concepts that are related (cohesion within the concepts) and how they relate to other groups (coupling between groups). For the maritime case, the basis for the analysis became the management structure. This resulted in capabilities such as “Maritime management”, “Ship financial management” and “Ship technical management”.

From the maritime case, we also experienced *the complexity of mapping capabilities to goals and processes*. For the goal-first and processes-first strategies, one of the mappings to goals or process are done as an integral part of the capability identification process, sometimes leading to a 1:1 mapping between goals/processes and capabilities. However, when applying the concept-first strategy, we discovered that it was common that one identified capability was mapped to several goals. To handle this, the mapping was visualised in matrices, linking capabilities to goals.

4 A Comparative Analysis of the Modelling Strategies

The goal-first strategy, naturally, has the primary view that capabilities exists as a way to fulfil long-term business objectives of an organisation. To get *started* with the goal-first strategy it is therefore beneficial, i.e. to capture the intentional aspect of a business and to merit it through well-defined KPIs. Since the strategy is focused on the elaboration of goals and KPIs, it provides at the same time the basis for run-time monitoring of the quality of capability delivery in respect to the established KPIs.

The primary view of the process-first strategy is that capabilities are delivered through enacting the business processes. This strategy assumes that specifications of the current business processes exist or, at least, that the organisational culture is oriented towards processes. Domain experts and product owners are required to identify context influences on the processes, and to (re)design process variants and adaptation business rules accordingly. Strategic management staff should be involved during goal model creation or update. Representative workers may also provide valuable input concerning current problems and operational needs. The process-first strategy suits organisations with well-conceived and stable processes, since any non-trivial reengineering of processes requires revising the capability designs. The advantage is that the resulting capability delivery fosters context awareness within the organisation and enables flexible variability management in the running processes.

The concept-first strategy views concepts as the primary means for capability identification and is assumed to be based on well-defined products or organisational structures. The concept-first strategy are likely to fit organisations that have a stable product or organisation structure, while not necessarily having well defined goals or strong process orientation.

Following the above discussion, we identified different aspects suitable for comparing the developed strategies. These aspects are:

- Primary view on capabilities. What are the bases for capability identification?
- Preconditions with respect to models. What kind of preconditions with respect to existing models or specification is needed for using the strategy?
- Stakeholders required. Besides the actual capability “modeller”, the strategies have different requirements regarding what stakeholders (domain experts, capability owners, product owners, etc.) need to be involved during the strategy.
- Effects on the succeeding steps. What are the next modelling or development steps followed by the strategy?

Table 2. Comparison of capability modelling strategies

Aspect of comparison	Strategy		
	Goal-first	Process-first	Concept-first
Primary view on capabilities	A capability fulfils key organisational goals	A capability is operationalised as a set of processes	A capability encompasses the management of key concepts
Preconditions with respect to models	Ideally, top-level organisational goals should be defined	Pre-existing business process specifications or process-oriented culture	Pre-defined management structures, product structures or other conceptual models
Stakeholders required	Different levels of management personnel	Domain experts, product owners, strategic management	Product managers/owners
Effects on the succeeding steps	Provides a comprehensive base for capability monitoring by the use of KPIs	Provides a detailed specification for context-aware variability management	Provides a base for having the concept of capabilities as the main subject for organisational analysis and change management
Characteristics of enterprise	Organisations with a high degree of adaptable/non-routine work	Mature organisations with well-established processes	Organisations with a well-defined and stable organisational or product structure
Degree of flexibility of the strategy	Can also start with visions or existing issues. Highly iterative and incremental modelling process	The strategy can cope with ill specified goal or concept models. Process reengineering requires thorough revision of capability designs	Can cope with different levels of concept granularity. The drivers for capability definition are slightly more complex. Is flexible with regard to the degree of specification of business processes
Impact on the organisational culture	Reinforces strategic vision and clarifies the IT-business alignment	Improves the perspective of the enterprise (or service) context	It brings perspective over the organisational concepts by identifying (highly-cohesive and lowly-coupled) groups of concepts

- Characteristics of enterprise. What are the characteristics of an enterprise (regarding size, domain or type) the strategy is expected to be useful for?
- Degree of flexibility of the strategy. To what extent does the strategy allow for adapting the flow of activities to project contingencies? (e.g. changes in the order of activities, dealing with missing or unreliable input, etc.)
- Impact on the organisational culture. What is the impact that the strategy is expected to have on the way the organisation conceives their services? The fact of introducing novel perspectives on the enterprise strategy, structure and work practice may alter how the own organisation views their current capabilities.

Table 2 summarizes how the three strategies relate to the above aspects.

5 Summary and Future Work

Based on the capability design and delivery approach developed in the CaaS project this paper has proposed three different strategies for capability modelling, presented initial application experiences and compared them.

During the application of the three strategies, several areas for improvement have been identified. For the goal-first strategy, it is evident that there is a need for guidelines when it comes to the identification of KPIs that can be used for the monitoring of capability delivery. The concept-first strategy, being intuitively easy to follow, also needs stricter guidelines for capability identification. The process-first strategy has shown that the guidelines and activities for analysing the inputs influencing the business processes are needed to (i) identify context elements accurately, (ii) distinguish variation points from decision logic and (iii) model process variants efficiently. Moreover, we observed that the tasks in the business process models reflect the knowledge of domain experts and exhibit insights to best practices of the organisations, which constitute a good starting point for pattern analysis.

Future work entails merging parts of the three approaches, to build on their respective strengths.

Acknowledgments. This work has been performed as part of the EU-FP7 funded project no: 611351 CaaS – Capability as a Service in Digital Enterprises.

References

1. Jiang, Y., Zhao, J.: An empirical research of the forming process of firm inter-organizational e-business capability: based on the supply chain processes. In: 2010 2nd International Conference on Information Science and Engineering (ICISE), pp 2603–2606 (2010)
2. Antunes, G., Barateiro, J., Becker, C., et al.: Modeling contextual concerns in enterprise architecture. In: 2011 15th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW), pp 3–10 (2011)
3. Eriksson, T.: Processes, antecedents and outcomes of dynamic capabilities. *Scand. J. Manage.* **15**(2), 142–155 (2013). doi:[10.1016/j.scaman.2013.05.001](https://doi.org/10.1016/j.scaman.2013.05.001)

4. Boonpattarakan, A.: Model of Thai small and medium sized enterprises' organizational capabilities: review and verification. *JMR* **4**(3), 1–28 (2012). doi:[10.5296/jmr.v4i3.1557](https://doi.org/10.5296/jmr.v4i3.1557)
5. Stirna, J., Grabis, J., Henkel, M., Zdravkovic, J.: Capability driven development – an approach to support evolving organizations. In: Sandkuhl, K., Seigerroth, U., Stirna, J. (eds.) *PoEM 2012. LNBIP*, vol. 134, pp. 117–131. Springer, Heidelberg (2012)
6. Chen, J., Tsou, H.: Performance effects of 5IT6 capability, service process innovation, and the mediating role of customer service. *J. Eng. Tech. Manage.* **29**(1), 71–94 (2012). doi:[10.1016/j.jengtecman.2011.09.007](https://doi.org/10.1016/j.jengtecman.2011.09.007)
7. Loucopoulos, P., Kavakli, V., Prekas, N., Rolland, C., Grosz, G., Nurcan, S.: Using the EKD approach: the modelling component (1997)
8. Bubenko, J.A.J., Persson, A., Stirna, J.: User Guide of the Knowledge Management Approach Using Enterprise Knowledge Patterns. Deliverable D3 IST Programme project Hypermedia and Pattern Based Knowledge Management for Smart Organisations, Project no. IST-2000-28401 (2001). [ftp://ftp.dsv.su.se/users/js/d3_km_using_ekp.pdf](http://ftp.dsv.su.se/users/js/d3_km_using_ekp.pdf). Accessed February 2015
9. Sandkuhl, K., Stirna, J., Persson, A., Wißotzki, M.: Enterprise Modeling – Tackling Business Challenges with the 4EM Method. The Enterprise Engineering Series. Springer, Heidelberg (2014). ISBN 978-3-662-43724-7
10. Brézillon, P., Cavalcanti, M.: Modeling and using context. *Knowl. Eng. Rev.* **13**(2), 185–194 (1998)
11. Hallerbach, A., Bauer, T., Reichert, M.: Capturing variability in business process models: The Provop approach. *Journal of Software Maintenance and Evolution* **22**(6–7), 519–546 (2010)
12. Milani, F., Dumas, M., Ahmed, N., Matulevicius, R.: Modelling Families of Business Process Variants: A Decomposition Driven Method, *CoRR* abs/1311.1322 (2013)
13. Bērziša, S., Bravos, G., Gonzalez Cardona, T., Czubyko, U., España, S., Grabis, J., Henkel, M., Jokste, L., Kampars, J., Koc, H., Kuhr, J., Llorca, C., Loucopoulos, P., Juanes Pascual, R., Sandkuhl, K., Simic, H., Stirna, J., Zdravkovic, J.: Deliverable 1.4: Requirements specification for CDD, CaaS – Capability as a Service for Digital Enterprises, FP7 project no 611351, Riga Technical University, Latvia (2014)
14. Sandkuhl, K., Koç, H., Stirna, J.: Context-aware business services: technological support for business and IT-alignment. In: Abramowicz, W., Kokkinaki, A. (eds.) *BIS 2014 Workshops. LNBIP*, vol. 183, pp. 190–201. Springer, Heidelberg (2014)
15. Bērziša, S., Bravos, G., Gonzalez Cardona, T., Czubyko, U., España, S., Grabis, J., Henkel, M., Jokste, L., Kampars, J., Koç, H., Kuhr, J.-C., Llorca, C., Loucopoulos, P., Juanes Pascual, R., Pastor, O., Sandkuhl, K., Simic, H., Stirna, J., Zdravkovic, J.: Capability Driven Development: An Approach to Designing Digital Enterprises. *Business and Information Systems Engineering (BISE)*, February 2015. doi:[10.1007/s12599-014-0362-0](https://doi.org/10.1007/s12599-014-0362-0)