Engineering Ontologies and Ontologies for Engineering

Giancarlo Guizzardi[1,2] and João Paulo A. Almeida[1]

[1] Ontology and Conceptual Modeling Research Group (NEMO), Federal University of Espírito Santo, Vitória, Brazil
{gguizzardi,jpalmeida}@inf.ufes.br

[2] Conceptual and Cognitive Modelling Research Group (CORE), Free University of Bozen-Bolzano, Bolzano, Italy
giancarlo.guizzardi@unibz.it

Abstract. This paper is written in honour of Ricardo de Almeida Falbo, on the occasion of his formal retirement. Some aspects of his career are discussed from two complementary directions. The first direction concerns his contributions from the area of ontologies to software engineering – helping to shape what is now known as Ontology-Driven Software Engineering. The second direction regards his contributions employing insights from Software Engineering to derive solutions to the area of Ontology Engineering.

Keywords: Pattern-Based Ontology Engineering · Ontology-Based Software Engineering

1 Introduction

Ricardo Falbo studied Mechanical Engineering, with graduate studies in Information Systems (System Analysis) and Industrial Engineering, as well as a master's degree in Environmental Engineering. Despite his formal training in engineering, his professional interests soon led him to the world of software. After working as systems developer, he initiated a career as a lecturer in Computer Science. He then went to pursue a PhD in Computer Science, in which he wanted to address the problem of knowledge integration in software engineering tools.

The engineering background has had a clear impact during his entire research and teaching career. Despite of his interest in theoretical work, he has always been obsessed with the development of practical engineering tools that practitioners could use for problem-solving. Simplicity and usability have always driven his involvement in development efforts.

During his PhD studies, he studied logics and knowledge representation, and was introduced by one of his supervisors (Crediné Menezes) to the work of the philosopher Mario Bunge [30]. At that time, he also came in contact with an emerging community of ontologies in computer science in the mid-1990s [45, 84]. From that point on, the backbone of his research career would develop in the interplay between ontologies and software engineering.

As an early ontologist, Falbo made seminal contributions to the area of what came later to be known as Ontology-Driven Software Engineering. These contributions, discussed in Section 2, include the creation of a network of ontologies capturing knowledge in a multitude on software engineering sub-domains. Moreover, by leveraging on these software engineering ontologies, he made contributions to developing methodological approaches for performing a number of software engineering activities, including a pioneering work in the area of Domain Engineering. Finally, still with the support of these ontologies, he led the development of a number of computational tools for supporting software engineering activities designed to support both automated reasoning as well as knowledge integration.

Falbo made several important contributions adapting mature techniques from Software Engineering to the then incipient area of Ontology Engineering. These contributions discussed in Section 3 include the SABiO method for ontology engineering, as well as computational tools developed for supporting the development of ontologies following that method. Furthermore, by adapting the notion of generic process models, he proposed a practical approach that revived the (by then, forgotten) notion of Task Ontologies. Finally, once more drawing on the software engineering tradition, he contributed to conceptual clarification work in the area of Ontology Design Patterns and proposed a seminal approach for developing Pattern Languages in this area.

After discussing these two streams of contributions, we present some final considerations in Section 4. Finally, Section 5 concludes the paper with some personal notes from the authors.

2 Ontology-Driven Software Engineering

2.1 Software Engineering as a Domain

Falbo was already a software engineer when he discovered the area of ontologies. For this reason, it was natural for him to treat Software Engineering as his first complex and vast domain of interest, and one whose knowledge should be explicitly captured as domain ontologies. Throughout his research career, he and his collaborators produced a multitude of domain and task ontologies addressing many sub-domains in Software Engineering. These include *Software Process* (appearing as early as [15], but later refined in [10], [53], and [69]); *Software Development Methodologies* [75]); *Software Requirements* [28,39,64,66], including *Run-Time Requirements* [38]; *Software Organizations* [23]; *Software Measurement* (including domain [25] and task ontologies [24]); *Software Configuration Management* (including domain [22] and task ontologies [32]); *Software Quality* [40], *Software Code and Design* [2,3], *Software Testing* [60,82], *Software Errors* [37], and *Software Project Management* [26,27].

This line of work later culminated on the *SEON (Software Engineering Ontology Network)* project (http://dev.nemo.inf.ufes.br/seon/). As indicated by the name, the idea of SEON is the creation of an integrated network of formal domain ontologies for supporting Knowledge Management in Software Engineering [74]. As depicted in Figure 1, the network is organized in layered structure in which a number of domain ontologies are created by extending Core Ontologies, which in turn are grounded in the UFO foundational ontology [49,55].

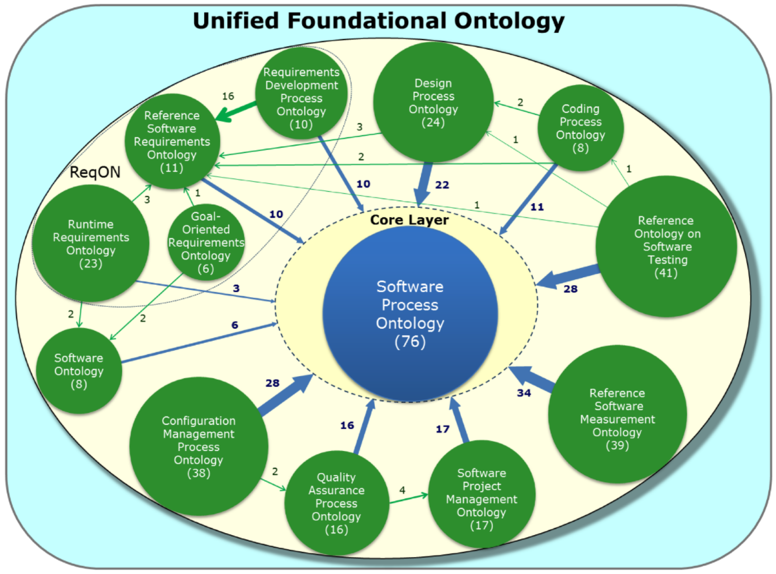

Fig. 1: The SEON Ontology Network [74].

2.2 Ontology-Based Methodological Support for Software Engineering Activities

In addition to his contribution developing knowledge artifacts for software engineering (see Section 2.1) and frequently building on these artifacts, Falbo made several methodological contributions to the area. These include ontology-based approaches for Requirements Engineering [14], for software documentation [26, 27], as well as for the semantic integration of Software Tools, including COTS (Components Off-the-Shelf) tools [31]. We focus here, however, on the contributions of Falbo to the topic of Domain Engineering.

In general, a domain engineering process is composed of the following subactivities: *domain analysis* and *domain design*, the latter being further decomposed in *infrastructure specification* and infrastructure *implementation*. Intuitively, domain engineering can be considered analogous to software application engineering, however, operating at a meta-level, i.e., instead of eliciting requirements, designing and implementing a specific application, the target is on a family of applications in a given domain [47, 67].

The product of a domain analysis phase is a *domain model*. A domain model defines objects, events and relations that capture similarities and regularities in a given domain of discourse. Moreover, it serves the purposes of a unified reference model to be used when ambiguities arise in discussions about the domain (communication), and a source of knowledge that can be used in a learning process about that domain. In summary, the specification produced by the domain modelling activity is *“a shared representation of entities that domain experts deem relevant in a universe of discourse, which can be used to promote problem-solving, communication, learning and reuse in a higher level of abstraction”* [21]).

As discussed in [49], the challenge in domain modeling is finding the best concepts that can be used to create representations of phenomena in a universe of discourse that are both as reusable as possible and still truthful to reality. The field, however, at the time was severely debilitated by a lack of concrete and consistent formal bases for making modeling decisions. Given the clear consonance between domain models in this context and a domain ontology, Falbo, Guizzardi and colleagues [11, 52] propose a domain engineering approach that could profit from the existing formal and theoretical maturity of the area of formal ontologies [48]. In this approach, they advocate the role of formal ontologies for software reuse and demonstrate how ontologies can support several tasks of a reuse-based software process. In order to support that process, they propose a new formalization for the LINGO language (see Section 3.1), and a constructive approach for deriving object-oriented frameworks from domain models (domain ontologies) represented in that language, but one which preserves the intended semantics of the original models. The framework derivation methodology proposed is supported by a spectrum of techniques, namely, mapping directives, design patterns, formal translation rules, and a framework using computational reflection mechanisms implementing the semantics of LINGO. In particular, in [52] they introduce a design pattern to preserve some ontological properties of part-whole relations (non-reflexivity, asymmetry, transitivity and shareability) in object-oriented implementations.

This approach was pioneer in introducing ontologies to domain engineering. It has been then employed for the creation of a number of object-oriented frameworks in many of the software engineering domains discussed in Section 2.1 (e.g., [11,13,52]). In fact, it has been employed for the construction and integration of basically all tools constituting the ODE Semantic Software Engineering Environment (see Section 2.3). However, besides being a method for Ontology-Based Domain Engineering, this was also an early method for Ontology Engineering, allowing the codification of ontologies in terms of object-oriented code. This Ontology Engineering aspect of the approach (including the aforementioned object-oriented code generation capabilities) was automated by the construction of the Ontology Development tool discussed in Section 3.2.

2.3 Ontology-Based Tools for Software Engineering

Another clear contribution of Ricardo Falbo to the area of Ontology-Based Software Engineering was formulated as as goal of *Semantic-Aware Model-Based Tools for Software Engineering* [13]. Falbo's idea was to approach software engineering as a domain (see Section 2.1), which can then be explicitly represented as a set of interconnected ontologies. By leveraging on these ontologies, one can, on one hand, (i) develop Software Engineering tools that can benefit from automated reasoning over these models; on the other hand, (ii) promote seamless integration of different software tools that are employed in software engineering processes.

In fact, since his PhD thesis [41], Falbo emphasized the idea that there can be no unique Software Engineering process suitable for developing all kinds of software in all kinds of contexts. To address that, he contributed to a meta-tool (called TABA, see Figure 2) that instantiates different software engineering tools each of which is suitable for different software engineering settings. This is done by leveraging on a set of ontologies for the software engineering domain [15].

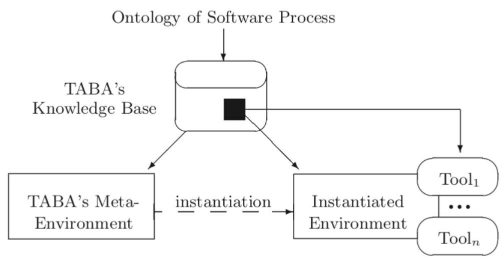

Fig. 2: The TABA meta-tool (from [15]).

After his PhD defense, Falbo founded in the Federal University of Espírito Santo a research group named LabES (Laboratory for Software Engineering). In that context, he took this project of a Semantic Software Engineering Environment (SSEE) to another level. In a long term project entitled ODE (Ontology-Based Software Development Environment) [20, 42, 77] (see Figure 3), Falbo and his group aimed at addressing the aforementioned challenges (i) and (ii), and produced an integrated suite of tools for software engineering to support ontology-based automated reasoning (in Prolog) and domain knowledge integration. ODE was designed as an ontology-based Knowledge Management environment for Software Engineering [7], supporting domain ontology creation and management, as well as ontology-based software engineering activities such as software process definition as well as quality control and documentation [40], software estimation [68], resource allocation, risk analysis [18, 33], human resource management [77], agent-based proactive knowledge dissemination [16], etc.

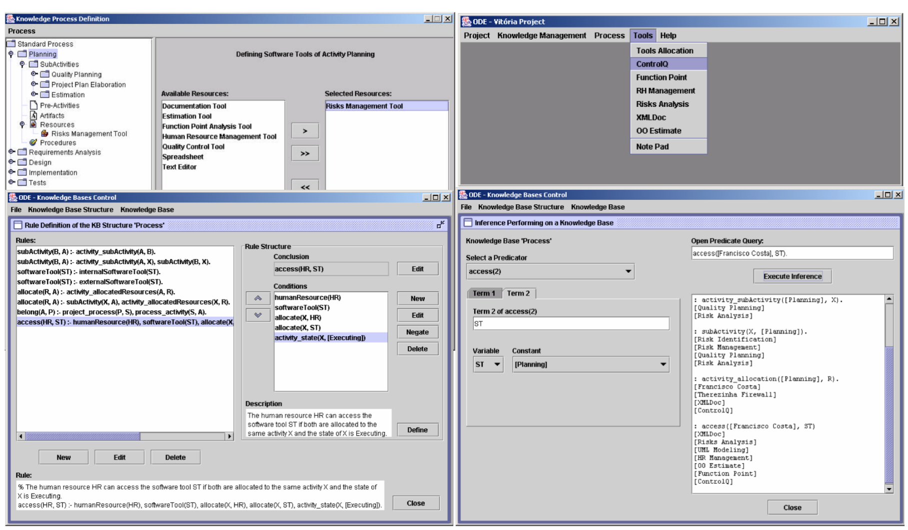

Fig. 3: The ODE Suite of Tools (from [77]).

3 An Engineering Approach to Ontology Development

3.1 A Systematic Method for Ontology Engineering

In the second half of the 1990s, we started to see the emergence of the first ontology engineering methodologies. These proposals were extrapolations from the experience of developing ontologies in specific domains. Examples included the TOVE methodology [44] and the METHONTOLOGY [43]. In that context, and drawing from their experience in building ontologies for the TABA Software Engineering Meta-Environment (see Section 2.3), Falbo and colleagues proposed an initial approach for systematizing the landscape of the state of the art in ontology engineering at the time [15]. This approach is illustrated in Figure 4.

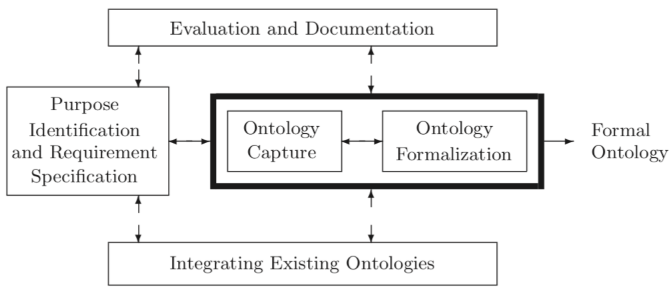

Fig. 4: An initial proposal of a method for Ontology Engineering (from [15]).

A distinctive feature of this approach, which differentiated it from other existing methods in the literature back then was the proposal of a visual language for ontology representation. Taking inspiration from the conceptual modeling literature, Falbo and colleagues propose LINGO [41][3], an *epistemological level* [50] diagrammatic language containing modeling primitives such as subtyping and parthood. Moreover, this work also proposes first-order logics rendering of these primitives such that the language can be used as a device for theory inclusion, i.e., whenever a language fragment is used, an equivalent first-order rendering of that fragment would be generated. The semantics of this language was advanced in [52] and [48]. The proposed set-theoretical semantics was formulated as to facilitate the translation from LINGO models to the Object-Oriented paradigm (see Section 2.2). Furthermore, by incorporating additional ontological distinctions in the set of modeling primitives of this language (e.g., those distinguishing types, roles and attributions), this approach later inspired the creation of a full-blown conceptual modeling language grounded in a foundational ontology, namely, OntoUML [49].

In fact, following his continuous interest in conceptual modeling and knowledge representation, Falbo contributed to the evolution of OntoUML [5, 78] as well its underlying foundational ontology UFO [56]. In particular, to the proposal of a version of

[3] Despite the meaning of "slang or jargon shared by a specific community", LINGO was also used as an abbreviation for *LINGuagem de Ontologias*, i.e., Language for Ontologies, in Portuguese. In [15], the authors use the English translation for LINGO (GLEO - Graphical Language for Expressing Ontologies).

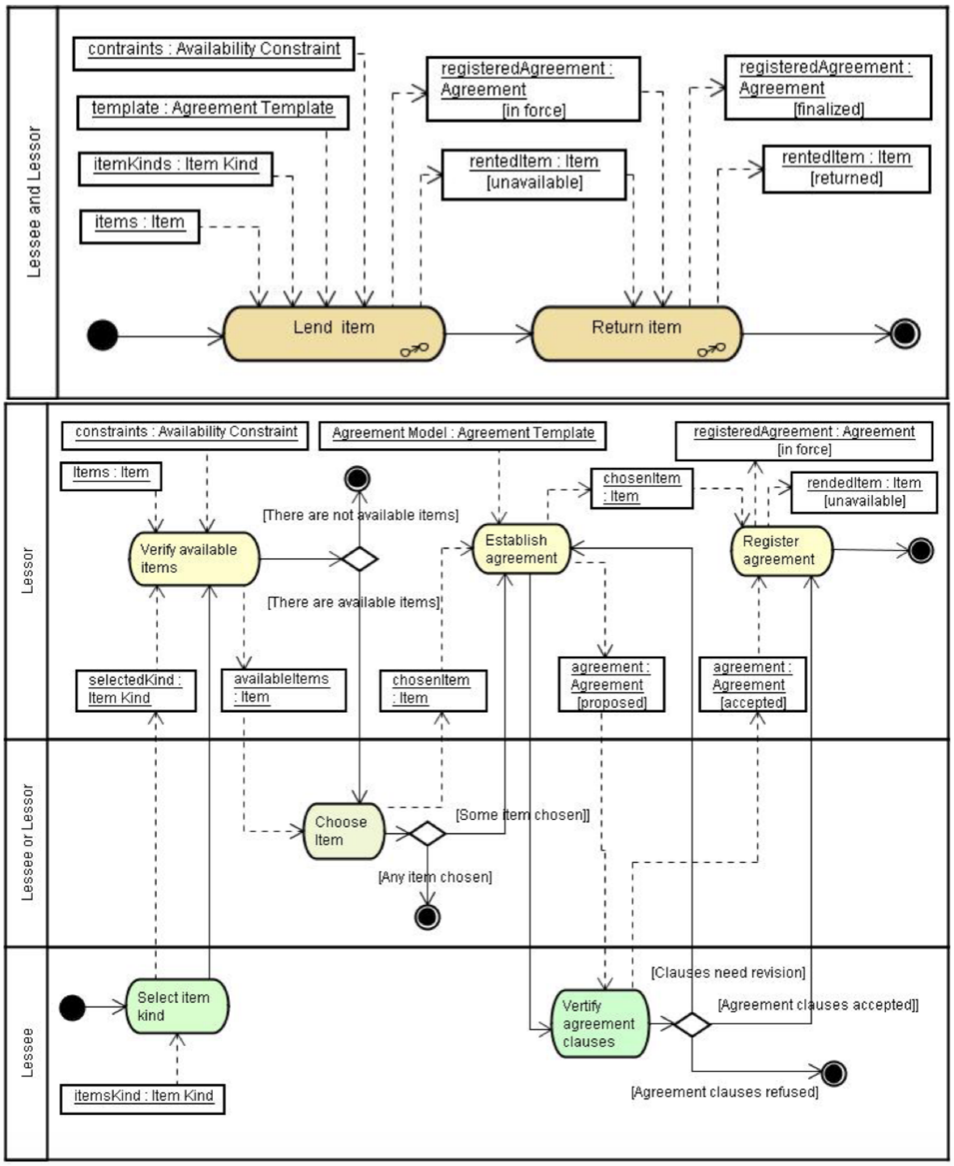

Fig. 5: Examples of a Task Ontology (from [62]).

OntoUML (dubbed E-OntoUML [61]) that was designed for the representation of the so-called Task Ontologies [46]. Task Ontologies were on a par with Domain Ontologies in early engineering methodologies. Besides, analogous to the way domain ontologies were intended to represent domain regularities independent of application, Task Ontologies were meant to represent a "generic task or activity (like diagnosing or selling), by specializing the terms introduced in the top-level ontology". Like in the case for domain ontologies, most ontology engineering approaches used logical languages for representing Task Ontologies. In fact, in earlier ontology engineering methods, Task Ontologies were associated with PSMs (Problem-Solving Methods) in Knowledge-Based Systems Engineering. However, once ontologies moved beyond that community, Task Ontologies were more and more neglected in the literature. By building on the literature of generic process models and even *workflow patterns* [36], Falbo contributes to proposing a version of UML Activity Diagrams, combined with OntoUML structural models for capturing generic and reusable task models [62]. Figure 5 depicts an example of a Task Ontology following Falbo's approach. At the top of the figure, we have the representation of a generic "Lending Task", with the refinement of the "Lend Item" sub-task in the lower part. Types such as *(chosen, rented) Item* or *Agreement* represent roles in these tasks that can be played by entities of multiple *kinds*, i.e., the so-called *role mixins* [49], and should be represented in a complementary OntoUML model.

Building on the experience acquired developing in a multitude of domains (e.g., see Section 2.1), Falbo then evolved his initially proposed method culminating in the Systematic Approach for Building Ontologies: SABiO[4] [6]. (See Figure 6 for an overview.)

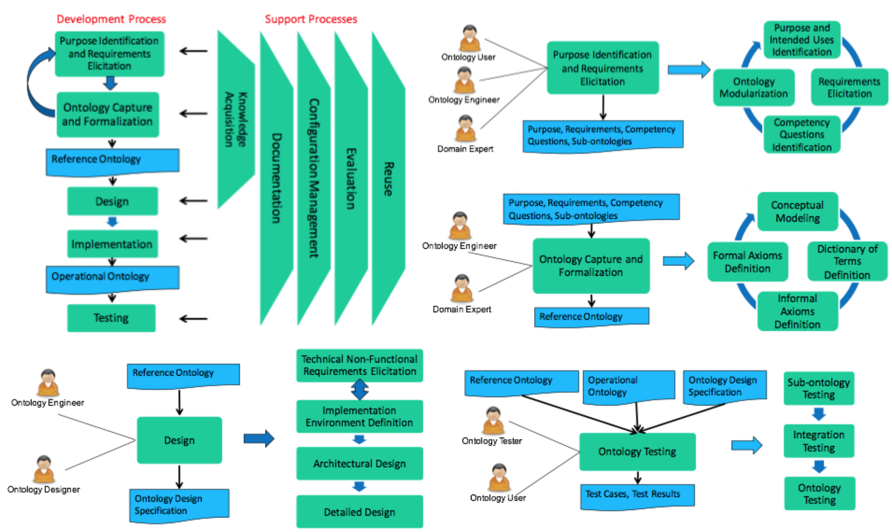

Fig. 6: An overall view of the SABiO methodology [6]).

[4] The term 'sábio' in portuguese means 'wise'.

Drawing on the methodological approach defended in [50] and later in [12], SABiO incorporates an explicit distinction between *Reference Ontologies* and *Operational Ontologies*: while the former are models created with the purpose of maximizing *domain appropriateness* (expressivity and truthfulness to the domain being represented) and *comprehensibility appropriateness* [49], the latter are codified versions of these designed with the purpose of addressing specific sets of non-functional requirements, architectural choices, and modeling languages. Another aspect of Falbo's work that influenced SABiO is his approach on Ontology Patterns as discussed in Section 3.3. Besides the development of several ontologies in software engineering (see Section 2.1), over the years, this methodology has been adopted in a number of initiatives for building ontologies in domains as diverse as aspect-orientation [58, 70], clinical reasoning [73], and knowledge explanability [29], among many others.

3.2 Software Engineering-inspired Tools for Ontology Engineering

Due to his engineering background, since his first attempts to propose an ontology engineering method (see Section 3.1), Falbo was aware of the importance of supporting such methods with computer-based tools. In [63, 83], he proposes a tool termed ODEd (Ontology Development Editor) (see Figure 7). ODEd was designed (i) to support the definition of concepts and relations, using graphical representations (based both on LINGO and on an initial UML profile for ontology representation); and (ii) to promote automatic generation of some classes of axioms from models created in these languages. Moreover, ODEd also supports the derivation of object-oriented frameworks from ontologies, following the method discussed in Section 2.2. In addition, it supports the creation of a hypertextual documentation for the ontology. Finally, it includes a software agent (OntoBoy) that assists the user in the ontology creation process.

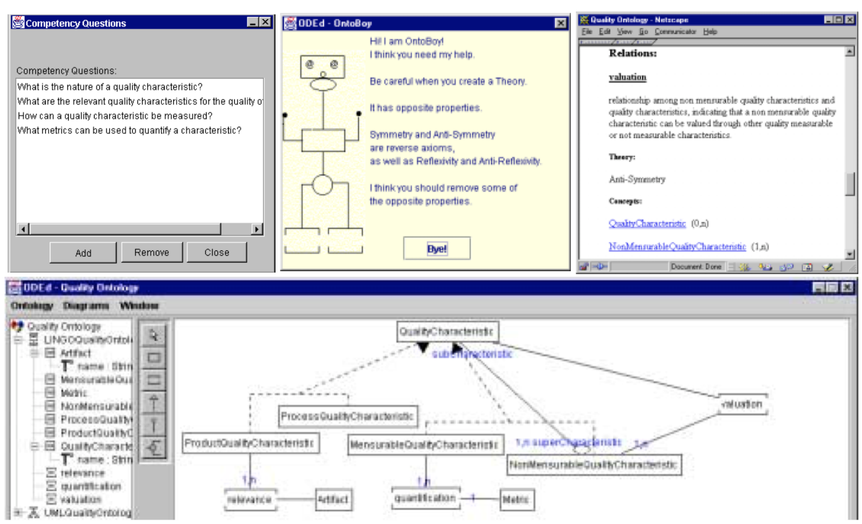

Fig. 7: The ODEd Ontology Development Tool (from [63]).

3.3 Pattern-Based Ontology Engineering

The connection between ontologies and design patterns appears in the work of Falbo and colleagues since the early 2000's [48, 51, 52]. In [12], building on the separation of reference and operational ontologies (see Section 3.1), and drawing on his software engineering background, Falbo collaborated to the terminological clarification regarding the notions of patterns in ontology engineering. Moreover, in this paper, Falbo and colleagues make the case that the confusion related to the term ontology patterns was connected to the lack of a clear separation of conceptual and operational concerns in areas such as Knowledge Engineering and the Semantic Web. In that work, they contribute to organizing the space of ontology patterns, showing that the ontology engineering literature at times abused the semantics of these terms borrowed from the Software Engineering literature. For example, the term Ontology Design Pattern is used not as a *standard solution to a recurrent design problem* but as a very general category, including patterns as different as what in software engineering is termed *Analysis Pattern* as well as idiom (or ontology coding pattern), the latter concerning solutions to circumvent language specific problems. In the proposed space, this work distinguishes the categories of: (a) *Conceptual Patterns* comprising *Foundational Ontology Patterns* (conceptual solutions extracted from Foundational ontologies), and DROPs or *Domain-Related Ontology Patterns* (conceptual solutions extracted from Reference Domain ontologies); (b) *Architectural Patterns* addressing architectural problems (e.g., ontology complexity management); (c) *Design Patterns* addressing design problems, such as guaranteeing some reasoning patterns, addressing ontology non-functional requirements (e.g., tractability), or circumventing limitations of a class of languages (e.g., how to model n-ary relations in languages that are limited to binary relations); (d) *idioms*. In that paper, they also investigate the different mechanisms through which patterns are reused, namely, *reuse by extension*, typically associated with DROPs, or *reuse by analogy*, which is the main mechanism associated with foundational patterns (but not only).

In [8], Falbo and colleagues show that, while in the ontology engineering community patterns were mostly (re)used as stand alone entities, a whole new range of benefits could be achieved if there were complete guidelines guiding the process of interrelating patterns forming solutions to larger problems. Borrowing the term from software engineering, they them propose the notion of Ontology Pattern Language (OPLs) (later refined in [9]) as a *network of interrelated ontology patterns along with a procedure for systematically using them in tandem*. A pattern language defines not only a temporal ordering for the application of patterns but also relations between these patterns, such as those showing that they require the presence of each other, or that they are variants as solutions to the same problem. In that paper, they also demonstrate the role of Core Ontologies for extracting DROPs that can them be easily combined to form pattern languages: *"as patterns move closer to a Domain ontology, they agglutinate to form a stable model, i.e., the constraints on how they can be inter-related become so strong that the very domain model is practically the only way they can appear together, thus, lacking the potential for recurrence which is part of the very definition of what a pattern is. That is why we advocate that DROPs occurring at the level of Core Ontologies are the best candidates for being organized as ontology pattern languages."*.

As initially demonstrated in [54], in a modeling language such as OntoUML that explicitly commits to a foundational ontology, its modeling primitives are not low granularity ones such as class, attribute or relationships, but conceptual patterns reflecting micro-theories of the underlying foundational ontology. Hence, such a language is a pattern language in the strong sense of language[5]. In fact, as shown in [85], the grammar of OntoUML can be expressed as a graph-rewriting pattern grammar.

In [78, 79], Falbo and colleagues demonstrate how an iterated combination of "pattern representation systems" at different levels can contribute to maximize the benefits of reuse of ontological structures. This idea (partially illustrated in Figure 8) amounts (in a nutshell) to the following strategy: (i) a domain-independent pattern grammar such as OntoUML (constituted by foundational patterns) can be used to directly build domain ontologies; (ii) by using the approach proposed in [8, 9], one can then build a Domain-Specific OPL by extracting DROPs from these core ontologies; (iii) these OPLs can be used to effectively create domain ontologies in the respective domains.

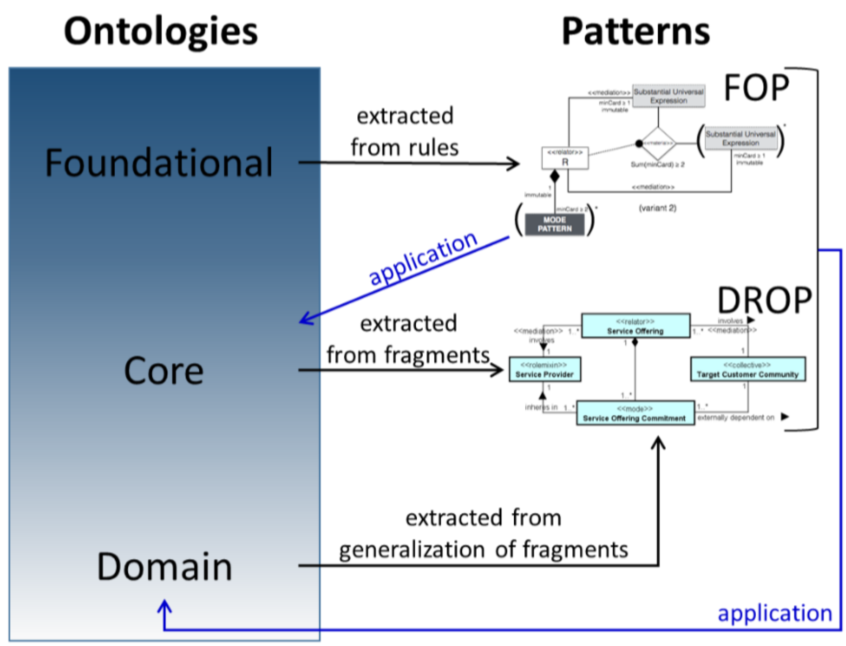

Fig. 8: Combining the benefits of Foundational Ontology Patterns and DROPs [6]).

[5] In [8], when discussing OPLs as systems of DROPs, Falbo and his co-authors write "the use of the term 'language' is, in fact, a misnomer, given that a pattern language does not typically define per se, a grammar with an explicit associated mapping to a semantic domain. However, if we focus on a more general concept of a representation system, we may consider the constituent patterns as an alphabet of higher-granularity primitives. Moreover, in this case, we can consider the procedural rules prescribing how these primitives can be lawfully combined as defining a set of valid possible instantiations for that representation system. Perhaps, a more appropriate name would be 'Pattern System'."

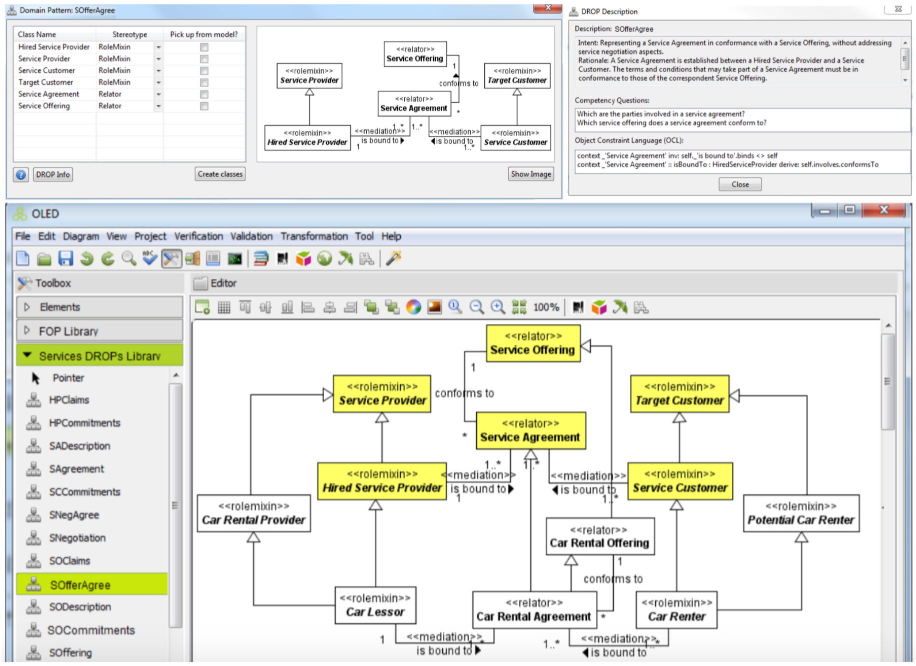

Fig. 9: Supporting for pattern definition and reuse in OLED (from [78]).

In line with their concern in developing computationally supporting tools for their approach (see Section 3.2, in [78, 79], this strategy is implemented in the OntoUML editor (OLED) (see Figure 9).

An example of the application of this strategy to create an ontology of Car Rentals is shown in Figure 10. As depicted there, we have the application, by analogy, of a the *Role* and *RoleMixin* OntoUML patterns, as well as the application, by extension, of the *Service Offering and Agreement* DROP (extracted from the UFO-S ontology, which is itself represented in OntoUML [65]).

Over the years, a number of OPLs have been developed following this approach addressing several domains, including *Service Modeling* [17, 72], *Software Process Harmonization* [76], *Enterprise Modeling* [19], *Software Testing* [81] *Measurement* [25], *Configuration Management* [4], among many others. As always, conscious of the need for providing engineering tools for operationalizing the use theoretical results, in [71, 80], Falbo and colleagues treat OPL design as a domain in itself and employ a systematic language engineering method to propose a domain-specific modeling language for representing OPLs (called OPL-ML). Figure 11, summarizes the visual notion of this language, whose application is then illustrated in Figure 12 where it is employed in the design of S-OPL (an OPL for service modeling based in UFO-S [17, 72]).

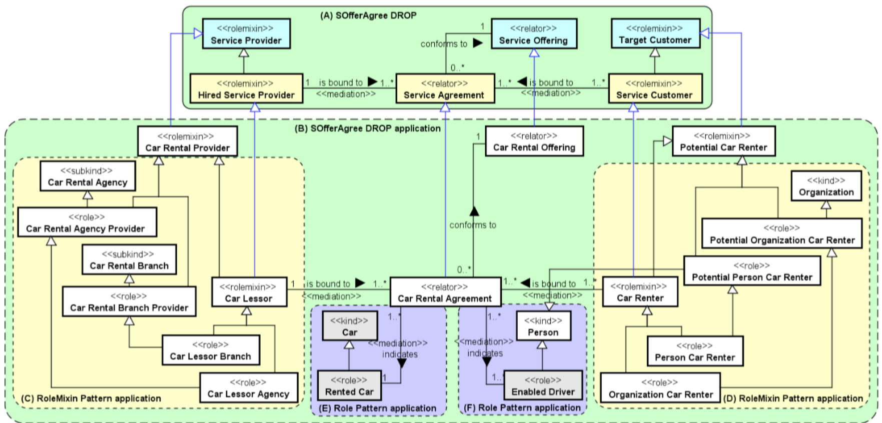

Fig. 10: Fragment of a Car Rental Agreement ontology built from patterns [78].

Structural Model	
Element	**Symbol**
Pattern	<<name>>
Pattern Group (expanded format)	<<name>>
Pattern Group (black box format)	<<name>>
Variant Pattern Group (expanded format)	<<name>>
Variant Pattern Group (black box format)	<<name>>
Relation "requires"	——▶
Relation "requires a pattern of"	- - - - ▶

Fig. 11: Summary of the OPL-ML Visual Notation (from [71]).

4 Final Considerations

In this paper, we review the long term research program carried out by Ricardo Falbo throughout his career. On one hand, as an engineer, Falbo has always been driven by concrete solutions and for the need of developing practical engineering tools. In particular, as a software engineer, he continuously adopted insights from software engineering to propose new and to improve existing solutions in Ontology Engineering. As shown in this paper, these include contributions to developing: a systematic method for ontology engineering; a set of pattern-based solutions for ontology design; and a number of computational tools for the discipline. On the other hand, as an ontologist, he brought the theoretically grounded methods of that discipline to continuously tackle conceptual problems in software engineering. This includes developing reference ontologies for a number of complex software engineering domains, as well as leveraging on these

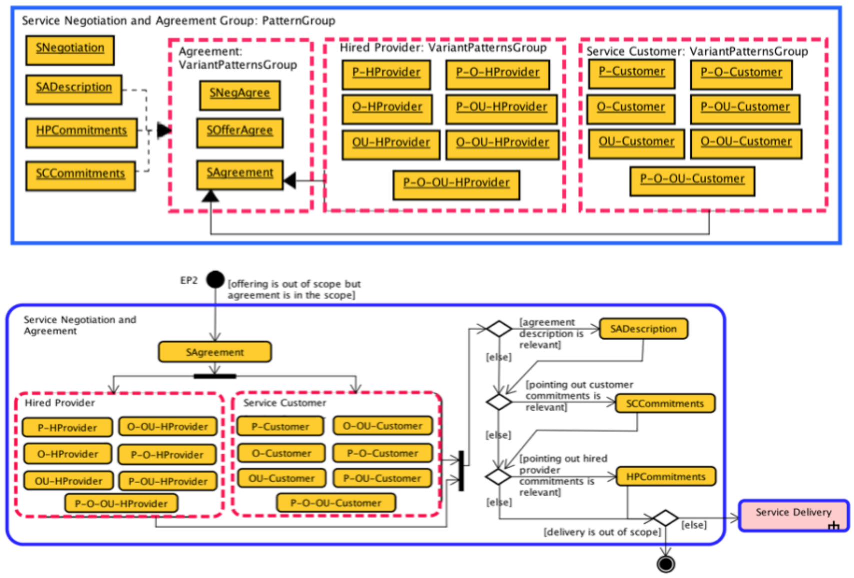

Fig. 12: Using OPL-ML to design S-OPL (from [80]).

ontologies to: improve the methods of that discipline regarding, for instance, requirements engineering, application integration, and domain engineering; develop semantic software engineering tools based on these ontologies for supporting effective knowledge management in Software Engineering.

Unswayed by buzzwords and fashion, Falbo consistently followed this research program, leading for many years a Laboratory for Software Engineering Research (LabES), and later co-founding the Ontology and Conceptual Modeling Research Group (NEMO). In these initiatives, he has always been a natural leader, combining intellectual sharpness and rigour, creativity, enthusiasm, kindness and humility, in a manner that is uncommon in science (as well as generally in life). In his research career, Falbo inspired many colleagues and students, and established a network of collaborators in different countries. Together they produced results that are constantly used by scholars internationally. However, despite these remarkable qualities as a researcher, Falbo used to say that if he had to choose between being a scientist, a practitioner, or a professor, he would chose the latter. This is because, he judged, in that way, he could perhaps have the chance of having a deeper and long lasting influence on people. That certainly worked on us.

5 Personal Notes

Giancarlo: I met Ricardo when I was still a bachelor student and he was working towards finishing his PhD thesis. In the mid 1990's, I was drawn by the problem of producing precise representations that could capture application-independent domain regularities. After attending one of his presentations, I decided to send him my very first paper on that topic, to which he replied "This is very interesting! What you are doing here is an ontology. Have you heard about ontologies?", then giving me a copy of Nicola Guarino's 'The Ontological Level' [45] (see Nicola Guarino's personal letter in this volume). This is how I was first introduced to the research topic that would be the cornerstone of my own research program for more than two decades. Since then, we have developed a pleasant, fruitful and long-lasting research collaboration that has so far resulted in 44 joint publications, in topics ranging from Domain and Ontology Engineering, Web Engineering, Ontologies in Software Engineering and Enterprise Modeling, Domain-Specific Visual Language Design, as well as in the foundations of the UFO ontology and the OntoUML language. But, first and foremost, I consider him a mentor, one of my dearest friends and, simply, one of the best people I know.

João Paulo: Towards the end of my undergraduate studies (around 1998), Ricardo and Crediné Menezes taught together a course on object-oriented software engineering. The course was naturally informed by their investigation into ontology-based approaches, which resulted in a very interesting experience to me as a student. It was an excellent first impression of him! Years later, when I was hired at UFES, I soon realized that he was not just a great teacher, but also a wonderful friend and an extraordinary colleague. He is generous, sincere, and one of the most sensible voices around. He thinks and speaks calmly, always focusing on the essence, with no room for distractions or pretension! I have the greatest honor and pleasure to work with him. Together, we supervised a number of Ph.D. students and worked on advances in UFO (particularly UFO-B and UFO-S) and OntoUML. Ricardo has a simple and practical attitude to work that gets things done with zero stress. This attitude is a key part of what we now call warmly "Falbo's approach". I am very grateful he has shared his approach with us. It is not only an approach to ontology but also an inspiring approach to life!

References

1. Proceedings of the 14th international conference on Software engineering and knowledge engineering, SEKE 2002, Ischia, Italy, July 15-19, 2002. ACM (2002). https://doi.org/10.1145/568760, `https://doi.org/10.1145/568760`
2. de Aguiar, C.Z., de Almeida Falbo, R., Souza, V.E.S.: Ontological representation of relational databases. In: Carbonera, J.L., Guizzardi, G. (eds.) Proceedings of the XI Seminar on Ontology Research in Brazil and II Doctoral and Masters Consortium on Ontologies, São Paulo, Brazil, October 1st-3rd, 2018. CEUR Workshop Proceedings, vol. 2228, pp. 140–151. CEUR-WS.org (2018), `http://ceur-ws.org/Vol-2228/paper9.pdf`
3. de Aguiar, C.Z., de Almeida Falbo, R., Souza, V.E.S.: OOC-O: A reference ontology on object-oriented code. In: Laender et al. [59], pp. 13–27. https://doi.org/10.1007/978-3-030-33223-5_3, `https://doi.org/10.1007/978-3-030-33223-5_3`
4. Almeida, A.C., Schwabe, D., Lifschitz, S., Campos, M.L.M.: Cm-opl: An ontology pattern language for configuration management task. In: ONTOBRAS. pp. 233–238 (2018)
5. Almeida, J.P.A., de Almeida Falbo, R., Guizzardi, G.: Events as entities in ontology-driven conceptual modeling. In: Laender et al. [59], pp. 469–483. https://doi.org/10.1007/978-3-030-33223-5_39, `https://doi.org/10.1007/978-3-030-33223-5_39`

6. de Almeida Falbo, R.: Sabio: Systematic approach for building ontologies. In: Guizzardi, G., Pastor, O., Wand, Y., de Cesare, S., Gailly, F., Lycett, M., Partridge, C. (eds.) Proceedings of the 1st Joint Workshop ONTO.COM / ODISE on Ontologies in Conceptual Modeling and Information Systems Engineering co-located with 8th International Conference on Formal Ontology in Information Systems, ONTO.COM/ODISE@FOIS 2014, Rio de Janeiro, Brazil, September 21, 2014. CEUR Workshop Proceedings, vol. 1301. CEUR-WS.org (2014), http://ceur-ws.org/Vol-1301/ontocomodise2014_2.pdf
7. de Almeida Falbo, R., Arantes, D.O., Natali, A.C.C.: Integrating knowledge management and groupware in a software development environment. In: Karagiannis, D., Reimer, U. (eds.) Practical Aspects of Knowledge Management, 5th International Conference, PAKM 2004, Vienna, Austria, December 2-3, 2004, Proceedings. Lecture Notes in Computer Science, vol. 3336, pp. 94–105. Springer (2004). https://doi.org/10.1007/978-3-540-30545-3_9, https://doi.org/10.1007/978-3-540-30545-3_9
8. de Almeida Falbo, R., Barcellos, M.P., Nardi, J.C., Guizzardi, G.: Organizing ontology design patterns as ontology pattern languages. In: Cimiano, P., Corcho, Ó., Presutti, V., Hollink, L., Rudolph, S. (eds.) The Semantic Web: Semantics and Big Data, 10th International Conference, ESWC 2013, Montpellier, France, May 26-30, 2013. Proceedings. Lecture Notes in Computer Science, vol. 7882, pp. 61–75. Springer (2013). https://doi.org/10.1007/978-3-642-38288-8_5, https://doi.org/10.1007/978-3-642-38288-8_5
9. de Almeida Falbo, R., Barcellos, M.P., Ruy, F.B., Guizzardi, G., Guizzardi, R.S.S.: Ontology pattern languages. In: Hitzler, P., Gangemi, A., Janowicz, K., Krisnadhi, A., Presutti, V. (eds.) Ontology Engineering with Ontology Design Patterns - Foundations and Applications, Studies on the Semantic Web, vol. 25, pp. 133–159. IOS Press (2016). https://doi.org/10.3233/978-1-61499-676-7-133, https://doi.org/10.3233/978-1-61499-676-7-133
10. de Almeida Falbo, R., Bertollo, G.: A software process ontology as a common vocabulary about software processes. IJBPIM **4**(4), 239–250 (2009). https://doi.org/10.1504/IJBPIM.2009.032281, https://doi.org/10.1504/IJBPIM.2009.032281
11. de Almeida Falbo, R., Guizzardi, G., Duarte, K.C.: An ontological approach to domain engineering. In: Proceedings of the 14th international conference on Software engineering and knowledge engineering, SEKE 2002, Ischia, Italy, July 15-19, 2002 [1], pp. 351–358. https://doi.org/10.1145/568760.568822, https://doi.org/10.1145/568760.568822
12. de Almeida Falbo, R., Guizzardi, G., Gangemi, A., Presutti, V.: Ontology patterns: Clarifying concepts and terminology. In: Gangemi, A., Gruninger, M., Hammar, K., Lefort, L., Presutti, V., Scherp, A. (eds.) Proceedings of the 4th Workshop on Ontology and Semantic Web Patterns co-located with 12th International Semantic Web Conference (ISWC 2013), Sydney, Australia, October 21, 2013. CEUR Workshop Proceedings, vol. 1188. CEUR-WS.org (2013), http://ceur-ws.org/Vol-1188/paper_11.pdf
13. de Almeida Falbo, R., Guizzardi, G., Natali, A.C.C., Bertollo, G., Ruy, F.F., Mian, P.G.: Towards semantic software engineering environments. In: Proceedings of the 14th international conference on Software engineering and knowledge engineering, SEKE 2002, Ischia, Italy, July 15-19, 2002 [1], pp. 477–478. https://doi.org/10.1145/568760.568843, https://doi.org/10.1145/568760.568843
14. de Almeida Falbo, R., Martins, A.F., Segrini, B.M., Baiôco, G., Moro, R.D., Nardi, J.C.: Um processo de engenharia de requisitos baseado em reutilização de ontologias e padrões de análise. In: Kong, M., Pow-Sang, J.A., Anticona, M.T., García, L.A.F. (eds.) VI Jornadas Iberoamericanas de Ingeniería del Software e Ingeniería del Conocimiento-JIISIC'07, 31 de Enero al 2 de Febrero del 2007, Lima, Perú. pp. 59–68. Facultad de Ciencias e Ingeniería and Departamento de Ingeniería, Pontificia Universidad Católica del Perú (2007)

15. de Almeida Falbo, R., de Menezes, C.S., Rocha, A.R.: A systematic approach for building ontologies. In: Coelho, H. (ed.) Progress in Artificial Intelligence - IBERAMIA 98, 6th Ibero-American Conference on AI, Lisbon, Portugal, October 5-9, 1998, Proceedings. Lecture Notes in Computer Science, vol. 1484, pp. 349–360. Springer (1998). https://doi.org/10.1007/3-540-49795-1_31, https://doi.org/10.1007/3-540-49795-1_31
16. de Almeida Falbo, R., Pezzin, J., Schwambach, M.M.: A multi-agent system for knowledge delivery in a software engineering environment. In: Chu et al. [35], pp. 253–258, http://ksiresearchorg.ipage.com/seke/Proceedings/seke/SEKE2005_Proceedings.pdf
17. de Almeida Falbo, R., Quirino, G.K., Nardi, J.C., Barcellos, M.P., Guizzardi, G., Guarino, N., Longo, A., Livieri, B.: An ontology pattern language for service modeling. In: Ossowski, S. (ed.) Proceedings of the 31st Annual ACM Symposium on Applied Computing, Pisa, Italy, April 4-8, 2016. pp. 321–326. ACM (2016). https://doi.org/10.1145/2851613.2851840, https://doi.org/10.1145/2851613.2851840
18. de Almeida Falbo, R., Ruy, F.B., Bertollo, G., Togneri, D.F.: Learning how to manage risks using organizational knowledge. In: Holz, H., Melnik, G. (eds.) Advances in Learning Software Organizations, 6th International Workshop, LSO 2004, Banff, Canada, June 20-21, 2004, Proceedings. Lecture Notes in Computer Science, vol. 3096, pp. 7–18. Springer (2004). https://doi.org/10.1007/978-3-540-25983-1_2, https://doi.org/10.1007/978-3-540-25983-1_2
19. de Almeida Falbo, R., Ruy, F.B., Guizzardi, G., Barcellos, M.P., Almeida, J.P.A.: Towards an enterprise ontology pattern language. In: Cho, Y., Shin, S.Y., Kim, S., Hung, C., Hong, J. (eds.) Symposium on Applied Computing, SAC 2014, Gyeongju, Republic of Korea - March 24 - 28, 2014. pp. 323–330. ACM (2014). https://doi.org/10.1145/2554850.2554983, https://doi.org/10.1145/2554850.2554983
20. de Almeida Falbo, R., Ruy, F.B., Moro, R.D.: Using ontologies to add semantics to a software engineering environment. In: Chu et al. [35], pp. 151–156, http://ksiresearchorg.ipage.com/seke/Proceedings/seke/SEKE2005_Proceedings.pdf
21. Arango, G.: Domain analysis methods. Software Reusability pp. 17–49 (1994)
22. Arantes, L.D.O., Falbo, R.D.A., Guizzardi, G.: Evolving a software configuration management ontology (2007)
23. Barcellos, M.P., de Almeida Falbo, R.: Using a foundational ontology for reengineering a software enterprise ontology. In: Heuser, C.A., Pernul, G. (eds.) Advances in Conceptual Modeling - Challenging Perspectives, ER 2009 Workshops CoMoL, ETheCoM, FP-UML, MOST-ONISW, QoIS, RIGiM, SeCoGIS, Gramado, Brazil, November 9-12, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5833, pp. 179–188. Springer (2009). https://doi.org/10.1007/978-3-642-04947-7_22, https://doi.org/10.1007/978-3-642-04947-7_22
24. Barcellos, M.P., de Almeida Falbo, R.: A software measurement task ontology. In: Shin, S.Y., Maldonado, J.C. (eds.) Proceedings of the 28th Annual ACM Symposium on Applied Computing, SAC '13, Coimbra, Portugal, March 18-22, 2013. pp. 311–318. ACM (2013). https://doi.org/10.1145/2480362.2480428, https://doi.org/10.1145/2480362.2480428
25. Barcellos, M.P., de Almeida Falbo, R., Moro, R.D.: A well-founded software measurement ontology. In: Galton, A., Mizoguchi, R. (eds.) Formal Ontology in Information Systems, Proceedings of the Sixth International Conference, FOIS 2010, Toronto, Canada, May 11-14, 2010. Frontiers in Artificial Intelligence and Applications, vol. 209, pp. 213–226. IOS Press (2010). https://doi.org/10.3233/978-1-60750-535-8-213, https://doi.org/10.3233/978-1-60750-535-8-213

26. Bastos, E., Barcellos, M.P., de Almeida Falbo, R.: Exploring ontologies for semantic documentation in project management. In: Freitas, F., Baião, F.A. (eds.) Proceedings of the Brazilian Seminar on Ontologies (ONTOBRAS 2015), São Paulo, Brazil, September 8-11, 2015. CEUR Workshop Proceedings, vol. 1442. CEUR-WS.org (2015), http://ceur-ws.org/Vol-1442/paper_9.pdf
27. Bastos, E., Barcellos, M.P., de Almeida Falbo, R.: Using semantic documentation to support software project management. J. Data Semantics **7**(2), 107–132 (2018). https://doi.org/10.1007/s13740-018-0089-z, https://doi.org/10.1007/s13740-018-0089-z
28. Bernabé, C.H., Souza, V.E.S., de Almeida Falbo, R., Guizzardi, R.S.S., Silva, C.: GORO 2.0: Evolving an ontology for goal-oriented requirements engineering. In: Guizzardi, G., Gailly, F., Maciel, R.S.P. (eds.) Advances in Conceptual Modeling - ER 2019 Workshops FAIR, MREBA, EmpER, MoBiD, OntoCom, and ER Doctoral Symposium Papers, Salvador, Brazil, November 4-7, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11787, pp. 169–179. Springer (2019). https://doi.org/10.1007/978-3-030-34146-6_15, https://doi.org/10.1007/978-3-030-34146-6_15
29. Bouter, C.: Constructing an Explanation Ontology for the Communication and Combination of Partial Explanations in a Federated Knowledge Environment. Master's thesis (2019)
30. Bunge, M.: Treatise on basic philosophy: Ontology I: the furniture of the world, vol. 3. Springer Science & Business Media (1977)
31. Calhau, R.F., de Almeida Falbo, R.: An ontology-based approach for semantic integration. In: Proceedings of the 14th IEEE International Enterprise Distributed Object Computing Conference, EDOC 2010, Vitória, Brazil, 25-29 October 2010. pp. 111–120. IEEE Computer Society (2010). https://doi.org/10.1109/EDOC.2010.32, https://doi.org/10.1109/EDOC.2010.32
32. Calhau, R.F., de Almeida Falbo, R.: A configuration management task ontology for semantic integration. In: Ossowski, S., Lecca, P. (eds.) Proceedings of the ACM Symposium on Applied Computing, SAC 2012, Riva, Trento, Italy, March 26-30, 2012. pp. 348–353. ACM (2012). https://doi.org/10.1145/2245276.2245344, https://doi.org/10.1145/2245276.2245344
33. de Carvalho, V.A., Coelho, A., de Almeida Falbo, R.: Apoio automatizado à gerência de riscos cooperativa. In: Losavio, F., Travassos, G.H., Pelechano, V., Díaz, I., Matteo, A. (eds.) Memorias de la X Conferencia Iberoamericana de Software Engineering (CIbSE 2007), Isla de Marguerita, Venezuela, Mayo 7-11, 2007. pp. 297–310 (2007)
34. Castro, J., Cernuzzi, L., Gordillo, S.E. (eds.): Memorias de la IX Conferenci a Iberoamericana de Software Engineering (CIbSE 2006), La Plata, Argentina, Abril 24-28, 2006 (2006)
35. Chu, W.C., Juzgado, N.J., Wong, W.E. (eds.): Proceedings of the 17th International Conference on Software Engineering and Knowledge Engineering (SEKE'2005), Taipei, Taiwan, Republic of China, July 14-16, 2005 (2005), http://ksiresearchorg.ipage.com/seke/Proceedings/seke/SEKE2005_Proceedings.pdf
36. van Der Aalst, W.M.: Workflow patterns. Encyclopedia of Database Systems pp. 3557–3558 (2009)
37. Duarte, B.B., de Almeida Falbo, R., Guizzardi, G., Guizzardi, R.S.S., Souza, V.E.S.: Towards an ontology of software defects, errors and failures. In: Trujillo, J., Davis, K.C., Du, X., Li, Z., Ling, T.W., Li, G., Lee, M. (eds.) Conceptual Modeling - 37th International Conference, ER 2018, Xi'an, China, October 22-25, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11157, pp. 349–362. Springer (2018). https://doi.org/10.1007/978-3-030-00847-5_25, https://doi.org/10.1007/978-3-030-00847-5_25
38. Duarte, B.B., de Castro Leal, A.L., de Almeida Falbo, R., Guizzardi, G., Guizzardi, R.S.S., Souza, V.E.S.: Ontological foundations for software requirements with a focus on require-

ments at runtime. Applied Ontology **13**(2), 73–105 (2018). https://doi.org/10.3233/AO-180197, https://doi.org/10.3233/AO-180197
39. Duarte, B.B., Souza, V.E.S., de Castro Leal, A.L., de Almeida Falbo, R., Guizzardi, G., Guizzardi, R.S.S.: Towards an ontology of requirements at runtime. In: Ferrario, R., Kuhn, W. (eds.) Formal Ontology in Information Systems - Proceedings of the 9th International Conference, FOIS 2016, Annecy, France, July 6-9, 2016. Frontiers in Artificial Intelligence and Applications, vol. 283, pp. 255–268. IOS Press (2016). https://doi.org/10.3233/978-1-61499-660-6-255, https://doi.org/10.3233/978-1-61499-660-6-255
40. Duarte, K.C., Falbo, R.A.: Uma ontologia de qualidade de software. In: Workshop de Qualidade de Software, João Pessoa. pp. 275–285 (2000)
41. Falbo, R.: Integração de Conhecimento em Ambientes de Desenvolvimento de Software. Ph.D. thesis, Federal University of Rio de Janeiro (COPPE/UFRJ) (1998)
42. Falbo, R.D.A., Natali, A.C.C., Mian, P.G., Bertollo, G., Ruy, F.B.: Ode: Ontology-based software development environment. In: IX Congreso Argentino de Ciencias de la Computación (2003)
43. Fernández-López, M., Gómez-Pérez, A., Juristo, N.: Methontology: from ontological art towards ontological engineering (1997)
44. Grüninger, M., Fox, M.S.: Methodology for the design and evaluation of ontologies (1995)
45. Guarino, N.: The ontological level. Philosophy and the cognitive sciences (1994)
46. Guarino, N.: Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy, vol. 46. IOS press (1998)
47. Guillermo, A., Ruben, P.D.: Domain analysis concepts and research directions (1991)
48. Guizzardi, G.: Software Development for and with Reuse: A Case Study with a Video on Demand System (also published as the technical report entitled: *A Methodological Approach for Development with and for reuse based on Formal Domain Ontologies*). Master's thesis (2000)
49. Guizzardi, G.: Ontological Foundations for Structural Conceptual Models. Telematica Instituut / CTIT (2005)
50. Guizzardi, G.: On ontology, ontologies, conceptualizations, modeling languages, and (meta)models. In: Vasilecas, O., Eder, J., Caplinskas, A. (eds.) Databases and Information Systems IV - Selected Papers from the Seventh International Baltic Conference, DB&IS 2006, July 3-6, 2006, Vilnius, Lithuania. Frontiers in Artificial Intelligence and Applications, vol. 155, pp. 18–39. IOS Press (2006), http://www.booksonline.iospress.nl/Content/View.aspx?piid=5421
51. Guizzardi, G., de Almeida Falbo, R., Filho, J.G.P.: From domain ontologies to object-oriented frameworks. In: Workshop on Ontologies (ONTO-2001), Joint German-Austrian Conference on Artificial Intelligence, Vienna, Austria (2001)
52. Guizzardi, G., de Almeida Falbo, R., Filho, J.G.P.: Using objects and patterns to implement domain ontologies. J. Braz. Comp. Soc. **8**(1), 43–56 (2002)
53. Guizzardi, G., de Almeida Falbo, R., Guizzardi, R.S.S.: Grounding software domain ontologies in the unified foundational ontology (UFO): the case of the ODE software process ontology. In: Lencastre, M., e Cunha, J.F., Valecillo, A. (eds.) Memorias de la XI Conferencia Iberoamericana de Software Engineering (CIbSE 2008), Recife, Pernambuco, Brasil, February 13-17, 2008. pp. 127–140 (2008)
54. Guizzardi, G., das Graças, A.P., Guizzardi, R.S.S.: Design patterns and inductive modeling rules to support the construction of ontologically well-founded conceptual models in ontouml. In: Salinesi, C., Pastor, O. (eds.) Advanced Information Systems Engineering Workshops - CAiSE 2011 International Workshops, London, UK, June 20-24, 2011. Proceedings. Lecture Notes in Business Information Processing, vol. 83, pp. 402–413. Springer (2011). https://doi.org/10.1007/978-3-642-22056-2_44, https://doi.org/10.1007/978-3-642-22056-2_44

55. Guizzardi, G., Wagner, G., Almeida, J.P.A., Guizzardi, R.S.S.: Towards ontological foundations for conceptual modeling: The unified foundational ontology (UFO) story. Applied Ontology **10**(3-4), 259–271 (2015). https://doi.org/10.3233/AO-150157, https://doi.org/10.3233/AO-150157
56. Guizzardi, G., Wagner, G., de Almeida Falbo, R., Guizzardi, R.S.S., Almeida, J.P.A.: Towards ontological foundations for the conceptual modeling of events. In: Ng, W., Storey, V.C., Trujillo, J. (eds.) Conceptual Modeling - 32th International Conference, ER 2013, Hong-Kong, China, November 11-13, 2013. Proceedings. Lecture Notes in Computer Science, vol. 8217, pp. 327–341. Springer (2013). https://doi.org/10.1007/978-3-642-41924-9_27, https://doi.org/10.1007/978-3-642-41924-9_27
57. Johannesson, P., Lee, M., Liddle, S.W., Opdahl, A.L., López, O.P. (eds.): Conceptual Modeling - 34th International Conference, ER 2015, Stockholm, Sweden, October 19-22, 2015, Proceedings, Lecture Notes in Computer Science, vol. 9381. Springer (2015). https://doi.org/10.1007/978-3-319-25264-3, https://doi.org/10.1007/978-3-319-25264-3
58. Júnior, P.A.P., Penteado, R.A.D.: Obascid (-tool): an ontologically based approach for concern identification and classification and its computational support. Journal of the Brazilian Computer Society **24**(1), 3 (2018)
59. Laender, A.H.F., Pernici, B., Lim, E., Palazzo M. de Oliveira, J. (eds.): Conceptual Modeling - 38th International Conference, ER 2019, Salvador, Brazil, November 4-7, 2019, Proceedings, Lecture Notes in Computer Science, vol. 11788. Springer (2019). https://doi.org/10.1007/978-3-030-33223-5, https://doi.org/10.1007/978-3-030-33223-5
60. Maciel, C.P.C., de Souza, É.F., Vijaykumar, N.L., de Almeida Falbo, R., Meinerz, G.V., Felizardo, K.R.: An empirical study on the knowledge management practice in software testing. In: Genero, M., Kalinowski, M., Molina, J.G., Pino, F., Conte, T., Marín, B., Brito, I., Giachetti, G. (eds.) Proceedings of the XXI Iberoamerican Conference on Software Engineering, Bogota, Colombia, April 23-27, 2018. pp. 29–42. Curran Associates (2018)
61. MARTINS, A., Falbo, R.A., Guizzardi, G., Almeida, J.P.A.: Uso de uma ontologia de fundamentação para dirimir ambiguidades na modelagem de processos de negócio. VII Simpósio Brasileiro de Sistemas de Informação (SBSI 2011) (2011)
62. Martins, A.F., de Almeida Falbo, R.: Models for representing task ontologies. In: de Freitas, F.L.G., Stuckenschmidt, H., Pinto, H.S., Malucelli, A., Corcho, Ó. (eds.) Proceedings of the 3rd Workshop on Ontologies and their Applications, Salvador, Bahia, Brazil, October 26, 2008. CEUR Workshop Proceedings, vol. 427. CEUR-WS.org (2008), http://ceur-ws.org/Vol-427/paper4.pdf
63. Mian, P.G., de Almeida Falbo, R.: Supporting ontology development with oded. In: de Leon Ferreira de Carvalho, A.C.P., Maldonado, J.C., Augusto, J.C., de Mendonça Neto, M.G., Acuña, S.T. (eds.) Proceedings of 2nd Ibero-American Symposium on Software Engineering and Knowledge Engineering (JIISIC'02), Salvador, Brasil, October 2002. pp. 1–11 (2002)
64. Nardi, J.C., de Almeida Falbo, R.: Uma ontologia de requisitos de software. In: Castro et al. [34], pp. 111–124
65. Nardi, J.C., de Almeida Falbo, R., Almeida, J.P.A., Guizzardi, G., Pires, L.F., van Sinderen, M., Guarino, N., Fonseca, C.M.: A commitment-based reference ontology for services. Inf. Syst. **54**, 263–288 (2015). https://doi.org/10.1016/j.is.2015.01.012, https://doi.org/10.1016/j.is.2015.01.012
66. Negri, P.P., Souza, V.E.S., de Castro Leal, A.L., de Almeida Falbo, R., Guizzardi, G.: Towards an ontology of goal-oriented requirements. In: Pastor, O., Guizzardi, R.S.S., Kaplan, G.N., Molina, J.G., Moreira, A., Brito, I.S., de la Vara, J.L., Genero, M., da Silva, F.Q.B., Cernuzzi, L. (eds.) Proceedings of the XX Iberoamerican Conference on Software Engineer-

ing, Buenos Aires, Argentina, May 22-23, 2017. pp. 469–482. Curran Associates (2017), http://cibse2017.inf.ufes.br
67. Neighbors, J.M.: Software construction using components. Ph.D. thesis, University of California, Irvine (1980)
68. de Oliveira Arantes, L., de Carvalho, V.A., de Almeida Falbo, R.: Uma ferramenta integrada de apoio a estimativas de tamanho e esforço em um ambiente de desenvolvimento de software. In: Castro et al. [34], pp. 493–496
69. de Oliveira Bringuente, A.C., de Almeida Falbo, R., Guizzardi, G.: Using a foundational ontology for reengineering a software process ontology. JIDM **2**(3), 511–526 (2011), http://seer.lcc.ufmg.br/index.php/jidm/article/view/164
70. Parreira, P.A., Penteado, R.A.D.: Obascid: an ontologically-based approach for concern identification and classification. In: 2016 X Brazilian Symposium on Software Components, Architectures and Reuse (SBCARS). pp. 141–150. IEEE (2016)
71. Quirino, G.K.S., Barcellos, M.P., de Almeida Falbo, R.: OPL-ML: A modeling language for representing ontology pattern languages. In: de Cesare, S., Frank, U. (eds.) Advances in Conceptual Modeling - ER 2017 Workshops AHA, MoBiD, MREBA, OntoCom, and QMMQ, Valencia, Spain, November 6-9, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10651, pp. 187–201. Springer (2017). https://doi.org/10.1007/978-3-319-70625-2_18, https://doi.org/10.1007/978-3-319-70625-2_18
72. Quirino, G.K., Nardi, J.C., Barcellos, M.P., de Almeida Falbo, R., Guizzardi, G., Guarino, N., Bochicchio, M.A., Longo, A., Zappatore, M.S., Livieri, B.: Towards a service ontology pattern language. In: Johannesson et al. [57], pp. 187–195. https://doi.org/10.1007/978-3-319-25264-3_14, https://doi.org/10.1007/978-3-319-25264-3_14
73. Radović, M., Tošić, M., Milošević, D., Milošević, M., Milošević, M.: Semantic approach to script concordance test
74. Ruy, F.B., de Almeida Falbo, R., Barcellos, M.P., Costa, S.D., Guizzardi, G.: SEON: A software engineering ontology network. In: Blomqvist, E., Ciancarini, P., Poggi, F., Vitali, F. (eds.) Knowledge Engineering and Knowledge Management - 20th International Conference, EKAW 2016, Bologna, Italy, November 19-23, 2016, Proceedings. Lecture Notes in Computer Science, vol. 10024, pp. 527–542 (2016). https://doi.org/10.1007/978-3-319-49004-5_34, https://doi.org/10.1007/978-3-319-49004-5_34
75. Ruy, F.B., de Almeida Falbo, R., Barcellos, M.P., Guizzardi, G.: An ontological analysis of the ISO/IEC 24744 metamodel. In: Garbacz, P., Kutz, O. (eds.) Formal Ontology in Information Systems - Proceedings of the Eighth International Conference, FOIS 2014, September, 22-25, 2014, Rio de Janeiro, Brazil. Frontiers in Artificial Intelligence and Applications, vol. 267, pp. 330–343. IOS Press (2014). https://doi.org/10.3233/978-1-61499-438-1-330, https://doi.org/10.3233/978-1-61499-438-1-330
76. Ruy, F.B., de Almeida Falbo, R., Barcellos, M.P., Guizzardi, G.: Towards an ontology pattern language for harmonizing software process related ISO standards. In: Wainwright, R.L., Corchado, J.M., Bechini, A., Hong, J. (eds.) Proceedings of the 30th Annual ACM Symposium on Applied Computing, Salamanca, Spain, April 13-17, 2015. pp. 388–395. ACM (2015). https://doi.org/10.1145/2695664.2695796, https://doi.org/10.1145/2695664.2695796
77. Ruy, F.B., Bertollo, G., de Almeida Falbo, R.: Knowledge-based support to process integration in ODE. CLEI Electron. J. **7**(1) (2004). https://doi.org/10.19153/cleiej.7.1.3, https://doi.org/10.19153/cleiej.7.1.3
78. Ruy, F.B., Guizzardi, G., de Almeida Falbo, R., Reginato, C.C., Santos, V.A.: From reference ontologies to ontology patterns and back. Data Knowl. Eng. **109**, 41–69 (2017). https://doi.org/10.1016/j.datak.2017.03.004, https://doi.org/10.1016/j.datak.2017.03.004

79. Ruy, F.B., Reginato, C.C., dos Santos, V.A., de Almeida Falbo, R., Guizzardi, G.: Ontology engineering by combining ontology patterns. In: Johannesson et al. [57], pp. 173–186. https://doi.org/10.1007/978-3-319-25264-3_13, https://doi.org/10.1007/978-3-319-25264-3_13
80. das Graças da Silva Teixeira, M., Quirino, G.K., Gailly, F., de Almeida Falbo, R., Guizzardi, G., Barcellos, M.P.: Pon-s: A systematic approach for applying the physics of notation (pon). In: Schmidt, R., Guédria, W., Bider, I., Guerreiro, S. (eds.) Enterprise, Business-Process and Information Systems Modeling - 17th International Conference, BPMDS 2016, 21st International Conference, EMMSAD 2016, Held at CAiSE 2016, Ljubljana, Slovenia, June 13-14, 2016, Proceedings. Lecture Notes in Business Information Processing, vol. 248, pp. 432–447. Springer (2016). https://doi.org/10.1007/978-3-319-39429-9_27, https://doi.org/10.1007/978-3-319-39429-9_27
81. de Souza, É.F., de Almeida Falbo, R., Vijaykumar, N.L.: Knowledge management applied to software testing: A systematic mapping. In: The 25th International Conference on Software Engineering and Knowledge Engineering, Boston, MA, USA, June 27-29, 2013. pp. 562–567. Knowledge Systems Institute Graduate School (2013), http://ksiresearchorg.ipage.com/seke/Proceedings/seke/SEKE2013_Proceedings.pdf
82. de Souza, É.F., de Almeida Falbo, R., Vijaykumar, N.L.: Roost: Reference ontology on software testing. Applied Ontology **12**(1), 59–90 (2017). https://doi.org/10.3233/AO-170177, https://doi.org/10.3233/AO-170177
83. Souza, V.E.S., de Almeida Falbo, R.: Supporting ontology axiomatization and evaluation in oded. In: Astudillo, H., Taramasco, C. (eds.) Memorias de la VIII Conferencia Iberoamericana de Software Engineering (CIbSE 2005), Valparaiso, Chile, Mayo 2-6, 2005. pp. 59–70 (2005)
84. Uschold, M., Gruninger, M.: Ontologies: Principles, methods and applications. The knowledge engineering review **11**(2), 93–136 (1996)
85. Zambon, E., Guizzardi, G.: Formal definition of a general ontology pattern language using a graph grammar. In: Ganzha, M., Maciaszek, L.A., Paprzycki, M. (eds.) Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017, Prague, Czech Republic, September 3-6, 2017. Annals of Computer Science and Information Systems, vol. 11, pp. 1–10 (2017). https://doi.org/10.15439/2017F001, https://doi.org/10.15439/2017F001