Modeling Resources and Capabilities in Enterprise Architecture: A Well-Founded Ontology-Based Proposal for ArchiMate

Carlos L. B. Azevedo[1,2], Maria-Eugenia Iacob[3], João Paulo A. Almeida[1],
Marten van Sinderen[2], Luís Ferreira Pires[2] and Giancarlo Guizzardi[1]

[1] Ontology and Conceptual Modeling Research Group (NEMO),
Federal University of Espírito Santo (UFES), Vitória, ES, Brazil
[2] Services, Cybersecurity and Safety Research Group, Centre for Telematics and Information
Technology, University of Twente, Enschede, The Netherlands
[3] Department of Industrial Engineering and Business Information Systems,
University of Twente, Enschede, The Netherlands
clbazevedo@inf.ufes.br, m.e.iacob@utwente.nl, jpalmeida@ieee.org,
{m.j.vansinderen, l.ferreirapires}@utwente.nl, gguizzardi@acm.org

Abstract— The importance of capabilities and resources for portfolio management and business strategy has been recognized in the management literature. Despite that, little attention has been given to integrate the notions of capabilities and resources in enterprise architecture descriptions. One notable exception is a recent proposal to extend the ArchiMate framework and language to include capability and resources and thus improve ArchiMate's coverage of portfolio management. This paper presents an ontological analysis of the concepts introduced in that proposal, focusing in particular on the resource, capability and competence concepts. We provide an account for these concepts in terms of the Unified Foundational Ontology (UFO). The analysis allows us to identify semantic issues in the proposal and suggest well-founded recommendations for improvements. We revise the proposed metamodel in order to address the identified problems, thereby improving the semantic clarity and usefulness of the proposed language extension. Two real-world cases are modeled with the resulting metamodel to show the applicability of the constructs and relations in an industrial setting.

Keywords: Capability, Resource, Enterprise Architecture Modeling, Ontology-based Semantics, ArchiMate

1. INTRODUCTION

A fundamental question in the field of strategic management is how organizations gain and sustain competitive advantages (Barreto 2009). Constant shifts in markets, competition, technology and regulation drive organizations to promote changes, and to continuously adapt and improve their organizational structures. Failing to address these shifts affects organizational performance negatively (Audia, Locke, and Smith 2000). Research has shown that the average period during which organizations are able to sustain their competitive advantage has decreased over time (Wiggins and Ruefli 2005). This suggests that organizations have to build successive temporary advantages (Eisenhardt and Martin 2000). In order to be able to accomplish this, an organization needs to employ its resources and capabilities in the most effective manner. This observation is not only valid at the level of one organization, but also within business networks, in which organizations establish partnerships with the goal of pooling their capabilities and resources. The suggestion that organizational resources and capabilities are key success factors for competitive advantage has been proposed already in the 90's (Barney 1991) and is still a predominant topic (Helfat and Winter 2011). A major concern of strategic management is the usage of organizational

capabilities and resources, in order to improve performance, quality and to reduce costs (Ray, Barney, and Muhanna 2004) (Baines et al. 2009).

In the work presented in (McKelvie and Davidsson 2009), many organizations have been analyzed in order to answer the question "to what extent do access and changes to resource bases influence the development of dynamic capabilities in new firms?". The work provides statistical evidence for the relationship between organizational resources and the subsequent capabilities of the organization. Different resources lead to different capabilities, and the changes of resources over time have a great impact on organizations' capabilities (McKelvie and Davidsson 2009).

Strategic management deals with survival and competitiveness in the long-term, despite unknown facts that will take place in the future. This has led to the formulation of multiple theories, with a focus on Resources (Barney 1991) and Capabilities ((Teece and Pisano 1994), (Eisenhardt and Martin 2000)) as a source of competitive advantage. Resources-centric theories regard an organization as a bundle of resources (Grant 1996). They suggest that the resources' properties (e.g., rare, valuable, non-substitutable or inimitable) confer organizations competitive advantage (Peteraf and Barney 2003). The idea is that enterprises with appropriate resources should be able to leverage the needed capabilities and to sustain competitive advantages regardless of scenario. Since the introduction of the resource-based theories, several major limitations have been identified. Amongst these, the most relevant are the fact that valuable, rare, inimitable, and non-substitutable (VRIN) resources (Barney 1991) are neither necessary nor sufficient for sustaining competitive advantage in a dynamic environment (Kraaijenbrink, Spender, and Groen 2010). Resources by themselves are not useful unless they are correctly employed. The way resources are used defines the outcome: (Penrose 1959) stated that "exactly the same resources when used for different purposes or in different ways and in combination with different types or amounts of other resources provide a different service or set of services". As a response to this criticism, the capability-based theories have taken shape. According to these theories, the enterprise needs to know the capabilities it wants to leverage in order to use and plan to acquire resources in an intended manner. Whereas resource-based theories focus on accumulating resources, the capability-based theories focus on "adapting, integrating, and re-configuring internal and external organizational skills, resources, and functional competences toward a changing environment" (Teece and Pisano 1994).

The need for capability-based planning in the context of organizations has become more apparent in the recent years. Recent developments have identified capabilities as the way to link business and IT ((Danesh and Yu 2014), (Stirna et al. 2012)), to link business outcomes to IT (Miklos 2012), and as a solution for improving the business and IT alignment ((Lee and Song 2011), (Zdravkovic et al. 2013)).

Given this increased interest in capabilities, both in theory and in practice, it comes as no surprise that this concept has recently surfaced in enterprise architecture (EA) following the need to align strategic decisions with their actual implementation at the level of processes, IT systems and infrastructure. EA frameworks such as TOGAF (The Open Group 2009), have already introduced basic notions of capability-based planning and its role in designing, planning and implementing organizational change.

One of the main issues addressed by capability-based planning, in the context of EA, is to provide enterprise architects with a common ground to initiate discussions with business leaders in terms of business outcomes (increased output, better quality, lower costs, revenue growth or improved market share) instead of projects, processes and applications (Scott, 2009). The problem is that processes are too detailed, applications are too technical and projects focus on short term results, usually having little strategic value. Despite the relevance of capabilities and resources to the success of enterprises, little attention has been given to integrate these notions of capability and resource in enterprise architecture descriptions. A notable exception is the work discussed in (Iacob, Quartel, and Jonkers 2012), which extends

the ArchiMate language (Iacob et al. 2012) with constructs to represent capabilities and resources, integrating these new constructs with those used to represent other aspects of an enterprise architecture (such as active structure and behavior). The objective is to empower enterprise architects to use these important notions in coherent enterprise architecture descriptions. We argue that modelling resources and capabilities for decision making purposes at strategic level must simplify models and hide the complexity of architecture models which is of no relevance at that abstraction level, where decision makers are mostly interested in means (i.e., resources & capabilities) and goals (i.e., motivation). This is due to the fact that both capabilities and resources are defined in the proposed extension as abstractions of complex behavior and architectural structure, respectively. This is also why important benefits to be reaped from modelling resources and capabilities are at the strategic level, while carrying out activities such as strategic alignment, capability-based planning, capability portfolio management, etc. In contrast, resources and capabilities can be linked to the architecture fragments they are abstracted from, thus enabling the end-to-end traceability from strategic decisions to implementation and architecture change.

A main challenge of incorporating the notions of capability and resource in enterprise architecture (EA) lies in identifying a precise conceptualization for these notions. Without such a precise conceptualization, rigorous definition of the semantics of the proposed modeling elements is problematic, and modeling and communication problems arise. For example, when various modelers share a model without a clear semantics, False Agreement (Guarino 1998) most likely ensues. In that case, different modelers come to different interpretations of the same model and are not aware of the conflict. This would result in enterprise architecture models that cannot serve their purpose as tools for communication between stakeholders, decreasing the value of enterprise architecture models in the pursuit of informed decision-making.

Our objective in this paper is to address this challenge, resulting ultimately in a well-founded approach to capture capabilities and resources in enterprise architecture descriptions. The novelty of our approach resides in the strong relation of enterprise architecture modelling to these concepts as introduced in strategic management, and, most importantly, in the clear semantics we provide for the proposed modeling elements through ontological analysis. We aim at addressing the strategic management level, allowing capability-based planning and better decision making. To accomplish that, we analyze the semantics of the Business Strategy and Valuation Concepts (BSVC) extension to ArchiMate (Iacob, Quartel, and Jonkers 2012) building up on our earlier work presented in (Azevedo et al. 2013). We identify issues with the original proposal, by interpreting it in terms of a reference foundational ontology. We then introduce a new ontologically well-founded version of the BSVC metamodel that addresses the identified issues.[1]

The remainder of this paper is structured as follows: Section 2 describes the current modeling of resources and capabilities in EA using the ArchiMate Business Strategy and Valuation Concepts extension as our starting point and introduces a running example. Section 3 introduces ontological analysis, in order to clarify the methodology we employ. Section 4 presents the ontological foundation concepts used in our analysis. Sections 5, 6 and 7 provide interpretations for the concepts of resource, capability and competence, respectively. Each concept is interpreted according to the metamodel, the textual definitions, and the examples of usage provided in (Iacob, Quartel, and Jonkers 2012), and in the related literature. Section 8 proposes a number of improvement recommendations, including a new metamodel proposal for the language. Section 9 illustrates the proposed extension and recommendations by means of two use cases: the Toyota case, which focuses on the conceptual modelling aspects, and the IT consolidation case, which is an example of capability-based planning, showing how the

---

[1] This paper extends (Azevedo et al. 2013), in which the issues with the original proposal were identified.

strategic decision of facilitating and improving the switching of customers (as result of an EU regulation) has led to an IT consolidation problem for which the implementation solution is based on the selection of an optimal portfolio of systems. Section 10 discuses related work. Finally, Section 11 presents our conclusions and outlines future work.

2. MODELING CAPABILITIES, RESOURCES AND COMPETENCES

Organizational resources and capabilities are key factors for organizational success. Our goal is to enable modelers to capture capabilities, resources and competences in an enterprise architecture modeling language. We use ArchiMate as a starting point.

The modeling of resources and capabilities for ArchiMate was proposed in (Iacob, Quartel, and Jonkers 2012). The Business Strategy and Valuation Concepts extension was based on the analysis and review of relevant business strategy and portfolio management literature. For this reason, many concepts present in other approaches, such as (Gordijn and Akkermans 2001) and (Zandi and Tavana 2010) have been incorporated in the proposed extension.

Similarly to other earlier ArchiMate extension proposals (e.g., the motivation extension presented in (Carlos L. B. Azevedo et al. 2011)) the initial development of the BSVC has been conducted without a rigorous definition of the semantics of the proposed modeling elements. The absence of such definitions could lead to several modeling and communication problems. Further, this would result in enterprise architecture models that cannot serve their purpose as tools for communication between stakeholders, and would decrease the enterprise architecture value in the pursuit of informed decision-making.

The BSVC introduces to ArchiMate the concepts of resource, capabilities, competence and risk. We focus on the resource, capability and competence concepts in this paper. The definitions and concrete syntax for the modeling elements introduced by the extension are shown in Table 1. In this section we have preserved the definitions as they were provided originally in (Iacob, Quartel, and Jonkers 2012). These are the objects of analysis in Sections 5, 6 and 7.

Table 1 - Concrete syntax and definitions for Resource, Capability and Competence as it is on the BSVC

Modeling element	Definition	Concrete syntax
Resource	"an asset owned or controlled by an individual or organization"	Resource
Capability	"the ability (of a static structure element, e.g., actor, application component, etc.) to employ resources to achieve some goal"	Capability
Competence	"the definition of competence […] is almost identical with that of personnel-based resources".	Resource [2]

Figure 1 shows the metamodel fragment, as proposed in (Iacob, Quartel, and Jonkers 2012), for the integration of the BSVC with the ArchiMate core metamodel and its extensions.

---

[2] The competence element has no individual concrete syntax and is represented by the resource concrete syntax in (Iacob, Quartel, and Jonkers 2012).
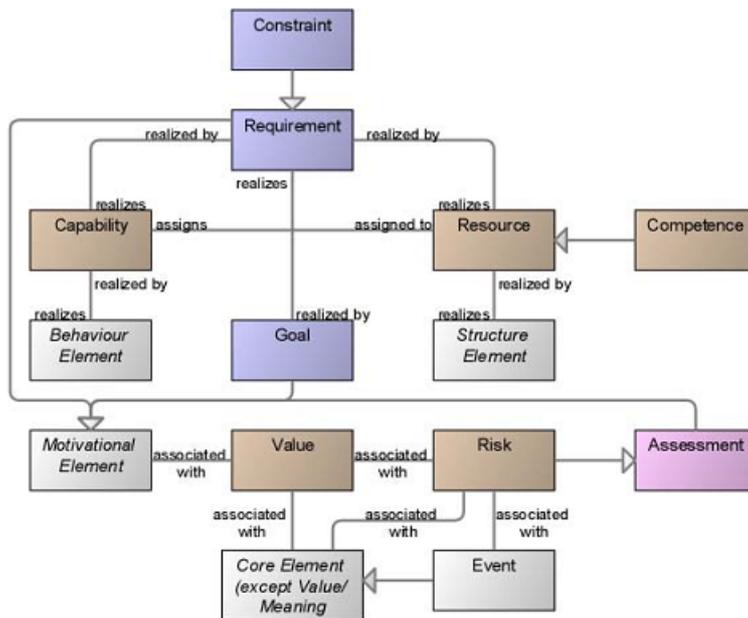
Figure 1 – BSVC Metamodel fragment (Iacob, Quartel, and Jonkers 2012)

The resource concept is prominently present in most valuation techniques, in business modeling approaches, and in constraint optimization models in which they are mathematically defined and constrained. This supports the importance given to the resource concept in the management literature (Iacob, Quartel, and Jonkers 2012).

A resource represents an asset owned or controlled by an individual or organization. Resources are related to the motivation extension, in particular to requirements and goals, through the realization relation. The argument for this relationship is that goal achievement assumes availability and (constrained) consumption of certain resources. This view is based on the mathematical formulation of constrained optimization models, in which a goal function is minimized/maximized subject to a system of constraints (expressed as inequalities) imposed on the resources to be consumed for the achievement of the goal. Thus, a resource may satisfy a requirement, which in turn, may realize a goal. Furthermore, a resource is realized by structure elements, and is regarded in (Iacob, Quartel, and Jonkers 2012) as an abstraction of these elements.

Capability is defined as the ability (of a static structure element, e.g., actor, application component, etc.) to employ resources to achieve some goal. Similarly to resource, capability is regarded in (Iacob, Quartel, and Jonkers 2012) as an abstraction of some behavior. The assignment relationship between resource and capability expresses the ability to employ (i.e., configure, integrate, etc.) resources.

The competence concept is introduced in (Iacob, Quartel, and Jonkers 2012) as a specialization of resource, based on the definition of competence proposed by (Prahalad and Hamel 1990), which equates competence to personnel-based resources.

Figure 2 shows an ArchiMate model that was defined before the ontological analysis to reflect the range of elements and relations used to represent capabilities and resources in the proposed ArchiMate extension. The example focuses on the ArchiSurance company and its 'Insuring Capability'. The example shows other capabilities that are associated to the 'Insuring Capability' ('Claim Handling Capability', 'Selling Capability', 'Damage Assessment Expertise Capability' etc.). The example also reviews the resources assigned to the capabilities ('Damage Assessment Resources', 'Money', 'Authorized Garage', etc.). The example reflects that capabilities can be decomposed into other capabilities, and can be realized by some behavior elements (as in the case of 'Insuring Capability' and 'Claim Handling Capability' being realized by the 'Assess claim' business process, by the 'Financial

administration' business function and by the 'Claim registration' application service). Similarly, resources can be realized by structure elements and resources can be assigned to capabilities (as in the 'Car Damage Assessment Resource' and 'Claim Handling Capability' case).
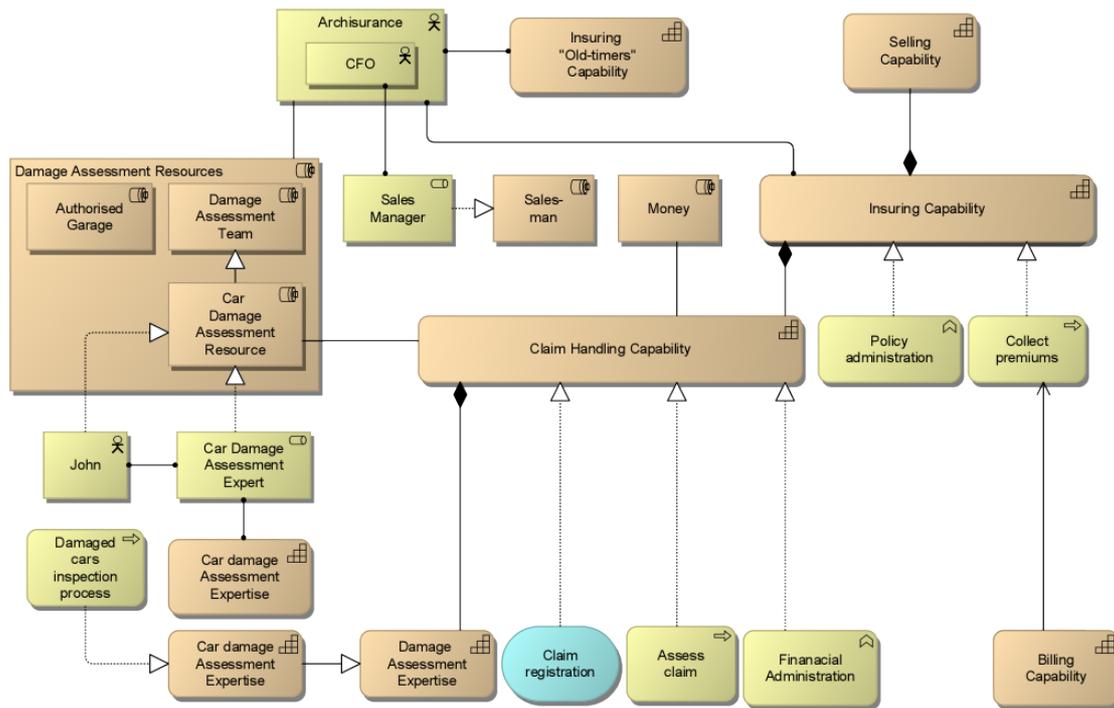

Figure 2 - Example model using the BSVC extension of ArchiMate

3. ONTOLOGICAL ANALYSIS

Since the last 80's there has been a growing interest in the use of foundational ontologies for evaluating and reengineering conceptual modeling languages and methodologies, as in (Wand and Weber 1989) and (Wand and Weber 1990). A foundational ontology defines a system of domain-independent categories and their ties, which can be used to articulate the conceptualizations of reality. The use of foundational ontologies aims to ensure ontological correctness of the language and of the models described with the language.

Empirical evidence (Bodart et al. 2001), (Gemino and Wand 2005), (Shanks et al. 2008), and (Recker et al. 2011) have corroborated the hypothesis that a suitable conceptual modeling language should comprise modeling elements that reflect conceptual modeling categories and relations defined in a foundational ontology. A number of enterprise modeling approaches have been subject to ontology-based analysis in recent years, e.g., (Azevedo et. al. 2011), (Santos Jr et al. 2013), (Davies, Rosemann, and Green 2000), (Gailly, Geerts, and Poels 2009), (Green and Rosemann 2005), (Rosemann, Green, and Indulska 2004), (Guizzardi et al. 2007), (Laurier and Poels 2009a), and (Laurier and Poels 2009b).

Ontological analysis is performed by considering a mapping between modeling constructs and the concepts in an ontology. "On the one hand, each modeling element can be interpreted using the ontological theory as a semantic domain. On the other hand, concepts of the domain of discourse (captured in the ontological theory) should be represented by modeling elements of the language being considered. According to (Weber 1997), there should be a one-to-one correspondence between the concepts in the ontology and modeling elements" (Santos Jr et al. 2013). When the correspondence cannot be obtained, the following language problems can be identified[3]:

---

[3] Adapted from (Santos Jr et al. 2013)

- Construct excess: exists when a modeling construct does not correspond to any ontological concept. Since no mapping is defined for the exceeding construct, its meaning becomes uncertain, hence, undermining the clarity of the specification. According to (Wand and Weber 1990), users of a modeling language must be able to make a clear link between a modeling construct and its interpretation in terms of domain concepts. Otherwise they will be unable to articulate precisely the meaning of the unclear construct and, consequently, the specifications they generate using the language. Therefore, a modeling language should not contain construct excess and every instance of its modeling constructs must represent an individual in the domain.
- Construct overload: exists when a single modeling construct can represent multiple ontological concepts. Construct overload impacts language clarity negatively. Construct overload is considered as an undesirable property of a modeling language since it causes ambiguity and, hence, undermines clarity. When a construct overload exists, users have to bring additional knowledge not contained in the specification to understand the phenomena which are being represented.
- Construct redundancy: exists when multiple modeling constructs can be used to represent a single ontological concept. Construct redundancy is a violation of parsimony. In (Wand and Weber 1990), the authors claims that construct redundancy "adds unnecessarily to the complexity of the modeling language" and that "unless users have in-depth knowledge of the grammar, they may be confused by the redundant construct. They might assume for example that the construct somehow stands for some other type of phenomenon." Therefore, construct redundancy can also be considered to undermine representation clarity.
- Construct deficit: exists when there is no construct in the modeling language that corresponds to a particular ontological concept. Construct deficit entails lack of expressivity of the modeling language, i.e., there are relevant phenomena in the considered domain (according to a domain conceptualization) that cannot be represented by the language. Alternatively, users of the language can choose to overload an existing construct, thus, undermining clarity.

Figure 3 illustrates the relation between modeling constructs (in a language's syntax) and ontological concepts.
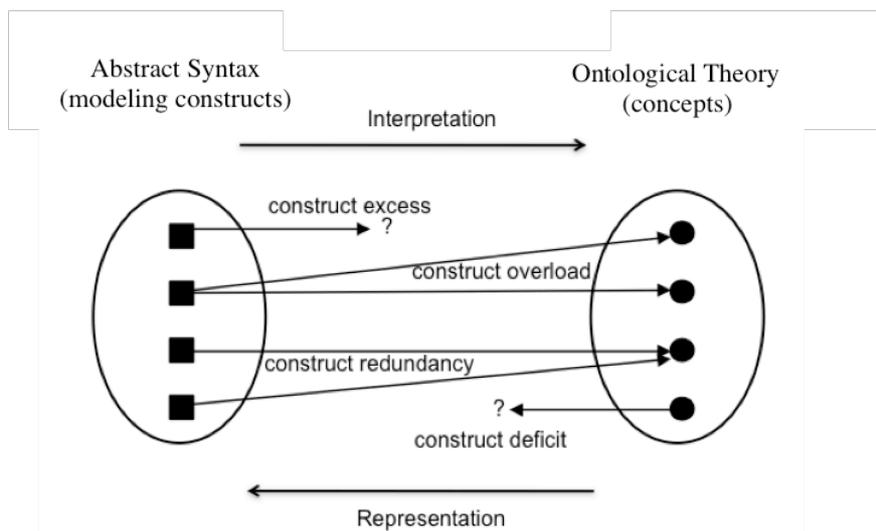

Figure 3 - Issues uncovered by ontological analysis - adapted from (Moody 2009)

Modeling languages aiming at wide adoption, such as ArchiMate, should avoid ontological deficiencies. Recker et al. (2011) reported results from a study with 528 modelers demonstrating that "users of conceptual modeling grammars perceive ontological deficiencies to exist and that these deficiency perceptions are negatively associated with usefulness and ease of use of these grammars".

In order to uncover any of the aforementioned problems, we discuss the interpretation of the resource, capability and competence modeling elements in terms of a foundational ontology. The issues uncovered in Sections 5, 6 and 7 are addressed with recommendations for improvements in Section 8.

4. ONTOLOGICAL FOUNDATIONS

In our ontological analysis we make use of the Unified Foundational Ontology (UFO) as semantic foundation. The UFO ontology has been developed based on theories from Formal Ontology, Philosophical Logics, Philosophy of Language, Linguistics and Cognitive Psychology (Guizzardi 2005). Like other foundational ontologies, such as DOLCE (Gangemi et al. 2002) and GFO (General Formalized Ontology) (Herre et al. 2006), it has the firm ontological grounding of the so-called Aristotelian Square (or Four-Category Ontology) (Lowe 2006). This allows for the construction of an ontology that is founded in a parsimonious set of essential ontological categories, while still being able to account both for natural science as well as linguistic and cognitive phenomena (Guizzardi 2005). However, differently from DOLCE and GFO, UFO was developed with the primary goal of providing foundations for conceptual modeling. As a consequence, many aspects that are essential for conceptual modeling have not received a sufficient attention in DOLCE and GFO. One example is the notion of material relations and relational properties (G. Guizzardi and Wagner 2008), which does not receive any treatment in DOLCE. Another example is the finer-grained distinctions that are needed for dealing with different categories of universals (e.g., kind, roles, roleMixins) in conceptual modeling, which are not addressed in GFO. In fact, a category such as roleMixin, which is so recurrent in conceptual modeling to the point of becoming a semantic design pattern (G. Guizzardi et al. 2004), is not present in any of these alternative foundational ontologies but UFO. A third example is the availability of a rich system of part-whole relations. Part-whole relations have received a richer treatment in UFO than in other foundational ontologies. Furthermore, UFO presents a treatment of dispositions (and their systematic connection to events and situations), which allows us to properly address many of the issues that are germane to the purposes of this article.

The UFO ontology has been employed in many semantic analyses, including ones in the topics of ARIS EPCs (Santos Jr et al. 2010), i*/TROPOS (Guizzardi and Guizzardi 2011), goals and business processes models (Cardoso et al. 2010) and role-related concepts in EA (Almeida, Guizzardi, and Santos Jr 2009). In particular, it has been previously used to analyze and interpret the semantics of the ArchiMate motivation concepts, having led to recommendations of that proposal that have been incorporated in the latest ArchiMate specification (Azevedo et al. 2011).

The UFO ontology has been structured in three main parts: UFO-A is the core of the ontology and is concerned with endurants (e.g., entities such as objects, qualities, relators, and dispositions); UFO-B is an ontology of events and, as such, makes a distinction between enduring and perduring individuals and elaborates on the possible connections between these two fundamental types of entities; UFO-C is built on top of UFO-A and UFO-B and focuses on social aspects of reality by dealing with notions such as plans, goals, agents, commitments, and normative descriptions.

For a more detailed discussion of the development and applications of UFO we refer to (Guizzardi 2005), (Guizzardi, Falbo, and Guizzardi 2008), (Almeida and Guizzardi 2013) and (Guizzardi et al. 2013). This section is based on the UFO description that appeared in (Azevedo et al. 2011).

4.1. Basic Elements

A fundamental distinction in this ontology is between the categories of *individuals* and *universals*. *Universals* are predicative terms that can be applied to a multitude of *individuals*,

capturing their general aspects. *Individuals* are entities that exist in reality possessing a unique identity and that can instantiate one or more *universals*.

Further, UFO makes a distinction between the concepts of *endurants* and *events*. *Endurants* are *individuals* that persist in time while keeping their identity, in the sense that if we say that in circumstance *c1* an *endurant e* has a property *p1* and in circumstance *c2* a property *p2* (possibly incompatible with *p1*), *e* is the same *endurant* in each of these situations. Examples can include a particular person (say Peter) weighting 70kg in one circumstance and 78kg in a different circumstance, while being the same *individual* (Peter) in these two circumstances. Other examples include organizations (the University of Twente, the Federal University of Espírito Santo, etc.) and everyday objects (a ball, an apple, etc.). *Events*, in contrast, are individuals composed of temporal parts, they happen in time, in the sense that they extend in time and accumulate temporal parts. Examples include a particular execution of a business process, a meeting or a soccer game. Whenever an *event* occurs, it is not the case that all of its temporal parts necessarily occur. For instance, if we consider a business process "Buy a Product" at different time instants, at each time instant only some of its temporal parts are occurring.

A *substantial* is an *endurant* that does not depend existentially on any other *individual*, what is usually referred by the common sense term "object". In contrast with *substantials*, *moments* (also known as 'abstract particulars' and 'tropes' (Lowe 2006), (Heil 2003)) are existentially dependent entities, i.e., for a *moment x* to exist, another *individual* must exist, named its bearer. Examples of *moments* include an apple's color, John and Mary's marriage, an electric charge on a conductor, etc. *Moments* in UFO include both *qualities* (e.g., color, weight, temperature) and *dispositions* (e.g., the fragility of a glass, the disposition of a magnet to attract metallic material) (Guizzardi et al. 2013). In the philosophical jargon, the category of dispositions typically subsumes properties such as powers, tendencies, potentials, capacities, capabilities, affordances, liabilities and propensities. In general, these properties have in common that they endow their bearers with the potential of exhibiting some behavior or bringing about a certain effect under certain conditions. Dispositions are only manifested in particular situations, but they can also fail to be manifested. When manifested, they are manifested through the occurrence of *events*. Take, for example, the disposition of a magnet *m* to attract metallic material. The object *m* has this *disposition* even if it is never manifested, for example, because *m* was never close to any magnetic material. Nonetheless, *m* can certainly be said to possess that intrinsic property (Heil 2003), (Molnar 2006), (Mumford 2003).

Existential dependence can be used to differentiate *intrinsic* and *relational moments*. *Intrinsic moments* are dependent on a single individual, while *relational moments* (also called *relators*) depend on a plurality of individuals. Examples of the first include an apple's weight and color, while examples of the latter include John and Mary's marriage, John's enrollment at the University of Twente. A *relator* is the truthmaker of a material relation.

A *universal* is rigid if it necessarily applies to its instances, i.e., if it applies to its instances in every possible world (e.g., Apple, Person). A *kind* is the rigid *substantial universal* that supplies a principle of identity for the *substantial individuals* that instantiate them. Every *substantial individual* must be an instance of exactly one *kind*. In contrast to rigid universals, a *universal* is anti-rigid if it does not apply necessarily to all its instances. *Roles* are anti-rigid and relationally-dependent *universals* (e.g., Student, Husband). This means that *roles* are played by a *substantial* in a relational context, i.e., it requires the role player to be connected by a *relator* to other individuals in that context or that the role player participates in a event playing what is termed a processual role.

Whenever entities of different *kinds* have similar properties they may be classified by *substantial universals* termed *Mixin universals.* Rigid Mixin Universals subsuming different Kinds are termed *categories* (e.g., Physical Object, Living Entity). Some Mixin Universals are anti-rigid and represent abstractions of common properties of *roles*. These are termed *role*

mixins. An example of *role mixin* is 'customer', which can be played by 'persons' (i.e., entities of the kind Person) and 'organizations' (i.e., entities of the kind Organization). In any case, Customer is an anti-rigid and relationally dependent type for all its players.

4.2. Intentional and Social Elements

An *agent* is a specialization of *substantial individual*, representing entities capable of bearing *intentional moments*. These include mental states, such as individual *beliefs*, *desires* and *intentions*. Intentionality should not be understood as the notion of "intending something", but as the capacity to refer to possible situations of reality (Searle 1983). Every *intentional moment* has an associated *proposition* that is called the *propositional content* of the *moment*. In general, the *propositional content* of an *intentional moment* can be satisfied (in the logical sense) by *situations* in reality. Every *intentional moment* has a type (*belief*, *desire* or *intention*). The *propositional content* of a *belief* is what an *agent* holds as true. Examples include one's belief that the Eiffel Tower is in Paris and that the Earth orbits around the Sun. A *desire* expresses the will of an *agent* towards a possible *situation* (e.g., a desire that Brazil wins the next World Cup), while an *intention* expresses desired states of affairs for which the *agent* commits to pursuing (internal commitment) (e.g., John's intention of going to Paris to see the Eiffel Tower).

Actions are *intentional events*, i.e., *events* with the specific purpose of satisfying (the propositional content of) some intention of an *agent*. The *propositional content* of an *intention* is termed a *goal*. Only *agents* are said to perform *actions* (Guizzardi, Falbo, and Guizzardi 2008), as opposed to *non-agentive objects* that *participate* (*non-intentionally*) in *events*.

Agents can be further specialized into *physical agents* (e.g., a person) and *social agents* (e.g., an organization). *Social agents* are further specialized into *institutional agents* and *collective social agents*. *Institutional agents* are composed of other agents, each one contributing to the functionality (or behavior) of the institution, also termed *functional complex* (Guizzardi 2005). In addition to *institutional agents*, UFO also acknowledges the existence of *collective social agents*, which are distinguished from *institutional agents* in that all its members play the same role in the collective.

Similarly to *agents*, *non-agentive objects* can be specialized into *physical objects* and *social objects*. A category of *social objects* of particular interest to us here is that of *normative descriptions*. *Normative descriptions* are *social objects* that create social entities recognized in that context. Examples of *normative descriptions* include a company's regulations and public laws. Examples of social entities that can be defined by *normative descriptions* include *social roles* (e.g., president, manager, sales representative), *social role mixins* (whose instances are played by entities of different kinds, e.g., customer, which can be played by persons and organizations), *social agent universals* (e.g., that of political party, education institution), *social agents* (e.g., the Brazilian Labour Party, the University of Twente), *social object universals* (e.g., currency) and other *social objects* (e.g., the US dollar) or even other *normative descriptions* (e.g., a piece of legislation). *Normative descriptions* are recognized by at least a *social agent*. Figure 4 shows a fragment of the specializations of individuals in UFO.
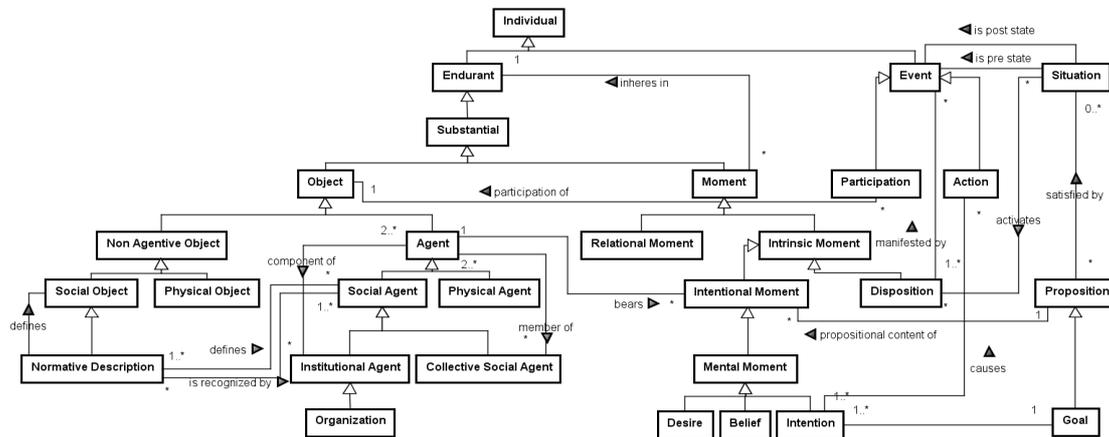
Figure 4- Fragment of Specializations of Individuals in UFO

5. ONTOLOGICAL ANALYSIS OF RESOURCE

In this section we discuss the ontological analysis and interpretation of the *resource* modeling element introduced in (Iacob, Quartel, and Jonkers 2012). We discuss possible interpretations in terms of UFO and consider the consequences of the various interpretations on the usage of the language. We make a thorough analysis, and the section is structured according to six problems that have been revealed in this analysis, which for convenience are named R1 to R6. The analysis of the revealed problems leads to the recommendations presented in Section 8.

5.1. Problem R1

The Oxford Dictionary defines *resource* as "a stock or supply of money, materials, staff, and other assets that can be drawn on by a person or organization in order to function effectively". In (Iacob, Quartel, and Jonkers 2012), a similar intuition is put forward when motivating the *resource* element in the BSVC extension: "the achievement of a goal assumes the availability and (constrained) consumption of certain resources". Further, *resources* are also characterized as "assets owned or controlled by an organization".

Since "assets" are (valuable) things, in a first examination, this characterization seems to suggest that *resources* represent specific individuals, such as business actors (e.g., in case of staff as *resource*) or business objects (agentive objects (*agents*) or *non-agentive objects*). Nevertheless, this interpretation would show a clear case of construct redundancy (G. Guizzardi 2005), as the additional *resource* modeling element would serve no purpose, being supplanted by the previously existing *structure elements* of the language (such as *business actor* and *business object*). We must conclude this is not the intention of the designers of the extension, which indicate further that a *resource* is "an abstraction of *structure elements*" and include a "realizes" relation that may be used to connect *structure elements* to the *resources* they "realize". This suggests that it is not the specific *structure element* that is represented using a *resource*, but some more abstract notion, which reveals the dependence on a *structure element* with certain characteristics without specifying the particular element involved. In other words, we understand that the *resource element* defines some type of *structure element* (a *universal*), and that the *structure element* that realizes the *resource* instantiates this type. An example of this would be a model that includes a *business actor* 'John' that realizes a *resource* called 'Car Damage Assessment Resource'.

If we further consider that *resources* are used in the scope of the efforts to achieve goals, we can understand that a *resource* refers to the *role* an *object* (*agentive* or *non-agentive*) plays when employed in the scope of these efforts. In the example above, 'John' functions as a 'Car Damage Assessment Resource' in some context of the organization (for example, that of the process of assessing damages).

Further, in order to play a particular *role*, an *object* may be required to instantiate some particular type (what is called an "allowed type" in (Bock and Odell 1998)). For example, any 'Car Damage Assessment Resource' may need to possess specific damage assessment skills, and thus instantiate some specific *universal* that is characterized by these skills.

Thus an intermediate conclusion is that *resource* represents an *externally dependent universal* (either a role or role mixin) that may be instantiated by *objects* of a particular allowed type. The fact that *resource* models both the *role* an *object* plays in a particular context of usage as well as its allowed type has some consequences to the terms used in the label of a *resource*. In some cases, such as 'Car Damage Assessment Resource', the context of usage is emphasized, focusing thus on the *role* or *role mixin* that is instantiated when the *resource* is used; in other cases, such as 'Money', the allowed type required for *role* playing is emphasized in the label.

An insight that comes out of this interpretation is that, as a *role* (*mixin*), a *resource* should have a context of usage, which in UFO is defined in the scope of a *material relation* (or in the scope of an *event*). This means some asset is a *resource* in a defined context, but not in others. For example, the 'Car Damage Assessments Expert' is a *resource* in the 'Damaged cars inspection process' because of the 'Car Damage Assessment Expertise', but it should not be considered a *resource* for the 'Collect premiums' process. This cannot be directly represented with the BSVC metamodel, because there is no notion of 'use' of *resources*. We label this as problem 'R1'.

5.2. Problems R2 and R3

The *resource* concept is also defined as "an asset owned or controlled by an individual or organization". Being "owned or controlled" is understood as being available for the organization, e.g., by an employment contract between employers and employees, or by having the right/ownership over a certain object. For the cases in which the *resource* is an *agentive element* (*agent*) we understand the 'controlling' in the context of the *social relator* that bounds the particular *individual* or organization with the first, e.g., the employment contract. *A* controls *B* means that there is a (possibly a set of) meta-commitment(s) of *B* towards *A*. In other words, *A* has meta-claims over *B* and, hence, the ability to delegate to *B* and, consequently to increase its social ability (Bringuente, Falbo, and Guizzardi 2010). For the interpretations in which the *resource* is a *non-agentive object*, we understand "the control or owning of the asset" as the ability to have that element to participate in an *event* of interest, in which the organization *A* has a certain right with respect to the object *O* (for example, a right to use, to consume, to destroy, to sell, etc.). The current metamodel does not allow the modeler to identify who controls the *resource*, aside from the use of the very general and abstract *associated with* relation, which has no specific semantics and can be applied between any constructs in the language. This could be an issue when there are multiple business actors (different organizations, business departments) that could control this *resource*. For example, the model user would not be able to know which organization controls the *resource* modeled, and as such, could not distinguish if that *resource* is or is not available at his organization. We call the lack of expressiveness of control relations 'R2'.

Furthermore, the extension does not distinguish between *resources* potentially played by *agents* from those potentially played by *non-agentive objects*. We call this lack of expressiveness 'R3'. Further, R3 has an impact on R2, since *non-agentive resources* are to be controlled by someone, and, as importantly, should not control any other element.

5.3. Problems R4 and R5

Other examples of usage indicate that not only *business actors* and objects may realize a *resource*, but also that *business roles* may be said to realize a *resource*. In our example, the

'Car Damage Assessment Expert' *business role* realizes the 'Car Damage Assessment Resource'. In this case, we should understand that whichever *object* instantiates the *role* represented by the *business role* may also instantiate the *role* (*mixin*) represented by the *resource*. Intuitively in the example, not only 'John' but also any other damage assessment expert is a 'Car Damage Assessment Resource'. For these cases, the language does not determine whether one or more *individuals* instantiating the *role* (*mixin*) represented by the *resource* are required, used or controlled in the particular context. In other words, it is not possible to express whether all the instances of that type are required, used or controlled, if just one instance of that type is required, used or controlled or if an arbitrary set of instances of that type are required, used or controlled. We label this as problem 'R4'.

Further, the current ArchiMate language does not address the cases in which the *resources* are objects of interest or raw materials (e.g., 'Money' as in the running example, or gold, diamond, gas), i.e., passive *non-agentive* elements. We believe the language designers have tried to cover this by stretching the *resource* element, and using it directly to represent such objects. However, these would be *resources* that do not have any *structure element* to realize them, since there are no *structure elements* that can represent these types of objects in ArchiMate (the passive structure of ArchiMate focuses primarily on information objects). Also, in these cases, the *resources* are *role* (*mixins*) and the language is not able to express if the same instance is to be considered a *resource* in various contexts (e.g., usage of the same amount of money in different contexts). Also, it is not possible to express any property associated to the element itself (e.g., quantity of money, gold carats). We label this as problem 'R5'.

The *resource* concept is related to the motivation extension through the realization relation, in which a '*resource* realizes a *requirement*'. According to (Azevedo et al. 2011), a *requirement* corresponds to a *normative description*, which states that if a system (in a broad sense) is to exist, then it must satisfy a particular *proposition*. In this case, we understand that *proposition* refers to the *object* (or objects) playing the *resource role*. Any instance of the *role* (*mixin*) represented by the *resource* must satisfy the *requirement's proposition*. To put it simply, a *requirement* adds characteristics to a *resource's* allowed type.

The proposal also states that "the achievement of a *goal* assumes the availability and (constrained) consumption of certain *resources*". However, *goals* are not associated directly to *resources*, and the proposal is silent on the issue of *resource* consumption. *Resources* and *goals* are only indirectly related through the '*goal* is realized by *requirement*' and the '*requirement* is realized by *resource*' relations. At this point, no interpretation can be given to the textual definition, and further language documentation would be required on the topic of *resources* availability and consumption.

5.4. Problem R6

Now we focus on the common ArchiMate relationships that apply to the *resource* concept (specialization and aggregation). We interpret the specialization relation between *resources* as subsumption between the *roles* or (*role mixins*) represented by the *resources*. The aggregation relation between *resources* suggests some sort of whole-part relationship, since aggregation in ArchiMate may be represented by containment (see in Figure 2 the relation between 'Damage Assessment Resources' as a whole, and 'Authorized Garage', 'Damage Assessment Team' and 'Car Damage Assessment Resource' as parts). Since *resources* may represent both *agentive* and *non-agentive objects*, it would be possible to combine these with aggregation. We interpret this as a very general sort of whole-part relation known as mereological sum. However, there is no distinction between AND or OR *resource* aggregations in ArchiMate. Thus, when *resources* are aggregated, it is unclear whether all the aggregated *resources* are required/used, or whether one or any arbitrary number of them is required. We label this as problem 'R6'.

6. **ONTOLOGICAL ANALYSIS OF CAPABILITY**

In this section we discuss the ontological analysis and interpretation of the *capability* modeling element introduced in (Iacob, Quartel, and Jonkers 2012). This section discusses the 5 problems revealed by the ontological analysis, named C1 to C5, although the ontological analysis is continued throughout the subsections. The revealed problems inspired the recommendations in Section 8.

6.1. **Problem C1**

The BSVC proposal defined *capability* as "the ability (of a static structure element, e.g., actor, application component, etc.) to employ *resources* to achieve some goal. [...] Also *capability* assumes the ability to employ (i.e., configure, integrate, etc.) *resources*". The definition also states that "*capability* (similarly to *resource*) can be seen as an abstraction of some behavior of the static *structure element*".

We intuitively understand that a *capability* is attributed to some *agent* and gives that *agent* its power to bring about some behavior in order to achieve a desired outcome. From the excerpt "of a *static structure* element, e.g., actor, application component, etc.", the *capability* appears to belong to the specific *individual* that is to bring about the desired outcome. This would lead us to interpret that *capabilities* are *dispositions* in UFO (*dispositions*, i.e., *moments* that are only manifested in particular situations and that can also fail to be manifested). However, carefully examining the "abstraction of some behavior" fragment, and considering the same pattern that was employed by the language designers with respect to *resources* (as "abstractions of *structural elements*"), we understand that *capabilities* should be interpreted as types of *dispositions* (*disposition universals* in UFO).

Often a *capability* represents a general *disposition* type. For example, the 'car damage assessment expertise' *capability* is a general *disposition* type that is implicitly specialized into a more specific type (e.g., the *capabilities* to assess car damage produced by fire, to assess car damage caused by flood, and to assess car damage after a crash).

This dispositional account is also applicable for cases in which some organization might hire a different organization to perform processes that realize a *capability*, and still state to have that *capability*, in this case because it has the *disposition* of delegating the capability (R. Guizzardi and Guizzardi 2011), (Bringuente, Falbo, and Guizzardi 2010)). This is related to the idea of what an organization can "socially perform". If *A* has a meta-commitment from *B* to execute *S*, then *A* (socially) can do *S*. An *object* can have *dispositions* which arise from its parts, or from the network of its delegation relations (Bringuente, Falbo, and Guizzardi 2010).

Preferably, the language should allow us to infer which *individuals* bear the *dispositions* that are related to that *capability*. However, the original metamodel does not include relations between the *capabilities* and the *structure elements* that are said to have the *capabilities*, such as a *business actor* or *business role*. As a consequence, the language does not allow one to identify the *individuals* or types of *individuals* that bear *dispositions* of the type presented by the *capability*, aside from using the generic and semantically-neutral *association* of ArchiMate. In other words, it is not possible to express in the language which *structure element* has a *capability*, including the *capabilities* an organization has, unless the capability is realized by some *behavior element*. We label this as 'problem C1'.

For the case of *resources*, these are assigned to a *capability*, in the sense that they are used in order to leverage *capabilities*), but *resources* themselves do not have *capabilities* in the original extension. This is represented in Figure 2, in which the organization has not assigned the 'selling capability' to any *resource*. Since the language cannot express which are the *capabilities* inhering in the 'Salesman' *resource* and the 'Sales Manager' *structure element* (*business role*), the organization is not able to know which *resource* or *structure element* has the required 'selling capability' to properly assign its performance.

6.2. Problems C2, C3 and C4

The *capability* concept has three defined relationships, according to the original metamodel. We now focus on "*capability* realizes *requirement*". Again, according to (Azevedo et al. 2011), a *requirement* corresponds to a *normative description*, which states that if a system (in a broad sense) is to exist, then it must satisfy a particular *proposition*. In this case, we understand that *proposition* refers to the *dispositions* that instantiate the *disposition universal* represented by the *capability*. The *dispositions* must be in accordance with the *requirement*, in order to satisfy its *proposition*.

We now focus on the relations '*capability* realized by *behavior element*' and '*resource* assigned to *capability*'. We understand that the first needs to be considered also with the participation of the *resource* (via the '*resource* assigned to *capability*' relationship). We understand that this pattern of relations can have two different interpretations. We label this as 'problem C2'. A first one is that the *resource object* has a *disposition* that instantiates the *disposition universal* represented by the *capability*, and that the *participation* of the *resource* manifesting its *disposition* is required in order to perform the *behavior element* (an *event universal*). For example, the 'Car damage Assessment Expertise' *capability* on Figure 2 is to be manifested in the 'Damaged cars inspection process' *business process* in order for the organization to perform that *process*. The second possible interpretation is that the *capability* is acquired (by the *resource*) with the performance of the *behavior element* (Molnar 2006), i.e., the *resource* acquires a *capability* after the process is performed. For example, the 'Car damage Assessment Expertise' *capability* is acquired with the occurrence of the 'Damaged cars inspection process' *process*. In UFO, this can be interpreted as follows: *s* is a *situation* in which the *object* has the *disposition d*, *e* is an *event* representing the *behavior element*, and *e* is a *pre-state* of *s*. If no *resource* is represented, the object that is acquiring or manifesting the *capability* is unknown. We label this as 'problem C3'. In this case, one can argue that it is an organization's *capability*, but it is not possible to clearly define it without relating it to the *object*, and the specification does not define this case. Even when related to the *resource* concept, since the *resource* that is acquiring the *disposition* represents a *universal*, a type element, the actual *object* that is acquiring the *disposition* is undetermined. The language is not expressive enough to state if one *individual*, all the *individuals* that instantiates the *universal* or an arbitrary combination of *individuals* instantiating the *universal* are acquiring the *disposition*. In none of these interpretations it is possible to know in advance if it is one *individual*, all the *individuals* that instantiates that *universal* or an arbitrary combination of them that are related to the *disposition*. In the first interpretation, is not possible to know how many *objects* are to manifest their *dispositions* in the *event* represented by the *behavior element*. In the later, it is not possible to know which *object* is to acquire the *disposition*. We label this as 'problem C4'.

6.3. Problem C5

Now we focus on the common ArchiMate relations that apply to the *capability* concept. The specialization relation between *capabilities* should mean that a *disposition universal* (type) subsumes other *disposition universal*, and the aggregation relation between *capabilities* is interpreted as (*complex*) *dispositions*, which are *dispositions* based on other *dispositions* (Molnar 2006). However, there is no distinction between AND or OR *capability* aggregations in ArchiMate, i.e., it is not clear whether the "aggregated" capabilities are all required or whether there are optional *capabilities*. The language also lacks expressiveness to state if all the *capabilities* associated to a *behavior element* are acquired or manifested, if just one of them is acquired or manifested, or if an arbitrary number of them are acquired or manifested. We label this as 'problem C5'.

7.   ONTOLOGICAL ANALYSIS OF COMPETENCE

In this section we discuss the ontological analysis and interpretation of the competence modeling element introduced in (Iacob, Quartel, and Jonkers 2012). This section discusses the 3 problems revealed by the ontological analysis, named C1.1, C2.1 and C6, although the ontological analysis is continued throughout the subsections. The revealed problems inspired the recommendations in Section 8.

7.1. Problems C1.1 and C2.1

The *competence* concept was introduced in (Iacob, Quartel, and Jonkers 2012) as a "specialization of *resource* (intangible or personnel-based)". The proposal states that "a core competence is a particular strength of an organization. Core competences are the collective learning in organizations, and involve how to coordinate diverse production skills and integrate multiple streams of technologies. Examples of core competences include technical/subject matter know-how, a reliable process and/or close relationships with customers and suppliers". This was "based on the fact that the definition of competence […] is almost identical with that of personnel-based *resources*". Personnel-based *resources* have been exemplified as "technical know-how, other knowledge assets including organizational culture, employee training, […]".

Based on the aforementioned statements regarding *competence*, we understand that a *competence* is something that an element, when provoked, is able to do, or to perform. It addresses the element's capacity of performing an activity. At a first examination, this characterization seems to suggest that *competence* is to be applied to specific *individuals*, and that it would be a *disposition* that *inheres* in the *individual*. However, the original metamodel shows that *competence* is a *resource*, i.e., a *universal*. Based on the *resource* interpretation as a *universal*, we conclude that the *competence* also represents a *universal*, which would be a *disposition type*, whose instances *inhere* in the *objects* that play the *role* represented by a *resource*.

Since a *competence* is a specialization of *resource* in the BSVC, it inherits the *resource's* relations. We now focus on the '*competence realizes requirement'*. According to *requirement* interpretation (Azevedo et al. 2011), we interpret this relation as 'the *disposition* of the *object satisfies* the *requirement's* proposition'. We understand that the *competence* is of the same *disposition type* as of the *capability* that is manifested (or acquired depending on the given interpretation) with the performance of the *behavior element* (*event*). The interpretation varies according to the interpretation given to the *capability* relationship. We label this as 'problem C2.1', since it is a consequence of C2. This can be interpreted as that: (i) by being able of executing a certain *behavior element* (an *event universal*), the *resource* object has a *disposition* d (of the *competence* defined type) that instantiates the type represented by the *capability* or that; (ii) s is a *situation* in which the *object* has the *disposition* d and e is an *event* representing the *behavior element*, e is a *pre-state* of s, in which the d is an instance of the *competence*. The relation '*competence* is *realized by structure element'* relates the *dispositions* to the elements that *bears* them. This is interpreted in UFO as that the *object* that represents the *structure element bears* a *disposition* of that *disposition type*. This is desirable, however, since it is not enforced by the language, we label the lack of the knowledge on which *object bears* the *disposition* as 'problem C1.1', since it is associated to C1.

7.2. Problem C6

The interpretation of the *competence* concept points to the same ontological construct as the interpretation of the *capability* concept. We label this as 'problem C6'. The *competence* concept appears to have been introduced to fill the gap in the proposal that it is unknown which *capabilities* a *resource* (or *structure element*) has. This shows a case of construct

redundancy in the language. According to (Weber 1997), "construct redundancy occurs when more than one grammatical construct can be used to represent the same ontological construct". Our analysis confirms and explicates the informal suspicions raised in the original proposal text when it states that "depending on the (interpretation of the) definition of competence, one may argue that, for example it is more natural to introduce competence in the metamodel as a specialization of a capability" (Iacob, Quartel, and Jonkers 2012). The original proposal also states that "the semantic distance between competence, on one hand, and either resource or capability, on the other hand, is too small".

8. WELL-FOUNDED LANGUAGE REVISION

In this section we propose improvements to the language based on the ontological analysis, the original intended meaning for the extension concepts and solutions to problems revealed in Sections 5, 6 and 7. We propose a revised metamodel to the language extension as part of the solution. We have attempted to preserve the original intended interpretations and original relations whenever possible, still addressing the semantic problems. Some additional expressiveness is made possible by using relations that were not initially employed in the BSVC metamodel, but most of which already existed in ArchiMate. We refrain from defining extensions to the ArchiMate core, focusing only on the BSVC capability and resource concepts and their relations. The competence construct was eliminated as a direct consequence of the ontological analysis.

Figure 5 presents a fragment of the revised metamodel. The constructs, relations and their semantics are discussed in the sequel.
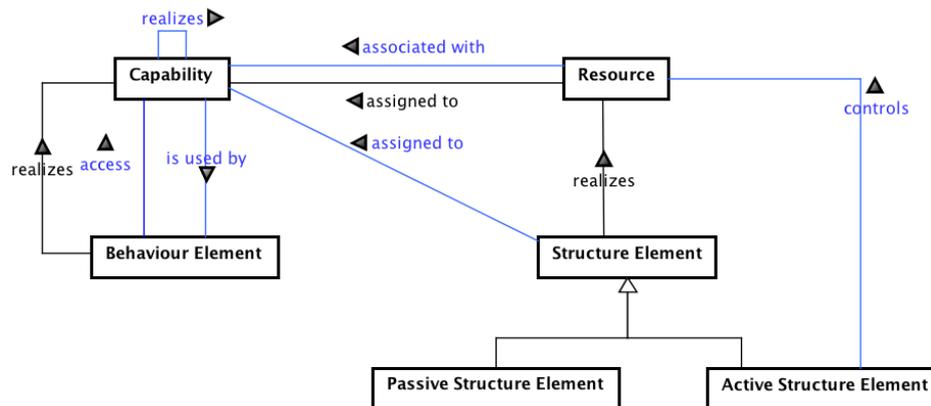

Figure 5 – Fragment of the proposed metamodel.

8.1. Resource

The resource concept in UFO represents a role or a role mixin that objects may play in particular contexts of usage. In Section 5, problem R1 stated that an element is a resource in a defined context, but it is not a resource in all situations, and that this was not enforced by the language. Thus, a resource should have a defined context.

The metamodel proposed here defines two relations between resources and capabilities, namely one that already existed in the extension (*assigned to*), and another due to the ArchiMate core (the generic *associated with* relation).

We propose that the *associated with* relationship between resource and capability should represent the context of usage of a resource. This relation should represent that a resource is allocated to perform the capability it is associated to, thus being the performance of that

capability the context of usage of the element that is playing the resource role. This relation addresses problem R1.

Another relation involving resources is the *assigned to* relation, which we use to denote that a resource has that capability. It means that the object that instantiates the role represented by the resource has a disposition of the type represented by the capability. Combining the two aforementioned relations enables the modeler to represent the prescriptive capabilities that resources should have in order to perform a specific capability (e.g., to perform c_1 a resource r should have c_a, c_b and c_c). These relations address problems C1, C1.1, C2 and C2.1, partially contributing to their solutions.

The metamodel also introduces a *controls* relation between *resources* and *structure elements*. The controls relation defines that a resource is controlled by an active structure element. Being controlled is understood as being available for the organization (e.g., by an employment contract between employers and employees, or by having the right/ownership over a certain object). Since the ArchiMate specification defines *passive structure elements* as "objects in which behavior is performed", usually denoting information and data elements, we understand that only *active structure elements* are to *control resources*. This relation addresses problem R2.

Problem R6 states that there is no distinction between AND or OR resource aggregations in ArchiMate and, as such, the language lacks the expressivity to state optional resources. Although ArchiMate does not distinguish between AND or OR aggregations, an extension allow the representation of optional resources is addressed and described in Section 8.5.

8.2. Structure Element and Its Specializations

The proposed metamodel further introduces the *assigned to* relation between structure element and capability. This relation defines that a structure element has a capability. It means that the object represented by the structure element has a disposition of the type represented by the capability. This relationship enables the organization to specify which capabilities it has, as a whole, and as a summary of the capabilities of its participating parts (organizational units, individual agents, etc.). This relation, in addition to the 'resource is assigned to capability relation', described in Section 8.1, solves problems C1 and C1.1.

The Structure Element *realizes* Resource relation means that a resource is realized/performed by the mentioned structure element. This enables the organization to be able to match the 'required' capabilities a resource should have, as in its prescriptive version, to the capabilities the organization's structure elements actually have.

Problem R4 stated that for the cases in which a resource is realized by a business role (a universal), the language does not determine whether one or more individuals that instantiate the role (mixin) represented by the resource are required in the context of usage. This limitation can be addressed by the addition of a replication attribute to the language, such as a cardinality constraint. However, this kind of extension is out of the scope of this work, since it deals with ArchiMate core. These limitations have been reported by industry, and some ArchiMate tools already implement a replication attribute for similar purposes, in which case R4 could be considered remediated.

The Structure Element is explicitly shown in the proposed metamodel with its specializations, Active Structure Element and Passive Structure Element. Problem R3 states that the language does not distinguish resources that are realized by agents from those realized by non-agentive objects. These limitations can be addressed by specializing the structure element concept to explicit agentive elements and non-agentive elements. The language is also originally unable to address the cases in which resources are passive non-agentive elements, which are objects of interest or raw material (e.g., 'Money', gold, diamond, gas), which corresponds to problem R5. This can be addressed by specializing the structure element concept to include these

objects of interest and raw material. Both these proposals, however, are out of the scope of this work, since they deal with the ArchiMate core. The ArchiMate core language or an extension that deals with the core elements of ArchiMate should address these limitations.

8.3. The Behavior Element Concept

The *behavior element realizes capability* relation means that elements that are capable of performing the behavior element have the capabilities that the behavior element is said to realize. Another reasoning is that some behavior elements might be seen as the simple ability to perform the specified behavior, as a business process might reflect the ability to organize elements in a specified manner in order to produce a desired outcome.

The *capability is used by behavior element* relation specifies that a *capability* of that type is required to the successful execution of the *behavior element*. In other words, for a behavior element to be performed, one (or more) specific capability (or capabilities) is required. This allows the organization to structure its necessities in terms of capabilities and then to prescribe the necessary capabilities that its resources should have in order to be able to perform the behavior element. Further, this scenario would allow the organization to match the structure elements it has to the required capabilities for the behavior element it needs to perform, thus improving organizational planning. The planning improves the organizational management of its current and future structure elements. It also helps the organization understand the elements it needs to acquire according to its plan. This enables the organization to work aligned to the methods suggested in resource-centric theories.

The *behavior element access a capability* relation specifies that a *capability* of that type is acquired with the *behavior element* performance. The element that acquires the *capability* is the one that is *assigned to* the capability. It means that the object assigned to the capability, represented by the structure element or the structure element that is playing the resource role, acquires a disposition of the type represented by the capability by performing the behavior element. This relationship allows the organization to specify which capabilities its resources and structure elements acquire and how they acquire that (e.g., through training). This relationship thus solves problem C2 (and as a consequence C2.1) in tandem with the following relations: 'resource *assigned to* capability' (section 8.1), 'structure element *assigned to* capability' (section 8.2) and 'behavior element *realizes* capability relation'.

8.4. The Capability Concept

The capability concept is a central concept in this proposal. The capability concept represents the power to bring about a desired outcome. This power should be understood in broad sense, as, for example, a mug has the power of constraining coffee, which is the desired outcome. Capabilities can be used to state a broad range of behaviors, ranging from simple ones as mug's behavior, to complex behaviors, as the 'damage assessment expertise' (shown on Figure 2) that can be assumed to inhere in an organization as well as in a specific person.

Based on the UFO concept of disposition, this interpretation allows the organization to model the capabilities it can "socially perform". This is applicable for a variety of cases that uses delegation, such as, for example, cases in which an organization might hire a different company to bring about some desired outcome and needs to state that the original (hiring) organization has the *capability* of bringing about that desired outcome (in this case because it has the *capability* of delegating it (R. Guizzardi and Guizzardi 2011), (Bringuente, Falbo, and Guizzardi 2010)). As stated in Section 6, "if *A* has a meta-commitment from *B* to execute *S* then *A* (socially) can do *S*. An *object* can have *dispositions* [capabilities] which arise from its parts (or from the network of its delegation relations)".

In the proposed metamodel, the Capability concept is related to the Resource, Structure Element and Behavior Element concepts, and with itself. The relations with the other

elements have been explained in Sections 8.1, 8.2 and 8.3, respectively. We now focus on the *capability realizes capability* relation. Let us assume that C_2 realizes C_1. This relation means that C_2 enables C_1, in the sense that C_2 partially realizes or is required for one to have capability C_1. As a capability can be enabled by multiple ways and might need different ('lower-level') capabilities to be enabled, the need for optional possibilities arises. The capability-enabling bundle concept is used to address this need.

8.5. The Capability-enabling Bundle Concept

The capability-enabling bundle concept is used to represent the diverse approaches that can be used to enable a desired capability. The bundle is used to represent optional enabling approaches to a specific capability. This concept solves the lack of expressivity stated in problem R6, as well as the lack of expressivity stated in problem C5. When no bundle is used, we assume a conjunction, i.e., that all elements are required. When more than one bundle that explicitly realize the same capability are modeled, each bundle is considered as an alternative. Figure 6 presents the metamodel fragment that shows the Capability-enabling Bundle concept and its relations to the other elements of the metamodel.
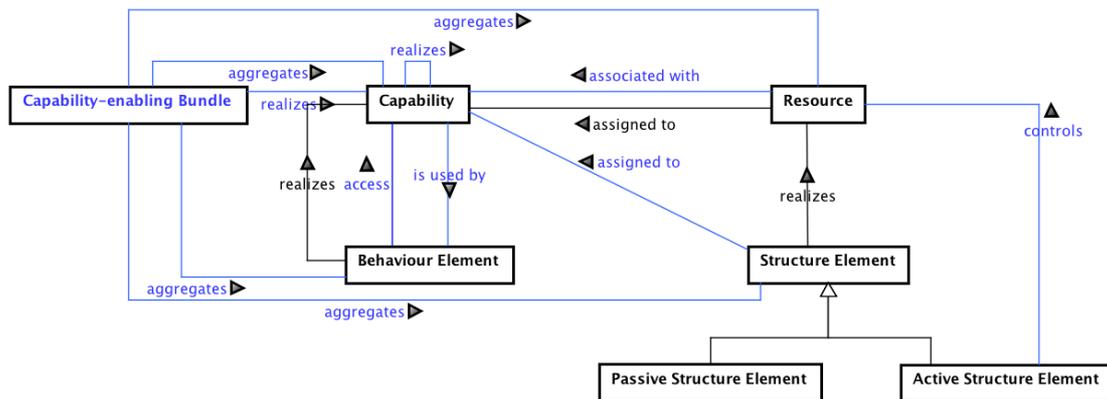

Figure 6 – Fragment of the proposed metamodel for the Capability-enabling Bundle

Figure 7 presents an example of use of the capability and the capability-enabling bundle. In the example, 'Archisurance' is assigned to an 'Insuring capability'. The 'Insuring capability' is realized by the 'Selling Capability', 'the Policy Administration process', the 'Collect Premium Business Function' and the 'Claim Handling Capability', in the sense that they, altogether, are required to enable the 'Insuring Capability'. To model alternative means of enabling a capability, the modeler should use a Capability-enabling Bundle like the 'Claim Handling Capability', which is realized by 'Capability Bundle 1' or by 'Capability Bundle 2'. In Figure 7, the 'Capability Bundle 2' option uses the 'Damage Assessment Expert', and relies on his capabilities to bring about the desired outcome. In contrast, the 'Capability Bundle 1' relies on its 'outsourcing capability' to outsource the required outcome to a third-party.
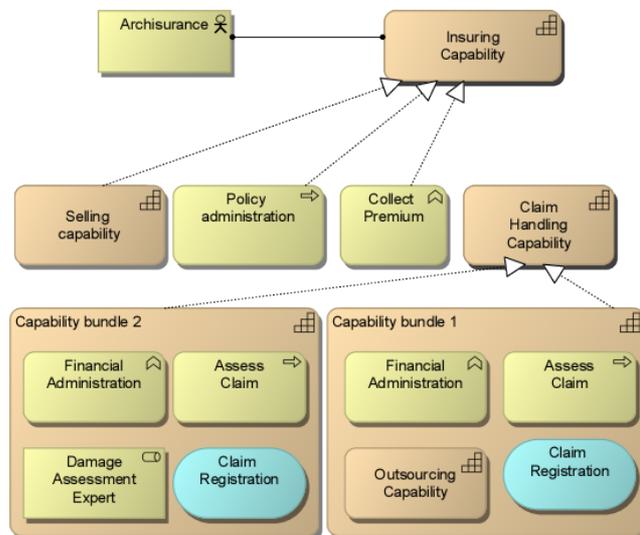
Figure 7 - Capability and Capability-enabling bundle examples

Problem C3 states that if no *resource* is represented, the object manifesting (or acquiring) the *capability* is unknown at the '*capability realized by behavior element*' and '*resource assigned to capability*' relations. The proposed metamodel relates *resources* to *capabilities* using the *associated with* relation for the purpose of indicating the resource that participates on the performance of the capability.

Problem C4 states that the language is not expressive to state the number/amount of *resources* related to the *capabilities* when the *resource* represents a *universal* (a type). For example, it is not possible to know how many instances of *resources* are to manifest their *capabilities* to perform a *behavior element*. In our proposal, resources should be related to behavior elements through capabilities. However, the solution to problem C4 is similar to the solution to problem R4, described in Section 8.1, so that the limitation can be addressed with the addition of a replication attribute to the language, such as a cardinality constraint,. Nevertheless, this is out of the scope of this work since it requires modifications to the ArchiMate core.

9. APPLYING CAPABILITY AND RESOURCE MODELING FOR STRATEGIC MANAGEMENT

The extension presented in this paper introduces modeling constructs to represent capabilities and resources for high-level strategic management, such as described in (Mintzberg 1994), (McKelvie and Davidsson 2009), (Ray, Barney, and Muhanna 2004), (Baines et al. 2009), (Barney 1991), (Grant 1996) and (Penrose 1959), among others. According to (Mintzberg 1994), most companies pursue a strategy informally termed as 'umbrella strategy', in which the general strategic guidelines are initially deliberated, and the details are left to be deliberated later on in the process.

In this setting, this extension aims to allow organizations to model their core capabilities and resources, both to understand their current status or to improve the organization, according to a capability-based planning approach. The enterprise then focuses on maintaining, improving and creating the required capabilities.

A further key characteristic for the extension is that it enables capabilities and resources to serve as abstractions for (more detailed) business process and structure elements. This allows us to define models that should be more stable and require less effort to maintain than models based on those more detailed elements. This allows the enterprise to plan on providing the required capabilities and resources in order to achieve a desired state without actually having to pursue a complete and extended view on the business processes and tasks that are meant to realize that state. The organization may selectively assign resources to the performance of the

capabilities and/or design new business processes in order to achieve the modeled desired situation.

In the remaining of this section, we discuss two real-world cases modeled using our proposal. The first case has been taken from the strategic management literature (Sako 2004) and concerns capability improvement in the automotive industry. This case shows the applicability of the capability concept and its relations with other elements of the enterprise architecture. The case is instructive in that it shows that the capability concept directly captures key aspects of Toyota's "organizational capability enhancement" approach as applied in one its suppliers (JECO). The second case concerns portfolio management for an energy supplier company. This case models a necessary reconfiguration of resources in which the enterprise is required to maintain the same capabilities while consolidating its IT systems. This case has been previously used in the original Business Strategy and Valuation Concepts extension proposal (Iacob, Quartel, and Jonkers 2012). Using this case, we are able to compare our proposal to the original one and present the improvements that the well-founded proposal provides to the language.

9.1. Toyota's JECO Supplier Use Case – Organizational Capability Enhancement[4]

This use case considers supplier development, which is a procedure undertaken by a company to help improve its supplier's capabilities. This specific case of supplier development can be understood as an organization's attempt to transfer (or replicate) some aspects of its in-house organizational capability across its boundaries. The organizational capabilities being replicated at suppliers include not only well-specified technical routines, "but also the relatively constant dispositions and strategic heuristics that shape the approach of a firm to the non-routine problems it faces" (Nelson and Winter 1982). As stated in (Sako 2004), "the ability to replicate such capability, is, in itself, also a capability".

We focus on the capability improvement efforts of Toyota in one of its suppliers, the JECO instrumentation supplier. Specifically, we take the diffusion of the Toyota Production System at JECO. For building up the case and modeling it, we refer to the work published by (Sako 2004), in which three major Japanese automotive industries were analyzed. To perform this analysis the authors had access to historical and contemporary documents provided by the companies and also conducted interviews with key respondents (purchasing managers and supplier development engineers) in the organization and at some of its suppliers.

The Toyota Production System (TPS) has its roots in the Toyota factories of the 1960's. The Operations Management Consulting Division (OMCD) was established as part of Toyota's Production Control Function, to facilitate a seamless transfer of knowledge between Toyota and its suppliers. OMCD is in charge of implementing TPS both within Toyota factories and its core suppliers, and it intends to guarantee that the same methods, procedures and heuristics are applied to both internal and external factories.

Within Toyota, *kojo jishuken* (factory autonomous study groups) take place as a culmination of education and training for Toyota's middle managers and first-line supervisors. They are considered the most important repository of know-how on the shop floor. Supervisors are given an incentive to make continuous improvement with concrete results, and are required to regularly present improvement ideas. The transfer of knowledge and reproduction of Toyota's capabilities using *jishuken* groups (self study autonomous groups) in its suppliers did not become public until the 90's (Sako 2004).
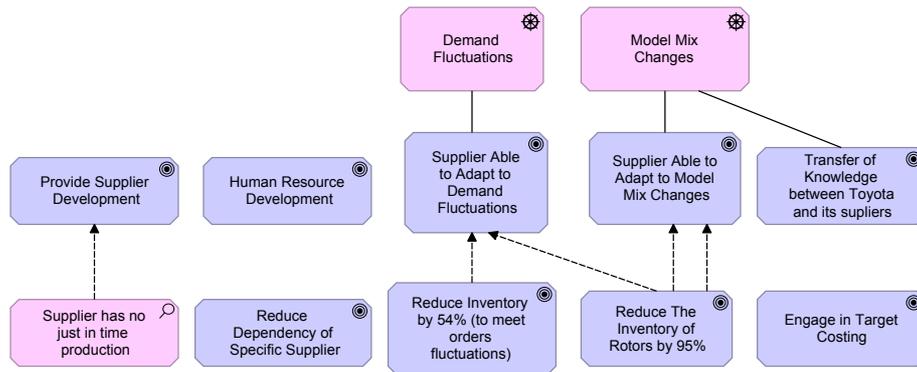
Figure 8 shows Toyota's intentions, represented by its goals, drivers and assessments, represented using ArchiMate's motivation extension notation. In our case, Toyota has the goals (i) 'Provide Supplier Development'; (ii) 'Human Resource Development'; (iii)

---

[4] This case and its description is taken and adapted from (Sako 2004).

'Supplier Able to Adapt to Demand Fluctuations'; (iv) 'Supplier Able to Adapt to Model Mix Changes'; (v) 'Transfer of Knowledge between Toyota and its Suppliers'; (vi) 'Reduce Dependency of Specific Supplier'; (vii) 'Engage in Target Costing'; and specific goals for the application of TPS in JECO, (viii) 'Reduce Inventory by 54% (to meet orders fluctuations)' and (ix) 'Reduce The Inventory of Rotors by 95%'. Toyota had some drivers for some of its goals, like the 'Demand Fluctuations' and the 'Model Mix Changes' drivers.

The performance of one goal might influence the other goals. In our example, goals viii and ix, influences goals i, iii and/or iv. An important assessment that has been made by Toyota was that 'Supplier has no just in time production'. The crisis in the automotive sector, in which many of Toyota's suppliers were experiencing difficulties adapting to demand fluctuations, enforced this concern.

Figure 8 - Toyota's Intentions

Figure 9 shows an ArchiMate model that represents an overview of Toyota's capability improvement efforts for JECO, including Toyota's capabilities, JECO's capabilities, the capabilities of the so-called autonomous study groups ('*Jishuken* study group'), and the key resources.

The '*Jishuken* study groups' were designed to help suppliers improve their shop floors by refining the application of TPS. The '*Jishuken* Study group' is required to choose a specific theme amongst the ones discussed by the OMCD, and to identify a specific factory area to be studied by the group. JECO study group chose the output fluctuations in the age of low demand ('Demand Fluctuations') and the 'Model Mix Changes'. The specific factory area chosen for the study was 'JECO Parts and Goods logistics'. Toyota has considered mainly geographical location and the absence of direct competitors when forming these groups, to help interaction and sharing of know-how during the '*jishuken* activities'.

The study section begins by setting concrete performance targets in terms of shop floor indicators, which should be achieved with the help of the '*Jishuken* activities'. Typical indicators to be considered are productivity, cost reduction and/or inventory turns. JECO's targets were to 'Reduce Inventory by 54%' and to 'reduce the inventory of rotors by 95%'. This is represented in the model by the dotted lines with a white arrow on the end between the activity and the goal it realizes.

From the Toyota side, trained experts ('Supplier Development Employees') help a *jishuken* group to achieve its results. A 'Senior OMCD engineer' with 'observation and analysis capability' is in charge of the '*jishuken* study group' and visits the company under study during the '*jishuken* Activity' period, in which he is responsible for making critical observations. 'Junior Supplier Development Engineers' also visit the company, more often though, to give more detailed guidance.
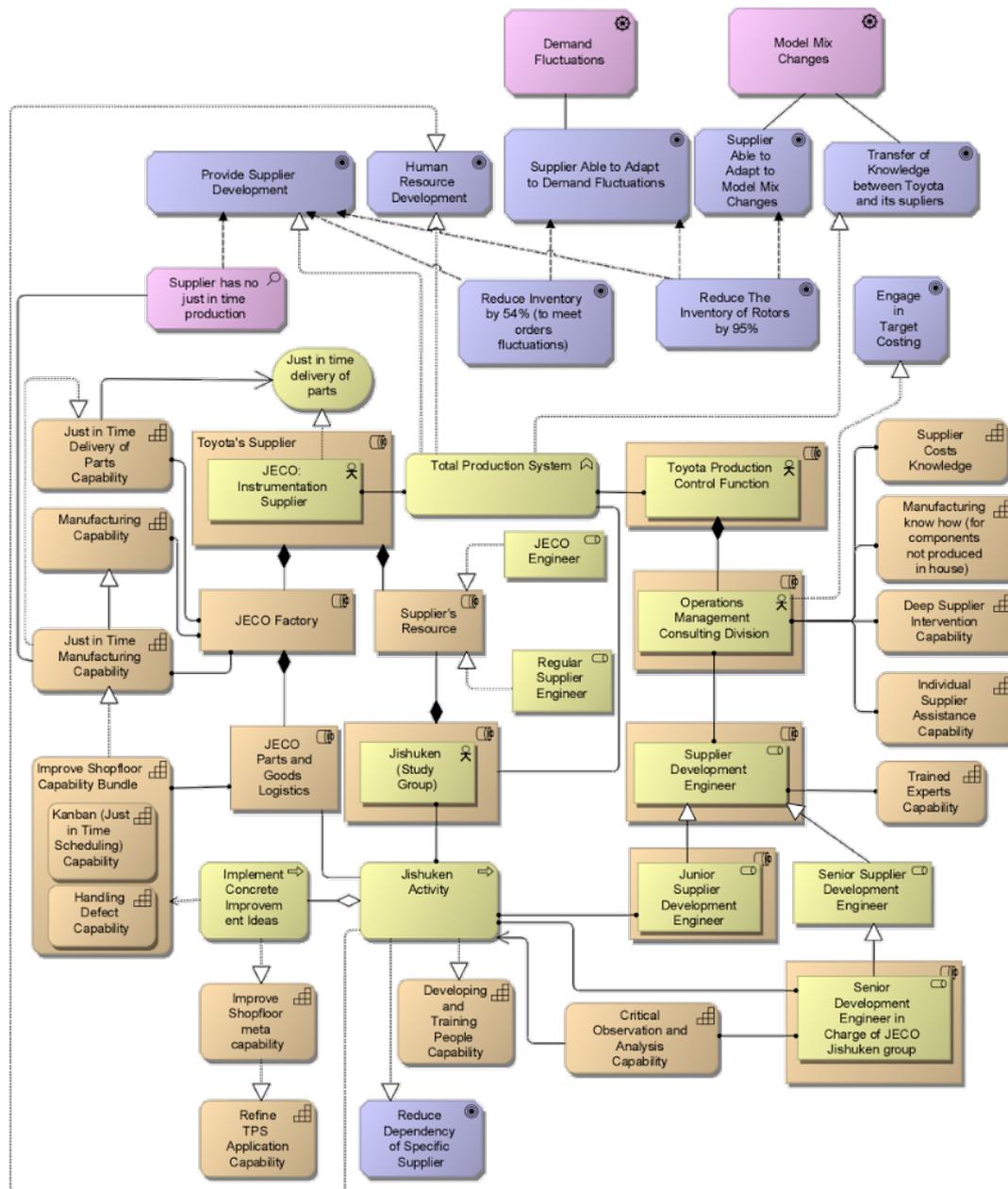
Figure 9 - Toyota's Supplier Capability Enhancement Case

Jishuken groups are responsible to put forward and 'Implement Concrete Improvement Ideas'. A typical *Jishuken* group would consist of engineers from various suppliers, not only for the company under study, since this facilitates knowledge sharing, which is important for the 'Developing and Training People Capability' realized by the '*Jishuken* Activity'. Most of the ideas are implemented in the study-host company. For example, JECO has implemented 222 of the 248 improvement ideas, an implementation rate of 90%. Many of the ideas implemented by JECO concerned the improvement of its '*Kanban* (Just in Time Scheduling) Capability' and its 'Handling Defects Capability', with the clarification of rules about when defects are discovered. These capabilities helped JECO accomplish the 'Just In Time Manufacturing Capability', a specialization of its 'Manufacturing Capability', which realizes its 'Just in Time Delivery of Parts Capability', used in its 'Just In Time Delivery of Parts' service. The 'Manufacturing Capability' by itself would allow the enterprise to realize the 'Just In Time Delivery of parts' *service*, since the organization might have all the required parts in its inventory. However, the 'Just In Time Manufacturing' capability has been enable by the 'Kanban (Just in time Scheduling) Capability' and the 'Handling Defect Capability'.

This helped the enterprise to diminish its inventory of rotors and its regular inventory, realizing the 'Just in Time Manufacturing Capability' and, thus, addressing Toyota's assessment that '*Supplier has no just in time production*', and realizing JECO's goals of 'reducing inventory by 54% (to meet order fluctuations)' and 'Reduce The Inventory of Rotors by 95%'. These goal realizations, in their turn, influence the realization of Toyota's goals of 'Provide Supplier Development', 'Supplier Able to Adapt to Model Mix Changes' and 'Supplier Able to Adapt to Demand Fluctuations'.

Besides *Jishuken*, Toyota's OMCD also provides individual assistance to suppliers on an if-and-when-necessary basis. For that, OMCD have the capability of 'Deep Supplier Intervention' and 'Individual Supplier Assistance'. Individual assistance is suitable whenever Toyota is looking for quick results. However, short-term deep intervention is said to come to a halt when the experts go home, since the suppliers do not understand why things are being performed as intervened and short-term deep intervention does not 'provide supplier development' or 'human resource development'. A Toyota OMCD expert understands that the "[i]t would most certain be quicker for an expert to take a lead and provide answers, but this would not result in developing the skills of those who are led. The strength of Toyota Production System lies in creating as many people who can implement and put into practice TPS on their own as possible. So the most important thing for the survival of TPS is 'Human Resource Development'."

Jishuken is a gathering of middle-level production technologists from a stable group of companies, who jointly develop better capabilities for applying TPS through mutual criticism and concrete application. *Jishuken* also has the benefit of giving Toyota 'Supplier Costs Knowledge', with an enormous access to detailed cost structures of its main suppliers, a capability that inheres in the OMCD. This contributes to Toyota's core capability of 'Engaging in Target Costing' and to retention of 'Manufacturing know-how for components not produced in-house'.

In order to evidence the role of the capability construct, a model *without* the capability construct is presented in Figure 10. This model is instructive in that it shows that without the capability construct the model is not able to capture key aspects of Toyota's organizational capability enhancement approach. Instead, it focuses on more operational aspects of the same enterprise setting, failing to reveal the capabilities of JECO that are at stake, the key capabilities of Toyota's OMCD, as well as the link between the activities of the study group and the capabilities improved.
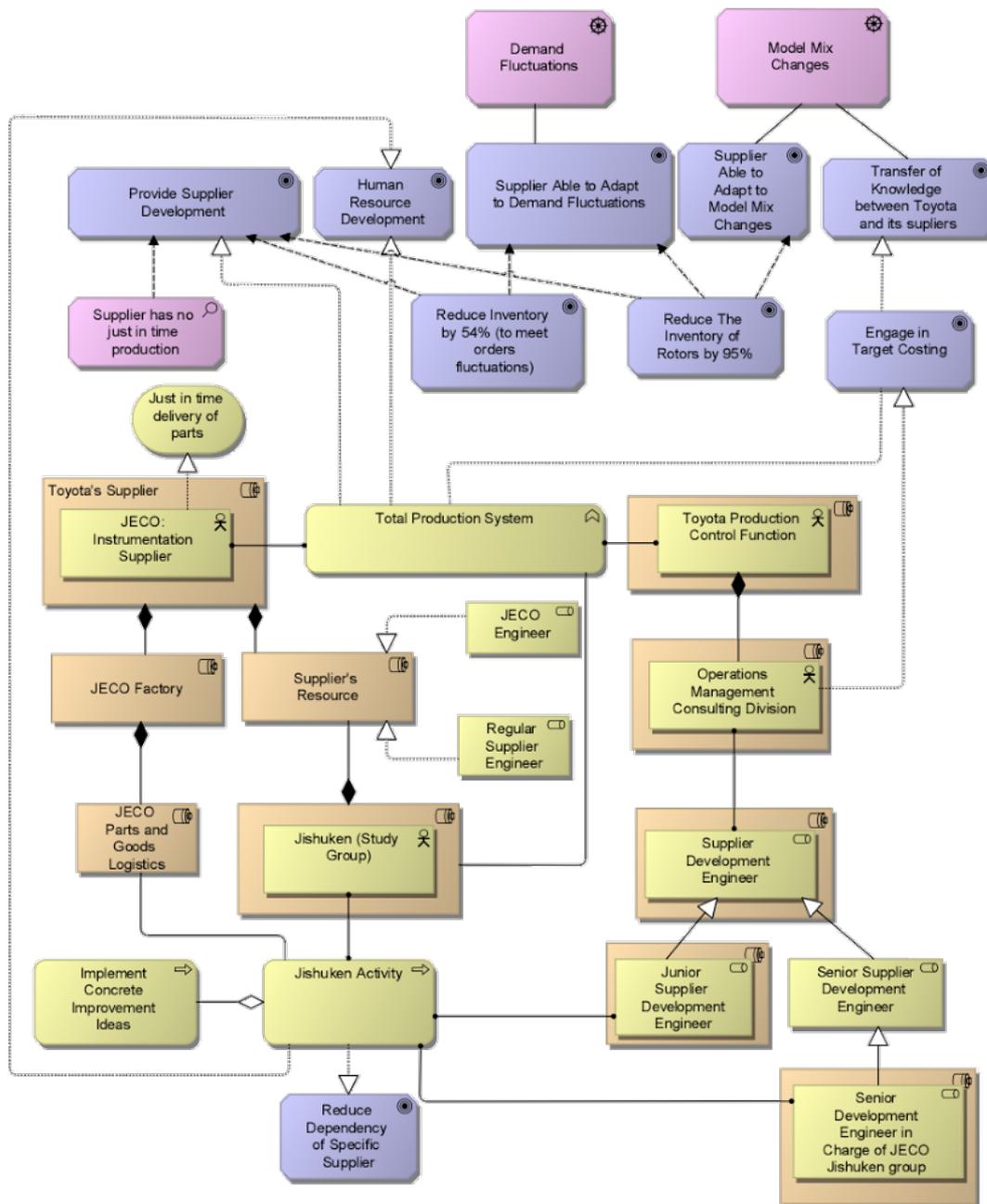
Figure 10 – Toyota's case modeled without the capability concept

9.2. IT Consolidation for an European Energy Supplier Company

The second case we use to illustrate our approach concerns an IT consolidation problem (Iacob, Quartel, and Jonkers 2012). Consolidation of software application portfolios is a typical situation in which portfolio management techniques are applied in *a capability–based planning* setting. The main goals of the consolidation of IT resources are the elimination of functional and data redundancies. Typical situations in which IT consolidation is necessary include the co-existence of different software systems in an organization that offer the same functionality, or the replication and storage of data by several different systems. The positive effect of IT consolidation on cost reduction has long been recognized in the literature, such as, e.g., in (Hancock, Humphrey, and Wilcox 1999)). In particular, we refer to (Franke et al. 2010) that used integer binary programming to solve this IT consolidation problem, while minimizing consolidation and maintenance costs. In this particular case, the organization

seeks IT consolidation to gain efficiency without affecting its business capabilities. Expressing business capabilities is thus required in order to make their coherent management possible and plan in a way that matches the organization's business model and strategy. The strategy in this case involves gaining efficiency through resource optimization, while the model requires the maintenance of the current capabilities.

The goal of this case is twofold:

- Firstly we demonstrate the benefits of the proposed improved metamodel fragment concerning the quality of models compliant to it. We do this by showing how some of the problems concerning the original metamodel are handled in a concrete setting with the revised construct and relations.
- Secondly, we demonstrate its usefulness in making the relation between business strategy and enterprise architecture explicit, allowing one to reason about it. We show how capability-based planning bridges the gap between strategy definition and strategy implementation.

The capability-based planning process we follow throughout the case consists of the following steps:

1. Identifying drivers & problem(s), business strategy and strategic capabilities;
2. Analyzing the baseline capability gaps and creating baseline heat capability map;
3. Relating capability gaps to the baseline architecture;
4. Using resource portfolio management approaches to create a target architecture and target capability map, and;
5. Implementing changes.

We consider and extend the case that was reported in (Iacob, Quartel, and Jonkers 2012), namely of a large European Energy Supplier (EES) that consists of three different units (formerly three different companies that have merged). As a result of the energy market liberalization, customers may now switch between energy suppliers, and thus this energy supplier must be able to ensure a fast and reliable switching process for new and leaving customers. A consequence of the previous merger is that currently the company has seven different systems that all take care of some part of customer switching process for three business units, and provide overlapping functionality. Information about customers, contracts and their consumptions is scattered over different systems and databases. The company aims at consolidating the system architecture, and at eliminating redundant functionality with minimal costs. We present the results of each step of the capability-based planning process in the sequel, employing ArchiMate models, some of which include the capability and resource constructs. We omit discussing step 5 (implementing the changes) as this is outside the scope of this paper.

9.2.1.Identifying drivers & problem(s), business strategy and strategic capabilities

In this case, the main driver for action was the energy market liberalization. An assessment of EES's current situation in the light of this driver is given in the Figure 11 using ArchiMate's motivation extension. It relates the *driver* ('Energy market liberalisation') and various *assessments* ('High IT maintenance costs', 'Inefficient switching process', etc.).
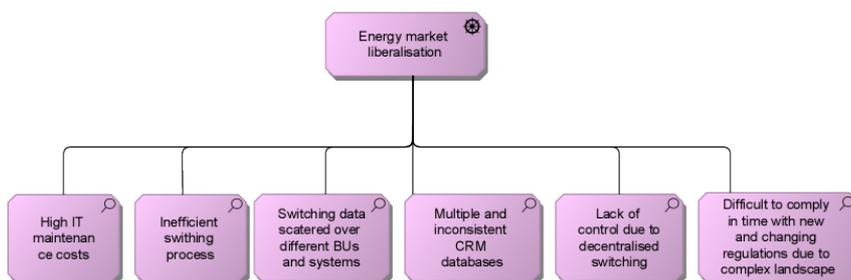

Figure 11 - Assessment and driver

The assessments are related to many of the goals that fall under the overarching "Achieving operational excellence" business strategy (Figure 12).
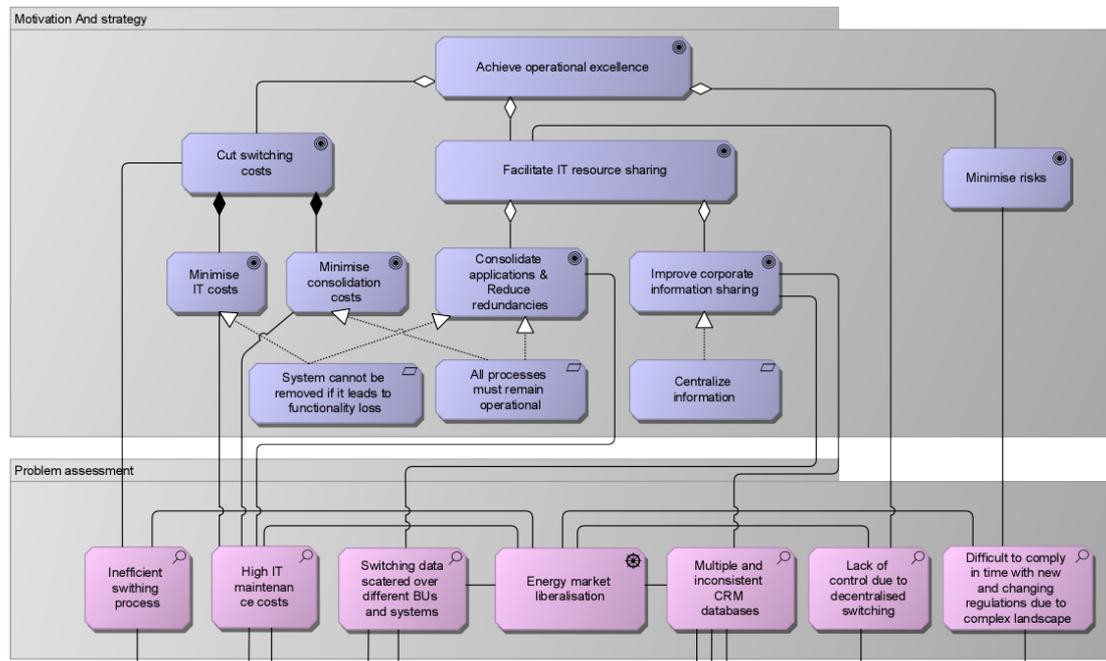

Figure 12 – Relating assessments and strategy

9.2.2. Analyzing baseline capability gaps and creating baseline heat capability map

Figure 13 shows the relation between the assessments and the capabilities whose current realization leads to the challenges identified in the assessments. The identified capabilities are 'Customer order management', 'Customer data management', 'Customer relation management' and 'Risk management'. The lower part of Figure 13 shows a number of operational capabilities that are related to the strategic capabilities. Figure 12 and Figure 13 are actually the upper and lower part of a single diagram, enabling us to trace from strategic elements to capabilities.
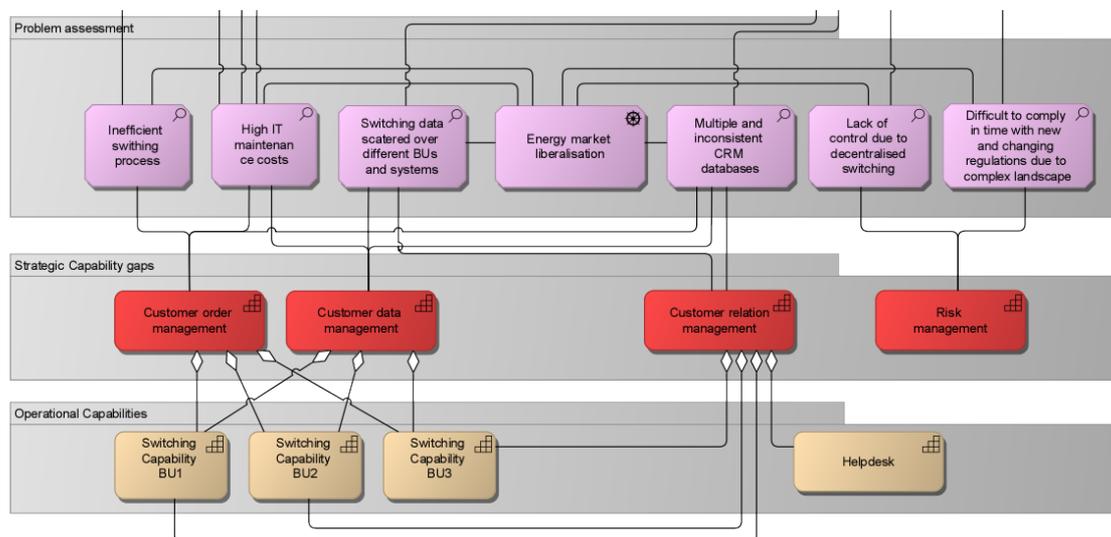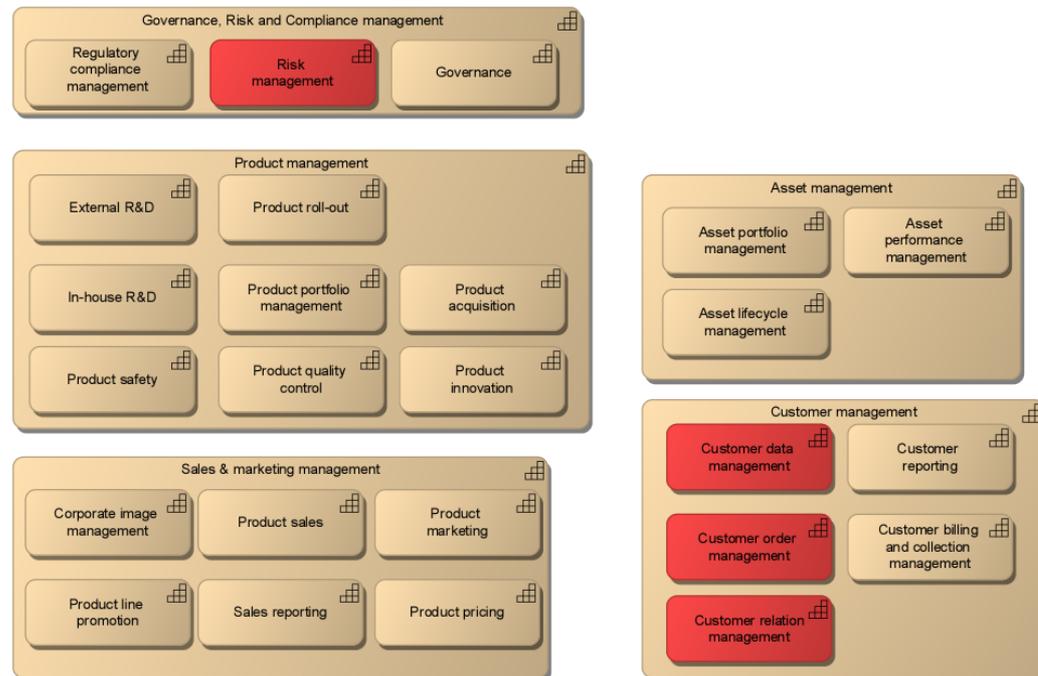

Figure 13 – Strategic input for capability-based planning: Assessment of capability gaps

The heat map shown in Figure 14 shows the identified strategic capabilities in context. Strategic capabilities depicted in red are those whose realization needs to be improved in the context of the addressed problem.


Figure 14 – Heat-map of strategic capabilities associated with the "Achieving operational excellence" strategy in the context of the current case

9.2.3. Relating capability gaps to the baseline architecture

Having identified the strategic capabilities gaps, the next step in the approach concerns relating the capability gaps to the baseline architecture, in order to find paths for possible improvement in capability implementation.

The baseline architecture that focuses on the switching process is shown in Figure 15. It includes seven different application systems that all take care of switching for three business units, and thus provide overlapping functionality.
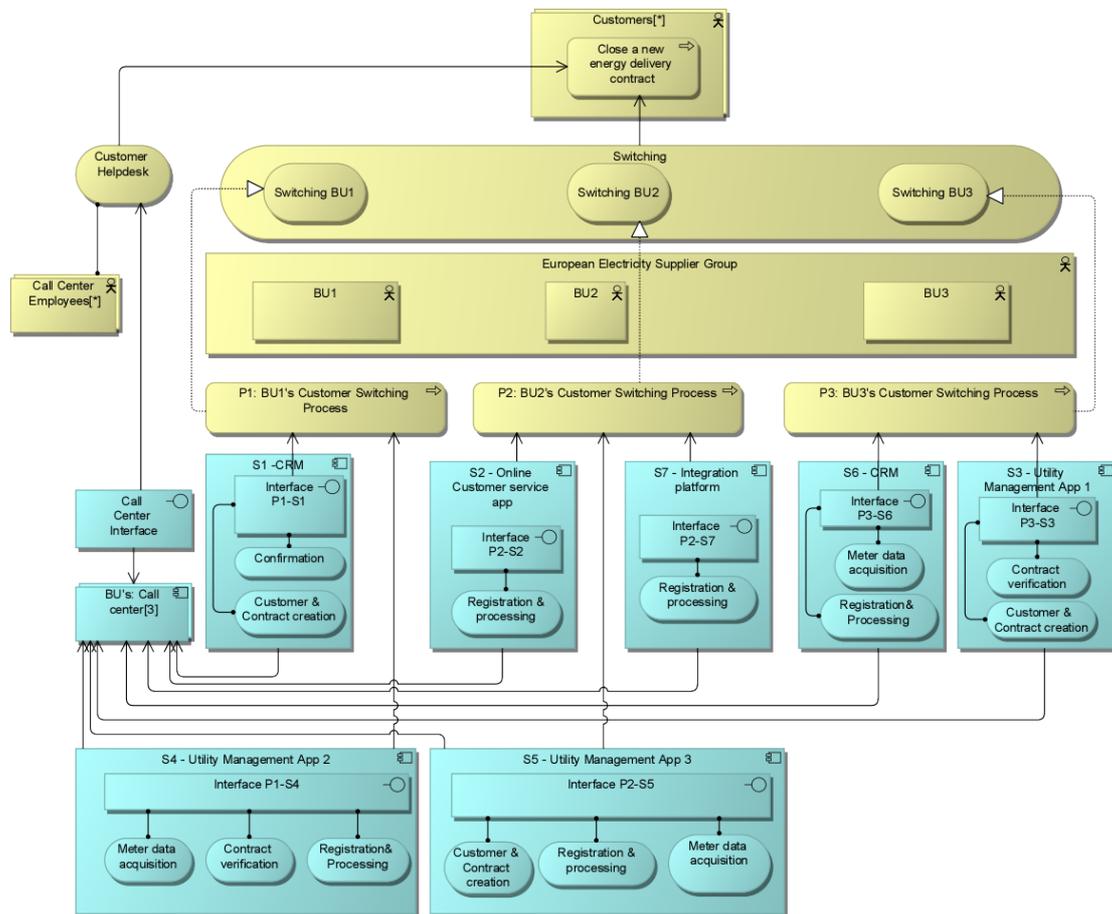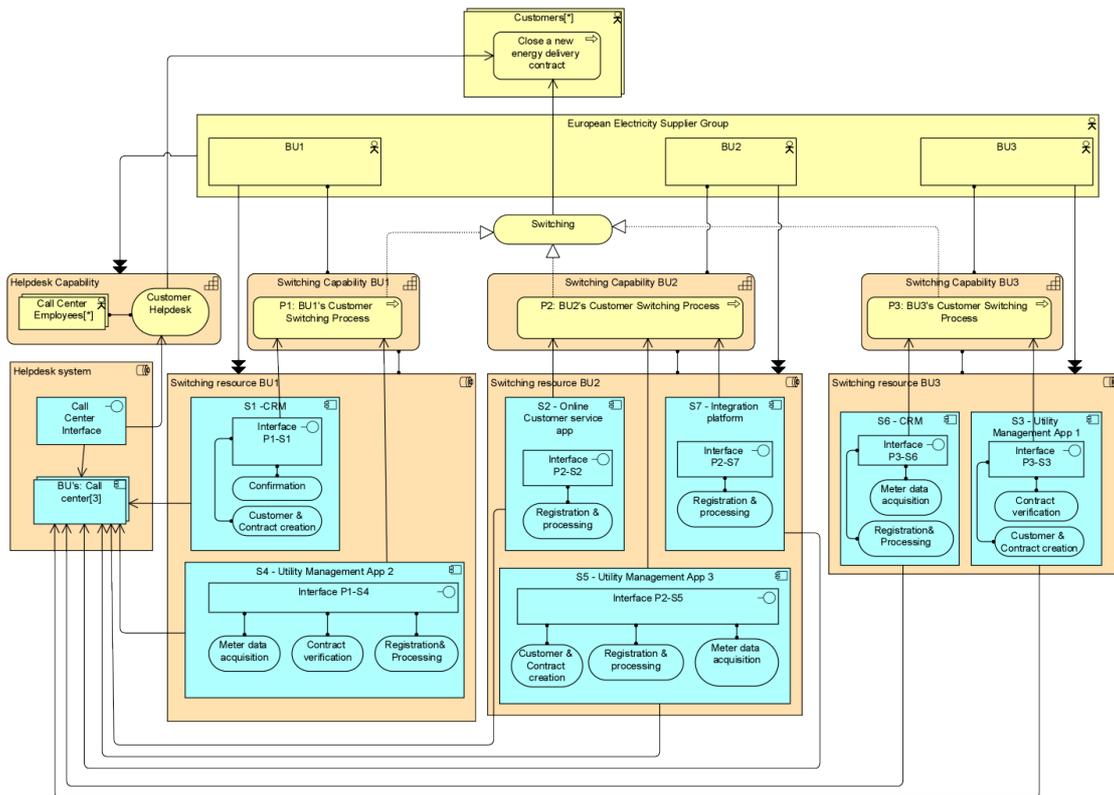
Figure 15 - Baseline architecture

The main goal of this step is to establish a relation between the previously identified capability gaps and those elements/parts of the baseline architecture that should be changed to resolve the capability gaps, and subsequently to solve the original problem. Nevertheless, examining the heat-map (Figure 14) and the baseline architecture (Figure 15) one can see that the distance between the two models in terms of abstraction level is large, making it unfeasible to establish a precise relationship between them as far as the traceability of capability gaps in the enterprise architecture is concerned.

In order to address this issue, we add an intermediary abstraction level between the baseline architecture and the heat map. This additional level focuses on the specification and refinement of capabilities and resources, so that traceability all the way down to the operational aspects of the baseline architecture becomes possible. In this process we make use of the links between the strategic capabilities and the operational capabilities shown in the bottom part of Figure 13. This refinement step is important because it allows operational capabilities to be easily linked to architecture fragments. Thus, Figure 16 depicts the configuration of resources and capabilities that are involved in the current decision making problem, and represents their usage context in a simple way.
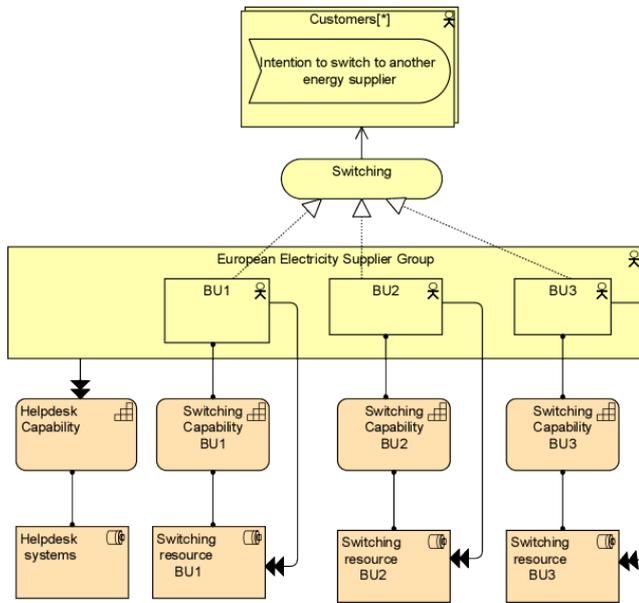
Figure 16. Resources & Capabilities configuration

In Figure 17 we show how the previously identified configuration of resources and capabilities is mapped onto the architecture by focusing on the IT components these capabilities require.

Figure 17 -Baseline architecture with capabilities and resources

9.2.4. Using resource portfolio management approaches and creating a target architecture and target capability map

As explained in (Iacob, Quartel, and Jonkers 2012), the model shown in Figure 17 can be used as basis for the application (in a model-based fashion) of integer binary programming (Franke et al. 2010) to enterprise architecture once it has been enhanced with the concepts covered by the proposed metamodel. The resource portfolio optimization problem to be solved can be shortly formulated as follows: the objective function is to minimize the switching costs, which are defined as the sum of maintenance and consolidation costs, subject to two constraints: (1) all processes must remain functional at all times, meaning that the BUs' switching processes must be all fully supported by the systems that will be selected in the future situation, and (2) no functionality loss is acceptable, meaning that the selected systems must provide all the functionality currently provided by all of them. Details on how maintenance and consolidation costs are calculated can be found in (Franke et al. 2010), (Iacob, Quartel, and Jonkers 2012).

The optimal solution provided by the solver for the cost minimization problem leads to the target switching architecture shown in Figure 18, which includes resources, capabilities and relations defined according to our proposed metamodel. Systems S2, S4, S5 and S7 have been removed from the target architecture, and the remaining systems have to be enhanced with new process interfaces (colored in red in Figure 18) to enable the execution of all BUs' switching processes in the future situation.
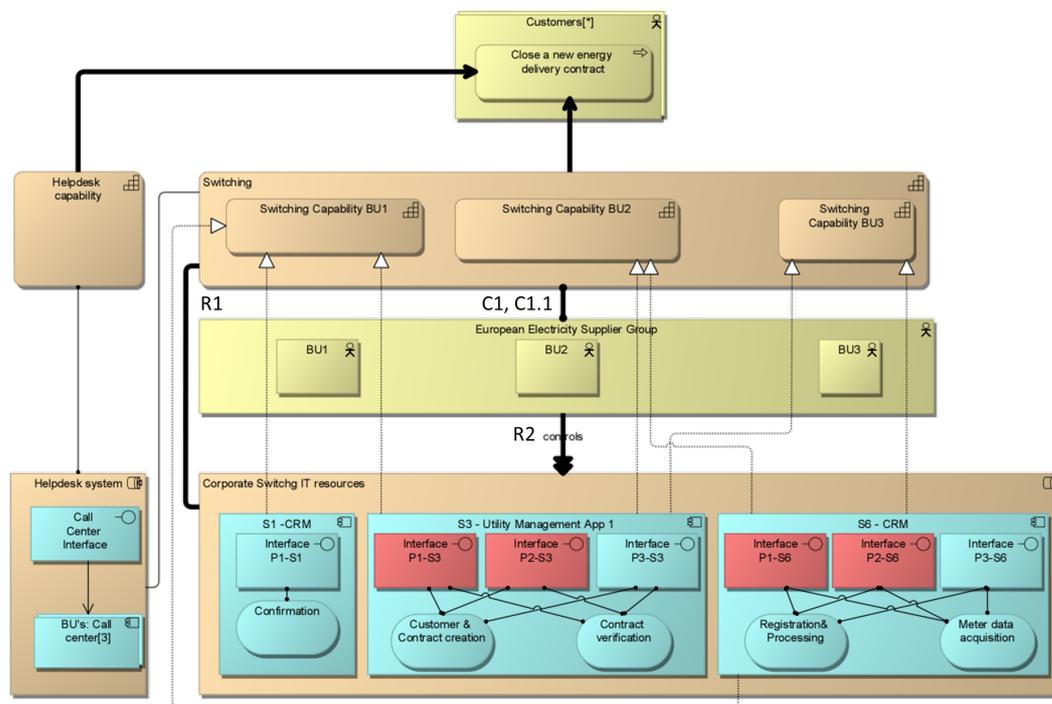

Figure 18 – Target consolidated architecture

9.2.5. Concluding remarks

The case demonstrates the usefulness of the resource and capability concepts to support a capability-based planning process, which is typically carried out at a strategic/tactical level. For such endeavors one should be able to abstract from the details of architecture models but still being able to trace planning decisions concerning resources and capabilities in these architecture models in order to correctly assess the impact they have on the different layers of the architecture. In this particular case we have shown how the problem of multiple, partly overlapping, and hardly maintainable distributed IT resources (Figure 17) can be solved by consolidating them into a single integrated resource (Figure 18), which can be controlled at a corporate level.

Figure 18 shows some relations that could not be represented with the original metamodel, which are drawn with thick lines. These relations facilitate the correct modeling and interpretation of different dependencies between resources, capabilities and architecture elements. In particular, the model shows the relation between the consolidated resources and the controlling structural element, the relation between the resource and its context of usage, and finally, the relation between the capabilities and the (customer-facing) processes that employ the capabilities. Together with the other models discussed here, traceability from business strategy, business capabilities, operational capabilities to (IT) resources has been made possible.

10. **RELATED WORK**

Some related approaches have addressed the use of capabilities and resources in enterprise architecture and enterprise modeling.

The REA accounting model (McCarthy 1982), e.g., has included a notion of resource since its inception. In REA, Resources are exchanged through economic Events in which Agents participate, hence the REA acronym. Since REA was conceived as a conceptual framework to support accounting practices, its main focus is on the notion of economic resource as a transactable entity of value. It considers, among other aspects, the "flow" of resources and the accountability for the "custody" of resources (in the "stock" or "inventory"). Similarly to our approach, the authors of REA have been concerned with the ontological foundations of their framework, which has been reported in (G. L. Geerts and McCarthy 2002) and (G. Geerts and McCarthy 2000). Differently from our work, REA does not aim at considering the properties or capabilities of resources, and it does not account for how resources and their capabilities may be employed in the organization in order to achieve higher-level capabilities.

The notion of resource is also adopted in the ARIS framework (Scheer 2000), in order to account for resource bottlenecks and resource availability in the scope of business process management. ARIS captures the relation between resources and organizational entities that controls them, as well as "input" and "output" relations between resources and events ("functions" in ARIS). The ARIS framework does not address the notion of capability.

Dryer et al. (2007) discuss the extension of DoDAF with a Capability Evaluation Model that include the concept of capability. In that work, capabilities are said to be provided by "systems of systems", which comprehend any combination of "doctrine, organization, training, materiel, leadership and education, personnel and facilities (DOTMLPF)". Some of these elements subsumed under DOTMLPF are incorporated into the DoDAF's capability viewpoints, DoDAF's DM2 model and grounded in the IDEAS foundational ontology, which is used as a foundation for the whole DM2. Further investigation is required in order to assess the relations between ArchiMate BSCV constructs and DoDAF Capability Viewpoints, and to establish potential semantic interoperability relations for the representation of capabilities and resources. This task can be made more manageable now that the BSCV constructs have been given a precise semantics in terms of a foundational ontology. That should enable comparison and harmonization of both frameworks.

In (Barroero, Motta, and Pignatelli 2010), TOGAF has been extended to support the modeling of the capabilities a Business Component (BC) can perform. A BC is a business unit that encompasses a set of activities, supported by assets including people, processes and technology. The approach uses capabilities as "an idealized conceptual structure that describes what a BC can do to create value for customers".

In recent years, a number of enterprise modeling approaches have been subject to ontology-based analysis. In (Recker et al. 2010), the authors performed an ontological analysis to the Business Process Modeling Notation (BPMN). Nine ontological deficiencies related to modeling when using the BPMN were found. In (Santos Jr et al. 2010), the authors have defined the semantics of the ARIS framework concepts and relationships in terms of UFO.

Problems regarding the ARIS Method were exposed, and possible solutions to these problems were proposed. (Azevedo et al. 2011) performed an ontological analysis of the ArchiMate motivation extension proposal, unveiling problems and proposing improvement recommendations. To the best of our knowledge there are no comparable analyses addressing the modeling of capabilities and resources.

11. **CONCLUSIONS AND FUTURE WORK**

In this paper, we have discussed an ontological analysis of the BSVC ArchiMate extension and the associated notions of capability, resources and competences. We have employed a comprehensive foundational ontology that incorporates concepts to deal with objects, relations, roles, events, dispositions, as well as social and intentional concepts. Our main aim has been to clarify the semantics of the proposed modeling constructs, which should contribute to the application of the language in practice as a communication tool for stakeholders involved in decision making. We have revised the original language metamodel, in order to accommodate the proposed recommendations.

We have been able to clarify that the resource element represents a type-level entity, capturing the role of an (agentive or non-agentive) object in a particular context of usage. Our well-founded recommendations should lead to a language that allows improved resource planning, as it allows the specification of the context of usage of resources and the ability to explicitly state mandatory and optional resources when related to capabilities.

By considering capabilities as a type of dispositions, we have been able to account for what it means for a behavior element to realize a capability. Our recommendations should lead to a language that allows improved resource and capability oversight, as it allows one to relate capabilities and structure elements that possess capabilities, as well as relate resources with capabilities.

By also considering that competences should be interpreted as types of capabilities, we have been able to identify a case of construct redundancy. We have traced the root of the problem to a lack of relations to express the capabilities of resources. Our recommendations seem to lead to a more regular and parsimonious solution for the expression of human resources and their capabilities, which was not fully addressed with the introduction of the competence construct. Our extension is intended to model the organization's core capabilities and key resources with a strategic management focus. With our extension, the enterprise can consider the required capabilities and resources to achieve a desired state without actually having to pursue a complete and extended view on the business processes and tasks that are necessary to realize that state. This means that at the strategic management level enterprise architects are able to focus on the proper level of abstraction, avoiding unnecessary commitments with lower level details. Thus, they are able to accommodate future changes in the operational parts of the enterprise architecture and at the same time realize the higher-level capabilities and resources. This creates a loose coupling between higher-level capabilities and other operational enterprise architecture elements, contributing to flexibility and maintainability of the resulting enterprise architecture descriptions. This approach is especially valuable to competitive and changing environments, which requires both planning and ability to adapt.

The contributions in this paper can be assessed from the perspectives of *rigor* and *relevance* (as proposed for design-science research by (Hevner et al. 2004)). From the perspective of *rigor*, we have adopted a well-established ontological analysis methodology (section 3) and clearly defined ontological foundations (section 4). From the perspective of *relevance*, there is ample support from the business literature discussing the key role of capabilities and resources in strategic management (Barney 1991) (Ray, Barney, and Muhanna 2004) (Baines et al. 2009) (Helfat and Winter 2011). Further, capability-based planning has had major interest from research efforts in the literature as well as from the practice in EA, with frameworks such as TOGAF introducing basic notions of capability-based planning and its

role in designing, planning and implementing organizational change (The Open Group 2009). Incorporating capability-based planning in EA is thus relevant for industrial and academic efforts.

In order to further stress the *relevance* of the approach, we have shown that it is able to represent key aspects of two real-world cases, one concerning capability improvement and the other concerning resource consolidation in capability-based strategic management. In the first case study, we show that without the capability construct the EA model is not able to capture key aspects of Toyota's organizational capability enhancement approach. The EA model without the capability construct is focused mostly on operational aspects loosing the link to 'what is relevant' for strategic management. In other words, the model shows operational elements without revealing the key capabilities that enable achieving enterprise's goals. In the second case study, we show the application of the approach in portfolio management for an energy supplier company. More specifically, we show the usefulness of the capability-based models in representing a real scenario that required the reconfiguration of resources when the enterprise faced major changes. In this case, the enterprise was required to maintain the same capabilities while consolidating its IT systems. The model captures capabilities that remained stable even in the face of changes in the operational elements of the architecture. It also captures traceability from business strategy, business capabilities and operational capabilities to (IT) resources. This case study has also served for us to compare our proposal to the original "Business Strategy and Valuation Concepts" extension (Iacob, Quartel, and Jonkers 2012) and present the improvements that the well-founded proposal provided to the language.

In our future efforts, amongst others we intend to investigate the relation between the concepts of resource and value. The latter deserves careful attention from the perspective of semantic definition and ontological analysis, given its subjective nature and usage flexibility.

Similarly to the discussion in (Azevedo et al. 2011) concerning the analysis of the motivation extension to ArchiMate, we do not intend to suggest that the terminology used in this paper should replace the terminology currently used in ArchiMate, and we do not intend to imply that the UFO conceptualization should be exposed directly to users of the standard. The main role of the ontological analysis has been to provide us with a rigorous framework to analyze the BSVC proposal. In this sense, the ontological analysis can be seen as a tool for hypothesis formulation, and the recommendations that we have identified here using ontological analysis should be considered as subject to further examination. For example, considering the pragmatic impact of amendments on the set of standards and its users. We have outlined the recommendations raised by the ontological analysis performed here, and we believe that they can have direct application in the revision of the proposal before it reaches the standardization effort. It is natural that, once the approach is incorporated into standards and into EA tools, new insights will arise from the appropriation of the technique by the industry. These insights should inform an *ex post* evaluation effort.

REFERENCES

Almeida, Joao Paulo A, Giancarlo Guizzardi, and Paulo Sergio Santos Jr. 2009. "Applying and Extending a Semantic Foundation for Role-Related Concepts in Enterprise Modelling." *Enterprise Information Systems* 3: 253–277. http://www.tandfonline.com/doi/abs/10.1080/17517570903046292.

Almeida, João Paulo a., and Giancarlo Guizzardi. 2013. "An Ontological Analysis of the Notion of Community in the RM-ODP Enterprise Language." *Computer Standards & Interfaces* 35 (3) (March): 257–268. doi:10.1016/j.csi.2012.01.007. http://linkinghub.elsevier.com/retrieve/pii/S0920548912000402.

Audia, Pino G, Edwin A Locke, and Ken G Smith. 2000. "The Paradox of Success: An Archival and a Laboratory Study of Strategic Persistence Following Radical Environmental Change." *Academy of Management Journal* 43 (5): 837–853.

Azevedo, Carlos L B, Maria-Eugenia Iacob, João Paulo A Almeida, Marten van Sinderen, Luís Ferreira Pires, and Giancarlo Guizzardi. 2013. "An Ontology-Based Well-Founded Proposal for Modeling Resources and Capabilities in ArchiMate." In *Enterprise Distributed Object Computing Conference (EDOC), 2013 17th IEEE International*, 39–48.

Azevedo, Carlos L. B., Joao Paulo A. Almeida, Marten van Sinderen, Dick A. C. Quartel, and Giancarlo Guizzardi. 2011. "An Ontology-Based Semantics for the Motivation Extension to ArchiMate." *2011 IEEE 15th International Enterprise Distributed Object Computing Conference*. Ieee. doi:10.1109/EDOC.2011.29. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6037557.

Baines, Tim S, Howard W Lightfoot, Ornella Benedettini, and John M Kay. 2009. "The Servitization of Manufacturing: A Review of Literature and Reflection on Future Challenges." *Journal of Manufacturing Technology Management* 20 (5): 547–567. doi:10.1108/17410380910960984. http://www.emeraldinsight.com/10.1108/17410380910960984.

Barney, Jay. 1991. "Firm Resources and Sustained Competitive Advantage." *Journal of Management* 17 (1): 99–120.

Barreto, I. 2009. "Dynamic Capabilities: A Review of Past Research and an Agenda for the Future." *Journal of Management* 36 (1) (December 28): 256–280. doi:10.1177/0149206309350776. http://jom.sagepub.com/cgi/doi/10.1177/0149206309350776.

Barroero, Thiago, Gianmario Motta, and Giovanni Pignatelli. 2010. "Business Capabilities Centric Enterprise Architecture." In *Enterprise Architecture, Integration and Interoperability*, 32–43. Springer.

Bock, Conrad, and James Odell. 1998. "A More Complete Model of Relations and Their Implementation: Roles." *Journal of Object Oriented Programming* 11: 51–54.

Bodart, François, Arvind Patel, Marc Sim, and Ron Weber. 2001. "Should Optional Properties Be Used in Conceptual Modelling? A Theory and Three Empirical Tests." *Information Systems Research* 12 (4): 384–405.

Bringuente, Ana C O, Ricardo A Falbo, and Giancarlo Guizzardi. 2010. "Using a Foundational Ontology for Reengineering a Software Process Ontology."

Cardoso, Evellin Cristine Souza, Paulo Sérgio Santos Jr, João Paulo Andrade Almeida, Renata S S Guizzardi, and Giancarlo Guizzardi. 2010. "Semantic Integration of Goal and Business Process Modeling." In *International Conference on Research and Practical Issues of Enterprise Information Systems (CONFENIS 2010)*.

Danesh, Mohammad Hossein, and Eric Yu. 2014. "Modeling Enterprise Capabilities with I * :" 112–123.

Davies, Islay G, Michael Rosemann, and Peter Green. 2000. "Exploring Proposed Ontological Issues of ARIS with Different Categories of Modellers." In *Proceedings of the Australasian Conference on Information Systems*. Hobart, Australia.

Dryer, David A, Thomas Bock, Maria Broschi, and Timothy D Beach. 2007. "DoDAF Limitations and Enhancements for the Capability Test Methodology." In *Proceedings of the 2007 Spring Simulation Multiconference - Volume 3*, 170–176. SpringSim '07. San Diego, CA, USA: Society for Computer Simulation International. http://dl.acm.org/citation.cfm?id=1404803.1404839.

Eisenhardt, Kathleen M., and Jeffrey A. Martin. 2000. "Dynamic Capabilities: What Are They?" *Strategic Management Journal* 21 (10-11) (October): 1105–1121. doi:10.1002/1097-0266(200010/11)21:10/11<1105::AID-SMJ133>3.0.CO;2-E. http://doi.wiley.com/10.1002/1097-0266(200010/11)21:10/11<1105::AID-SMJ133>3.0.CO;2-E.

Franke, Ulrik, Oliver Holschke, Markus Buschle, Per Närman, and Jannis Rake-Revelant. 2010. "IT Consolidation - An Optimization Approach." In *Proceedings - IEEE International Enterprise Distributed Object Computing Workshop, EDOC*, 21–26. doi:10.1109/EDOCW.2010.11.

Gailly, Frederik, Guido Geerts, and Geert Poels. 2009. "Ontological Reengineering of the REA-EO Using UFO." In *International OOPSLA Workshop on Ontology-Driven Software Engineering*.

Gangemi, Aldo, Nicola Guarino, Claudio Masolo, Alessandro Oltramari, and Luc Schneider. 2002. "Sweetening Ontologies with DOLCE." In *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web, Lecture Notes in Computer Science, Vol. 2473*, 223–233. doi:10.1007/3-540-45810-7_18. http://dx.doi.org/10.1007/3-540-45810-7_18.

Geerts, GL, and WE McCarthy. 2000. "The Ontological Foundation of REA Enterprise Information Systems." *Annual Meeting of the American Accounting Association*: 1–34. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.91.8191&rep=rep1&type=pdf.

Geerts, Guido L., and William E. McCarthy. 2002. "An Ontological Analysis of the Economic Primitives of the Extended-REA Enterprise Information Architecture." *International Journal of Accounting Information Systems* 3: 1–16. doi:10.1016/S1467-0895(01)00020-3.

Gemino, Andrew, and Yair Wand. 2005. "Complexity and Clarity in Conceptual Modeling: Comparison of Mandatory and Optional Properties." *Data & Knowledge Engineering* 55 (3): 301–326.

Gordijn, Jaap, and Hans Akkermans. 2001. "Designing and Evaluating E-Business Models." *IEEE Intelligent Systems* 16 (4): 11–17. doi:10.1109/5254.941353. http://dx.doi.org/10.1109/5254.941353.

Grant, Robert M. 1996. "Toward a Knowledge-Based Theory of the Firm." *Strategic Management Journal* 17: 109–122.

Green, Peter, and Michael Rosemann. 2005. "Ontological Analysis of Business Systems Analysis Techniques: Experiences and Proposals for an Enhanced Methodology." *Business Systems Analysis with Ontologies*: 1–27.

Guarino, Nicola. 1998. "Formal Ontology and Information Systems" (June): 3–15.

Guizzardi, Giancarlo. 2005. "Ontological Foundations for Structural Conceptual Models." Univeristy of Twente. http://doc.utwente.nl/50826.

Guizzardi, Giancarlo, Ricardo Falbo, and Renata S S Guizzardi. 2008. "Grounding Software Domain Ontologies in the Unified Foundational Ontology (ufo): The Case of the Ode Software Process Ontology." In *Proceedings of the XI Iberoamerican Workshop on Requirements Engineering and Software Environments*, 244–251.

Guizzardi, Giancarlo, and Gerd Wagner. 2008. "What's in a Relationship: An Ontological Analysis." In *Conceptual Modeling-ER 2008*, 83–97. Springer.

Guizzardi, Giancarlo, Gerd Wagner, Ricardo de Almeida Falbo, Renata S S Guizzardi, and João Paulo A Almeida. 2013. "Towards Ontological Foundations for the Conceptual Modeling of Events." In *Conceptual Modeling*, 327–341. Springer.

Guizzardi, Giancarlo, Gerd Wagner, Nicola Guarino, Marten van Sinderen, and Marten Van Sinderen. 2004. "An Ontologically Well-Founded Profile for UML Conceptual Models." In *Advanced Information Systems Engineering*, 112–126.

Guizzardi, Renata S S, Giancarlo Guizzardi, Anna Perini, and John Mylopoulos. 2007. "Towards an Ontological Account of Agent-Oriented Goals." In *Software Engineering for Multi-Agent Systems V*, 148–164. Springer.

Guizzardi, RSS, and Giancarlo Guizzardi. 2011. "Ontology-Based Transformation Framework from TROPOS to AORML." *Social Modeling for Requirements Engineering, Cooperative Information Systems Series*: 547–570. http://ufesmestradoanabringuente.googlecode.com/svn-history/r278/trunk/4per/Artigos/CE9917E7d01.pdf.

Hancock, Diana, David B. Humphrey, and James A. Wilcox. 1999. "Cost Reductions in Electronic Payments: The Roles of Consolidation, Economies of Scale, and Technical Change." *Journal of Banking & Finance.* doi:10.1016/S0378-4266(98)00108-3.

Heil, John. 2003. *From an Ontological Point of View*. Clarendon Press Oxford.

Helfat, CE Constance E, and Sidney G SG Winter. 2011. "Untangling Dynamic and Operational Capabilities: Strategy for the (N) Ever-Changing World." *Strategic Management Journal* 1250 (June): 1243–1250. doi:10.1002/smj. http://onlinelibrary.wiley.com/doi/10.1002/smj.955/full.

Herre, Heinrich, Barbara Heller, Patryk Burek, Robert Hoehndorf, Frank Loebe, and Hannes Michalek. 2006. "General Formal Ontology (GFO)--a Foundational Ontology Integrating Objects and Processes." *Onto-Med Report* 8.

Hevner, A.R., S.T. March, J. Park, and S. Ram. 2004. "Design Science in Information Systems Research." *MIS Quarterly* 28 (1): 75–105. doi:10.1007/s12599-012-0246-0. http://dl.acm.org/citation.cfm?id=2017217.

Iacob, M-E, H Jonkers, M M Lankhorst, E Proper, and D A C Quartel. 2012. "ArchiMate 2.0 Specification: The Open Group."

Iacob, M-E, Dick A. C. Quartel, and Henk Jonkers. 2012. "Capturing Business Strategy and Value in Enterprise Architecture to Support Portfolio Valuation." *2012 IEEE 16th International Enterprise Distributed Object Computing Conference* (September): 11–20. doi:10.1109/EDOC.2012.12. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6337268.

Kraaijenbrink, Jeroen, J-C Spender, and Aard J Groen. 2010. "The Resource-Based View: A Review and Assessment of Its Critiques." *Journal of Management* 36 (1): 349–372.

Laurier, Wim, and Geert Poels. 2009a. "Enterprise Ontology-Based Structuring of Conceptual Data Modeling Patterns."

———. 2009b. "Extending REA Models with a Reference Model for Abstraction Mechanisms." In *4th International Workshop on Value Modeling and Business Ontologies, Amsterdam*.

Lee, Hyeoncheol, and Yeong T. Song. 2011. "Bridging Enterprise Architecture Requirements to Archimate." In *Studies in Computational Intelligence*, 365:63–78. doi:10.1007/978-3-642-21375-5_6.

Lowe, E Jonathan. 2006. *The Four-Category Ontology: A Metaphysical Foundation for Natural Science.* Clarendon Press.

McCarthy, William E WE. 1982. "The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment." *Accounting Review* 57: 554–578. doi:8. http://www.jstor.org/stable/246878\nhttp://www.jstor.org/stable/10.2307/246878.

McKelvie, Alexander, and Per Davidsson. 2009. "From Resource Base to Dynamic Capabilities: An Investigation of New Firms." *British Journal of Management* 20 (March): S63–S80. doi:10.1111/j.1467-8551.2008.00613.x. http://doi.wiley.com/10.1111/j.1467-8551.2008.00613.x.

Miklos, Josef. 2012. "A Meta-Model for the Spatial Capability Architecture." *Journal of Theoretical and Applied Information Technology* 43 (2).

Mintzberg, Henry. 1994. "The Rise and Fall of Strategic Planning." *Harvard Business Review* 72: 107–114. doi:10.1016/0024-6301(94)90173-2. http://www.hotlinkfiles.com/files/996068_cfudn/Strategic_Planning-Mintzberg.pdf.

Molnar, George. 2006. *Powers: A Study in Metaphysics*. Edited by Stephen Mumford. Oxford University Press. http://www.lavoisier.fr/livre/notice.asp?id=OR6W63AOKRLOWH.

Moody, Daniel. 2009. "The 'physics' of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering." *Software Engineering, IEEE Transactions on* 35 (6): 756–779.

Mumford, Stephen. 2003. *Dispositions*. Oxford University Press.

Nelson, Richard R, and Sidney G Winter. 1982. *An Evolutionary Theory of Economic Change*. *The Economic Journal*. Vol. 93. doi:10.2307/2232409. http://books.google.com/books?id=6Kx7s_HXxrkC&pgis=1.

Penrose, T. 1959. *The Theory of the Growth of the Firm*. Oxford University Press.

Peteraf, Margaret A, and Jay B Barney. 2003. "Unraveling the Resource-Based Tangle." *Managerial and Decision Economics* 24 (4): 309–323.

Prahalad, C K, and Gary Hamel. 1990. "The Core Competence of the Corporation." *Resources, Firms, and Strategies: A Reader in the Resource-Based Perspective*: 235–256.

Ray, Gautam, Jay B. Barney, and Waleed a. Muhanna. 2004. "Capabilities, Business Processes, and Competitive Advantage: Choosing the Dependent Variable in Empirical Tests of the Resource-Based View." *Strategic Management Journal* 25 (1) (January 3): 23–37. doi:10.1002/smj.366. http://doi.wiley.com/10.1002/smj.366.

Recker, Jan, Marta Indulska, Michael Rosemann, and Peter Green. 2010. "The Ontological Deficiencies of Process Modeling in Practice." *European Journal of Information Systems* 19 (5) (June 15): 501–525. doi:10.1057/ejis.2010.38. http://www.palgrave-journals.com/doifinder/10.1057/ejis.2010.38.

Recker, Jan, Michael Rosemann, Peter Green, and Marta Indulska. 2011. "Do Ontological Deficiencies in Modeling Grammars Matter?" 35 (1): 1–9.

Rosemann, Michael, Peter Green, and Marta Indulska. 2004. "A Reference Methodology for Conducting Ontological Analyses." In *Conceptual Modeling--ER 2004*, 110–121. Springer.

Sako, M. 2004. "Supplier Development at Honda, Nissan and Toyota: Comparative Case Studies of Organizational Capability Enhancement." *Industrial and Corporate Change* 13 (2) (April 1): 281–308. doi:10.1093/icc/dth012. http://icc.oupjournals.org/cgi/doi/10.1093/icc/dth012.

Santos Jr, Paulo Sérgio, A Almeida, Giancarlo Guizzardi, Paulo Sérgio Santos, and João Paulo a. Almeida. 2013. "An Ontology-Based Analysis and Semantics for Organizational Structure Modeling in the ARIS Method." *Information Systems* 38 (5)

(July): 690–708. doi:10.1016/j.is.2012.09.004. http://linkinghub.elsevier.com/retrieve/pii/S0306437912001263.

Santos Jr, Paulo Sérgio, João Paulo Andrade Almeida, Giancarlo Guizzardi, Paulo Sérgio, and Santos Jr. 2010. "An Ontology-Based Semantic Foundation for ARIS EPCs." In *Proceedings of the 2010 ACM Symposium on Applied Computing*, 124–130. http://dl.acm.org/citation.cfm?id=1774114.

Scheer, August-Wilhelm. 2000. *ARIS - Business Process Modeling. ARIS Business Process Modeling*. Vol. 2., comple. http://books.google.com/books?id=Jzuwm2uV7CQC&pgis=1.

Searle, John R. 1983. *Intentionality: An Essay in the Philosophy of Mind*. Cambridge University Press.

Shanks, G., Elizabeth Tansley, Jasmina Nuredini, Daniel Tobin, and Ron Weber. 2008. "Representing Part–whole Relations in Conceptual Modeling: An Empirical Evaluation." *MIS Quarterly* 32: 553–573. doi:Article. http://csz.csu.edu.tw/pp/98/MISQ/2008/2008-3/REPRESENTING PART-WHOLE RELATIONS IN CONCEPTUAL MODELING.pdf.

Stirna, Janis, J\=anis Grabis, Martin Henkel, and Jelena Zdravkovic. 2012. "Capability Driven Development--an Approach to Support Evolving Organizations." In *The Practice of Enterprise Modeling*, 117–131. Springer.

Teece, David, and Gary Pisano. 1994. "The Dynamic Capabilities of Firms: An Introduction." *Industrial and Corporate Change* 3: 537–556. doi:10.1093/icc/3.3.537-a.

The Open Group. 2009. *TOGAF Version 9. Architecture*. http://books.google.com/books?hl=en&lr=&id=dNIf8TZaylcC&oi=fnd&pg=PT2&dq=TOGAF+Version+9&ots=W6dnwd5wtj&sig=C4emZBzdp5pkoMOeE9sy1FtaYX4.

Wand, Yair, and Ron Weber. 1989. "An Ontological Evaluation of Systems Analysis and Design Methods." *Information System Concepts: An In-Depth Analysis. Elsevier Science Publishers BV, North-Holland* 1989.

———. 1990. "Mario Bunge's Ontology as a Formal Foundation for Information Systems Concepts." *Studies on Mario Bunge's Treatise, Rodopi, Atlanta*: 123–149.

Weber, Ron. 1997. *Ontological Foundations of Information Systems*. Coopers & Lybrand and the Accounting Association of Australia and New Zealand Melbourne.

Wiggins, Robert R., and Timothy W. Ruefli. 2005. "Schumpeter's Ghost: Is Hypercompetition Making the Best of Times Shorter?" *Strategic Management Journal* 26 (10) (October): 887–911. doi:10.1002/smj.492. http://doi.wiley.com/10.1002/smj.492.

Zandi, Faramak, and Madjid Tavana. 2010. "A Multi-Attribute Group Decision Support System for Information Technology Project Selection." *International Journal of Business Information Systems* 6 (2): 179–199.

Zdravkovic, Jelena, Janis Stirna, Martin Henkel, and J\=anis Grabis. 2013. "Modeling Business Capabilities and Context Dependent Delivery by Cloud Services." In *Advanced Information Systems Engineering*, 369–383.