A System Core Ontology for Capability Emergence Modeling

Rodrigo F. Calhau[1,2,3], Tiago Prince Sales[3], Ítalo Oliveira[3], Satyanarayana Kokkula[4], Luís Ferreira Pires[3], David Cameron[5], Giancarlo Guizzardi[3], and João Paulo A. Almeida[1]

[1] Ontology & Conceptual Modeling Research Group, Federal University of Espírito Santo, Vitória, Brazil
[2] LEDS, Federal Institute of Espírito Santo, Serra, Brazil
[3] Semantics, Cybersecurity & Services, University of Twente, Enschede, The Netherlands
[4] Department of Science and Industry Systems, University of South-Eastern Norway, Kongsberg, Norway
[5] SIRIUS Centre, Department of Informatics, University of Oslo, Norway
calhau@ifes.edu.br; t.princesales@utwente.nl; i.j.dasilvaoliveira@utwente.nl; satyanarayana.kokkula@usn.no; davidbc@ifi.uio.no; g.guizzardi@utwente.nl; jpalmeida@ieee.org

Abstract. To properly understand organizational adaptation and innovation, it is critical to understand the emergence phenomenon, i.e., how the capabilities of a system emerge after changes. However, for this, we should be able to explain systems, their structure, behavior, and capabilities. In pursuit of an understanding of the emergence phenomenon and the nature of those new kinds of systems in organizations, we propose a well-founded system core ontology based on the Unified Foundational Ontology. The ontology is also grounded in system science definitions and disposition theories. For a more integrated explanation of emergence, the proposed ontology considers distinct perspectives of a system, such as its composition, structure, properties, and functions. In the end, we discuss the applications and implications of the proposed ontology on the enterprise architecture area and emergence modeling.

Keywords: system · emergence · ontologies · enterprise architecture.

1 Introduction

In recent years, we experienced rapid development of information technologies, such as artificial intelligence and cloud computing, which stimulated the emergence of new kinds of social and technical systems within enterprises [6, 22, 27]. In this context, enterprises are compelled to constantly adapt and innovate to remain competitive. This requires certain organization-wide capabilities that do not exist in specific individuals but are the result of a complex phenomenon through which capabilities *emerge* from the interaction of organizational parts. An organization with innovative professionals does not necessarily imply an innovative organization, since this depends on the kind of relationships and interactions among them.

Emergence is a complex phenomenon that cannot be explained by just one cause. It is a result of the way the system's parts are related, the properties of parts, relational properties, and constraints, among other factors [9, 26, 28, 31]. This phenomenon has been studied by system science researchers since the rise of the General System Theory (GST) [1, 9, 12]. These researchers consider the notion of *system* present in distinct fields in order to identify common principles among various system types (e.g. physical, chemical, biological, and social systems) [11]. According to these authors, an enterprise can be seen as a socio-technical system composed of interrelated technical and social parts whose capabilities emerge from the interaction between its parts [17].

At the same time, the structure and capabilities of enterprises have been studied in the Enterprise Architecture (EA) discipline [8]. In a similar way, EA notations support the representation of enterprises, their parts, relationships, and behavior, which, in theory, could be used to model capabilities resulting from the emergence phenomenon. However, there are no guidelines on how to properly model emergence. For example, how should one structure a team into roles in a way that maximizes the overall performance of the team? Or what is the best combination of functions for a team to be more productive? Since it helps to explain the emergence phenomenon, system science can ground EA with its theoretical foundation to model systems and help answer these questions. As addressed by [34], ontologies can play a fundamental role in this task since system models concern a distinct paradigm in organizational context.

Ontologies can improve the *expressiveness* and *domain appropriateness* of EA notations, as is shown in [3, 14]. According to [20], ontologies have been useful in the computing field for representing and formalizing the semantics of various types of artifacts. Many ontologies have been proposed to model different types of systems, such as systems-of-systems [5], enterprise systems [27], smart systems [2], cyber-physical systems [35], to name a few. All these system-related ontologies focus on solving technological and practical issues related to a specific context and generally lack a wide system concept understanding, failing in particular to address the emergence phenomenon.

In this paper, we contribute to bridging this gap by applying concepts from system science authors such as Bunge [10, 12], authors from GST [1, 9] and systems engineering (SE) [16] areas, by leveraging the contributions related to emergence from system science. In [13][6] we proposed an ontology-based language pattern to ArchiMate to represent capability emergence based just on human capabilities. This previous work was grounded in disposition theories [7, 19], to explain the emergence of capabilities, without considering system distinctions, their parts, functions, and connections. In this paper, we consider not just human capability but the capabilities of distinct system types in organizations. We also consider the phenomenon of emergence from the perspective of systems, their components, and their relations. In order to provide a comprehensive account of the emergence phenomenon in a system, we consider distinct perspectives, including its composition, structure, properties, and function, based mainly on Bunge's "systemist" model [11, 12]. To properly account for these system distinctions in EA notations, we propose a system core ontology based on the Unified Foundational Ontology (UFO). The ontology aims to: (i) improve the capability emergence modeling (in EA notations) of socio-technical systems into enterprises and; (ii) facilitate the identification

[6] under review

of capability emergence patterns by using some pattern recognition technique. To regard this, the proposed ontology is also grounded on theories of parts and parthood [18, 29], other system ontologies, system models [10, 11, 16, 25], literature reviews on the system concept [16], emergence explanations [21, 26, 28, 31], and disposition theories [7, 19].

This paper is structured as follows: Section 2 presents an overview of the literature related to system theory concepts; Section 3 presents the Unified Foundational Ontology, used to create the proposed system ontology presented in Section 4, which is the main result of this work; Section 5 discuss the application of the ontology in the EA context and its implications; Section 6 shows an application of the ontology in the Spotify case, and; Section 7 presents the related works, and Section 8 concludes with our final remarks.

2 Emergence and Systems

The concept of "system" is strongly associated with emergence. Very often, a system is defined as a whole composed of related parts that allow the emergence phenomenon. For this reason, comprehending systems is fundamental to an account of emergence. System's definitions have distinct perspectives and most often vary from one author to another [16]. Despite this variation, authors who worked on the system concept recently or those whose work traces back to the origin of the General-System Theory, such as [1, 9, 11], converge on an understanding of a system as a kind of 'complex' (or 'organized' whole) composed of 'connected' (or interacting) elements. In this sense, a system is understood as a collection of things that, through their connections (or interactions), creates something new, such as behavior or emergent properties [9, 16]. Reinforcing this understanding, Dori and Sillitto [16] presented the following generic definition of a system based on the extensive literature review by analyzing more than one hundred definitions:

> *A system is a set (or combination, group, collection, arrangement, organization, assemblage, assembly, ensemble) of parts (or components, elements, objects, subsystems, entities) combined (or integrated, organized, configured, arranged) in a way that creates (or enables, motivates) properties (or functions, processes, capabilities, behaviors, dimensions) not possessed (or exhibited, presented) by the separated (or individual, single) parts (or components, elements, objects, subsystems, entities).*

Dori and Sillitto's generic definition captures the many terms used to refer to the different aspects of the complex notion of a system. It alludes to the inevitable *plurality* of things that in *combination* make up a system ("a set of parts combined..."), also referring to the existence of properties that characterize the system *as a whole*, i.e., beyond the properties of parts in isolation ("... in a way that creates properties not possessed by the separated parts").

Bunge [10, 11] classifies system properties into *resultant* and *emergent*. Resultant properties are those that can be decomposed, explained, or reduced into properties of a system's parts. For example, the total mass of a system is defined by the simple sum of the masses of its components. Unlike resultant properties, emergent properties are those that, while related to the properties of parts, are not present in isolation in the separated parts. For example, the buoyancy of a ship cannot be reduced to the buoyancy of its parts (an arbitrary piece of a steel hull is typically not buoyant by itself). In the words of [28], the *emergent properties supervene* on the components' properties. According to [21,26], the emergent properties are also the result of *system constraints*, which limit it on the one hand but enable the arising of new characteristics on the other. [21] exemplifies this through the restrictions caused by the knee in the femur and tibia movements, which contribute to the emergence of the walking capability. In the same way, according to [31], *emergent properties* are a direct consequence of the relationships among parts. For instance, as the author exemplifies, what distinguishes diamond and graphite is the way that carbon molecules are associated. Based on these differences, distinct emergent properties appear as transparency, in the case of a diamond, and electric conductivity, in the case of graphite.

In system definitions, emergence is also associated with system functions. As [30] states, the concept of *function* is generally related to a teleological perspective on a system. In this sense, *functions* are manifested through some goal-oriented result or behavior (process, action) [1,15]. *Functions* are frequently defined as a kind of *property* related to the capability (or disposition) concept [15,27]. As they have a relational aspect, some authors, such as [15], also see *functions* as a kind of *role* played through behaviors that are required by the system's capability definition. In general, *functions* are seen as a system-dependent aspect, called "system function" by [15], inherent specifically to one bearer. However, according to the authors, *functions* can also be seen in a more "generic" way, independently of a specific system (called "ontological function" by them). In this case, these "generic" *functions* are useful for designers to describe an intended system, before building or acquiring it. As [15,30] state, initially, the system's *functions* description starts identifying the system *macro-function*, as it relates to the whole system. Then, as the authors describe, this *macro-function* is decomposed into *sub-functions* (often to be assigned to the system parts), a process known as (logical) *functional decomposition*.

As explained above, the emergence phenomenon results from related parts and not isolated ones. Bunge [11] defines system structure as a set of all relationships between a system's parts. According to [12], the relationships between system elements can be *"bonding"* or *"non-bonding"*. *Bonding* relationships (also termed *"connections"*) are those in which one element somehow causes changes and impacts the other [11]. These relationships can also be characterized by the flow of energy, material, or information between elements [11]. Otherwise, *non-bonding* relationships are those that do not impact the *relata*, such as comparative relationships as temporal or spatial relationships [12] (e.g., "higher than" or "younger than"). The set of the *bonding* relationships connecting all system's elements at a specific time shapes the system's *bonding structure* (or "configuration") [11]. In this context, the *bonding structure* represents the system's form, organization, or arrangement [16]. It allows the system elements to interact with others and, consequently, the whole system to display its behavior. [12] defends that an object needs a *bonding structure* to be considered a system. [1] remarks that each system's element must be connected to every other one, either directly or indirectly. Hence, in

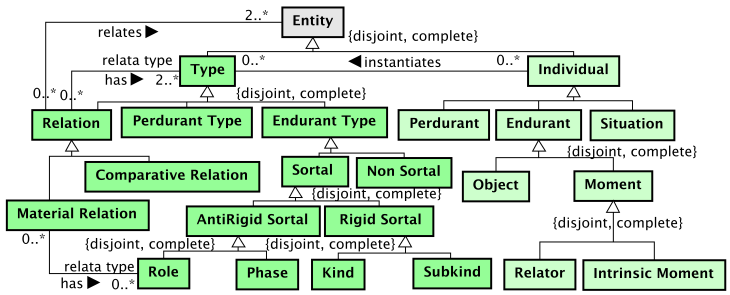

Fig. 1. Unified Foundational Ontology fragment

a system, there must be no isolated subset of elements. In this sense, not connected or independent elements should not be considered as a system [30].

3 Foundational Baseline

Foundation Ontologies are both Formal and Reference Ontologies. Guizzardi [20] proposes in his work the *Unified Foundational Ontology* which describes domain-independent and general concepts and that we will adopt in this work to perform ontological analysis. Figure 1 presents a UFO fragment that shows its fundamental distinctions, among *individuals* and *types*. These two concepts basically represent types and their instances. For example, *person* is a *type*, and Karl and John are *individuals*, instances of the *person* type.

UFO divides *individuals* into *endurants*, *situations*, and *perdurants* (or events). *Endurants* are *individuals* that persist in time, maintaining their identity (i.e. John, The Beatles, Spotify Technology S.A.). *Perdurants* are *individuals* that manifest themselves through time (e.g. John's birthday party, the inauguration of the Pope Francis). *Situations* are individuals composed (possibly) of many other individuals (including other situations) that may trigger *events*.

Endurants are divided into *objects* and *moments*. *Objects* are *endurants* that do not depend on another *individual* to exist (e.g., the Earth, John, an apple). In contrast, *moments* depend on another *individual* to exist (e.g., Mary's age, Gerald's headache, the reddish color of an apple). A *relator* is a specific type of *moment* capable of connecting two or more entities (e.g., a marriage that connects spouses). In contrast, an *intrinsic moment* is one that is existentially dependent on a single individual. Intrinsic moments include *qualities*, categorical properties such as color, height, weight, etc., and *modes*, which are moments that can bear their own moments and can vary independently. *Modes* include *dispositions*, which are moments that can be manifested through *events* in certain *situations*. Examples of dispositions include John's ability to speak English, and an airplane's flying capability. Based on UFO-C (extension of UFO approaching social aspects), *agents* are considered *objects* that perceive *events* and perform *actions* based on a background of *beliefs*, *desires*, and *intentions* (special categories of *intrinsic* *moments* termed *intentional moments*, omitted from the figure for brevity). As depicted in the model, *agents* can be *physical* (e.g., humans and animals) or *social* (e.g., teams, organizations, communities, etc.), and all of these are considered potential bearers of *capabilities* and *intentional moments*.

Although UFO does not include the concept of *system*, it includes the concept of *functional complex*, which is similar to the concept of *system* from a mereological perspective. Functional complexes are *objects* whose parts play distinct *functional roles* with respect to the *whole*. In this case, the *parthood relation* is defined in UFO by the *"is a component of"* relationship [20]. The *"is a component of"* relationship is a type of mereological relationship between *functional complexes* that establishes the part in a functional complex's whole. In this case, it establishes the *functional role* played by the *parts* (components) in the *functional complex*. For example, in a chair, each wooden piece has a different *functional role*: front leg, back legs, seat, etc. Besides not being the focus on [20], these *parts*, which play distinct *functional roles*, should be related between them. Besides that, UFO also considers another whole, namely *collectives*. Differently from *functional complexes*, collectives are wholes whose parts perform the same *role* type. Examples of collectives include a group of students performing a group assignment or a collection of books in a library.

Types are predicative entities whose instances share common features. In the taxonomy of types in UFO, there are *endurant types*, *perdurant types*, *object types*, etc., according to the ontological nature of their instances. *Types* also include *relations* between two or more *entities*. As shown in Figure 1, relations in UFO are associated with two or more relata types. *Material relations* are those that apply in the presence of a *relator* mediating the *relata*, e.g., the "married with" relation requires a *marriage*, the "enrolled in" relation requires an *enrollment*, and so on. As illustrated in Figure 1, *material relations* have *roles* as *relata type*. For example, the "marriage with" *material relation* has the spouses' roles as its *relata types*. *Comparative relations* are another kind of relation in UFO, as Figure 1 depicts. They are called *formal relations* since they involve two or more entities directly, without the intervention of a mediator (e.g. "taller than" and "younger than" relations).

Guizzardi [20] also categorizes the types according to the identity principle that entities maintain. Based on this, *types* are classified as *rigid* and *anti-rigid*. *Rigid* types are those that apply necessarily to their instances through their whole existence, and include *kinds* and *subkinds* (e.g. "Person", "Car", "Pineapple"). In contrast, *Anti-rigid* types are those that classify their instances contingently (or "dynamically"). *Roles* are anti-rigid types whose contingent classification conditions are relational (e.g., "Student" and "Employee") [20]. UFO also considers *non-sortal* types representing common properties of individuals of multiple *kinds*: (i) *categories* subsuming multiple kinds rigidly (e.g. mammal); and (ii) *role mixins* which subsume various *roles* with distinct *kinds* (e.g., "Customer", subsuming "Personal Customer" and "Organizational Customer").

4 The System Core Ontology

For the ontology modeling, we use the OntoUML notation, proposed by Guizzardi [20], as a UML extension that addresses the foundational distinctions from UFO in UML class diagram through stereotypes. The ontology requirements were identified based on GST literature and in Bunge's Composition, Environment, Structure, and Mechanism (CESM) model [12]. Therefore, the system ontology must account for a: **R1) system's composition** concerning its *components*, *subsystems*, and their hierarchical relations; **R2) system's structure** concerning the notions of (internal and external) *connection* and *non-bonding relationship*; **R3) system's function** concerning the *roles* played by the system and components in a *functional decomposition*; and **R4) system's characterization** concerning the system's *emergence* phenomenon.

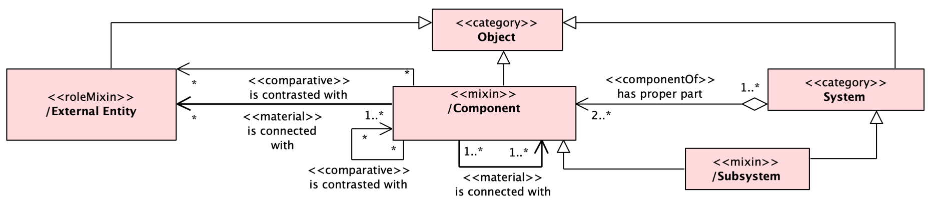

Fig. 2. Well-founded System Ontology (Composition and Structure)

Concerning system connectivity, *systems* are considered in this work as *complexes* of related elements [9] or *integral wholes* [29] since they need a "unifying condition" to exist. When it comes to UFO distinctions, we consider *systems* as being a *category* of *object*, as Figure 2 depicts. By being an *object* in UFO distinctions, systems can also be (social or physical) *agents* with intentions, desires, beliefs, perceiving events, and performing actions. Based on this distinction, an organization or a team are considered a *social agent* and also (socio-technical) systems formed by other agents (e.g., teams, humans, etc.) and a non-agentive objects (e.g., equipment and other resources). Specifically, based on [20], systems are *functional complexes* since the system's components also perform different *functional roles* in the respective whole. In this case, as *functional complexes*, we consider the following criteria that differentiate *systems* from "simple" (atomic) *objects*: (i) the complexity degree (number of components and connections); (ii) the integrated (bonding and non-bonding) structure formed by components' connections; (iii) the heterogeneity and complementarity of components' functional roles (and capabilities); and (iv) the emergence of new properties and new behaviors.

In relation to its *composition*, the *system* "has proper parts" called *components* (a derived class), as shown in Figure 2. In this case, *components* represent *mixins* of interrelated *objects* (including *functional complexes*, *quantities*, and *collectives*), in UFO terms. In this ontology, they are generic concepts that correspond to all kinds of system's parts (or elements), such as *units*, *blocks*, *modules*, *interfaces*, *ports*, and even other *systems*, called *subsystems* (a derived class), as depicted in Figure 2. These *components* could be arranged hierarchically, through the part-hood relation and they are interrelated. For example, the wooden stool *ws* has as proper parts the legs *l*1, *l*2, *l*3, and *l*4, besides the seat *st*.

We consider in the ontology the two senses of the "function" of the system (and components) addressed by [15]: one related to its "*position*" and the other to its "*ca-*

pabilities". Regarding the former, it is a more "generic" perspective, independent of a specific system, as [15] approach. In UFO terms, this perspective is closely related to *roles* (specifically *functional roles*), due to their generic, anti-rigid, and relational nature. As a result, in this sense, the *function* (in the sense of "position") of the system (or its component) is a *functional role* it instantiates, as depicted in Figure 3. As shown, the *functional roles* are "characterized by" *moment types*. As depicted, *moment types* can "complement" (and also trigger or block) others, based on the same disposition theories above. In addition, based on [15], we also regard the (macro) *function* of a system can be decomposed into (sub) *functions*. This aspect is shown in Figure 3 by the "constitutes" relationship between *functional roles*, forming a functional *role's* "hierarchical structure". In this case, to be a system, its sub-functional roles must be heterogeneous but also complementary, based on the *moment type's* complementary relations. Finally, the decomposition of a *functional role* and the parthood relationships between components and the system must match.

Concerning the second sense of *function*, as something intrinsic to a specific bearer, we consider that *systems* and their *components* bear *capabilities* (i.e., subtypes of *system moment* or *component moment* with dispositional nature) which can be specialized in this context to perform *functions*. So, in this sense, we mean *functions* as specialized *capabilities* (*dispositions*) that bring "benefits" to other entities in the system context, as defined by [16, 24]. As illustrated in Figure 3, for a *system* or *component* to perform some *functional role*, it must have certain *capabilities* (*moments*) in compliance with the *moment types* that characterize this *functional role*. For example, the wooden stool ws performs the stool (macro) *functional role* of "supporting a person"; the leg $l1$ performs the front-right leg *functional role*, $l2$ performs the front-left leg *functional role*, $l3$ performs the back-right leg *functional role*; $l4$ performs the back-left leg *functional role*, and the seat st performs the seating *functional role*. While performing these roles, the components must have specialized *capabilities* (functions): $l1$ must support seat st; $l2$ must support seat st too; seat st must support the person.

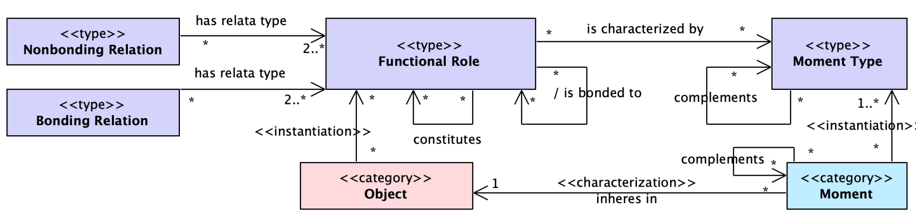

Fig. 3. Well-founded System Ontology (Functional Aspect)

In this work, we adopt the *bonding* and *non-bonding relations* definitions of Bunge [11, 12], concerning that one relatum impacts the other in the *bonding relation* and in the *non-bonding relation* it does not. We regard these definitions from the two UFO perspectives: *types* and *individuals*. In the former, *bonding* and *non-bonding relations* correspond to "types of relations" between *functional roles* (types). This aspect is shown in Figure 3, in which both *bonding* and *non-bonding relations* have two or more *functional roles* as their relata. Concerning these relations from the *individuals'* perspective, they follow the same principle. When the *functional roles* are instantiated by some entity (a system, its components, or external entities), the *bonding relation* type is "embodied" in (or instantiated by) these entities. This instantiation is illustrated in Figure 2 by the "connected with" relationships between *components* and between *components* and *external entities*. In this context, *connections* between only components in a system are called *internal connections*, and *connections* between components and *external entities* are called *external connections*. In a *non-bonding relation*, as two or more *functional roles* are related as a kind of restriction, to instantiate this *functional role*, the *component* must satisfy these restrictions established by the *non-bonding relation*. The instantiated *non-bonding relation* is represented in Figure 2 through the "is contrasted with" relationships between *components* and between *components* or *external entities*.

Concerning the ontological nature of *bonding relations*, when an *object* instantiates a *functional role*, its *dispositions* (capabilities) required for this role are specialized to attend to the system's peculiarities, in order to perform *functions*. So, for meeting the system's unifying condition, those "specialized" *dispositions* need to become externally (and existentially) dependent on other objects (components or external entities) and interact in the system context. As a consequence, relationships between connected objects are mediated by these externally dependent *dispositions* (capabilities). Based on this, we consider in this work that *bonding relation* is a kind of *material relation* in UFO terms. In contrast, Bunge's *non-bonding relations* are not *material relations* in UFO terms. Since they just regard constraints or restrictions, they are considered a kind of descriptive relationship that relies on the relatas' intrinsic properties (UFO), unable to change the relatas' state. As a result, they are related to the UFO *comparative relation* type. In the case of the wooden stool ws, the legs $l1$, $l2$, $l3$, and $l4$ are "connected with" the seat st (bonding relation); the legs "is contrasted with" each other since they are parallel, and they are perpendicular to the seat st (non-bonding relations).

Based on [10], systems have "global properties", not founded in their parts. We call them *system moments*, as shown in Figure 4. In UFO terms, this work considers *system moments* as a *category* of *moments*. As a consequence, *system moments* represent *extrinsic moments* (e.g., "*relators*", "mutual properties") and *intrinsic moments*, as *qualities* (e.g., "attributes") and *modes* (e.g., dispositions and capabilities). We also adopted here the distinction of *emergent* and *resultant properties* addressed by [10]. As discussed, emergence is not a simple phenomenon that appears from one and only factor, but it is a result of the component's properties combination, constraints (represented by comparative relationships), and connections [21, 26, 28, 31].

According to that, we consider in this work that an (emergent) *system moment* "emerges in" (i.e., stands in) a specific *system situation*, as depicted in Figure 4. In this case, the *system situation* represents the *components* of a *system* and their relationships in a certain occasion. Besides emerging "in a *situation*", *system moments* also "emerge from" (component) *moments*, as shown in Figure 4 (including *dispositions*, *qualities*, *relators*). In a complementary way to the emergence explanations, we also consider the relationships between *moments* (i.e., *dispositions*) to explain emergence, as approached in [13]. As a result, for a *system moment* to emerge from (components) *moments*, the latter must be (inter) related. In this sense, based on disposition theories, we consider that components' moments can be complementary, i.e., reciprocal (mutually activated) [19] or additional (additionally activated) [7]. As shown in Figure 4, both types of relations are considered in the "complements" relationship. We also consider which *component moments* can have relationships of triggering [7] and blocking [19] (not shown in the model as these relationships are derived). For example, the wooden stool ws has the "supporting capability" as an emergent *system moment* (emerged from the "supporting capability" from legs and seat). Concerning the disposition relation in this example, the "supporting capability" of each leg is additional to each other and they are mutually reciprocal to the "be supported capability" of the seat.

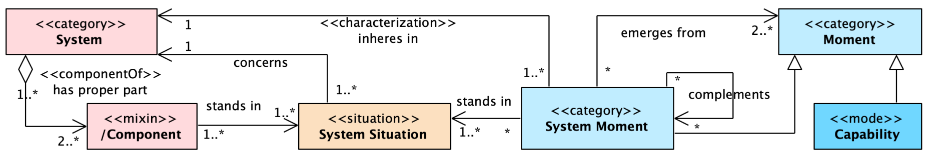

Fig. 4. Well-founded System Ontology (Capability)

5 Capability Emergence Modeling in Enterprise Architecture

As exemplified, a wooden stool must follow some "guidelines" in order to allow the emergence of its "comfortably supporting humans" capability and thereby fulfill its function. Based on the ontology distinctions, it must be composed of rigid pieces performing legs and seat roles; these roles must be complementary; these rigid pieces must attend quality criteria (e.g., have certain resistance); the legs must have the same height; must be parallel to each other and perpendicular to the seat. These guidelines are a "replicable" generalization that establishes patterns of emergence that a stool must follow.

Similarly to simple systems such as a stool, a socio-technical system such as a team, or an organization, also has certain "emergence patterns". For example, an organization can determine that productive teams are those in which developers (functional role) collaborate (connection); have complementary skills but share the same values (comparison); and the tech leader (functional role) supports the developers (connection) and is more experienced than them (comparison). In sum, a successful organization that evolves and adapts is one that is able to identify and replicate these "emergence patterns", remaining and creating new capabilities even when it changes.

The system's ontological distinctions can help with identification and representation of these patterns. They can facilitates the description of components their connections, properties, capabilities, functional roles, etc. These distinctions form guidelines that can be used to create: (i) *capability emergence models* for a particular system (e.g., a specific team), used to understand the emergence for create, change, or analyze it; or (ii) "general" *capability emergence patterns*, using *functional roles* to generalize the emergence phenomenon from different systems (e.g., all teams of an enterprise) and reply (specialize) it in distinct situations. Examples include appropriate combinations of (complementary) professional roles for building performative teams; the capability types each professional needs to have for product development with high quality; the types of relationships these professional roles need to have to allow collaboration; the more appropriate equipment type for a type of professional role to increase safety, and so on.

Embedded in these guidelines, the ontology also provides some "modeling patterns" to be satisfied by these *capability emergence models*, improving the system and emergence representation. In a *capability emergence model* for a particular team, for example, all team members must be connected (directly or indirectly); team members must satisfy constraints; team members must have complementary *capabilities* (reciprocal and additional); these related *capabilities* must follow the *connections* between the team's members; the *connections* must also happen just between team's members with related *capabilities*; team's members must have a function (*functional role*) in the team; and team members must satisfy the *functional role* criteria. Regarding the emergence patterns, it must satisfy the following principles for a team, for example, the team's macro-functional role must be constituted of at least two or more team member's sub-functional roles; all team member sub-functional roles must be related to *bonding relations* (directly or indirectly); team member sub-functional roles must be complementary/additional (based on the moment types); team member sub-functional roles must have constraints; all team member *functional roles* must be characterized by *moment types;* and non-bonding relations between the team's member functional represent restrictions to be satisfied.

These emergence patterns can be identified from the literature, success cases of organizations, experts, or even pattern recognition techniques which can analyze organizational data to identify these patterns. In this case, for example, the better team's structure (roles and relationships) and professional characteristics (desired capabilities for each role) can be identified from data of the more performative teams (with better KPIs) of an organization. To facilitate the identification and application of the emergence patterns, these tasks can be included into the knowledge management process of the organization. In this case, the identified patterns can even be considered as a knowledge items into the knowledge repository of the organization and forming a "library" of patterns.

Another important implication of this work is the improvement of EA notations through ontological analyses. As a result, it could be used as the basis for the creation of language patterns to increase the expressiveness of these notations, incorporating system-related concepts and emergence-related aspects into them. The use of the system ontology to create language patterns in EA notations will be detailed in the following case study.

6 Emergence in the Spotify Company

To show the benefits of the system ontology and the guidelines shown above, we have applied them to a real-life case study from the literature to improve the capability emergence modeling. This study case was addressed originally by Bäcklander [4] and concerns the Spotify company. Bäcklander [4] focuses specifically on understanding the emergence patterns followed by the company, such as (1) how adaptability and related capabilities (e.g., self-organization, learning, collaboration, etc.) emerge in the company, and (2) how the agile coach position contributes to these capabilities. Bäcklander [4] performed an ethnographic study inside the company, observing and interviewing the agile coaches. After the interviews, the author identified the main characteristics, practices, interactions, and motivations of the agile coaches that contribute to the emergence of these organizational capabilities.

The Spotify company is well-known for having a unique structure, which distinct authors describe [4]. Spotify is a socio-technical system composed basically of guilds and tribes. Guilds represent cross-cutting study groups focused on employee development and which anyone can join. In contrast, tribes are focused on the development of solutions. Tribes are composed of squads, a kind of development team in Spotify. Tribes are also formed by chapters in Spotify. They are "local" study groups that anyone can join, similar to guilds. In this case, while tribes and squads are result-oriented and highly coupled groups, guilds and chapters are learning-oriented and loosely coupled groups. Each squad has members (e.g., the developers) and is related to a product owner. Besides this, each tribe has its own agile coach supporting the developers of each squad.

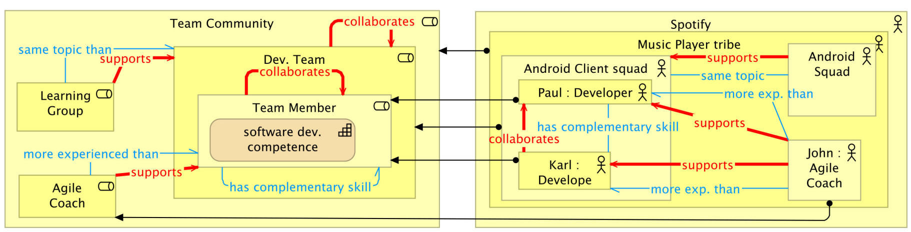

Fig. 5. Spotify as a System

To allow the emergence of the adaptability capability, Spotify considers some capability emergence patterns to be followed by the company, based on successful experiences. In this pattern, the company considers not only professional capabilities but also functions performed by them, and the connection between them, among other elements. These patterns are replicated in all subsystems (squads, guilds, etc), at all levels of the organization, impacting the whole company. As a result, they contribute to creating a flexible organizational structure, a condition for the emergence of adaptability. Each of these components of Spotify mentioned above performs a certain *functional role* in the system: (i) *team community* functional role, played by "tribes"; (ii) *learning community* functional role, played by "guilds"; (iii) *development team* functional role, played by "squads"; and (iv) *learning group* functional role, played by "chapter". *Team community*, *learning community*, *development team*, and *learning group* are examples of *functional roles* played by social entities. They are constituted by *sub-functional roles* played by people, such as *agile coaches* and *developers* functional roles.

These functional roles also must be characterized by *capability types* and have *bonding* and *non-bonding relations* as part of Spotify's *emergence pattern*. For example, (i) learning groups (chapters) support development teams; (ii) development teams support other development teams; (iii) agile coaches support team members; (iv) team members collaborate with other team members, among others. Regarding non-bonding relations, they describe distinctions that should be attempted by Spotify's components, for example (i) learning groups (chapters) have the same topic as some team (squads); (ii) agile coaches must be "more experienced than" team members; (iii) team members "have complementary competence than" other team members; (iv) chapter member "has similar competence to" other chapter members, among others. Concerning the characterization of these *functional roles*: (i) the agile coach must have supporting, leader boosting, and communication skills; and (ii) the developer must have agile development competence and communication skills; Spotify and its components should instantiate these functional roles in order to replicate the emergence pattern and attempt the "unifying conditions" to form a whole (system).

To improve the representation of this emergence pattern, we created an illustrative language pattern in ArchiMate specific to this case, as shown in Figure 5. As illustrated, *functional roles* are represented as BUSINESS ROLE construct (when it is performed by an agent) or RESOURCE construct (when it is performed by an object, as equipment for example) using the assignment relation; *capability types* are represented as CAPABILITY constructs related to *functional roles*; *bonding relations* and *connections* are represented using TRIGGERING, FLOW, or SERVING relationships between *functional roles* (highlighted in red); *non-bonding relations* and *comparisons* are represented using ASSOCIATION relations related to *functional roles* (in blue); and *system*, *component*, *external entities* are represented using STRUCTURAL ELEMENTS, related to functional roles through realization relation; and parthood relationships are represented using COMPOSITION or ASSOCIATION relationships.

The emergence phenomenon in Spotify is one of the main aspects explained in this case study, even because Bäcklander [4] considers complex and adaptive system theory as a foundation. As a result, Spotify is seen by the author as a complex system that belongs to a changeable environment. In this case, the work explains how agile coaches play a special position since they contribute to the emergence of some capabilities, especially the adaptability and evolution of Spotify. The author associates adaptation and evolution capabilities with learning, open dialogue, and creativity capabilities. According to the study case, these capabilities in Spotify are a result of the agile coaches acting as enabling leaders and creating adaptation spaces in the company.

However, the author explains the influence of the agile coach in this case, but does not explain how emergence happens. As stated, the emergence in Spotify's context is a result of an emergence pattern that addresses relationships and properties to be satisfied by the professionals performing certain functional roles. In order to reach this pattern, those professionals establish connections between themselves, in accordance with the functional role relations, as detailed above. Based on these connections, the professionals are able to interact and perform tasks. These connections are a result of their complementary capabilities (additional and reciprocal). They are also a consequence of enabling relationships between capabilities.

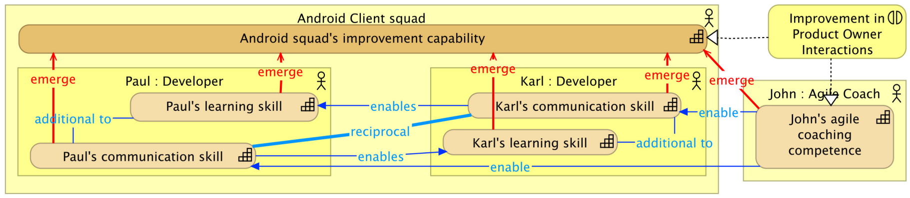

Fig. 6. Emergence phenomenon in Spotify

Figure 6 illustrates how the agile coach contributes practically to the emergence phenomenon in a squad in this emergence pattern. As depicted, in this case, one of the main contributions of the agile coaches is enabling better communication between the squad members through the supporting connection. Consequently, the developers can share more of their opinions, among other communication skills. As it is shown, the more the (reciprocal) communication skills of the developers are enabled by agile coaches, the more their learning (and reflection skills) are enabled through these interactions, allowing them to create new solutions. In summary, as a result of these dynamics, the improvement capability of the squad emerges, as illustrated in Figure 6. Therefore, as a result of the emergence of the improvement capability in each squad stimulated by the agile coach, the improvement capability of the tribe also emerges, contributing to the emergence of adaptability in the whole organization.

7 Related Work

One of the main related work in the context of our work is the Systemic Enterprise Architecture Methodology (SEAM) [34]. SEAM is concerned with a method for assisting in the modeling of businesses as complex systems. SEAM enterprise modeling addresses multiple levels that can guide emergence phenomenon modeling. Aside from that, the methodology is based on an ontology that addresses fundamental concepts such as object, action, state, location, time, space, and characteristics. Nonetheless, the SEAM method does not center on emergence phenomenon modeling nor on capabilities. Therefore, no modeling guidelines are provided in this case. Furthermore, the ontological distinctions fail to take into account basic concepts from system science such as system, function, component, connection, and other relationships. Otherwise, the present work considers these basic distinctions in a well-founded manner.

Concerning system ontologies, a number of them focus on *engineered systems* [25] as *cyber-physical systems* [6,35] and systems-of-systems [27]. Most of these ontologies focus on defining systems, components (subsystems), and their parthood relationships. A part of these models also considers system characteristics, such as attributes, properties, and capabilities [5,6,25]. However, almost none of the models consider the representation of the *emergent property*. Some exceptions [23, 33] define an *emergent property* as a property that belongs to the whole system, not its components. Besides this, these models do not relate the *emergent properties* to the basic properties (or a kind of situation), which are inherent to the system parts.

System function is not a well-covered aspect in related work. Many of them [25, 27, 35] link the (whole) system to a "generic" function concept (more related to an intrinsic aspect of it). In addition, in these works, the functional decomposition of the system is not considered. Otherwise, some of the works [6] relate the system to a kind of role that it can perform (or position that it occupies). They also allow a hierarchical representation of the system's *functional roles*. As a result, these ontologies enable the functional decomposition representation, including not only the system's *functional role* as a unit but also the component's *functional role*. Most of the works [5, 6, 23, 25, 27, 33] consider some kind of structural relationships among systems (or their components). Some of these works define this structural relation explicitly through a concept like *connection* [25], structural relation [27], connector, link [5], interaction, or binding mutual [33]. Others also consider some kind of mediators as connection points, ports, or interfaces [6]. One aspect not considered by the ontologies is the *non-bonding relationships* between systems and components. The only exception is in [33], which defines a kind of mutual property called non-binding property. Most of these system-related ontologies are focused on addressing technological and practical challenges in particular situations and, overall, lack a broad and comprehensive well-founded system notion, failing to deal with the emergence phenomenon.

8 Final Remarks

The complexity of the systems in society is increasing considerably as a consequence of technological development. It has given rise to new kinds of more complex and diverse systems. In this context, ontologies are crucial to a better understanding of these systems. To address this issue, this work aimed at proposing a well-founded *system ontology* based on Unified Foundational Ontology [20]. This ontology was proposed based on GST principles, allowing the broad representation of the distinct kinds of socio-technical systems including their composition, function, structure, and properties.The major implication for Enterprise Architecture of this ontology is to provide guidelines for capabilities emergence modeling and emergence pattern identification in EA notations (e.g., ArchiMate). This system ontology can also be used as a reference to integrate ontologies of distinct kinds of systems such as cyber-physical systems, system-of-systems, and digital twins, besides contributing to interoperability and data integration in different knowledge areas.

Future work can proposes a language pattern to OntoUML, the UFO-based UML extension for ontology modeling. In this context, the system ontology could be used to create ontological perspectives to represent the system composition, functional decomposition, system structure, system mechanism, system characterization, and variation over time. Based on the OntoUML for system modeling, a language pattern could also be proposed in ArchiMate to improve the representation of emergence, levels, variation over time, and structural aspects. With this, GST-based modeling notations could be integrated into ArchiMate, such as the Causal Loops diagram [32]. This integration could help better understand the systemic aspects of an organization in a practical way. An important computational application is apply the system ontology to support the use of pattern recognition tecniques to identify emergence patterns from organizational data, specially in complex networks models. Another future work could be the *system ontology* extension to include system behavior and variation over type to better understand how system capabilities manifest through events and how they evolve. In this context, the relations among dispositions (and also capabilities), mentioned in this work, could be more detailed. This ontology could help in system capabilities detailing and digital requirements specification in complex contexts, which involve different kinds of systems, such as system-of-systems, cyber-physical systems, or digital twins. In this context, an implementation of the system ontology in OWL would be usefull in semantic web applications in these kind of system, such as semantic annotation and interoperability. Finally, based on the system capability, other future work is to improve the Competence Ontology [14] to represent better the emergence phenomenon. In this case, the ontology could be used to represent the emergence of organizational capabilities from personal competencies. Besides emergence, another possible improvement concerns the representation of competence development over time.

Acknowledgment. This study was supported in part by CNPq (313687/2020-0), FAPES (281/2021, 1022/2022), and the DSYNE INTPART network (Research Council of Norway project number 309404).

References

1. Ackoff, R.L.: Towards a system of systems concepts. Management Science **17**(11), 661–671 (Jul 1971)
2. Aquino, N.M.R.: A smart assessment of business processes for enterprises decision support. Ph.D. thesis, Université de Lorraine (2021)
3. Azevedo, C.L.B., et al.: An Ontology-Based Well-Founded Proposal for Modeling Resources and Capabilities in ArchiMate. In: 17th IEEE International EDOC Conference (EDOC 2013). pp. 39–48. IEEE Computer Society Press (2013)
4. Bäcklander, G.: Doing complexity leadership theory: How agile coaches at spotify practise enabling leadership. Creativity and Innovation Management **28**(1), 42–60 (Jan 2019)
5. Baek, Y.M., et al.: A meta-model for representing system-of-systems ontologies. In: Drira, K., Oquendo, F. (eds.) 6th IEEE/ACM International Workshop on Software Engineering for Systems-of-Systems, SESoS@ICSE 2018, Gothenburg, Sweden, May 29, 2018. pp. 1–7. ACM (2018)
6. Bakirtzis, G., et al.: An ontological metamodel for cyber-physical system safety, security, and resilience coengineering. Software and Systems Modeling **21**(1), 113–137 (Jun 2021)
7. Barton, A., et al.: A taxonomy of disposition-parthood. In: Workshop on Foundational Ontology in Joint Ontology Workshops: JOWO 2017. vol. 2050, pp. 1–10. CEUR-WS: Workshop proceedings (2017)
8. Bernus, P.: Enterprise models for enterprise architecture and ISO9000:2000. Annual Reviews in Control **27**(2), 211–220 (Jan 2003)
9. von Bertalanffy, L.: General system theory. George Braziller, New York (1968)
10. Bunge, M.: Treatise on Basic Philosophy: The Furniture of the World. Ontology I. Reidel Pub. (1977)
11. Bunge, M.: Treatise on Basic Philosophy. Ontology II: A World of Systems. Springer Netherlands, Dordrecht, Netherlands (1979)
12. Bunge, M.: Emergence and convergence: Qualitative novelty and the unity of knowledge. University of Toronto Press (2003)

13. Calhau, R., et al.: Competences in ontology-based enterprise architecture modeling: Zooming in and out. International Journal on Software and Systems Modeling (2023), https://vixra.org/abs/2306.0052, under review
14. Calhau, R.F., et al.: Towards Ontology-based Competence Modeling in Enterprise Architecture. In: 2021 IEEE 25th International Enterprise Distributed Object Computing Conference (EDOC). IEEE (Oct 2021)
15. Compagno, F., et al.: Towards a formal ontology of engineering functions, behaviours, and capabilities. Semantic Web Journal (forthcoming)
16. Dori, D., et al.: What is a system? an ontological framework. Syst. Eng. **20**(3), 207–219 (2017)
17. EMERY, F.: Socio-technical systems. Systems Thinking (1969)
18. Fine, K.: Things and their parts. Midwest Studies in Philosophy **23**, 61–74 (1999)
19. Galton, A., et al.: Dispositions and the infectious disease ontology. In: Formal Ontology in Information Systems: Proceedings of the Sixth International Conference (FOIS 2010). vol. 209, p. 400. Ios Press (2010)
20. Guizzardi, G.: Ontological Foundations for Structural Conceptual Models. No. 15 in Telematica Institute Fundamental Research Series, Telematica Instituut, Enschede, The Netherlands (2005)
21. Juarrero, A.: Dynamics in action: Intentional behavior as a complex system. Emergence **2**(2), 24–57 (2000)
22. Leal, G., et al.: An ontology for interoperability assessment: A systemic approach. Journal of Industrial Information Integration **16**, 100100 (Dec 2019)
23. Lukyanenko, R., et al.: Foundations of information technology based on bunge's systemist philosophy of reality. Software and Systems Modeling **20**(4), 921–938 (jan 2021)
24. Merrell, E., et al.: Capabilities (2022)
25. Morbach, J., et al.: Onto CAPE 2.0 —a (re-)usable ontology for computer-aided process engineering. In: Computer Aided Chemical Engineering, pp. 991–996. Elsevier (2008)
26. Mossio, M., et al.: Emergence, closure and inter-level causation in biological systems. Erkenntnis **78**(S2), 153–178 (2013)
27. Naudet, Y., et al.: Towards a systemic formalisation of interoperability. Computers in Industry **61**(2), 176–185 (feb 2010)
28. O'Connor, T.: Emergent properties. American Philosophical Quarterly **31**(2), 91–104 (1994)
29. Simons, P.M.: Parts: A study in ontology (1987)
30. Skyttner, L.: General systems theory: ideas & applications. World Scientific (2001)
31. Spencer-Smith, R.: Reductionism and emergent properties. In: Proceedings of the Aristotelian Society. pp. 113–129. JSTOR (1995)
32. Sterman, J.: Business dynamics. Irwin/McGraw-Hill c2000.. (2010)
33. Wand, Y., et al.: On the deep structure of information systems. Information Systems Journal **5**(3), 203–223 (Jul 1995)
34. Wegmann, A.: On the systemic enterprise architecture methodology (seam). pp. 483–490 (2003)
35. Yilma, B.A., et al.: Systemic formalisation of cyber-physical-social system (CPSS): A systematic literature review. Comput. Ind. **129**, 103458 (2021)