Applicability of a Foundational Ontology to Semantically Enrich the Core and Domain Ontologies

Luis Olsina[a]
GIDIS_Web, Engineering School, UNLPam, General Pico, LP, Argentina

Keywords: Thing Ontology, Foundational Ontology, Core Ontology, Domain Ontology, Ontological Architecture.

Abstract: This paper analyses the key terms, relationships and axioms of ThingFO (Thing Foundational Ontology), which is an ontology devoted for particular and universal things and assertions. It is placed at the foundational level in the context of a five-tier ontological architecture. This architecture groups together foundational, core, top-domain, low-domain, and instance levels, making ThingFO the single ontology at the top level. Thus, the ontologies at lower levels reuse and specialize, for example, its terms and relationships. To illustrate the applicability of ThingFO, this work also discusses enriched terms and specialized relationships for a core ontology, particularly for situation, where its concepts are themselves cross-cutting concerns for different domain terminologies. In addition, verification and validation issues are addressed as well.

1 INTRODUCTION

A foundational ontology –also known as top-level or upper– is independent of any domain. It is at the highest level of reference terminologies useful for the sciences. Even core ontologies such as situation, event, or process are domain-independent reference terminologies, but they can semantically extend or reuse foundational elements. Note that domain-dependent ontologies are the most massive terminologies built to date, such as for software, health, or mechanic. But most of the existing domain ontologies are not based on core and/or foundational terminologies. Or, if they do, there is often no clear separation of concerns considering ontological levels. As indicated in (Horsch et al., 2020), top-level ontologies are becoming increasingly important for integrating heterogeneous knowledge bases coming from different sources and domains of sciences.

The main reason for adopting, adapting or creating an upper ontology should be that it has a minimum set of particular and universal concepts of the described world so that they can be reused accordingly across domains. As a consequence, a large number of lower-level ontologies can fall under the umbrella of such a top-level ontology.

Considering the endeavor of developing ontologies, (Schneider, 2003) indicates that most knowledge engineers are unaware of the challenges of building an upper ontology, because it involves issues that are unusual for the practice of representing concrete knowledge for specific domains. Thus, to build an upper ontology a transdisciplinary knowledge is required not only in various areas of Information Systems and Artificial Intelligence, but also in Cognitive Sciences and Philosophy.

In fact, although thousands of domain ontologies have been developed so far, only fewer than a dozen well-known upper ontologies have been built in the last three decades, such as Cyc (Lenat et al., 1990), BFO (Arp et al., 2015), DOLCE (Masolo et al., 2002), PROTON (Casellas et al., 2005), GFO (Herre, 2010), SUMO (Pease, 2011), and UFO (Guizzardi, 2005).

In the light of these efforts, this work discuss ThingFO, which is a foundational ontology for particular and universal things and assertions placed at the highest level in the context of a five-tier ontological architecture called FCD-OntoArch (Foundational, Core, Domain, and instance Ontological Architecture). Why the need to build another foundational ontology is also justified later.

The main contribution of this paper is to analyze key features of ThingFO in the context of the above mentioned architecture. It also illustrates the applicability of ThingFO to enrich concepts of a core

[a] https://orcid.org/0000-0001-8969-4376

Olsina, L.
Applicability of a Foundational Ontology to Semantically Enrich the Core and Domain Ontologies.
DOI: 10.5220/0010641000003064
In *Proceedings of the 13th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2021) - Volume 2: KEOD*, pages 111-119
ISBN: 978-989-758-533-3; ISSN: 2184-3228

ontology, where its concepts are themselves cross-cutting concerns for domain ontologies.

Note that the first version (v1.0) of ThingFO was published at a national conference (Olsina, 2020). Since then the current version (v1.2) has been validated with external experts, which allowed adding, for example, new non-taxonomic relationships, properties and three axioms, among other improvements. Additionally, this work deals with verification aspects, which were not covered before.

The rest of the paper is organized as follows: Section 2 provides an overview of FCD-OntoArch, in which ThingFO and lower-level ontologies are placed. Section 3 discusses the elements of ThingFO. Section 4 illustrates the usefulness of ThingFO for enriching terms and relationships of a couple of lower-level ontologies. Section 5 provides a discussion, and Section 6 summarizes conclusions.

2 ThingFO AND ITS CONTEXT

As mentioned above, ThingFO is placed at the foundational level into the FCD-OntoArch architecture. Figure 1 depicts its five tiers, which entails foundational, core, top-domain, low-domain and instance levels. Each level is populated with ontological components, i.e., ontologies.

Figure 1 also shows that ontologies at the same level can be related to each other, except for the Foundational Ontological Level where only ThingFO is found. Additionally, ontologies' terms and relationships at lower levels can be semantically enriched or specialized by ontologies' terms and relationships from higher levels. Since ThingFO is at the highest level, ontologies at lower levels benefit from reusing and specializing its concepts.

ThingFO has three key terms namely Thing, Thing Category and Assertion that semantically enrich terms of components at lower levels. For example, TestTDO, a software testing ontology placed at the top-domain ontological level is enriched by concepts of SituationCO and ProcessCO placed at the core ontological level. In turn, both are enriched by the abovementioned terms of ThingFO.

The concepts of ThingFO are independent of any domain. From top to bottom, the next level is called Core Ontological Level. Ontologies such as ProcessCO, GoalCO, SituationCO and PEventCO are located at this level, among others not shown in the figure such as ProjectCO. Their concepts are also independent of any domain, but they are closer to different domains. For example, the term Activity in ProcessCO (Becker et al., 2021) is specialized in each domain accordingly. Thus, we have specific-domain Activities for measurement, testing, or development. On the other hand, ProcessCO includes terms with the semantics of Thing such as Work Entity (Work Process, Activity, Task), or Artefact. It is important to remark that ontological components at the same level may reuse terms with each other entirely.

Looking at Figure 1, the next level is called Top-Domain Ontological Level. Ontologies such as TestTDO (Tebes et al., 2020), FRsTDO (FRs stands for Functional Requirements), NFRsTDO (NFRs stands for Non-Functional Requirements), and MEvalTDO (MEval stands for Measurement and Evaluation) are located at this level, among others not shown in the figure. Note that the terminological coverage of a top-domain ontology can serve as the basis for the development of low-level (more specific) domain ontologies. For instance, at the Low-domain Ontological Level, the MetricsLDO and IndicatorsLDO components are depicted, but as the reader may surmise, many others can be conceived at this level.

Lastly, at the Instance Ontological Level, we can place ontologies such as instances of units (UnitIO in Figure 1), instances of quality characteristics, to name just a few.

Therefore, the described multitier architecture promotes a clear separation of concerns by considering the ontological levels and allotting built ontologies in the right place. This also fosters the modularity, extensibility and reuse of ontological elements throughout the levels.

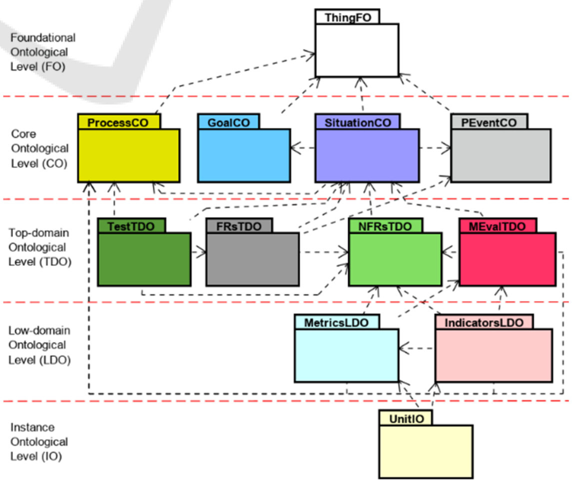

Figure 1: The five-tier ontological architecture named FCD-OntoArch, where ontological components are placed. ThingFO and ontological components at the core level are domain independent. Note that the figure does not depict all the developed components to date.

3 REPRESENTING ThingFO

As previously mentioned, building a foundational ontology requires a transdisciplinary knowledge. This is so because we are dealing with mental representations of human agents (subjects as Things), who explicitly make claims about the essentials of Things (objects) and their invisible links between them in particular and universal situations of the world. To put it elegantly, to represent a top-level ontology, the eyes of the subject's mind must be at the highest level.

Developing an upper ontology involves challenges that are unusual for the common practice of knowledge representation (Schneider, 2003). On one hand, the need for descriptive adequacy requires a considerable subtlety of conceptual analysis based on sound philosophical and cognitive grounds. On the other hand, the usefulness of an upper ontology depends on the greatest possible formal simplicity and transparency, as well as the completeness and conciseness of the elements included.

Foundational ontologies are representations about domain-independent top-level primitive constructs such as thing, property, power, relations, thing category, as well as assertions that deal with them. Hence, the main goal and requirement to conceive an upper ontology is to have a minimum set of particular and universal concepts of the target world, that is, key terms, properties, relationships and constraints that represent the world so that they can be reused and specialized, and ultimately can be useful and easy to adopt or adapt across all domains of the different sciences.

Therefore, in sub-section 3.1, we first discuss the terms Thing, Property and Power and their main relationships. Then, in sub-section 3.2, we analyse the terms Thing Category and Assertion. Particularly, we discuss types of Assertions that a human agent can formulate about things and categories. Note that we describe the ThingFO conceptualization of Figure 2 using the following text convention: ontology terms begin with capital letters, properties are italicized, and relationships are underlined.

The reader can check the entire ThingFO documentation at http://bit.ly/ThingFO.

3.1 Thing, Property and Power

The term Thing represents a particular, tangible or intangible object of a given particular world, but not a universal category, which is modelled by the term Thing Category. A particular object or entity represents and implies unique individuals or instances. Therefore, a particular Thing results in instances, whereas a universal Thing, i.e., a Thing Category does not result in instances, at least with valuable meaning of individual.

A Thing is not a particular object without its Properties and its Powers, so "things, properties and powers all emerge simultaneously to form a unity" […] "Things, properties and powers are necessary and sufficient for the existence of this unity" (Fleetwood, 2009). Moreover, a Thing cannot exist or be in spatiotemporal isolation from other Things. This principle of non-isolation is represented among Things in Figure 2 through the relationship <u>relates with</u>, in which the cardinality is at least one.

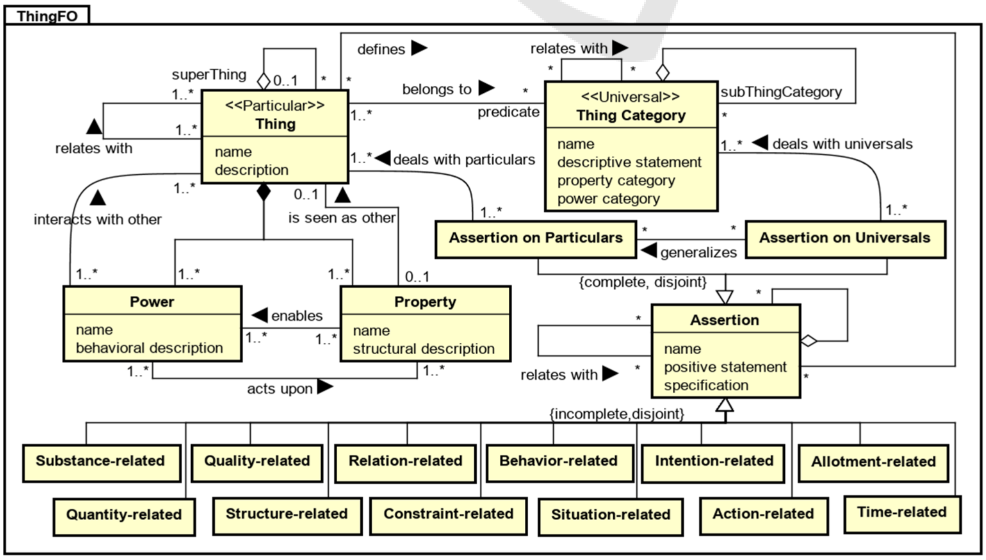

Figure 2: UML diagram of the terms, properties and relationships of the Thing Foundational Ontology (ThingFO).

Thus, in a particular situation of the represented world, a Thing (or many) in the role of the `target` is always surrounded by other Things in the role of the `environment`. This is modeled in SituationCO by including the terms Target Entity and Context Entity, as we will exemplify in sub-section 4.1. Note, however, under the principles of simplicity and conciseness, we tried to delegate most of the responsibilities to the core ontologies so as not to overload ThingFO with derivable terms, relationships and axioms. This lack of conciseness often occurs in other related work, as we will discuss in Section 5.

Property has a *structural description* that refers to the intrinsic constitution, structure, or parts of a particular Thing, whereas Power has a *behavioral description* that refers to what a particular Thing does, can do or behave. Thus, the *behavioral description* portrays the Power of a Thing in terms of responsibilities, operations or actions.

According to (Fleetwood, 2009) "Powers are the way of acting of a things' properties; powers are a things' properties in action". Also, he states that "Things have properties, these properties instantiate [...] acting powers, and this ensemble of things, properties and powers cause any events that might occur". These Fleetwood's statements are represented in the following relationships. One or more Properties enable one or more Powers. In turn, Powers act upon Properties, as well as can interact with other Things. For ThingFO to be actionable at lower levels, three axioms were defined, which were not available in (Olsina, 2020). They are specified in first-order logic in the linked documentation referenced above, so only the textual description follows:

- All Property of a Thing enables only its Powers;
- The Power of a Thing only acts upon its Properties;
- The Power of a Thing only interacts with other Things.

Powers and Properties are two members of the triad that conform the particular entity named Thing. Hence, there is no Power or energy alone floating in the air that can be dissociated from a Thing. Lastly, it is important to note that a Property, which is a member of the triad that makes up a given Thing, most of the time, is seen as other particular Thing outside of it, with its own Properties and Powers.

3.2 Thing Category and Assertions

Thing Category represents a universal of a set of particulars conceived by the human being's mind for classification purposes. Whereas a Thing represents a concrete object or entity, which implies unique instances, a Category of Thing represents an abstract or universal entity in which the instances do not have valuable meaning of individual. Therefore, a Thing Category does not exist, is or can be in a given particular world as a Thing does. On the contrary, it can only be mentally formed or developed by human beings as an abstract or generic construct, which in turn, hierarchies of sub-categories can be developed.

Ultimately, a Thing Category predicates on related particular objects. That is, it `predicates` on the common essence of Things which, therefore, belong to the intended Category of Thing.

Lastly, the third key term in ThingFO is Assertion. This construct has a great conceptual impact when a human agent intentionally represents and models particular and universal Things and situations of the world in question.

Assertion is defined as "A positive and explicit statement that somebody makes about something concerning Things, or their categories, based on thoughts, perceptions, facts, intuitions, intentions, and/or beliefs that is conceived with an attempt at furnishing current or subsequent evidence".

Regarding a particular or universal, a *positive statement* refers to what it is, was, or will be. Therefore, it contains no indication of approval (e.g. I like it) or disapproval. Assertions are conceptualized consequences of persons' mental models of the represented world, phenomenon, or situation at hand.

Considering the part of the previous phrase that indicates "*...statement that somebody makes about ...*" means that for instance a concrete human being – as a particular Thing- defines or conceives Assertions. And considering the part of the same phrase that indicates "*…about something concerning Things…*" means, for example, about the substance, structure, behavior, situation, quantity, quality, among other aspects of Things or Thing Categories.

To be valuable, actionable and ultimately useful for any science, an Assertion should largely be verified and/or validated by theoretical and/or empirical evidence. Assertions can be represented by informal, semiformal or formal specification languages. Thus, a *specification* can include text in natural language, mathematical or logical expressions, well-formed models and diagrams, among other representations.

There is only Assertion on Particulars for Things and Assertion on Universals for Thing Categories. Notice the constraint with the label `{complete, disjoint}` set in the inheritance relationship in Figure 2. Hence, Assertion on Particulars deals with particulars (Things), whereas Assertion on Universals deals with universals (Thing Categories).

In addition, Figure 2 shows 12 types of Assertions that allow specifying and representing the substance, situation, relations, structure, behavior, intention, quantity and quality, among other aspects related to Things and Thing Categories. Notice also the constraint with the label `{incomplete, disjoint}` set in the inheritance relationship.

In the sequel, we describe a subset of types of Assertions for space limits. The reader can see all the term definitions at http://bit.ly/ThingFO. For the sake of illustration, the reader can realize that a conceptualization of an ontology as an artefact (e.g., the ThingFO UML diagram in Figure 2, plus the linked document with the definition of terms, properties, and non-taxonomic relationships as well as specifications of axioms) represents a combination of Substance-, Relation-, Structure-, Intention-, Situation- and Constraint-related Assertions.

A Substance-related Assertion is related to the ontological significance and essential import of a Thing as a whole entity, or a set of Things. Substance aspects can be specified for Particulars and can also be abstracted for Universals.

A Relation-related Assertion refers to logical or natural associations between two or more Things and their categories. As abovementioned, a Thing cannot exist or be in spatiotemporal isolation from other Things in a given particular world. Therefore, a Thing is related to other Things. Also, it can be specified for both Particulars and Universals.

A Structure-related Assertion is related to the Property term, which represents the intrinsic constitution, structure, or parts of a Thing. Structural aspects can be specified for Particulars and can also be abstracted for Universals.

An Intention-related Assertion is related to the aim to be achieved by some agent. The statement of an Intention-related Assertion considers the propositional content of a goal purpose in a given situation and time frame. It can be specified for both Particulars and Universals.

A Situation-related Assertion is related to the combination of circumstances, episodes, and relationships/events between target Things and context entities that surround them, or their categories, which is of interest or meaningful to be represented or modeled by an intended agent. A Situation can be represented statically or dynamically depending on the intention of the agent. The conceptualization of an ontology embraces a static representation. It can be specified for Particulars and generalized for Universals.

Finally, a Constraint-related Assertion is related to the specification of restrictions or conditions imposed to Things, Properties, relationships, interactions or Thing Categories that must be satisfied or evaluated to true in given situations or events. It can be specified for both Particulars and Universals.

4 ThingFO APPLICABILITY

To analyse its applicability and usefulness, the present work illustrates semantically enriched terms and specializations of non-taxonomic relationships for an ontology at the core level like SituationCO. Its concepts (along with ProcessCO) are themselves cross-cutting concerns primarily for domain terminologies. Furthermore, we showcase the utility of ThingFO together with these core ontologies, by which domain ontologies reuse some of their conceptual blocks or patterns.

It is worth mentioning that this work uses stereotypes as a mechanism to semantically enrich the terms. Stereotypes are, in some cases, a more suitable mechanism that inheritance relationships, since they generate a loose coupling between a lower-level component and a higher-level component. Also, stereotypes can reduce the complexity of the model, promoting comprehensibility and communicability.

Next, in sub-section 4.1, we describe how some ThingFO terms are stereotyped in SituationCO. Also, we see how ThingFO relationships are specialized. To do this, we address aspects of the SituationCO vs. ThingFO non-taxonomic relationship verification matrix. Then, in sub-section 4.2, we briefly point out how these foundational and core concepts and patterns are extended or reused by domain ontologies.

It is important to note that we are not going to discuss the content of the SituationCO ontology, but rather the enrichment and reuse mechanism of some of its terms and relationships. The reader interested in SituationCO can consult all the documentation at http://bit.ly/SituationCO.

4.1 SituationCO Reuses ThingFO

Figure 3 depicts a fragment of the conceptualization of SituationCO with most of its elements reused from ThingFO. SituationCO is placed at the Core Ontological Level in the context of FCD-OntoArch (Figure 1). This ontology, which mainly deals with Particular and Generic Situations for a given Goal and problem at hand, was developed rather recently.

Situation is defined as a Situation-related Assertion that explicitly states and specifies the combination of circumstances, episodes and relationships/events embracing particular entities and their surroundings, or categories of entities and their related generic context, which is of interest and relevant to be represented by a Human Agent/Organization with an established Goal. Its concepts primarily extend from ThingFO, and it also borrows some core concepts from the GoalCO, ProcessCO and ProjectCO components.

The term Thing semantically enriches the terms Target Entity and Context Entity, in addition to the completely reused terms of the core components mentioned above, such as Project, Organization and Human Agent. The term Thing Category semantically enriches the term Entity Category and Context Category. So concrete Context Entities pertain to category Context Category, as seen in Figure 3.

Besides, the term Situation has the semantics of Situation-related Assertion. In turn, a Particular Situation has the semantics of Assertion on Particulars, while a Generic Situation of Assertion on Universals. Also, Goal –term reused from GoalCO- has the semantics of Intention-related Assertion. For the sake of a summary, a Human Agent/Organization conceives /establishes a Goal that implies a Situation, which is represented by a Situation Model. A Project operationalizes Goals and specifies a Situation.

Note that non-taxonomic relationships were verified with those of ThingFO. Table 1 represents an excerpt from ThingFO's non-taxonomic relationships that SituationCO specializes. Table 2 shows the resulting correspondence. The complete non-taxonomic relationship verification matrix is found in the end of the document at http://bit.ly/SituationCO.

Table 1: An excerpt from ThingFO's non-taxonomic relationships, which SituationCO specializes in Table 2.

ThingFO Term 1	Relationship	ThingFO Term2
Assertion on Particulars	deals with particulars	Thing
Thing	relates with	Thing
Thing	defines	Assertion
Assertion	relates with	Assertion
Thing	belongs to	Thing Category

Table 2: Correspondence of SituationCO (SCO)'s non-taxonomic relationships with those of ThingFO in Table 1.

SCO Term 1	Relationship	SCO Term 2
Particular Situation	deals with target	Target Entity
Target Entity	is surrounded by	Context Entity
Human Agent	conceives	Goal
Goal	implies	Situation
Context Entity	pertains to category	Context Category

4.2 Benefits at the Domain Level

As mentioned above, foundational and core terms, relationships, and conceptual blocks are reused and extended by domain ontologies. For example, TestTDO is a top-domain ontology for software testing activities, methods and projects, which is terminologically benefited from ThingFO, SituationCO and ProcessCO. For space reasons, we focus only on a few conceptual blocks. The comments below also apply to some other domain ontologies.

The term Particular Situation (Figure 3) enriches the term Test Particular Situation. Hence, TestTDO specializes the Particular Situation pattern including the three terms renamed Test Particular Situation, Testable Entity and Test Context Entity. SituationCO relationships such as deals with target/environment, is surrounded by and influences are also mirrored in the Test Particular Situation conceptual block.

Furthermore, the Project pattern is also reflected in TestTDO, in which Test Project operationalizes the Test Goal and specifies the Test Particular Situation.

Lastly, TestTDO extends from ProcessCO, the consumes/produces pattern (Becker et al., 2021).

5 DISCUSSION

This Section summarizes related work and provides a discussion on upper ontologies. As pointed out in the Introduction Section, although thousands of domain ontologies have been developed so far, only fewer than a dozen known upper ontologies have been built in the last three decades, with somewhat impact.

(Mascardi et al., 2006) provide a description and comparison of 7 upper ontologies, namely: BFO, Cyc, DOLCE, GFO, PROTON, SUMO and Sowa (Sowa, 2005), which were the most referenced within the research community at the time of their study. To summarize the information, they have designed a template with fields like: Status of this description; Home page; Developers; Description; History; Dimensions; Modularity; Applications; among others. In addition, they also provide a summary of existing comparisons drawn among subsets of the top 7 cited ontologies previously made by other authors.

Besides, (Guizzardi, 2005) developed UFO. It is made up of three ontologies: UFO-A (endurants), UFO-B (perdurants, or events) and UFO-C (social entities, built on top of UFO-A and B). This ontology was not preselected in the Mascardi et al.'s comparison surely for chronological reasons.

Another contemporary initiative is COSMO, which is an upper ontology that can serve to enable broad general semantic interoperability. COSMO development started as a merger of basic elements from Cyc, SUMO, and DOLCE adding new features

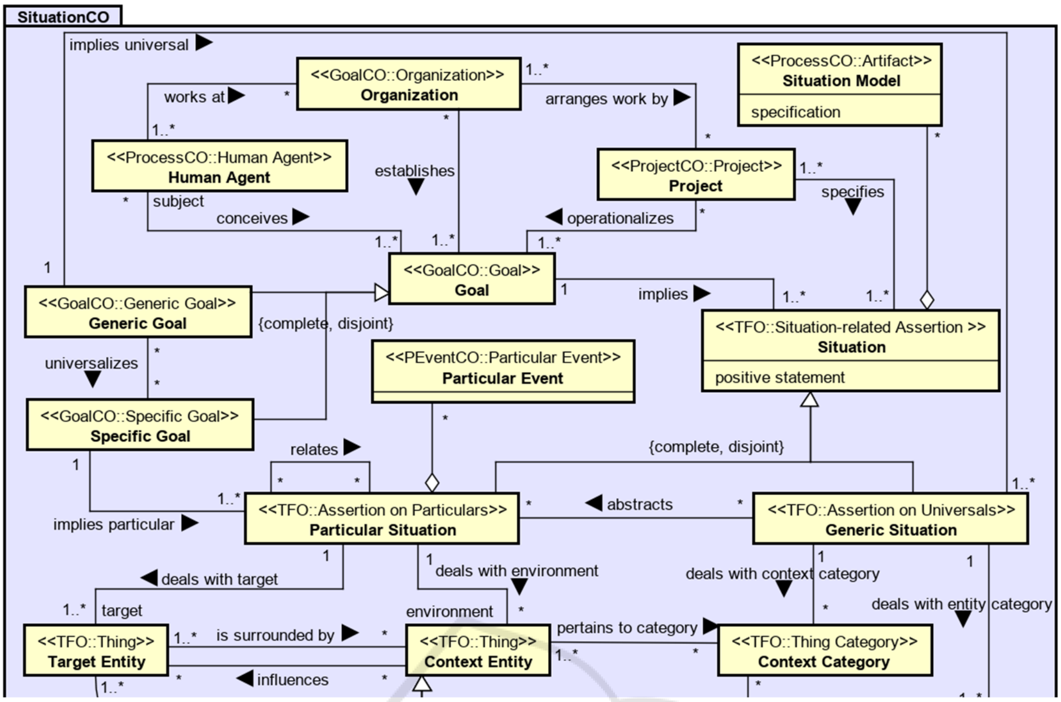

Figure 3: Excerpt from the SituationCO ontology with terms, properties and relationships enriched from ThingFO (TFO).

as well. Note that all the documentation of this open project can be accessed at micra.com/COSMO/. Its current OWL version has 24,059 types (classes), over 1,300 properties, and over 21,000 restrictions.

Some principles and quality criteria that benefit the understandability, usefulness and potential adoption of upper ontologies that guided the ThingFO construction process are, namely: formal simplicity and transparency promoting also the use of graphical representations for the conceptualization; coverage completeness but, at the same time, conciseness and self-intuitiveness of the elements included; balanced representation of both taxonomic and non-taxonomic relationships; and, under the principle of modularity and loose coupling, a clear delegation of responsibilities to core ontologies. Some of these quality criteria are in (D'Aquin and Gangemi, 2011).

In brief, qualitatively analysing the quoted foundational ontologies, none of them simultaneously satisfy all the above criteria. As for the numbers, the smallest are Sowa (with 30 classes, 5 relationships and 30 axioms), and BFO (36 classes linked via the taxonomic relation is_a, which makes it a taxonomy rather than an ontology). While the Cyc figures are approximately 300,000 concepts, 3,000,000 assertions (facts and rules), and 15,000 relationships, including in these numbers micro-theories. COSMO numbers as mentioned above are also huge.

The foundational ontology that its conceptualization is best represented graphically is UFO, whereas most of the remainder use other formal logic-based representations that are not easy to convey and even to understand for many stakeholders. On the other hand, frequently, a clear delegation of responsibilities to core ontologies is not observed. For example, among the BFO 36 classes, are terms such as Process, Quality, Temporal region (ontology.buffalo.edu/bfo/BFO2.png) that ThingFO delegates to lower levels. Similarly, UFO is made up of three ontologies at the upper level; however, ThingFO is the single ontology at that level that delegates the Event and Time issues at the core level.

Considering the terms, there is often a lack of consensus on semantic matching. For example, the DOLCE distinction between "endurant" and "perdurant" does not fully correspond to that established in GFO. Moreover, COSMO's great effort began as a way to tackle the problem of semantic interoperability by merging basic elements of Cyc, SUMO and DOLCE, and adding new ones.

In building ThingFO, we have adhered to the principles and quality criteria stated above. Its three key terms are Thing (particular), Thing Category (universal) and Assertion, which are used in the ontologies at lower levels.

Lastly, it is worthy remarking that the types of Assertions shown in Figure 2 are not represented in this way in any of the quoted ontologies at the upper level. In summary, the figures for the current version (v1.2) of ThingFO are 19 defined terms, 13 defined properties, 3 specified axioms in first-order logic, and 12 defined non-taxonomic relationships that are well balanced with the taxonomic ones.

6 CONCLUSIONS

This work has analysed ThingFO and its applicability. It is an ontology for Particular and Universal Things placed at the foundational level regarding a five-tier ontological architecture. This multitier architecture promotes a clear separation of concerns by considering the ontological levels that allow the allocation of components accordingly. Since ThingFO is at the highest level, ontologies at lower levels benefit from reusing or specializing its three key terms, namely: Thing, Thing Category and Assertion. So the main aim is to have a large number of core and domain ontologies accessible under the umbrella of this foundational ontology.

In order to analyse its applicability, this work has illustrated the semantically enriched terms of the SituationCO ontology, where they are, in turn, cross-cutting concerns primarily for top-domain or low-domain terminologies of any science. Particularly, to show the applicability of ThingFO alongside this core ontology, we have also addressed the mechanism to not only enrich terms, but also to reuse and specialize relationships for a top-domain software testing ontology. Moreover, the non-taxonomic relationships of SituationCO were verified for their correct correspondence considering them as refinements (not as subsets) of those of ThingFO. This exercise allowed us to find a missing relationship (now called defines) between the terms Thing and Assertion.

Last but not least, ThingFO was validated by two external experts, outside the members of the present research group. Based on their recommendations, some changes were made to the conceptualization of ThingFO. In short, their comments gave us evidence of its potential utility. Moreover, the aforementioned external experts, working on discrete event simulation, after the validation effort, plan to adopt ThingFO and populate FCD-OntoArch with new ontologies for events and simulation. Ultimately, if, as a produced artefact, the ThingFO ontology were adopted step by step by the academia and industry, this will be a promising fact of its utility and validity.

As future work, we are going to quantitatively compare and evaluate the conceptualization of ThingFO with a set of preselected conceptualized foundational ontologies, considering the quality criteria mentioned in the Discussion Section and using the strategy illustrated in (Tebes et al., 2018).

REFERENCES

Arp, R., Smith, B., Spear, A., 2015. *Building Ontologies with Basic Formal Ontology*, MIT Press.

Becker, P., Papa, M.F., Tebes, G., Olsina, L., 2021. Analyzing a Process Core Ontology and its Usefulness for Different Domains, In *Springer book, CCIS (to appear): Int'l Conference on the Quality of Information and Communication Technology, QUATIC*, pp. 1–14.

Casellas, N., Blázquez, M., Kiryakov, A., Casanovas, P., Poblet, M., Benjamins, R., 2005. OPJK into PROTON: Legal domain ontology integration into an upper level ontology. In *3rd International Workshop on Regulatory Ontologies*, Springer, LNCS 3762. pp. 846–855.

D'Aquin, M., Gangemi A., 2011. Is there beauty in ontologies? Applied Ontology, 6:(3), pp. 165–175.

Fleetwood, S., 2009. The ontology of things, properties and powers. Journal of Critical Realism, 8:(3), pp. 343–366.

Guizzardi, G., 2005. Ontological foundations for structural conceptual models. PhD thesis, University of Twente, Enschede, The Netherlands, ISBN 90-75176-81-3.

Herre, H., 2010. *General formal ontology (GFO): A foundational ontology for conceptual modelling*. Theory and Applications of Ontology, vol. 2. Springer.

Horsch, T., Chiacchiera, S., Seaton, A., Todorov, T., Schembera, B., Klein, P., Konchakova, A., 2021. Pragmatic Interoperability and Translation of Industrial Engineering Problems into Modelling and Simulation Solutions. TR. No. 2020–A, 2nd Revised Version, doi: 10.5281/zenodo.4749106.

Lenat, D., Guha, R. V., 1990. CYC: A Midterm Report. AI Magazine, 11(3), 32–59.

Mascardi, V., Cordì, V., Rosso P., 2006. A Comparison of Upper Ontologies. Technical Report DISI-TR-06-21.

Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A., Schneider, L., 2002. The WonderWeb library of foundational ontologies (D17), Available at http://wonderweb.man.ac.uk/deliverables.shtml

Olsina, L., 2020. Analyzing the Usefulness of ThingFO as a Foundational Ontology for Sciences. In: *Argentine Symposium on Software Engineering, ASSE'20, 49 JAIIO*, CABA, Argentina, pp. 172–191.

Pease, A., 2011. *Ontology: A Practical Guide*. Articulate Software Press, Angwin, CA.

Schneider, L., 2003. How to Build a Foundational Ontology: The Object-Centered High-Level Reference Ontology, *A. Gunter et al. (Eds.): KI 2003, Springer, LNAI 2821*, pp. 120–134.

Sowa, J., 1999. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Brooks Cole Publishing.

Tebes, G., Peppino, D., Becker, P., Papa, M. F., Rivera, M.B., Olsina, L., 2018. *Family of Evaluation Strategies: A Practical Case for Comparing and*

Adopting Strengths, Journal of Computer Science & Technology, 18:(1), pp. 48–60.
Tebes, G., Olsina, L., Peppino, D., Becker, P., 2020. TestTDO: A Top-Domain Software Testing Ontology, In *Conferencia Iberoamericana de Software Engineering (CIbSE)*, *Curran Associates*, pp. 364–377.