SP-ACT: A Hybrid Framework for Complex Activity Recognition Combining OWL and SPARQL Rules

Georgios Meditskos, Stamatia Dasiopoulou, Vasiliki Efstathiou, Ioannis Kompatsiaris
Information Technologies Institute, Centre of Research and Technology Hellas, Greece
Email: {gmeditsk, dasiop, vefstathiou, ikom}@iti.gr

***Abstract*—In this paper we describe SP-ACT, a hybrid framework for the derivation of high-level activity interpretations in context-aware environments, by defining a combination of OWL ontologies and SPARQL `CONSTRUCT` graph patterns. More specifically, the native semantics of OWL is used to formally represent and integrate activity-related information originated from different data sources, whereas SPARQL (SPIN) rules further aggregate activities so as to derive high-level activity abstractions. The goal of the hybrid framework is to address the limitations of the ontology-based context modelling paradigm in domains that require the recognition of complex context elements, namely, the lack of support for (i) temporal reasoning and (ii) new named individual assertions.**

***Keywords*-context, OWL, SPARQL, activity recognition**

I. Introduction

In the recent years, the demand for intelligent, customized user task support has proliferated across a multitude of application domains, ranging from healthcare and smart spaces to transportation and energy control. A key challenge in such applications is to abstract and fuse the captured context in order to elicit an understanding of the situation and to afford services tailored to the user needs; e.g., inferring that an Alzheimer's disease patient left the kitchen to answer an incoming call but failed to resume lunch afterwards, in which case a respective reminder needs to be issued.

The representational and reasoning power afforded by the ontology languages developed for the Semantic Web, and in particular of OWL DL [1] and more recently of OWL 2 [2], have motivated a growing body of research into ontology-based frameworks for modelling and reasoning about context [3]. Under this paradigm, ontologies are used to describe the context elements of interest (e.g. persons, events, activities, location, time), their pertinent logical associations, as well as the background knowledge required to infer additional context information. The captured context information, usually referred to as *low-level context*, is mapped into respective class and property assertions; standard DL inference services are then used to automatically derive the logically implied assertions, also referred to as *higher-level context*.

Despite the benefits entailed by the OWL family of ontology languages, namely modelling complex logical relations and sharing information coming from heterogeneous sources, and the availability of reasoning engines, e.g. [4], [5], that allow to derive higher-level context abstractions, ontology-based context models come with two main shortcomings. The first refers to OWL's lack of support for temporal reasoning; the second shortcoming is that, within OWL, it is not possible to infer and assert new named individuals. The implications are particularly evident in domains that require the recognition of complex context elements, such as human activities that are generally characterized by intricate temporal associations, and where it is often the case that the aggregation of individual activities entails the existence of a new (composite) activity.

To alleviate the lack of inherent temporal reasoning support, typical approaches espoused in the literature include the adoption of an *a-temporal* approach to conceptualization [6], [7], as well as the combined use of ontologies and rules, where the latter are used to establish the relative temporal extensions [8], [9]. As for the inability to assert new named individuals, the majority of relevant approaches suppresses it, implying the existence of some external module that creates new individuals when needed.

In this paper, we propose SP-ACT, a hybrid framework for complex activity recognition that combines ontologies and SPARQL rules (SPIN [10]). Ontologies are used to provide the common vocabulary for representing activity-related contextual information, whereas SPARQL rules derive high-level activity interpretations. SPARQL is used as a standardized declarative language able to address the limitations of the standard OWL semantics mentioned previously. More specifically, the temporal relations among activities are handled by SPARQL functions, whereas the derivation of new composite activities exploits the native capabilities of SPARQL to update the underlying activity model.

The rest of the paper is organized as follows. Section II presents related work in the domain of ontology-based context reasoning architectures. Section III describes the abstract architecture and the provided functionality by SP-ACT. Section IV analyzes the semantics of the SPARQL rules, whereas Section V describes the hybrid architecture that combines OWL ontologies and SPARQL rules in order to derive high-level activity interpretations. Section VI illustrates the reasoning capabilities of SP-ACT through a use case from the healthcare domain and finally, Section VII, concludes our work.

978-1-4673-5077-8/13/$31.00 ©2013 IEEE

II. Related Work

Several ontology-based context reasoning architectures and prototypes have been proposed in the recent past, with a substantial body of work focused on activity recognition [11]. OWL has been widely used within the community for building ontologies; for instance, in a smart home [7], concrete situations correspond to OWL individuals and realization is used to determine into which context concepts a specific situation individual falls. A similar approach is followed in [6], [12], [13], where complex activities are recognized based on subsumption reasoning.

With the emergence of OWL 2, a number of expressivity limitations have been overcome, offering the grounds for modelling and reasoning with complex human activities [14]. In order to cope with the uncertainty aspect in human activities, [15] adopts *log-linear* DLs that support the same operators as the OWL 2 language. Still, one of the main limitations of ontological reasoning for activity recognition is that it does not support temporal reasoning. This shortcoming is partially addressed in [12] by using a sliding time window to aggregate contextual data and generate an activity description instance that serves as input to the subsumption reasoner for (a-temporal) activity recognition.

The need for reasoning with temporal information in activity recognition is discussed in [16] where the issue is addressed by extending ontological reasoning with a temporal characterization of activities, taking into account information on actions recently performed by an actor. Although this approach provides some sort of temporal characterization of a composite activity in terms of the subsequent related activities, the fact that it does not support interval-based temporal reasoning is limiting for capturing complex temporal relations among activities. In [8], a hybrid approach that combines ontological and temporal modelling is presented where OWL ontologies are used for activity modelling and the representation of temporal interval relations. SWRL [17] rules are used for generating composite activity models. However, SWRL rules do not allow for assertion of new individuals, therefore this has to be done externally, similar to [16] and [12]. In [9], the nRQL language of RacerPro is used to detect and assert temporal relations among events. However, the framework only works with RacerPro, using the non-standardized nRQL query language. The SP-ACT framework can be realized on top of any ontology reasoner that supports the execution of SPARQL queries.

Finally, many extensions to the SPARQL language have been proposed for working with temporal streaming data, such as [18], [19] and [20]. Although the temporal processing of RDF streams is currently out of the scope of this work, it is worth noting that the common underlying core rule language (SPARQL) allows for the seamless integration of such frameworks in SP-ACT.

III. The SP-ACT Framework

SP-ACT is an ontology-based activity interpretation framework that relies on the combination of the OWL reasoning paradigm and the execution of SPARQL rules for the recognition of complex activities. More specifically, the native semantics of OWL (and OWL 2) is used to formally represent and integrate activity-related information originated from different data sources (referred to as *atomic activities*), whereas SPARQL rules further aggregate activities, describing the contextual conditions and the temporal relations that drive the derivation of *complex activities*.

The abstract architecture of SP-ACT is depicted in Figure 1 and consists of the *representation* and *interpretation* layers. The representation layer provides the ontology vocabulary for modelling activity-related information, such as *activity types* (in terms of concepts hierarchies), *who* performs the activity, *where* and *when* the activity takes place. Moreover, it supports the representation of two types of activity correlations, namely *classifications* and *compositions*. These correlations derive by the interpretation module that infers complex activities through an iterative hybrid combination of the OWL reasoning and SPARQL rule execution processes. In the following, we briefly present the provided functionality of each layer.

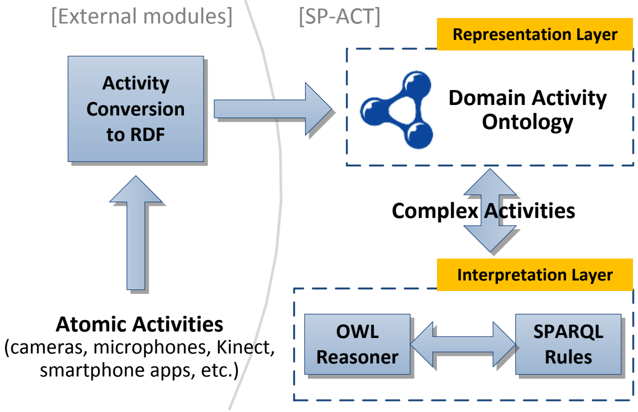

Figure 1. The abstract architecture of SP-ACT.

A. Representation Layer

The representation layer encapsulates a lightweight domain activity model for capturing information relevant to activities. Activity types can be modelled as specializations of the `Activity` core class that allows property assertions to be stated relevant to:

- **Temporal boundaries**: An activity is assumed to take place always inside a time interval, which is defined using the `hasStartTime` and `hasEndTime` datatype (`xsd:dateTime`) properties.

- **Actor**: The actor is the person who performs the activity and they are specified using the `hasActor` property, e.g. a patient.
- **Activity location**: Any information about the location of an activity is defined using the `hasArea` property, e.g. a room.
- **Classifiers**: A detected activity may be further classified in the activity hierarchy based on temporal correlations among other activities, called *classifiers* (see section IV). An activity is associated with its classifiers through the `hasClassifier` property.
- **Sub-activities**: The representation layer allows the modelling of compositions, that is, associations of composite activities with their sub-activities (see section IV). Such associations are represented through the `hasSubActivity` property.

The aforementioned modelling capabilities have been designed with a minimum of semantic commitment to guarantee maximal interoperability. As such, the vocabulary of the representation layer can be aligned with relevant foundational ontologies, such as the SEM [21] and Ontonym [22] ontologies, reusing existing vocabularies for modelling different aspects of activities, such as entities, places, etc.

B. Interpretation Layer

The interpretation layer derives complex activities by meaningfully aggregating and interpreting detected activities through the combination of the OWL ontology reasoning paradigm and the execution of SPARQL `CONSTRUCT` queries. Essentially, the aim of the hybrid architecture is to define a reasoning framework able to deliver key inferencing tasks important in many activity interpretation domains, but not supported by the standard semantics of OWL, such as:

- **Temporal reasoning**: The ability to reason over the temporal extensions of activities is crucial for the successful identification of activity correlations. However, OWL provides no support for temporal reasoning. In order to address this shortcoming in SP-ACT, SPARQL rules are used to handle the temporal dependencies among activities, describing the temporal relations (Allen's temporal operators [23]) and the way contextual information can be combined in order to infer high-level activities.
- **Complex activity correlations**: The schema-level axioms in OWL can model only domains where individuals are connected in a tree-like manner [24][25]. In the activity interpretation domain, however, there is a need to model general relational structures among individuals, i.e. relations among individuals that are not connected. In SP-ACT, this expressive limitation of OWL is addressed by utilizing SPARQL rules [26] for the description of the complex activity correlations that drive the activity recognition procedure.
- **Assertions of named individuals**: In most cases, the derivation of composite activities requires the assertion of new individuals. With OWL, such assertions are only feasible by external reasoning services, since OWL semantics does not allow the modelling of ABox assertions that refer to named individuals not present in the KB. In SP-ACT, this requirement applies in the case of activity compositions, where SPARQL rules generate composite activity individuals by aggregating relevant sub-activities.

IV. High-Level Activity Recognition Rules

SPARQL is used in SP-ACT as a declarative language recommended by the W3C for extracting and updating information in RDF graphs, able to address the limitations of the standard OWL semantics described in the previous section, relevant to activity monitoring domains. SPARQL rules are defined in terms of a `CONSTRUCT` and a `WHERE` clause: the former defines the set of triple patterns that should be added to the underlying activity model upon the successful pattern matching of the triple patterns in the `WHERE` clause. The following example rule implements the `isActorOf` property as the inverse of the `hasActor` property (triple variables are marked by the use of "?").

```
CONSTRUCT {
  ?p isActorOf ?x .
}
WHERE {
  ?x hasActor ?p .
}
```

The SPARQL rules in SP-ACT can be classified in two categories, namely *classifications* and *compositions*:

- **Classification rules**: They are used to further specialize activity instances in the activity hierarchy based on temporal dependencies with other activities, called *classifiers*. Essentially, the classification procedure can be considered as a temporal-centric rule-based variant of the DL instance realization procedure [27], allowing temporal information to drive the computation of the set of instances that belong to a concept.
- **Composition rules**: They generate composite activities, that is, activities composed of other activities, called *sub-activities*. The representation of the composite activities requires the generation of new individuals that are asserted in the activity knowledge base.

V. Hybrid Reasoning Architecture

The standard (a-temporal) semantics of the domain activity ontology presented in Section III-A, such as class subsumption, property restrictions, instance class memberships, property relationships, e.g. transitive, inverse, etc., can be efficiently handled by OWL ontology reasoners (e.g. Pellet

```
Require: $KB_{atomic} \neq \emptyset$
1: repeat
2:     $KB' \leftarrow \emptyset$
3:     $KB_{atomic} \leftarrow KB_{atomic} \cup R_{OWL}(KB_{atomic})$
4:     $KB' \leftarrow R_{SPARQL}(KB_{atomic})$
5:     $KB_{atomic} \leftarrow KB_{atomic} \cup KB'$
6: until $KB' = \emptyset$
```

Figure 2. The hybrid complex activity recognition algorithm

[4] and OWLIM [28]). For example, the inverse relationship between the `isActorOf` and `hasActor` properties mentioned in Section IV can be handled directly by Pellet, without needing to implement custom reasoning services.

The hybrid reasoning architecture of SP-ACT combines the standard reasoning services of OWL reasoners and the extended reasoning services of the interpretation layer. Assuming that KB_{atomic} is a set of atomic activity assertions, R_{OWL} is the OWL reasoning module and R_{SPARQL} is the complex activity recognition module, the algorithm in Figure 2 describes the hybrid reasoning procedure that extends the KB_{atomic} set with additional activity assertions. More specifically, the architecture follows an iterative combination of the results of the two reasoning modules. Initially, the OWL reasoning module is used over the KB_{atomic} set to derive inferences based on the standard OWL semantics ($R_{OWL}(KB_{atomic})$). These inferences are added back to the KB_{atomic} set (line 3) that is subsequently used as the underlying model of the complex activity recognition module (line 4). The additional activity assertions are further added to the KB_{atomic} set (line 5), completing a reasoning iteration. If R_{SPARQL} does not produce any inferences, then the procedure terminates (line 6) with the R_{atomic} set containing both the atomic and the inferred complex activities. Otherwise, a new reasoning iteration begins.

VI. Implementation and Use Case

The SP-ACT framework has been realized using the OWLIM repository as the underlying OWL reasoner, since it provides efficient reasoning and SPARQL-based querying services. In practice, however, any OWL 2 reasoner can be used that supports SPARQL queries, e.g. Pellet.

The SPARQL-based complex activity recognition procedure has been realized using the SPARQL Inferencing Notation (SPIN [10]). In SPIN, SPARQL queries can be stored as RDF triples together with any RDF domain model, enabling the linkage of RDF resources with the associated SPARQL queries, as well as sharing and reuse of SPARQL queries. SPIN supports the definition of SPARQL inference rules that can be used to derive new RDF statements from existing ones through iterative rule application, serving as a ready-to-use framework for the implementation of the activity recognition procedure of SP-ACT.

```
CONSTRUCT {
  ?y a BedExit;
    hasClassifier ?x.
}
WHERE{
  ?x a NightSleep;
    hasStartTime ?st1;
    hasEndTime ?et1;
    hasActor ?p.
  ?y a OutOfBed;
    hasStartTime ?st2;
    hasEndTime ?et2;
    hasActor ?p.
  FILTER(
     :contains(?st1, ?et1, ?st2, ?et2)
  )
}
```

Figure 3. The classification rule for deriving bed exits

In the following, we illustrate the basic capabilities of SP-ACT through a use case from the healthcare domain relevant to night sleep monitoring.

A. Night Sleep Monitoring Use Case

In this scenario, we are interested in monitoring elderly people about nocturia incidences[1] during the night sleep. The setting involves the following atomic activities:

- **Night sleep**: The overall night sleep duration of the person (instance of the `NightSleep` class).
- **Out of bed**: It is detected when the person is out of the bed (instance of the `OutOfBed` class).
- **In bathroom**: It is detected when the person is inside the bathroom (instance of the `InBathroom` class).

The SP-ACT framework can be applied in order to define the rules that semantically interpret and combine the aforementioned atomic activities, so as to derive:

- **Bed exits**: Refer to out of bed activities performed during the night sleep (classification semantics).
- **Nocturia incident**: Inferred when a bed exit activity involves a bathroom visit (composition semantics).

Figure 3 presents the SPARQL rule that implements the classification of `OutOfBed` instances in the `BedExit` class. The rule matches night sleep activities that contain out of bed activities (`contains` temporal SPARQL function) and classifies the latter as `BedExit` activities. Furthermore, the classification dependency between the bed exit and night sleep activity instances is modelled through the `hasClassifier` property.

Figure 4 depicts the composition rule that generates instances of the `Nocturia` class. The `newURI` function is

[1] Nocturia refers to the frequent need to urinate during the night.

```
CONSTRUCT {
  ?new a Nocturia;
    hasStartTime ?st1;
    hasEndTime ?et1;
    hasActor ?p;
    hasSubActivities ?x;
    hasSubActivities ?y.
}
WHERE{
  ?x a BedExit;
    hasStartTime ?st1;
    hasEndTime ?et1;
    hasActor ?p.
  ?y a InBathroom;
    hasStartTime ?st2;
    hasEndTime ?et2;
    hasActor ?p.
  FILTER(
    :contains(?st1, ?et1, ?st2, ?et2)
  )
  BIND(:newURI(?x, ?y) as ?new)
  FILTER NOT EXISTS {?new a [] .}
}
```

Figure 4. The composition rule for inferring nocturia instances

responsible for generating a unique URI for the new individual in order to ensure the termination of the procedure. The new individual is initialized with the same time interval as the bed exit activity (`?st1` and `?et1`) and is associated with its sub-activities through the `hasSubActivity` property.

VII. Conclusion

The combination of ontologies and rules is a key prerequisite for effectively meeting the expressivity requirements when modelling and reasoning about context. In this paper, we presented our approach towards the definition of a hybrid framework for complex human activity recognition, combining the standard reasoning semantics of OWL and the execution of SPARQL rules. The temporal relations among activities, as well as, the generation of new individuals during the recognition of composite activities are handled by the native expressivity of SPARQL that allows the definition of custom temporal SPARQL functions and the update of the underlying data model.

The framework, namely SP-ACT, has been realized using the OWLIM repository as the underlying ontology reasoner and the SPIN vocabulary to represent and execute the SPARQL rules. Our future work will focus on defining an abstraction layer that would support the *dynamic* generation of the SPARQL-based activity recognition rules in terms of an activity metamodel.

Acknowledgment

This work has been supported by the EU FP7 project Dem@Care: Dementia Ambient Care – Multi-Sensing Monitoring for Intelligent Remote Management and Decision Support under contract No. 288199.

References

[1] D. L. McGuinness and F. van Harmelen, "OWL web ontology language overview," World Wide Web Consortium, W3C Recommendation, Feb. 2004.

[2] B. C. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider, and U. Sattler., "OWL 2: The Next Step for OWL," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 6, no. 4, pp. 309–322, October 2008.

[3] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, and D. Riboni, "A survey of context modelling and reasoning techniques," *Pervasive Mob. Comput.*, vol. 6, no. 2, pp. 161–180, Apr. 2010.

[4] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, and Y. Katz, "Pellet: A practical OWL-DL reasoner," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 5, no. 2, pp. 51–53, 2011.

[5] V. Haarslev and R. Möller, "Racer: A Core Inference Engine for the Semantic Web," in *Proceedings of the 2nd International Workshop on Evaluation of Ontology-based Tools (EON2003), located at the 2nd International Semantic Web Conference, Sanibel Island, Florida, USA*, 2003, pp. 27–36.

[6] D. Riboni and C. Bettini, "Cosar: hybrid reasoning for context-aware activity recognition," *Personal Ubiquitous Comput.*, vol. 15, no. 3, pp. 271–289, Mar. 2011. [Online]. Available: http://dx.doi.org/10.1007/s00779-010-0331-7

[7] L. Chen and C. D. Nugent, "Ontology-based activity recognition in intelligent pervasive environments," *International Journal of Web Information Systems*, vol. 5, no. 4, pp. 410–430, 2009.

[8] G. Okeyo, L. Chen, W. Hui, and R. Sterritt, "A hybrid ontological and temporal approach for composite activity modelling." in *TrustCom*, G. Min, Y. Wu, L. C. Liu, X. Jin, S. A. Jarvis, and A. Y. Al-Dubai, Eds. IEEE Computer Society, 2012, pp. 1763–1770.

[9] M. Wessel, M. Luther, and M. Wagner, "The difference a day makes - recognizing important events in daily context logs," in *C&O:RR*, 2007.

[10] H. Knublauch, J. A. Hendler, and K. Idehen, "SPIN - overview and motivation," World Wide Web Consortium, W3C Member Submission, Feb. 2011.

[11] L. Chen, J. Hoey, C. Nugent, D. Cook, and Z. Yu, "Sensor-based activity recognition," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 42, no. 6, pp. 790 –808, nov. 2012.

[12] L. Chen, C. D. Nugent, and H. Wang, "A knowledge-driven approach to activity recognition in smart homes," *IEEE Trans. on Knowl. and Data Eng.*, vol. 24, no. 6, pp. 961–974, Jun. 2012.

[13] X. Wang, D. Zhang, T. Gu, and H. Pung, “Ontology based context modeling and reasoning using owl,” in *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on*, march 2004, pp. 18 – 22.

[14] D. Riboni and C. Bettini, “Owl 2 modeling and reasoning with complex human activities,” *Pervasive and Mobile Computing*, vol. 7, no. 3, pp. 379 – 395, 2011.

[15] R. Helaoui, D. Riboni, M. Niepert, C. Bettini, and H. Stuckenschmidt, “Towards activity recognition using probabilistic description logics,” in *Activity Context Representation: Techniques and Languages, AAAI Technical Report WS-12-05*, 2012, pp. 26 – 31.

[16] D. Riboni, L. Pareschi, L. Radaelli, and C. Bettini, “Is ontology-based activity recognition really effective?” in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, march 2011, pp. 427 –431.

[17] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, “SWRL: A Semantic Web Rule Language Combining OWL and RuleML,” W3C Member Submission, National Research Council of Canada, Network Inference, and Stanford University, Tech. Rep., May 2004.

[18] J. Tappolet and A. Bernstein, “Applied Temporal RDF: Efficient Temporal Querying of RDF Data with SPARQL,” in *Proceedings of the 6th European Semantic Web Conference on The Semantic Web*. Heraklion, Crete, Greece: Springer-Verlag, 2009, pp. 308–322.

[19] A. Bolles, M. Grawunder, and J. Jacobi, “Streaming SPARQL extending SPARQL to process data streams,” in *Proceedings of the 5th European semantic web conference on The semantic web: research and applications*, ser. ESWC’08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 448–462.

[20] D. F. Barbieri, D. Braga, S. Ceri, E. D. Valle, and M. Grossniklaus, “Querying rdf streams with c-sparql,” *SIGMOD Rec.*, vol. 39, no. 1, pp. 20–26, Sep. 2010.

[21] W. R. van Hage, V. Malaisé, R. Segers, L. Hollink, and G. Schreiber, “Design and use of the Simple Event Model (SEM),” *J. Web Sem.*, vol. 9, no. 2, pp. 128–136, 2011.

[22] G. Stevenson, S. Knox, S. Dobson, and P. Nixon, “Ontonym: a collection of upper ontologies for developing pervasive systems,” in *Proceedings of the 1st Workshop on Context, Information and Ontologies*, ser. CIAO ’09. New York, NY, USA: ACM, 2009, pp. 9:1–9:8.

[23] J. F. Allen, “Towards a general theory of action and time,” *Artif. Intell.*, vol. 23, no. 2, pp. 123–154, Jul. 1984.

[24] M. Y. Vardi, “Why is modal logic so robustly decidable?” in *Proceedings of a DIMACS Workshop on Descriptive Complexity and Finite Models*, ser. DIMACS Series in Discrete Mathematics and Theoretical Computer Science. American Mathematical Society, Jan. 1996, pp. 149–184.

[25] B. Motik, B. Cuenca Grau, and U. Sattler, “Structured objects in OWL: representation and reasoning,” in *Proceedings of the 17th international conference on World Wide Web (WWW ’08)*. New York, NY, USA: ACM, 2008, pp. 555–564.

[26] J. Pérez, M. Arenas, and C. Gutierrez, “Semantics and complexity of SPARQL,” *ACM Trans. Database Syst.*, vol. 34, no. 3, pp. 16:1–16:45, Sep. 2009.

[27] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, Eds., *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[28] B. Bishop, A. Kiryakov, D. Ognyanoff, I. Peikov, Z. Tashev, and R. Velkov, “OWLIM: A family of scalable semantic repositories,” *Semant. web*, vol. 2, no. 1, pp. 33–42, Jan. 2011.