Modeling Enterprise Capabilities with *i**: Reasoning on Alternatives

Mohammad Hossein Danesh[1] and Eric Yu[2]

[1] Department of Computer Science, University of Toronto, Toronto, Canada
[2] Faculty of Information, University of Toronto, Toronto, Canada
{danesh,eric}@cs.toronto.edu

Abstract. In a dynamic world, information technology (IT) systems are expected to provide capabilities that can be used to address evolving needs. Recent work has adopted notions of capability to model how IT systems meet enterprise goals. In this paper we draw upon theories of dynamic capabilities from strategic management to model enterprise capabilities, reason on their development choices, orchestration alternatives and deployment configurations. The modeling approach builds upon *i** and proposes to model capabilities as actors. *i** modeling supports reasoning about intangible and tangible requirements of capabilities and trade-offs among alternatives. We illustrate with examples from the insurance industry. The examples show how social and non-functional dependencies among capabilities affect decisions about development, orchestration and configuration alternatives.

Keywords: Enterprise Modeling, Capability Modeling, Dynamic Capability, Resource Orchestration, iStar.

1 Introduction

Today's rapidly changing business environment requires dynamically evolving IT enabled competencies [1]. This dynamic requirement has changed the focus of software architecture from functional composition to dynamic configuration [2]. Approaches such as Service-Oriented Architecture (SOA), Model Driven Development (MDD), and software ecosystems facilitate more agile system development to support dynamic requirements of enterprises [1,2]. However a gap still exists between enterprise-level business requirements analysis on the one hand, and software engineering approaches that produce IT artifacts on the other [1]. Technical approaches for achieving adaptability such as context-aware and service-oriented systems allow run-time configuration in response to dynamic functional and non-functional requirements. The focus of the adaptation is primarily based on technical performance criteria rather than on higher-level business values and strategies [1].

In this paper, we focus on the need for IT systems to respond to the dynamic nature of strategic business requirements. Concepts and theories from strategic management have been adopted in enterprise modeling, raising the level of abstraction to analyze

L. Iliadis, M. Papazoglou, and K. Pohl (Eds.): CAiSE 2014 Workshops, LNBIP 178, pp. 112–123, 2014.

IT architectures and to better achieve business-IT alignment [1]. Capabilities, defined as "an organization's ability to appropriately assemble, adapt, integrate, reconfigure and deploy valued resources, usually, in combination or co-presence", have been recognized as a primary source of business profitability and competitive advantage [3,4]. IT capabilities have been shown to create competitive advantage when they form rare, valuable and difficult to replicate orchestrations [5,6].

Enterprise "core" capabilities that lead to competitive advantage include knowledge and skills embodied in people, business processes, as well as technical systems. In developing capabilities, an enterprise faces many choices and alternatives. These alternatives and choices are limited and influenced by the organization's past, its environment, governance structure, and organizational cultures and norms [4], [7,8].

Based on a review of the literature, we consider alternatives that occur in three stages: (1) Capability Development: How to build or acquire resources to form a capability. The choice to either develop a sales system in-house, or to adopt a software-as-a-service solution is an example of such alternative. The skill sets and resources required for each are significantly different. (2) Capability Orchestration: How to bundle the capabilities and which bundles to choose. For example, use the enterprise data warehouse or data virtualization servers for the in-house sales system. (3) Capability Deployment Configuration: How to configure the capability at deployment time. For example, for in-house implementation, to rely on the existing IT department, or to allow the sales team to hire their own IT staff.

Recognizing the socio-technical nature of such reasoning, we aim to explore the potential of *i** to express and reason about the three types of capability alternatives. The *i** modeling framework has been developed to capture dependencies and rationales of actors' strategic interests [9]. The representation of actors, dependencies, intentions and their alternatives has the potential to illuminate the social context of capabilities, their reliance on one another to create competitiveness, and the alternate choices available. The analysis will include how capability development alternatives affect capability orchestration and whether *i** dependencies can effectively represent such relations.

In section 2, related work on capability modeling in enterprises is reviewed. In section 3, we illustrate why capability is important in strategic management and what are the management processes and drivers of decision making. Section 4 discusses the suitability of *i** to model enterprise capabilities and their alternatives. In section 5 we discuss hypothetical alternatives for capability development, orchestration and deployment configuration of an insurance service provider. The paper is concluded in section 6.

2 Related Work

Capability modeling has recently been used to represent business investment profile, facilitate business-IT alignment, and support service design and mapping. In this section we review recent approaches to capability modeling, and consider their capacity to analyze alternative decisions regarding capability development, orchestration and deployment configuration.

Iacob et al. [3] propose an extension to ArchiMate V2 and use capability and resource modeling to facilitate modeling business strategy concepts and architecture-based approaches to IT portfolio valuation. ArchiMate is an enterprise architectural modeling language that facilitates integration of business, application, and technology architectures. A recent extension proposed by Iacob et al. [3], aims to capture the business value of IT artifacts and projects in order to achieve better alignment with business strategy. The capability construct is used to facilitate the alignment. The modeling framework can model constraints imposed on capabilities and align implementation architecture accordingly. However reasoning on sources of the constraint is not modeled. The models cannot depict relations among capabilities, stakeholders and the value creation logic. Furthermore, reasoning on capability alternatives or the influence of the alternatives on one another is not considered.

The DoD Architecture Framework (DoDAF) version 2 [10] emphasizes the importance of capabilities and has dedicated a viewpoint to facilitate capability deployment planning, implementation, monitoring and preservation. This viewpoint consists of capability vision, taxonomy, phasing, dependencies, organizational development mapping, operational activity mapping and service mapping to business capabilities. DoDAF facilitates the description of capabilities and their dependencies, and their mapping to operational components and services, but does not facilitate reasoning on capability alternatives, their orchestration and intentions of the dependencies between capabilities.

Capability maps [11] are used to present a library of organizational capabilities at different levels of granularity. Capability heat maps use color codes to visualize hot spots within the capability landscape. The Value Delivery Model Language (VDML) [12] extends the use of capability maps and links capability offerings to organizational value network. Capabilities in VDML are mostly treated as resources required to perform an activity. The dependencies modeled in VDML do not illustrate why a dependency exists and what kind of dependencies are required. VDML treats capabilities as assets that are needed to realize a business model. However they do not consider challenges concerning integration of two or more capabilities [2], [7].

Barroero et al. [13] present a capability-centric Enterprise Architecture (EA) that extends TOGAF with a business capability viewpoint. The authors use capability and business component maps to identify a modularization of the enterprise business portfolio. They use business components as IT clusters that provide and consume services and propose modularizing IT architecture accordingly. The collaboration diagrams used to describe interactions and dependencies between business components does not capture intensions behind the collaboration. The approach does not facilitate reasoning on alternatives available and how they affect one another.

Capability Driven Development (CDD) aims to facilitate smooth (nearly automated) transition to software development by modeling capabilities and the contexts in which they operate. CDD facilitates run-time adjustments to changing requirements by implementing contextualized patterns of capability execution. This approach allows selection among different service providers at design-time or run-time based on functional and non-functional requirements specified in the context [1], [14]. However CDD does not capture the socio-technical notion of capabilities, their relations and dependencies, alternatives influence on other capabilities and business goals. Capabilities in CDD are considered in isolation but not all compositions of capabilities and software artifacts work seamlessly without social and managerial support [2], [15].

3 Dynamic Capability and Strategic Management

Current modeling approaches as discussed in section 2 lack the ability to facilitate reasoning on how capability alternatives affect one another and organizational value creation. For conceptual foundations, we draw on the literature in strategic management, which focus on sustainable competitive advantage. The Resource Based Theory (RBT) [16] argues that sustainable competitive advantages is obtained by creating Valuable, Rare, Inimitable and Non-substitutable (VRIN) resources. Within the RBT, the Dynamic Capability View (DCV) [4] argues that VRIN resources are not sufficient. Organizations require a dynamic capability that can continuously integrate and reconfigure an organization's resource base to create strategic capabilities that are valuable, rare and difficult to replicate. RBT and DCV have been used extensively to analyze the role of IT in creating competitive advantage [5,6], [15].

To attain competitive advantage, an enterprise is faced with choices in multiple stages of the capability lifecycle and development [12]. Decisions on capability development are shaped by available physical capital, human capital, social capital, cognition, and the history of capabilities [8]. Sirmon et al. [17] identify capability management as including acquiring, building and retiring capabilities; bundling resources and processes to form capabilities; analyzing the combination of capabilities to use; and leveraging the right deployment strategy. Over time, capabilities acquire social identity and autonomy, particularly in decentralized organizations [4], [8].

Evaluation of capabilities and their strategic fit is challenging, especially when the capability in question is intangible or contains intangible elements [18]. The complementary nature of capabilities and reliance on one another to create competitive advantage [4] complicates decision making further. The complementary nature plays a more significant role when reasoning on IT capabilities as studies indicate that synergetic relations of IT capabilities can be sources of sustained competitive advantage. Managerial and social support is required to successfully integrate diverse capabilities (such as IT and business capabilities) [5,6], [15].

4 Suitability of *i** to Model Enterprise Capabilities

Drawing on the strategic management literature as outlined in the preceding section, we treat capabilities as intentional autonomous bundles of organizational resources that are built and evolved over time. The enterprise needs to decide among: (1) capability development alternatives: what to include and exclude and what resources to bundle into a capability (2) capability orchestration alternatives: which capabilities are complementary and what coordination mechanism suits them, and (3) Deployment configurations. There are four criteria (adapted from Molloy et al [18]) to consider when deciding on each alternative at any stage. (1) How does it affect the value creation logic? (2) How does the generated value benefit stakeholders? (3) How will it influence other capabilities that depend on it or that the capability depends on? (4) In what context is the alternative viable?

In this section we explore whether *i**, a socio-technical modeling framework [9], can facilitate capability modeling, reason about development alternatives and deployment configurations, and study the influence of alternate choices on one another. The *i** framework facilitates socio-technical exploration of enterprises by providing a graphical depiction of actors, intensions, dependencies, responsibilities and alternatives. *Actors* including agents and roles and associations between them (*is-a*, *part-of*, *plays*, *covers*, *occupies*, *instantiates*) represent the social aspect of *i**. Actor intentions are expressed within the actor boundary in *i** using actor's desired *goals* and *softgoals*, performed *tasks* and available *resources*. Softgoals in *i** are goals without clear-cut satisfaction criteria. In an *i** Strategic Dependency (SD) model, actors *depend* on each other to accomplish tasks, provide resources, and satisfy goals and softgoals. The reasoning of each actor is revealed in more detail in the Strategic Rationale (SR) model. *Decomposition* of a task within the boundary of an actor depicts elements required to accomplish it. *Means-ends* links illustrate alternatives available to achieve a certain goal. *Contribution* links, which can be *Make/Break*, *Help/Hurt*, *Some+/Some-* or *unknown*, show the effects of *i** elements on softgoals [19]. Horkoff and Yu [19] propose a qualitative, interactive evaluation procedure to reason on *i** goal models and alternatives.

We use *i** goal models to reason about how capabilities are constructed and achieved. In particular, we use *i** actors to model core capabilities which embody an identity that can act independently and is built over time, as emphasized by the dynamic capability view of strategic management [4], [8]. Capabilities are modeled as specialized actors (indicated with intertwined circles added to the i* symbol for actor). A position within the enterprise is responsible for a capability (in case of collaborating partners the position can be in a partner enterprise). A role within the organization can be associated with the position that is responsible for a capability and be dependent or depend on a capability. Examples of such associations are provided in the following section in Figures 4 and 5. In addition, by using *i** actors to model capabilities, we can reason on how a capability can resist or facilitate a development choice which is inspired by Leonard-Barton [7] and is presented in the next section (Figure 4).

Modeling capabilities as *i** actors allows us to: (1) analyze how different capabilities relate to one another. A map of capabilities dependencies will facilitate understanding consequences of a change to the capability. (2) Model capability dependencies alongside the social relations of influential actors within an organization that will facilitate reasoning on IT capability alternatives. This is supported by research findings that indicate alignment of IT capabilities to other organizational capabilities can have significant contribution to competitive advantage [5,6], [15]. The social dynamics and collaboration required for such alignment can affect the flexibility of the capabilities dependencies and their performance [20]. (3) Use SD models of capabilities and their relation with stakeholders (modeled as roles, positions and agents) as a roadmap to analyze how capabilities participate in organizational value creation and how the value is appropriated to stakeholders. Both value creation logic and value appropriation are dimensions proposed by Molloy [18] for evaluating the contributions of capabilities.

5 Illustrative Example

We illustrate with an example from the inventory of insurance industry capabilities provided by ACORD [21]. Figure 1 shows the SD model for the *Product Management* capability of an insurance company. The *Sales Representative*, shown as an *i** Role, depends on the *Product Development* capability to deliver a new product. That product along with other insurance services is provisioned to the consumer. Later in Figure 5 we present how the *Sales Representative* depends on the *Sales Management* capability to be paid. Figure 5 also depicts how the insurance enterprise makes money from the premium paid by the consumer. The aim of the SD model is not to quantify value creation and appropriation as done in value-based modeling approaches, but to illustrate the strategic dynamics that can facilitate reasoning on alternatives.

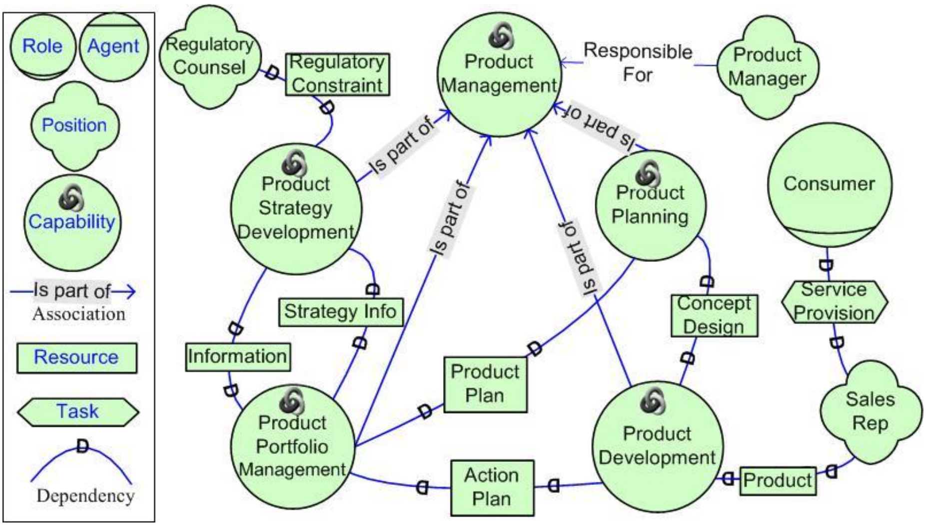

Fig. 1. Strategic Dependency Model - The Product Management Capability

Figure 2 depicts the *i** model of the *Product Management* capability and the sub-capabilities dependencies on one another. These dependencies are based on our understanding of the description provided by ACORD [21]. Some were explicitly mentioned and some were implicit. One would expect that some soft-goal dependencies exist between the capabilities but in the ACORD model none were specified. Such dependencies are exemplified in subsequent models in this paper.

Fig. 2. Capability Model for Insurance Product Management - Adopted from ACORD

It is our understanding that *Product Management* depends on its sub-capabilities and the integration of these capabilities requires further steps that are not mentioned in ACORD [21]. Therefore in Figure 2 the *Product Management* capability is not further explored. In the top left corner of Figure 2, the *Product Strategy* capability is modeled. The alternative presented in this model to achieve the goal *Define Corporate Direction for Product and Service Offering* is based on ACORD. There is a potential to change this implementation and develop new alternatives to achieve the goal. *Product Strategy* has two sub-capabilities: *Product Strategy Development* and *Product Research and Analysis.* . These third layer capabilities can be modeled as *i** actors and associated with *Product Strategy,* if their social and capability dependencies are significantly different from their parent. In this model we show the sub-capabilities as tasks as they describe business processes. Modeling the sub-capabilities as tasks allows specification of their requirements through decomposition and analysis of their dependencies on other capabilities. The decomposition presented in Figure 2 depicts our understanding of the description provided for the sub-capabilities. The *Product Strategy* capability has to produce strategies that are able to compete in the industry hence the softgoal of *Competitiveness* was added as an element. The modeled resources *Understanding of Internal Capacity* and *Regulatory Constraints* were mentioned as skills and knowledge requirements. *Monitoring Industry Trends and Opportunities* was mentioned as the main activity of the research and analysis process. Assets required to perform capabilities are modeled as *i** resources

The capabilities and their alternatives presented from this point onwards are hypothetical and not adopted from ACORD. These examples illustrate how by modeling dependencies between capabilities and analyzing their relations, one can facilitate decision making on capability alternatives. Figure 3 depicts how Product Strategy Development depends on Social Media Analytics. Legends are same as provided in Figure 2.

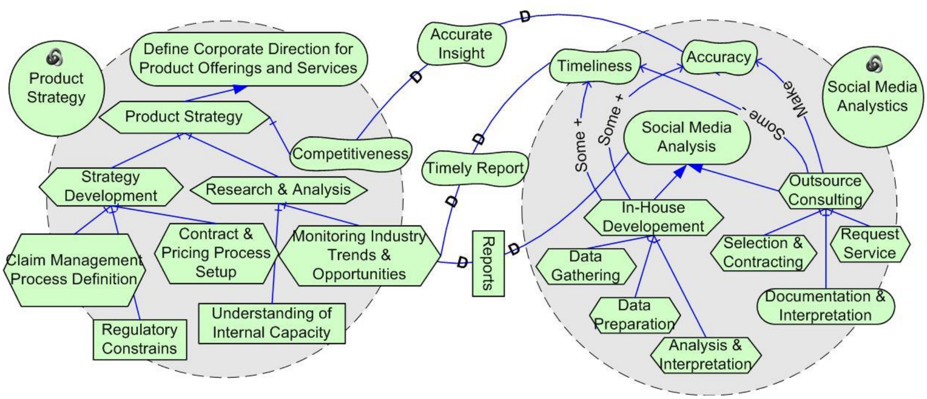

Fig. 3. Social Media Analytics - Capability Development Alternatives

The organization has two alternatives to satisfy its social media analytics requirements: (1) establish in-house processes that go through *Data Gathering*, *Data Preparation* and *Data Analysis and Interpretation;* or (2) hire a consulting firm that requires the following elements: *Request Service*, *Selection and Contracting* and *Documentation and Interpretation*. If the right consulting firm is chosen the accuracy of information would be guaranteed. On the other hand, following an in-house implementation will generate reports faster (softgoals of *Accuracy* and *Timeliness*). Considering dependencies in Figure 3, both alternatives can satisfy the reporting requirements. However the outsourcing option will provide better *Accuracy* and therefore better satisfy the *Competitiveness* of product strategy development. We would argue that since competitiveness plays a significant role in the organization's value generation, prioritizing its dependencies when making decision regarding capability development is justified. However in cases where their priority is not as trivial as competitive product offering, we need models that present a complete set of capabilities and trace value creation network. Such example is provided in Figure 5.

In this hypothetical case, it is assumed that the in-house *Social Media Analytics* capability exists in the organization (the left alternative in Figure 3). In this case it is highly likely that people behind the scenes will resist outsourcing. This resistance will harm the *Timeliness* softgoal in Figure 3 seriously which can cause interruption in product strategy development. Therefore when deploying the new alternative, the organization should use its in-house skill set to analyze and interpret data. How the *Social Media Data Analyst* within the organization would react towards each alternative is presented in Figure 4.

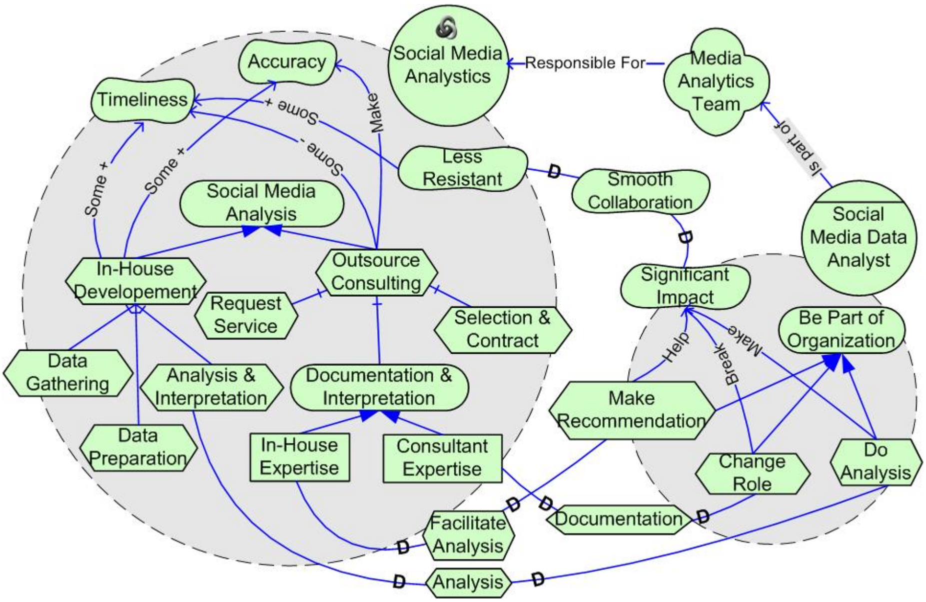

Fig. 4. Social Media Analytics - Capability Deployment Alternatives - Resistance to Change

In Figure 5, we introduce *Automated Sales Operation* which is an IT enabled capability that is part of sales management. The enterprise has two options: either purchase an *In-House* sales management product or use a *Software-as-a-Service* solution. If in-house implementation is chosen, the solution can rely on a *Data Virtualization Server* or the organization's *Data Warehouse*. The automated sales management is dependent on different IT capabilities in the two cases. The three alternatives also affect the softgoal dependency of the *Portfolio Management* capability. The decision in this case between software-as-a-service and in-house data virtualization is not trivial as both alternatives provide satisfactory levels of flexibility to access data. The *Sales Representative* depends on *Timely Operation* from the *Automated Sales Operation* capability which influences how the organization makes money as illustrated in Figure 5. This dependency will increase the importance of the *Fast Implementation* softgoal. If development and maintenance of this capability in the sales management department causes performance degradation in sales, then the software-as-a-service solution would be a better choice. In the presented analysis, one can further breakdown the alternatives (the two in-house implementation choices or software-as-a-service) if they influence decision making regarding capability development, orchestration or deployment configuration. However if no such influence exists, approaches such as CDD [14] (discussed in the related work section) can be used to facilitate design-time and run-time adaptation of capability implementation alternatives.

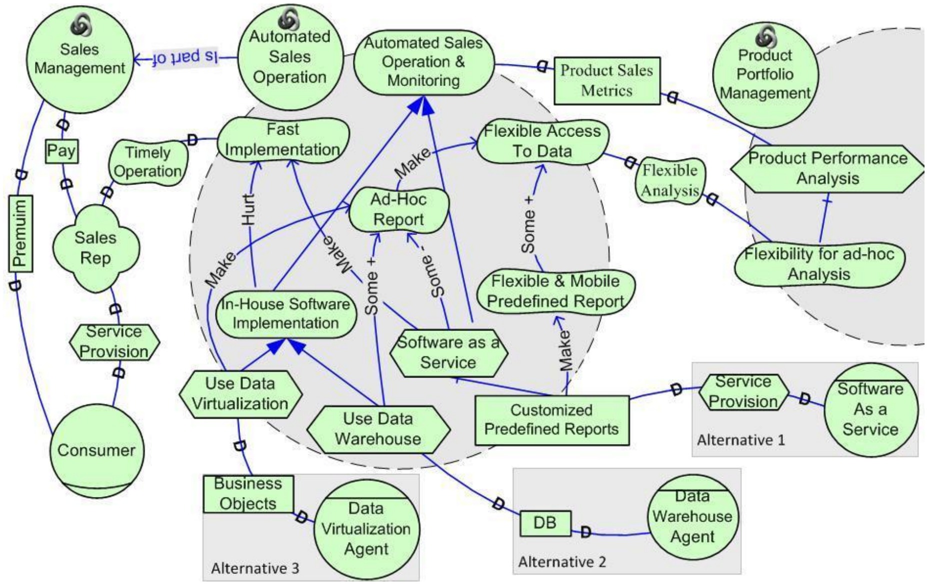

Fig. 5. Automated Sales Operation - Capability Development & Orchestration Alternatives

6 Discussion and Conclusion

Capability modeling is used to facilitate alignment of business architecture with IT artifacts and to identify associated risks with adoption of a certain capability deployment configuration [3]. Several approaches [1], [3], [10] aim to capture strategic intentions of capabilities through association to goals. Researchers propose monitoring Key Performance Indicators (KPIs) associated with goals to evaluate and maintain IT capabilities and projects [1], [3]. Mapping capabilities to service-oriented implementation is proposed to facilitate architectural alignment [3], [10]. Capability modeling is also used to identify capability context and allow run-time adaptation of alternatives [14]. However our review of the literature on the impact of IT on competitive advantage indicates the need to explore orchestration alternatives. Considering implementation alternatives for capabilities in isolation disregards such a need. Current approaches of capability modeling lack the ability to reason on the social dynamics, complementarities and influence of alternatives on one another. These approaches do not facilitate reasoning on alternatives with different quality outcomes to allow trade-off analysis.

In this paper we illustrated the *i** framework's potential to capture the socio-technical aspect of capabilities and allow expression of both top-down strategic intentions and bottom-up integration of organizational resources and skill sets. The *i** notion of softgoals allows expression of intangible drivers of capability alternatives. The mean-ends and decomposition links of *i** allow granular analysis of sub-capabilities. The social context in which capabilities are developed and evolve are modelled with the help of actor associations and their dependencies on capabilities. Reasoning on capability dependency alternatives allows the modeler to design more effective orchestrations. The *i** framework can capture the essence of decentralization by modeling capabilities as autonomous actors that can react to new capability development decisions. A limitation of the presented modeling is that one cannot show constraints imposed on capability dependencies and how they influence alternatives.

The modeling approach presented is not yet tested in real life although the illustrative example is based on a well-defined sector. The quality of the reasoning capacity of the *i** capability models is highly dependent on accurate identification of capabilities. Identifying the boundaries of capabilities and what to exclude or include in a capability model can be challenging. It is also difficult to capture the social aspect of capabilities accurately. Therefore a methodology that guides capability modeling and iterative improvements of the models is required. The methodology should help discover how alternatives affect each other and which combination better suits the organization at any given time. In future, we will also explore how different techniques and tools for supporting reasoning in goal models [22] can be applied to capability modeling.

References

1. 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)
2. Bosch, J.: From Software Product Lines to Software Ecosystems. In: Proceedings of the 13th International Software Product Line Conference, pp. 111–119. Carnegie Mellon University, Pittsburgh (2009)
3. Iacob, M.-E., Quartel, D., Jonkers, H.: Capturing Business Strategy and Value in Enterprise Architecture to Support Portfolio Valuation. In: 16th International Enterprise Distributed Object Computing Conference (EDOC 2012), pp. 11–20. IEEE, Beijing (2012)
4. Teece, D.J.: Explicating dynamic capabilities: the nature and microfoundations of (sustainable) enterprise performance. Strategic Management Journal 28, 1319–1350 (2007)
5. Bhatt, G.D., Grover, V.: Types of Information Technology Capabilities and Their Role in Competitive Advantage: An Empirical Study. Journal of Management Information Systems 22, 253–277 (2005)
6. Nevo, S., Wade, M.: The Formation and Value of IT-Enabled Resources: Antecedents and Consequences. Management Information Systems Quarterly 34, 163–183 (2010)
7. Leonard-Barton, D.: Core capabilities and core rigidities: A paradox in managing new product development. Strategic Management Journal 13, 111–125 (1992)

8. Helfat, C.E., Peteraf, M.A.: The dynamic resource-based view: capability lifecycles. Strategic Management Journal 24, 997–1010 (2003)
9. Yu, E.S.: Social Modeling and *i**. In: Borgida, A.T., Chaudhri, V.K., Giorgini, P., Yu, E.S. (eds.) Mylopoulos Festschrift. LNCS, vol. 5600, pp. 99–121. Springer, Heidelberg (2009)
10. DoD, D.C.I.O.: Department of Defense Architecture Framework (DoDAF) Version 2.0. DoD Deputy Chief Information Officer (2010)
11. Keller, W.: Using capabilities in enterprise architecture management, White Paper, Object Architects (2009)
12. Berre, A.J., de Man, H., Lindgren, P.: Business Model Innovation with the NEFFICS platform and VDML. In: Short Paper Proceedings of the Second Workshop on New Generation Enterprise and Business Innovation Systems, pp. 24–30. CEUR-WS proceedings, Valencia (2013)
13. Barroero, T., Motta, G., Pignatelli, G.: Business Capabilities Centric Enterprise Architecture. In: Bernus, P., Doumeingts, G., Fox, M. (eds.) EAI2N 2010. IFIP AICT, vol. 326, pp. 32–43. Springer, Heidelberg (2010)
14. Zdravkovic, J., Stirna, J., Henkel, M., Grabis, J.: Modeling Business Capabilities and Context Dependent Delivery by Cloud Services. In: Salinesi, C., Norrie, M.C., Pastor, Ó. (eds.) CAiSE 2013. LNCS, vol. 7908, pp. 369–383. Springer, Heidelberg (2013)
15. Aral, S., Weill, P.: IT Assets, Organizational Capabilities, and Firm Performance: How Resource Allocations and Organizational Differences Explain Performance Variation. Organization Science 18, 763–780 (2007)
16. Barney, J.B., Ketchen, D.J., Wright, M.: The Future of Resource-Based Theory Revitalization or Decline? Journal of Management 37, 1299–1315 (2011)
17. Sirmon, D.G., Hitt, M.A., Ireland, R.D., Gilbert, B.A.: Resource Orchestration to Create Competitive Advantage Breadth, Depth, and Life Cycle Effects. Journal of Management 37, 1390–1412 (2011)
18. Molloy, J.C., Chadwick, C., Ployhart, R.E., Golden, S.J.: Making Intangibles "Tangible" in Tests of Resource-Based Theory A Multidisciplinary Construct Validation Approach. Journal of Management 37, 1496–1518 (2011)
19. Horkoff, J., Yu, E.: Evaluating Goal Achievement in Enterprise Modeling – An Interactive Procedure and Experiences. In: Persson, A., Stirna, J. (eds.) PoEM 2009. LNBIP, vol. 39, pp. 145–160. Springer, Heidelberg (2009)
20. Byrd, T.A., Turner, D.E.: Measuring the Flexibility of Information Technology Infrastructure: Exploratory Analysis of a Construct. Journal of Management Information Systems 17, 167–208 (2000)
21. Jones, D., Schmitz, D., France, N., Orlandi, M.: The ACORD Capability Model. ACORD Corporation (2010)
22. Horkoff, J., Yu, E.: Comparison and evaluation of goal-oriented satisfaction analysis techniques. Requirements Engineering 18, 199–222 (2013)