



HAL
open science

Towards a Framework for Shaping & Forming Enterprise Capabilities

Mohammad Hossein Danesh, Eric Yu

► **To cite this version:**

Mohammad Hossein Danesh, Eric Yu. Towards a Framework for Shaping & Forming Enterprise Capabilities. 11th IFIP Working Conference on The Practice of Enterprise Modeling (PoEM), Oct 2018, Vienna, Austria. pp.188-202, 10.1007/978-3-030-02302-7_12 . hal-02156450

HAL Id: hal-02156450

<https://inria.hal.science/hal-02156450v1>

Submitted on 14 Jun 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Towards a Framework for Shaping & Forming Enterprise Capabilities

Mohammad Hossein Danesh¹ and Eric Yu^{1,2}

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 this era of rapid change and major technology-enabled transformations, information systems design needs to take into account the specific context of the organizational setting and the strategic direction of the enterprise. To this end, researchers and practitioners have built on the concept of *capability* to analyze what a business can and should do to manage its strategic trajectories. This paper describes four categories of modeling and analysis requirements to deal with capability formation. The requirements are identified through a review of the origins of the capability concept in the strategic management literature. A set of guidelines is proposed as part of a modeling framework based on the *i** language. Enterprise Capabilities are modeled as a specialized type of intentional actor so that their socio-technical characteristics can be specified and analyzed. This approach to modeling capabilities enables reasoning about (1) why a capability is needed, (2) how it is achieved, (3) how it fits within the organizational and social setting of the enterprise, and (4) what relationships are required for its success. The applicability of the guidelines and associated viewpoints are demonstrated on a chatbot example.

Keywords: Enterprise Capability, Dynamic Capability, *i**, Enterprise Modeling, Capability Modeling

1 Introduction

Enterprises compete in a fast-paced changing global environment that demands customer-driven innovation in which information technology and software systems play a key role [1]. Use of software systems in enterprises continues to transform businesses and plays a prominent role in their competitiveness [1, 2]. To achieve the transformative role and enable innovation in the organizational context, software systems need to be co-designed with the business [2, 3].

The concept of capability has been used by practitioners and researchers alike to conceptualize adoption and integration of new technologies and systems into existing organizational and technological fabric [4–7]. For example, capability maps and heatmaps have been used to communicate strengths and weaknesses and prioritize in-

vestments [6, 7]. Other formulations of the concept have been proposed to enable alignment of resources and capabilities to enterprise architecture [5], or to empower adaptation of business processes to changes in contextual parameters [4, 8].

In the field of strategic management, capabilities have long been studied to account for strengths and weaknesses in competitive positioning [9]. The literature views a capability as an enterprise-specific combination of resources and processes which has to be continuously renewed in order to attain and sustain competitive advantage [9–12].

In this paper we build on the theoretical foundations from strategic management to propose four aspects essential in understanding and analyzing the formation and evolution of an enterprise capability. These aspects result in modeling viewpoints and guidelines that (1) explicate why investments are made to develop and evolve a capability, (2) illustrate alternative couplings of resources and processes that can form the capability, (3) demonstrate how organizational and social setting shapes the capability, and (4) uncover capability relationships that enable co-creation of value.

The paper provides guidelines and examples on how to represent each of the mentioned aspects as part of a modeling framework. A meta-model was presented in an earlier publication [13] to provide an integrative view on capability and its related concepts. The meta-model serves as the conceptual foundation for modeling enterprise capabilities. Two analysis techniques that support reasoning on capability alternatives [14] and their flexibility [15] accompany the meta-model. The modeling notation of the framework builds on the *i** modeling language [16] and involves a number of extensions, including the introduction of the capability concept as a specialized kind of strategic actor. While some of these extensions are explained through the examples, it is beyond the scope of this paper to present these extensions in any detail.

The expressiveness of the framework in representing and analyzing capability formation is demonstrated on an example. The context of the example deals with the strategy to adopt a chatbot agent as part of a customer interaction management capability. This example is inspired by the first author’s experiences in applying the framework for designing and adopting a conversational (chatbot) platform in a Canadian company.

Section 2 of this paper briefly describes the illustrative example. In section 3 of the paper, the four aspects of capability formation and their requirements are presented based on the review of the literature. Section 3 discusses four viewpoints to demonstrate how the *i** framework is adopted and adapted to satisfy the requirements. In section 4, the related work and their ability to model capability formation is reviewed. Finally, the paper is concluded in section 5 with a discussion about components of the proposed framework and future work.

2 A Chatbot Example

To illustrate the use of the framework and its guidelines, an evolutionary scenario for a Customer Interaction (CI) capability is investigated that aims to add a chatbot agent to its communication channels. The capability is administered by the contact center department with the aim to effectively manage customer interactions on all channels.

To enable the evolution of the CI capability, one would need to (a) understand and justify investments on the technology and business processes required to support the new communication channel, (b) investigate what stakeholders need to be involved and what is needed to onboard them, (c) identify the required resources and information systems that should be acquired or developed to support the extended capability, (d) examine different sourcing options for major technical components, (e) recognize suitable organizational deployment configurations, and (f) plan for likely changes that are required in other domains and capabilities.

Without the ability to answer the above questions, a well-defined strategy and architecture that can support development and management of the new technology is not possible.

3 Modeling Specific Aspects of Capability Formation

Following the literature in strategic management [10, 17–20], Enterprise Capabilities are defined as *intentional combinations (orchestrations) of firm-specific assets, organizational routines (business processes), and human knowledge (skillset/know-how) that take advantage of complementary relations and are created and evolved overtime through social collaboration and learning*. We use the term Enterprise Capability (EC) in this paper to distinguish the notion from other usages of the term, such as in capabilities of an individual or of a technology system.

To represent different aspects of the above definition, an agent-oriented modeling paradigm has proven to be useful [21]. Hence, the framework adopted in this paper represents enterprise capabilities as actors with strategic intent and social properties that co-evolve with organizational systems and norms.

Each of the sub-sections below elaborates on one aspect of capability formation and starts with a set of questions as requirements. The questions indicate what a modeling framework should address and serve as guidelines to steer the modeler in identifying and extracting the kinds of information required for decision making. Each sub-section discusses a review of the literature that explains the rationale and the origins of the questions. Once the theory is reviewed, the section illustrates how the modeling framework can be applied in the chatbot example to answer the questions.

3.1 Why Should We Invest in the Capability?

Requirements: To enable decision making and reasoning on capability development, the ability to explicate and analyze why a capability is needed and what strategic role it plays for the firm is necessary. For example, why is a chatbot agent that responds to customer inquiries and handles their interaction beneficial? More specifically a decision maker should be able to answer (a) Why should the enterprise invest in the EC? (b) How do different stakeholders with different responsibilities understand and break-down the objectives? (c) Are there qualitative aspects about the objectives that require attention? (d) What competing objectives exist that are difficult to satisfy and require a compromise?

Theoretical Basis: the literature emphasizes the need to understand and analyze intentions for investing in ECs as they can account for differences in strength and weaknesses across firms [11, 12, 18]. ECs require co-design of IT and business as IT is playing a major role in differentiating enterprises in the market [1–3]. To this end, one needs to represent and reason on objectives that account for capability intentions from multiple perspectives such as technical, business, and organizational [3, 22]. Often qualitative aspects of intentions are responsible for differentiating ECs across firms resulting in different competitive positioning [23]. Hence, representing goals and their tradeoffs is crucial for steering capability development and evolution efforts.

Example: To model an EC, one should start by asking why the capability is beneficial while looking at multiple stakeholders and their viewpoints (focusing on questions (a) and (b)). For example, adding a chatbot agent to existing CI channels entails (1) a technical goal to ensure appropriate understanding and response to customer queries and intents, (2) a business goal to enhance response time and provide cost savings, and (3) an organizational goal to ensure policy enforcement with two distinct perspectives of addressing regulatory compliance, and maintaining and enhancing brand image.

The i^* framework enforces separation of soft and hard goals to enable specification and refinement of quality attributes (answering question (c)). For example, in **Fig. 1**, the requirement of “Enforce Organizational Policy” is broken down to the softgoal of “Enhance Brand Image” and the hardgoal of “Comply With Regulations”. Furthermore, contribution of the chatbot goals to “Effective Customer Interaction” through “Enhanced Response Time” and “Response to Customer Queries & Intents” are explicated to elaborate the role of the new channel in satisfying the CI capability objectives. Enforcing the use of means-end relationships provokes the modeler and designer to investigate alternative breakdowns of the intentions. By illustrating different contribution levels of alternatives to softgoals, one can reason on tradeoffs among the strategic intents of an EC (responding to question (d)).

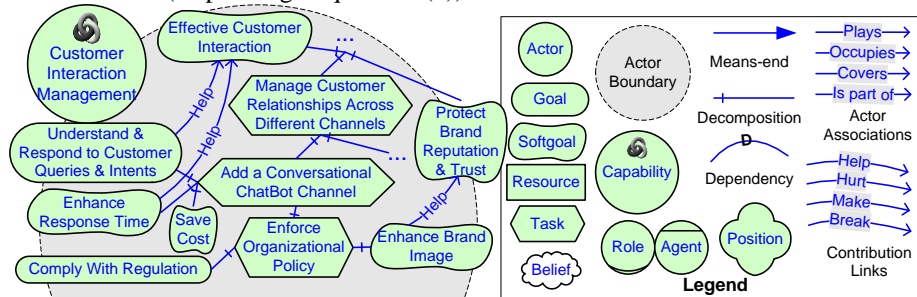


Fig. 1. Representing Capability Intentions for Customer Interaction

3.2 What is the Right Combination of Processes and Resources

Requirements: ECs are formed by acquiring and coupling processes and resources. In the chatbot example, the software components of language processing and intent management must be coupled and coordinated with CRM processes and policies to ensure effective customer interactions. In this regard, a capability manager and designer

should be able to answer (a) what resources are required to satisfy capability intentions? (b) What processes are needed to enable the capability? (c) What are the options available in coupling resources and processes? (d) Are there qualitative characteristics or enterprise-specific interests when coupling resources and processes? (e) How can one represent and reason on commitments of a given coupling option, e.g., how will outsourcing the intent management of the chatbot limit future choices for evolving the agent?

Theoretical Basis: Satisfying EC intentions require an intelligent coupling of enterprise-specific resources and processes [17, 24] that often entail long-lasting commitments (referred to as path dependency in the literature) [11, 20]. The effort is twofold: (a) *Structuring* the portfolio of enterprise resources and processes i.e., what to invest in and what to retire, and (b) *Selecting and Bundling* a particular set of resources and processes within the portfolio to form ECs [17, 24]. The coupling alternatives have different impacts on the intentions and will be preferable under different circumstances.

Example: Aside from the interaction interface of a chatbot, any conversational agent has three main components: (1) Natural Language Processing (NLP) engine, (2) knowledge repository, and (3) intent identification engine (algorithms) [25]. Two approaches in sourcing and coupling such processes and resources are illustrated in **Fig. 2**, each relying on a specific way of coupling resources and processes (answering to question (c)). One can model decisions to acquire a resource/process using i^* tasks and resources within the actor boundary, or to rely on others to provision the resource/process demonstrated by dependencies (responding to questions (a) and (b)).

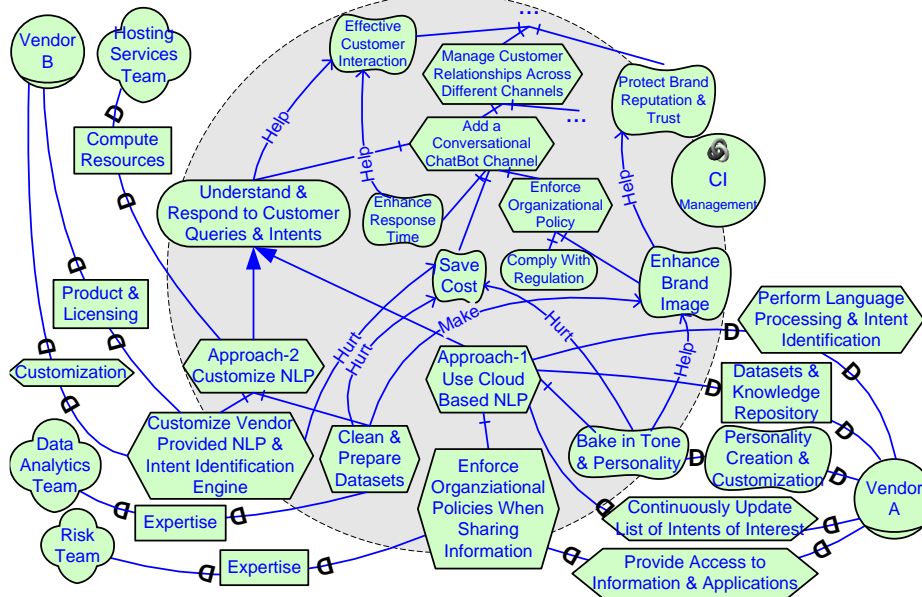


Fig. 2. Coupling Resources & Processes to Shape EC – Legend as in **Fig. 1**

The impact of coupling alternatives on EC intentions is demonstrated through their contributions to softgoals (enabling answering to question (d)). For example, if an external data source is used to train a vendor-provided NLP engine, it is going to be much more expensive to incorporate enterprise-specific tone and personality compared to using an in-house dataset. To use an in-house dataset, one needs to clean and prepare the training and test data which requires data analysis expertise and will be costly. **Fig. 2** illustrates these tradeoffs using contribution links to softgoals of “*Save Cost*” and “*Enhance Brand Image*”.

Aside from coupling of the three components, one needs to understand organizational processes and policies when accessing enterprise systems and integrating with them. Such integration requires technical, regulatory, and organizational expertise that is unique to the industry and often the enterprise. This is represented by the “*Expertise*” dependency to the “*Risk Team*” and the process of “*Enforce Organizational Policies When Sharing Information*” in **Fig. 2**. Such commitments have long-term consequences and can limit future options for evolving the EC. It is therefore important to analyze them when making decisions (relating to question (e)). An example of such a commitment is the involvement of the “*Risk Team*” and their best practices when interacting with “*Vendor A*”.

3.3 Shaping the Capability to Fit the Organization

Requirements: As ECs reside and operate in the context of the enterprise, they are heavily impacted by organizational norms and values. In the chatbot example, the organization’s experience in using vendors and sharing data with them while managing risk and regulatory compliance has a big impact on sourcing resources, employing processes, and choosing vendors. To enable understating of such context, one needs to answer: (a) Who is responsible for making decisions regarding the capability? (b) What social relationships are required to ensure effective collaboration and stakeholder onboarding? (c) What managerial decisions about incentive or organizational structures can impact the capability? (d) Are there organizational norms that enable or inhibit capability intentions and alternatives? (e) What are the domain-specific principles that a capability should build on? (f) What information systems does the EC rely on and how much influence and control does it have over them?

Theoretical Basis: ECs operate in the organizational context shaped by four interdependent kinds of systems - namely technical, human, managerial and value systems [20]. These systems co-evolve with ECs overtime, and have bi-directional impact on each other as they enable or inhibit decision alternatives [12, 20].

Technical systems refer to knowledge and domain-specific principles embedded in processes and information systems [20]. Hence, a modeling framework should be able to explicate the relationships of capabilities to such processes, systems and principles.

Human systems refer to the teams and their managers’ social capital and relationships that enable stakeholder onboarding and resource acquisition [26]. Therefore, a capability modeling framework should enable representation of the human systems.

Managerial systems refer to the formal and informal ways of cascading goals throughout the organization. The formal aspect usually refers to the organizational

structure and its means of enforcing goals while the informal aspect relates to incentive structures set in place [20].

Value systems include norms and values that emerge from accumulated organizational decisions, experiences, and policies. They are considered essential for attaining superior results by teams and managers [20]. Together, these four kinds of systems constitute important context that must be taken into account in EC decision-making.

Example: Responsibility for the CRM capability could be given to a central team with all necessary expertise, or alternatively to distinct teams with expertise in IT, business (e.g., marketing and sales), and organization development. The latter scenario is illustrated in **Fig. 3**. A new type of relationship titled “*Responsible-For*” (an extension to i^*) is introduced to capture decision making rights for ECs, responding to questions (a) and (c).

Organizational norms can impact social relationships among stakeholders of a capability [13]. They are modeled as i^* beliefs. The impact of norms that extend beyond an actor boundary are represented with dashed contribution links, following the work of Yu et al [27]. As an example, the belief that “*Build vs Buy Protects Our Interest & Competitiveness*” will positively impact the relationships with “*Vendor B*” as outlined in **Fig. 3**.

Social relationships of actors are modeled as softgoal dependencies. In **Fig. 3**, “*Vendor B*” depends on the “*Data Analytics Team*” for “*Trust with Cleaned & Vetted Proprietary Data*” to enable customization of their NLP engine. The representation of social relationships and the impact of organizational norms on such relationships enables addressing questions (b), (d) and (e).

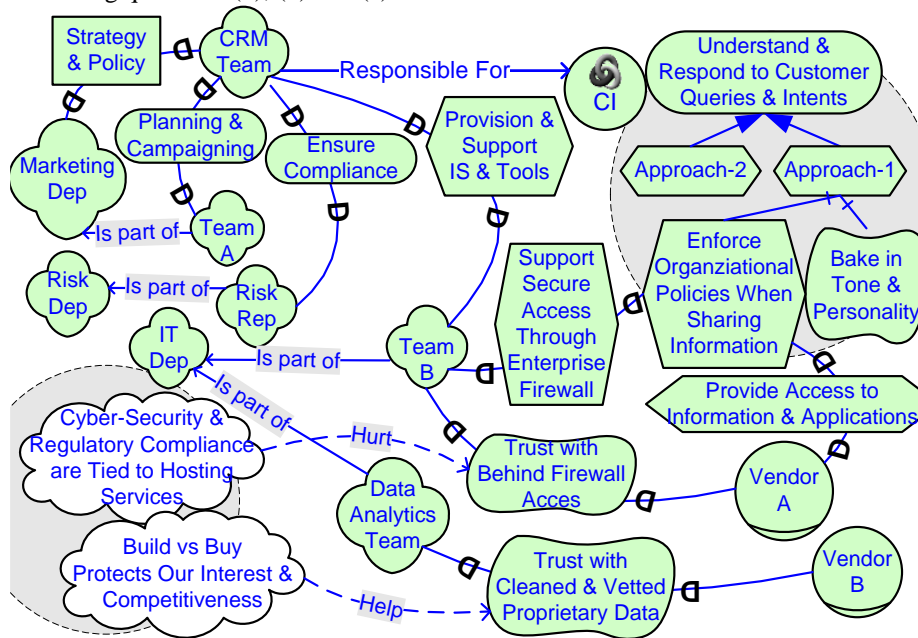


Fig. 3. Visualizing the Impact of Organizational Structure & Norms – Legend as in **Fig. 1**

Fig. 3 demonstrates how the organization would favor “*Approach-2*” as its dependencies on “*Vendor B*” give the organization more control over how it shares data¹. This becomes evident by representing organizational structure and norms of departments and teams.

Flexibility of the “*CRM Team*” in making decisions is significantly higher when they do not rely on teams of other departments and are centrally managed. This alternative was not presented in **Fig. 3** in the interest of space.

The questions presented earlier in the requirements section serve as guidelines for the modeler to navigate the enterprise context and investigate its impact on capability alternatives. The final question in that list was not instantiated in the model as it overlaps with some of the requirements in the coming section and will be discussed in section 3.4 and **Fig. 4**.

3.4 Making Capabilities Complement Each Other

Requirements: Network of ECs work together to serve customers. When designing and evolving ECs it is important to analyze and create complementary relationships among them. A capability modeling framework should enable answering (a) What does the capability have control over? What does it rely on others or share responsibilities with? (b) How do the relationships enable capability intentions and do they impose tradeoffs in achieving them? (c) What restrictions and limitations do the relationships entail and how do they impact strategic intentions? (d) How do changing (1) capability relationships, or (2) resource and process couplings supporting them, impact one another?

Theoretical Basis: ECs often form complementary and interlocking relationships in order to create superior value [10, 17, 24]. Some relationships can have a negative impact or pose constraints that inhibit evolution. Such relationships are referred to as suppressing relations [23]. With today’s enterprises competing in global ecosystems, the requirement for complementary relationships and formation of co-creating networks is ever more pressing [1, 28]. On the other hand, due to the fast paced changes and highly dynamic environments, ECs need to operate in near autonomy in order to quickly respond to evolving requirements and foster innovation [10].

As a result, a modeling framework should be able to reason about the autonomy in making decisions and the interdependencies that enable or suppress capabilities. Such reasoning will allow a better understanding and alignment of the interdependent and localized intentions of capabilities.

Example: To achieve localization and faster decision making, the organization can decompose the “*CI*” capability into specialized sub-capabilities that manage different channels of communication. In the example given in **Fig. 4**, the “*CI Management*” capability is decomposed to “*Managing Telephone Services*” and “*Managing Chatbot Services*”. Decompositions are indicated by “*is-part-of*” associations and justified through dependencies, i.e., the intentions behind the decomposition should be explicated as dependencies. With explicit decomposition of intentions, one can answer what-

¹ Dependencies for “*Approach-2*” are omitted from **Fig. 3** to save space but are shown in **Fig. 2**.

if questions about distributing responsibilities and resources among capabilities and teams (answering questions (a) and (e)). For example, in **Fig. 4** the decision to separate operation and management of the two main customer interaction channels, i.e., *Chat* and *Telephone*, is supported by the need to apply distinct principles and management mechanisms. The dependencies with numbers one to five in **Fig. 4** outline the intentions behind the proposed decompositions.

Customers expect a seamless experience across multiple interaction channels (illustrated through dependencies 2 and 3 in **Fig. 4**), yet it can be challenging to orchestrate changes across chatbot and human phone services. For one, when renewing responses to customer queries, the update cycles and mechanisms differ between a team of human agents and chatbot software. The chatbot might go through update cycles every few weeks while the human agents retrain once or twice a year. In addition, changing the features and offerings of the “*CRM Software*” every few weeks will make it impossible for agents to keep up with all the changes in a big organization with multiple services/products. Surely a compromise is needed, but one needs to identify the right compromise that is acceptable to all parties and advances “*CI*”. Understanding the impacts of a capability choice on other capabilities, information systems, processes and actors is a pre-requisite in identifying such compromises.

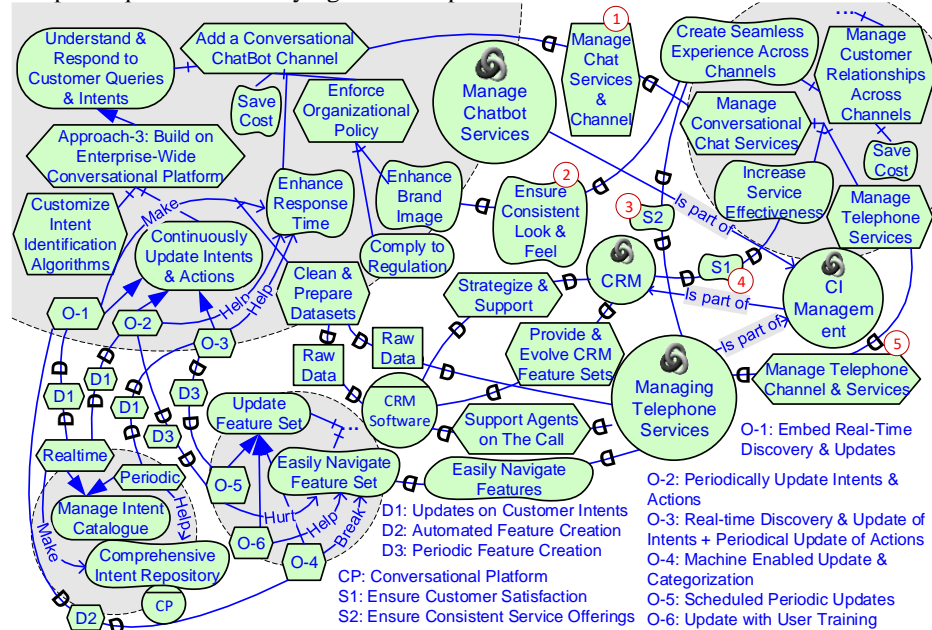


Fig. 4. Demonstrating Capability Relationships – Legend as in **Fig. 1**

In **Fig. 4**, the “*Easily Navigate Feature Set*” represents the interest of human agents supporting “*Telephone Services*”. The alternative mechanism in updating the “*CRM Software*” represented as “*O-4*”, “*O-5*”, and “*O-6*” contribute to “*Easily Navigate Feature Set*”. Furthermore, the choice among these three options will impact how “*Intents*”

& Actions” of the “*Chatbot Services*” is updated as it can be traced through dependencies “*D2*” and “*D3*” which impact alternatives “*O-1*”, “*O-2*”, and “*O-3*”. The impact and coordination needs do not end at the “*Chatbot Services*” capability as the choice on “*O-1*”, “*O-2*”, and “*O-3*” will impact the “*Comprehensiveness of Intent Repository*” which is a softgoal of the “*Conversational Platform (CP)*”. Depending on how crucial this softgoal is to the organization, one will have to prioritize and choose among the presented options. The dependency graph proposed in a related publication [15] can be used to trace the impacts of dependencies and ease the orchestration of choices.

Explication of alternatives for satisfying capabilities alongside their complementary relationships enables better decision making. It facilitates reasoning on how capabilities, actors and information systems should be orchestrated across the organization. Such representation and analysis enable answering questions (b), (c) and (d).

4 Discussion and Related Work

Many approaches have been built using the concept of capability to enable relating strategic intentions and operational choices of an enterprise [29]. Among these approaches, four ways of representing capability and its constituents are identified and will be evaluated against the proposed requirements of section 3. The first one and the most popular representation is a capability map that illustrates what a capability does, and its only constituents are sub-capabilities. The map serves as a taxonomy of enterprise functions and is used to communicate and prioritize enterprise investments [6, 7, 29]. The second representation adds a strategic perspective to the ArchiMate language to enable better alignment of enterprise architectures to business trajectories [5, 30]. In this view, a capability may consist of both behavioral and structural elements of ArchiMate [5].

The third representation builds on the goal-oriented Enterprise Knowledge Development (EKD) and enables adaptation of software services and processes to the changing context of the capabilities. A capability in this view is created from one or more processes, fulfills a goal, and responds to a specific context. The approach provides a methodology for Capability Driven Development (CDD) to guide the modeling activities [4, 8]. The fourth representation as discussed in this paper focuses on a socio-technical representation of capabilities building on the i* language. The capability in this view is formed by coupling resources and processes to pursue a series of business goals. It heavily relies on organizational actors and norms, information systems, and other capabilities to perform.

4.1 Representing and Analyzing Strategic Intentions

Capability Maps: The representation of strategic intentions behind a capability in this view is through a textual description. Some approaches link capabilities to outcomes and value chains using multiple viewpoints. However, outcomes are the result achieved in pursuing a strategic goal and not the goal itself and hence do not help in

capturing multiple perspectives. The maps are not well equipped to analyze and manage competing goals.

ArchiMate: The extended ArchiMate language enables association of capabilities to motivational elements including goals. As such, it can demonstrate the strategic intents and requirements that a capability satisfies. However, the notation does not differentiate qualitative goals and different levels of satisfaction, therefore limiting its ability to explicate compromises.

CDD: Business goals play a major role in defining capabilities in CDD. The goals help in defining measures that shape the context parameters of capability implementations. Such parameters will determine the adaptation criteria for capability delivery. Qualitative goals are not distinguished in this approach and a capability is developed in response to a single business goal. Capability evolution decisions depend on how one defines performance measures and contextual parameters i.e., they are not necessarily driven by the strategy but are related to the capability delivery mechanisms.

The i* Framework: The approach allows representation and reasoning on why a capability is needed from multiple perspectives while demonstrating how goals are re-defined and related to one another. Furthermore, softgoals in i* enable representation of qualitative goals and their satisfaction levels. This allows reasoning on tradeoffs and necessary compromises. The qualitative aspect of capabilities often accounts for unique and firm-specific characteristics that differentiate capabilities across enterprises.

4.2 Representing and Analyzing Resource and Process Coupling

Capability Maps: Depending on the approach, a capability map may be linked to business processes but is not actively investigated for options of coupling resources and processes. The linkages and multiple viewpoints are used to communicate priorities and do not empower reasoning about development, evolution or retirement choices.

ArchiMate: ArchiMate enables modeling how resources and processes are related to capabilities and represents their choices of coupling through the concept of “capability enabling bundle”. The approach is capable of elaborating choices and their associations to goals. However, the language does not investigate qualitative contributions and no methodological support is provided to analyze commitments.

CDD: Business processes are the means to deliver a capability and therefore shape the capability delivery pattern (alternative). Resources are only used by processes and are indirectly related to capabilities. Qualitative goals are not a primary concern in the methodology unless they impact the contextual parameters and therefore cannot determine tradeoffs for capability design. Capability commitments in provisioning resources and processes are not modeled in this approach. The CDD approach has the appropriate tool support accompanying its meta-model to track changes in contextual parameters throughout the capability lifecycle.

The i* Framework: By representing capabilities as intentional actors, one can illustrate their strategy in sourcing required processes and resources. The coupling alternatives are demonstrated using means-end relationships. Contribution links empower understanding the impact of alternative couplings on qualitative goals. Internal and external commitments are shown through dependencies. However, the i* framework is only

focused on point in time representation of commitments and cannot reason on their evolution. A solution would be to create a new/evolved instance of the model and enable tracking such instances throughout the capability lifecycle with tool support.

4.3 Representing and Analyzing Organizational Fit

Capability Maps: No representation of organizational actors and structure is present in capability maps. Suggestions have been made to link other viewpoints that represent organizational roles and their structure to capabilities [7]. However, the association does not empower and support the modeler to explore social relationships, organizational norms, and their impacts on one another. Therefore, it does not enable understanding of the fit between capability alternatives and organizational choices.

ArchiMate: The representation allows association of actors with capabilities, which is illustrative of an ownership relationship. Therefore, it does not help in understanding the social aspects of capabilities and their fit to enterprise context.

CDD: Actor, organizational situation and norms are not modeled.

The i* Framework: The approach focuses on representing and analyzing the organizational structure and its possible variations with respect to EC. It elaborates the tangible and intangible gains of teams and individuals in relation to a capability, and the norms and values impacting it. Such representation and reasoning are essential to manage social expectations and resistance to evolving strategies [20].

4.4 Representing and Analyzing Complementarities

Capability Maps: Capability relationships to one another and other entities such as actors and information systems are not investigated or represented. Mapping applications' contribution to capabilities are done through heatmaps but the nature and intentions behind the contributions are not specified. Hence, the heatmap cannot answer questions and analysis inquiries.

ArchiMate: Aggregation is the only capability relationship in this approach i.e., a series of capabilities and resources can be aggregated to form a new capability. The approach can express constraints imposed on capabilities but does not provide methodological support or guidance in identifying them.

CDD: Capabilities have no relationships with one another. Information systems impact capabilities through business processes. Constraints that limit capabilities can be explicated by contextual parameters, although the approach does not guide how one should identify such constraints.

The i* Framework: The framework uses dependencies to demonstrate relationships among capabilities, actors and information systems, enabling coordination of choices among them. Positive and negative impacts of relationships on strategic intents are depicted through contributions links to soft goal. However, the framework lacks the ability to explicate constraints imposed by dependencies.

5 Conclusion and Future Work

The notion of capability has gained attention as a concept to link and align operational decisions to strategic intents, and therefore plays an important role in modeling enterprises [29]. Building on the strategic management literature, this paper contributed to the field by (1) identifying a set of requirements for reasoning and analyzing on how capabilities are formed, (2) transforming the requirements into questions that guide modelers in gathering information about capability formation, and (3) illustrating how the i* language can be used to enable reasoning on different aspects of capability formation.

The contributions presented in this paper are part of an i* based capability modeling framework that consists of (1) an integrative meta-model [13] serving as a conceptual foundation, (2) the guidelines for representing and analyzing formation of capabilities (as presented in this paper), and (3) analysis techniques that enable reasoning on capability alternatives [14] and their flexibility [15]. The design of the framework follows a Design Science Research (DSR) methodology [31] and some of the framework's components have been tested in few cases [14, 15, 32, 33]. The guidelines presented in this paper were demonstrated in a case inspired by real world requirements. Additional empirical validations will be addressed in future publications.

The baseline i* framework presented in this paper needs to be formally described with clear specification of extensions and their conceptual justifications. The formalization should include a specification of how i* will represent concepts of the integrative meta-model. Tool support for the framework and its analysis techniques is essential to ease known scalability issues. Further guidelines on how and to what extent one should drill down when analyzing enterprise capabilities is needed to address possible impracticality of modeling full capability network(s).

References

1. J. Bosch: Speed, Data, and Ecosystems: The Future of Software Engineering. *IEEE Softw.* 33, 82–88 (2016).
2. Jeanne W. Ross, Ina M. Sebastian, Cynthia M. Beath, Lipsa Jha, Technology Advantage Practice of The Boston Consulting Group: Designing Digital Organizations - Summary of Survey Findings. MIT CISR, Boston, MA (2017).
3. Nevo, S., Wade, M.: The Formation and Value of IT-Enabled Resources: Antecedents and Consequences. *Manag. Inf. Syst. Q.* 34, 163–183 (2010).
4. Bērziša, S., Bravos, G., Gonzalez, T., Czubayko, U., España, S., Grabis, J., Henkel, M., Jokste, L., Kampars, J., Koç, H., Kuhr, J.-C., Llorca, C., Loucopoulos, P., Pascual, R., Pastor, O., Sandkuhl, K., Simic, H., Stirna, J., Valverde, F., Zdravkovic, J.: Capability Driven Development: An Approach to Designing Digital Enterprises. *Bus. Inf. Syst. Eng.* 57, 15–25 (2015).
5. Azevedo, C.L.B., Iacob, M.-E., Almeida, J.P.A., Sinderen, M. van, Pires, L.F., Guizzardi, G.: Modeling resources and capabilities in enterprise architecture: A well-founded ontology-based proposal for ArchiMate. *Inf. Syst.* 54, 235–262 (2015).

6. Burton, B.: Eight Business Capability Modeling Best Practices. *Gart. Res. ID.* (2012).
7. Business Architecture Guild: A Guide to the Business Architecture Body of Knowledge. (2014).
8. Zdravkovic, J., Stirna, J., Henkel, M., Grabis, J.: Modeling Business Capabilities and Context Dependent Delivery by Cloud Services. In: Salinesi, C., Norrie, M.C., and Pastor, Ó. (eds.) *Advanced Information Systems Engineering*. pp. 369–383. Springer Berlin Heidelberg (2013).
9. Teece, D.J., Pisano, G., Shuen, A.: Dynamic Capability and Strategic Management. *Strateg. Manag. J.* 18, 509–533 (1997).
10. Teece, D.J.: Explicating dynamic capabilities: the nature and microfoundations of (sustainable) enterprise performance. *Strateg. Manag. J.* 28, 1319–1350 (2007).
11. Helfat, C.E., Peteraf, M.A.: The dynamic resource-based view: capability lifecycles. *Strateg. Manag. J.* 24, 997–1010 (2003).
12. Winter, S.G.: Understanding dynamic capabilities. *Strateg. Manag. J.* 24, 991–995 (2003).
13. Danesh, M.H., Loucopoulos, P., Yu, E.: Dynamic Capabilities for Sustainable Enterprise IT – A Modeling Framework. In: Johannesson, P. (ed.) *Conceptual Modeling*. pp. 1–9. Springer International Publishing (2015).
14. Danesh, M., Yu, E.: Modeling Enterprise Capabilities with i*: Reasoning on Alternatives. In: Iliadis, L., Papazoglou, M., and Pohl, K. (eds.) *Advanced Information Systems Engineering Workshops*. pp. 112–123. Springer International Publishing (2014).
15. Danesh, M., Yu, E.: Analyzing IT Flexibility to Enable Dynamic Capabilities. In: Persson, A. and Stirna, J. (eds.) *Advanced Information Systems Engineering Workshops*. pp. 53–65. Springer International Publishing (2015).
16. Yu, E.: Modeling Strategic Relationships For Process Reengineering, <ftp://ftp.cs.toronto.edu/pub/eric/DKBS-TR-94-6.pdf>, (1995).
17. Helfat, C.E., Finkelstein, S., Mitchell, W., Peteraf, M., Singh, H., Teece, D., Winter, S.G.: *Dynamic Capabilities: Understanding Strategic Change in Organizations*. John Wiley & Sons (2009).
18. McKelvie, A., Davidsson, P.: From Resource Base to Dynamic Capabilities: an Investigation of New Firms. *Br. J. Manag.* 20, S63–S80 (2009).
19. Sirmon, D.G., Hitt, M.A., Ireland, R.D., Gilbert, B.A.: Resource Orchestration to Create Competitive Advantage Breadth, Depth, and Life Cycle Effects. *J. Manag.* 37, 1390–1412 (2011).
20. Leonard-Barton, D.: Core capabilities and core rigidities: A paradox in managing new product development. *Strateg. Manag. J.* 13, 111–125 (1992).
21. Danesh, M.H., Yu, E.: Representing and Analyzing Enterprise Capabilities as Specialized Actors - A BPM Example. In: *Advances in Conceptual Modeling Workshops*. Springer International Publishing - To be published (2018).
22. Frank, U.: Multi-perspective enterprise modeling: foundational concepts, prospects and future research challenges. *Softw. Syst. Model.* 1–22 (2012).
23. Sirmon, D.G., Hitt, M.A., Arregle, J.-L., Campbell, J.T.: The dynamic interplay of capability strengths and weaknesses: investigating the bases of temporary competitive advantage. *Strateg. Manag. J.* 31, 1386–1409 (2010).

24. Sirmon, D.G., Hitt, M.A., Ireland, R.D.: Managing Firm Resources in Dynamic Environments to Create Value: Looking Inside the Black Box. *Acad. Manage. Rev.* 32, 273–292 (2007).
25. Brian Manusama, Frances Karamouzis, Tom Austin: Seven Decision Points for Success With Virtual Customer Assistants. Gartner (2016).
26. Adner, R., Helfat, C.E.: Corporate effects and dynamic managerial capabilities. *Strateg. Manag. J.* 24, 1011–1025 (2003).
27. Yu, E., Liu, L.: Modelling Trust for System Design Using the i * Strategic Actors Framework. In: Falcone, R., Singh, M., and Tan, Y.-H. (eds.) *Trust in Cyber-societies*. pp. 175–194. Springer Berlin Heidelberg (2001).
28. Augier, M., Teece, D.J.: Dynamic Capabilities and the Role of Managers in Business Strategy and Economic Performance. *Organ. Sci.* 20, 410–421 (2009).
29. Zdravkovic, J., Stirna, J., Grabis, J.: A Comparative Analysis of Using the Capability Notion for Congruent Business and Information Systems Engineering. *Complex Syst. Inform. Model. Q.* 1–20 (2017).
30. 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, China (2012).
31. Gregor, S., Hevner, A.R.: Positioning and Presenting Design Science Research for Maximum Impact. *MIS Q.* 37, 337–355 (2013).
32. Loucopoulos, P., Stratigaki, C., Danesh, M., Bravos, G., Dimitrakopoulos, G.: Enterprise Capability Modeling: Concepts, Method and Application. In: *Enterprise Systems Conference (ES)*, 2015 (2015).
33. Loucopoulos, P., Kavakli, E., Anagnostopoulos, D., Dimitrakopoulos, G.: Capability-oriented Analysis and Design for Collaborative Systems: An Example from the Doha 2022 World Cup Games. In: *Proceedings of the 2018 10th International Conference on Computer and Automation Engineering*. pp. 185–189. ACM, New York, NY, USA (2018).