# An Ontology-Based Framework for Semantic Data Preprocessing Aimed at Human Activity Recognition

Rosario Culmone, Marco Falcioni, Michela Quadrini
Computer Science Division, School of Sciences and Technologies, University of Camerino
Email: {firstname.lastname}@unicam.it

***Abstract*—Over the last few years, complex systems which collect data from a considerable number of sources are increasing. However, it is not always possible to have a clear overall view of the information contained within data, due to both their granularity and to their wide amount. Since an analysis procedure able to take into account the semantics of records is often needed, ontologies are becoming widely used to describe the domain and to enrich the acquired data with its significance. In this paper, we propose an ontology-based methodology aiming to perform semantic queries on a data repository, whose records originate from a network of heterogeneous sources. The main goal of such queries is the pattern matching process, i.e., recognition of specific temporal sequences in fine-grained data. In our framework, benefits deriving from the implementation of a domain ontology are exploited in different levels of abstraction. Thereafter, reasoning techniques represent a preprocessing method to prepare data for the final temporal analysis. Our proposed approach will be applied to the ongoing AALISABETH, an Ambient Assisted Living project aimed to discover and manage the behaviour of monitored users.**

***Keywords - Ontology; Semantic Reasoning; Complex Event Processing.***

## I. INTRODUCTION

In complex data-acquisition environments, the storing of data as well as the information carried by such records become more and more important. When data are generated by many heterogeneous sources, it turns out to be important both the integration of information and the interoperability of applications that process the data. Usually, these records are collected in a data repository and it sometimes results difficult to have a clear view of the whole acquired information. Therefore, it could be even more hardly to proceed with an analysis which do take into account the semantics of data. For this reason, ontologies are becoming more and more utilized to address this issue, because they are able to describe instances of a real-world system.

An example of the described situation could be represented by an Ambient Assisted Living (AAL) context. In such domestic environment a wide network of smart objects is installed, whose task is to provide the possibility to monitor the user lifestyle. In order to reach this aim, the Smart Home (SH) relies on many different types of objects: from clinical devices for the user health to indicators of presence, from temperature and humidity measurements to fridge and door opening sensors. Considering that the storing data repository, usually a Database (DB), often shows a lack of semantic information and relationships among the smart home components, acquired data from smart objects need to be treated according to their significance. Hence, data processing cannot prescind from the implementation of a domain ontology, whose primary scope is to entirely describe actors belonging to the smart home, i.e., user, smart objects and their relationships. Thereby, data can be treated according to their semantic, which is formalized in the domain ontology. Subsequently, the same ontology can be enriched by rules for a further analysis phase of the system. In fact, it can happen that several concepts are known, but they are not yet present in the data repository nor in the ontology. If such knowledge is needed for the successive phase of analysis, it can be introduced in the ontology.

In this paper, a framework capable to address the illustrated context is presented. The described methodology, in addition to pattern discovery techniques, has been developed to answer to the requirements of an Ambient Assisted Living project. The ongoing Ambient-Aware LIfeStyle tutoring for A BETter Health (AALISABETH) project aims to analyse the user's lifestyle by means of a non pervasive sensor network, which can monitor and detect well-specific daily activities. In particular, the main goal of this project is to detect a set of abnormal behaviours that could eventually bring to an onset of the most common diseases. In the present paper, we intend to discuss a novel methodology that consists of comparing the observed activities to those formalized in the ontology. Hence, the final task of the framework is to determine whether the prearranged patterns are matched, and thereafter communicate such results to caregivers.

This paper is structured as follows: Section II examines the related literature concerning the topics addressed in this work. Section III firstly explains the motivation of the proposed methodology, then provides a detailed description of the framework architecture and lists the tools used to implement each component. Finally, Section IV illustrates the work in progress and the nearly future development of the ongoing project.

## II. RELATED WORK

The approach presented here includes different areas of research: ontology-based description of a domain, mapping a Database to an existing ontology and enrich external data with their significance, semantic data preprocessing, pattern matching and identification in a sequence of data.

Ontologies are commonly used to explicitly formalize and specify a domain of knowledge [1]. Furthermore, they improve the automation of integration of heterogeneous data sources, also providing a formal specification of the vocabularies of concepts and the relationships among them [2]. Many are

 ISBN: 978-1-61208-355-1 

the publications in which ontologies are employed to achieve information integration over various domains. An example for Intelligent Environments are found in [3] [4] [5] , where an ontology is essentially implemented for both formally expressing the domotic environment (e.g., sensors, gateways and network) and providing reasoning mechanisms. This reasoning allows to support automatic recognition of device instances and to verify the formal correctness of the model. Further works presenting ontologies finalized to AAL activities are Mocholi et al. [6] and Fleury et al. [7].

Techniques of mapping an external Database to a local ontology are suggested by Sedighi and Javidan [8] and Barrasa et al. [9]. Also, tools that automatically generate OWL ontologies [10] from database schemas have been presented, for instance by Cullot et al. [11] and Rodriguez-Muro et al. [12].

Ontologies may also support a semantic approach to applications involving Business Process Management (BPM) techniques and analyses of processes based on a list of recorded events, i.e., Process Mining. In this case, a possible procedure is to enrich the event logs coming from external data sources by using ontology based data integration, as observed by Tran Thi Kim and Werthner [13]. Furthermore, a similar methodology used to integrate semantic annotation to the event log is illustrated in a BPM context by Ferreira and Thom [14], where semantic reasoning is used to automatically discover patterns from the recorded data.

Since in [14] only sequences of determined data are relevant, time constraints among events may not be strictly taken into account. Considering the temporal nature of activities as a succession of actions admits several feasible approaches, such as the probabilistic [15] and the statistical one [7].

Instead, in the field of activity recognition, time interval restrictions become essential. Cases of dealing with complex events are rapidly increasing. To address this issue, ontologies are used as a basis to preserve information and relationships among events. Thereafter, they are temporally managed by a Complex Event Processor (CEP), yielding to a semantic complex event processing technique [16].

## III. Methodology

### A. Motivation

Our proposed methodology originates from the necessity to deal with the significance of a wide amount of heterogeneous data, which are commonly stored in a data repository. Since the beginning of the entire procedure, the final goal of the analysis is well-known, as well as a detailed awareness of the whole system and records thereby acquired. Furthermore, one should focus not only on the single values of data, rather than on its meaning within the context. In order to take into account such relationships and formalize the knowledge of the whole context, the implementation of an ontology results to be actually mandatory. The general approach can be illustrated by Figure 1. On top, the real-world system is composed of both static knowledge and data generated by the considered system. As the former is fixed, the user is allowed to directly transfer his domain knowledge into the corresponding ontology. On the other hand, the latter produces a stream of data which is collected by the repository. Since in this step data are usually registered as a list of records, they show a fine-grained nature, carrying generally their value, originating device, data type, timestamp, and so on. In a similar context, the granularity features of acquired data are a stumbling block for the contained semantic information, which may be eventually lost. Also, a further verifiable aspect is data redundancy; that is, there can be several devices which apparently output different results, but they provide the same information. Hence, the ontology is introduced to somehow circumvent such technical aspects and to form a bridge from the real-world system and its formal representation. In fact, it is able to merge the static knowledge and the dynamic parts by means of classes and their instances, rebuilding the whole context. Therefore, the advantages of a semantic technique are exploited twice. Once the ontology-based method has provided a conceptualization and specific description of the real-world system, such formalization drives the analysis phase. In our specific case, it is needed to look for well-determined set of data. It is worth noting that such research has to be performed according to the own semantics of the desired set. This requirement represent the main reason why an ontology-based technique is introduced.

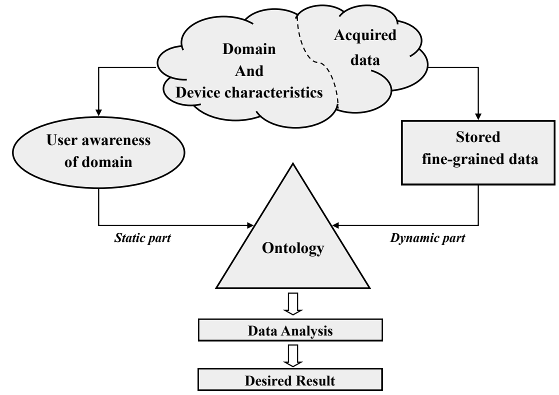


Figure 1. General approach: from real world to ontology

### B. Architecture of the framework

In order to address the presented situation, we propose the framework depicted in Figure 2. One of the most common methods to collect data from a network of heterogeneous sources is to store them in a DB. Therefore, our first **necessity** is the possibility to somehow find a correspondence between the elements of a DB and the ones of the previously implemented ontology. Such a semantic model is built following a precise structure, as described in detail later. Once data are reorganized according to their meaning, the ontology plays a preprocessing role. In fact, the user can express semantic queries in order to extract from the ontology a well-specific aspect of the entire environment. It is worth noting that **some** particular views could not be previously retrieved from the fine-grained nature of the data stored in the DB. These different views may be considered as the output of sensors which are not physically present in the system, and we can label them as *virtual* sensors.

As far as time constraints are not taken into account, an ontology is sufficient to classify and organize data produced

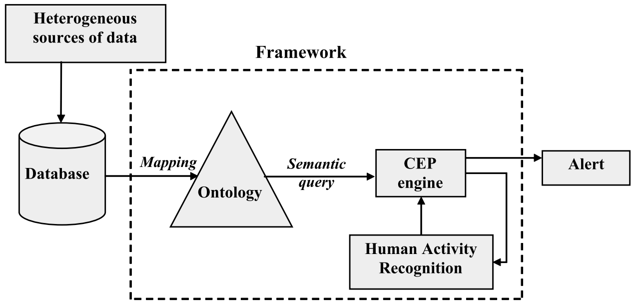


Figure 2. Multi step methodology of the framework

from both physical and virtual sensors. However, since our final aim is to obtain a specific time-dependent output, we need to introduce in our framework a component able to manage these time restrictions. This issue is solved by the use of a Complex Event Processing (CEP) engine, that is, a technique concerned with timely detection of compound events within streams of simple events [17].

### C. Ontology structure

In our proposed framework, the main element is represented by the ontology that clearly defines the semantics of the considered domain and is used as a shared knowledge base for all the related components.

This ontology, called OntoAALISABETH, has a particular approach, as illustrated in Figure 3. Four main domain ontology systems - User, Environment, Activity and Device - represent the knowledge base in AAL context. User describes the concepts related to user's profile, while Activity describes several domestic activities that are necessary to detect abnormal behaviour. These two parts play the central role. Consequently, the appliances within the AAL environment should adapt to the user, and not vice versa. Then, Environment and Device describe user's house and the sensors network installed.

Furthermore, this ontology shows different abstraction layers

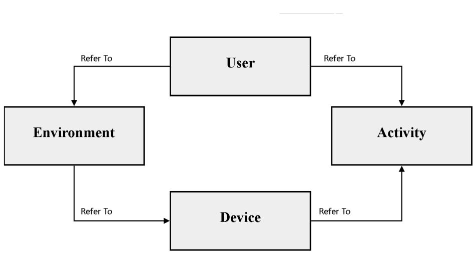


Figure 3. Context ontology overview.

that composed together form a pyramid-like structure, where each lower level specialises the one on the next upper layer.

The architecture, as reported in Figure 4, is realized by the following main components:

- A static layer (domain and domain-specific ontology);
- A dynamic layer (data and view ontology).

Each part of our ontology plays a specific role in order to respond to different requirements of the project, as described below.

*1) Domain ontology:* Initially, an upper domain ontology is built. One should note that this higher level of abstraction can be considered as a ready-to-use ontology for any other analogue domain. In other words, it consists of an ontology which generally formalizes concepts present in some context, and is thought to be commonly valid. In fact, concepts are described as much generally as possible, carrying *static* information. Since our instance is an AAL context, as the literature suggests, we implemented a domain ontology extending and reusing an existing one. In our case, the starting ontology has been chosen to be DogOnt [3]. It has been built in a smart home context, but does not take into account several elements of an AAL environment. Therefore, we have formalized classes and relationships about the SH, its architecture and furniture, the presence and activities of one or more users, the introduction of smart objects with a communication network, sensors and clinical devices, and so on.

*2) Domain-specific ontology:* This first middle layer places below the previous upper ontology, extends several static properties and focuses on the structure of the considered domain. In our domain-specific ontology, we formalize the various components belonging to the home environment: the real structure of the ambient and disposition of rooms, the personal information about who lives in the house, which sensors are installed in the network and how they communicate. Also, the complete knowledge of the domain allows the developer to add new elements and relationships in the ontology, which cannot be described in the technology of data storing.

*3) Data ontology:* The data ontology extends the previous domain-specific layer introducing the concept that each device generates fine-grained data. In this level, the described classes are instantiated with individuals that present a one-to-one correspondence with each record stored in the DB. This procedure is allowed by a technique known as Ontology-Based Data Access (OBDA) approach [12]. It consists of a *mapping* that associate data from the data sources with concepts in the ontology. In particular, by means of suitable SQL queries over the DB one extracts records and propagates them into concepts. Hence, the whole data ontology is implemented taking into account the sensor network, formalized in the previous layer, and is continuously updated. In this step, the semantic information about the fine-grained data is partially recovered, but the following layer permits to have custom specific views of the system.

*4) View ontology:* In our system, data are generated by the non pervasive network which is installed to monitor user lifestyle. In particular, such records may assume different meanings depending on the specific context. For instance, if a presence in the bedroom is followed by one in the kitchen, it has a different meaning from the same followed by one in the bathroom. Since a particular record deserves different semantic treatments, the view ontology takes into account such various circumstances. More frequently, one must evaluate the presence in the bedroom from different points of view. In terms of an ontology, this necessity converts to the implementation of

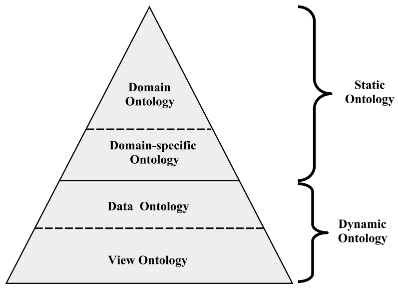


Figure 4. Pyramid-like structure of the ontology

new view classes where individuals are inferred. So, alternative views provided by this lower layer are needed in order to reorganize instances of data ontology. These views are defined by the expression of several equivalent classes. They are driven by the main scope to classify instances having well-determined properties and relationships; that is, these classes are populated by the desired individuals and carry the same knowledge replicated several times. The whole process of reorganization is allowed by the use of the reasoning tools, which represents the formal basis for the expressive strength of OWL. In fact, through this instrument, one can obtain additional statements that are inferred from the facts and axioms previously asserted. This reviewing step is the grounding of the *preprocessing* procedure. Thereafter, the reasoning tool allows to perform semantic queries on the ontology and extract the desired information for the following effective analysis, as reported in Figure 2. One should note that querying the ontology in this final step of the proposed methodology corresponds to select an amount of data generated by virtual sensors, i.e., a group of data following the user *interpretation* of the system. Moreover, this approach developed by means of inference classes has the important advantage to be extensible and additive.

In order to better explain the advantages deriving from the classification of the view ontology, let us consider the following cases. One of the most relevant aspects for our project is monitoring if the user gets up during the night for eating or toileting. In order to recognize these activities, we proceed creating two views, i.e., macro ontology classes. Each class contains all inferred individuals that allow the eventual recognition of the considered activity. In this particular case, the information about getting up and exiting from the bedroom are common. Instead, presence and utilization of the toilet is found in the first case, while presence in the kitchen and opening a sideboard or refrigerator belong to the second view. Furthermore, in both cases we require that the person comes back to the bedroom after some time and continues to sleep. Hence, these sets of individuals populating the view classes are selected as input for the following step of analysis. It is worth noting that processing data with the described technique allows to preserve relationships and constraints introduced by the previous domain-specific layers of the ontology. Contents of each layer of the pyramid-like structure are shown in Figure 5.

### D. Process analysis

The first component of the framework previously described employs traditional Semantic Web (SW) techniques, e.g., query languages and automated reasoning. However, for a

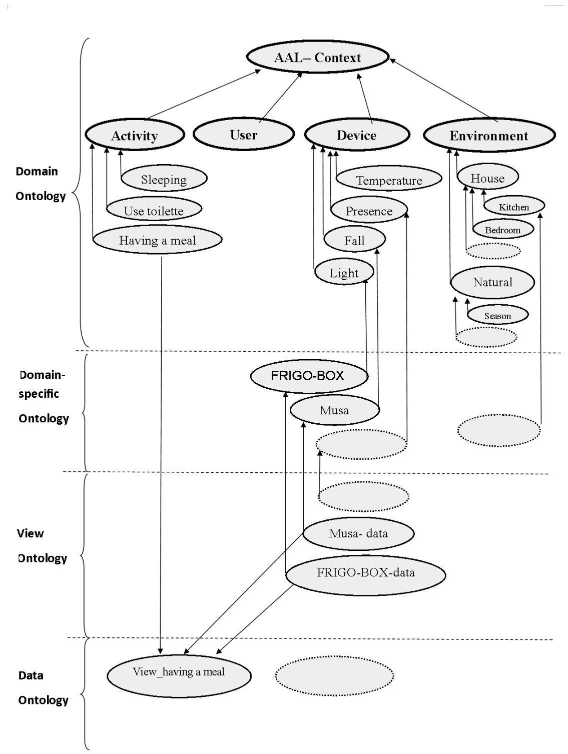


Figure 5. Class hierarchy diagram of OntoAALISABETH

dynamically changing dataset such traditional methods do not allow to perform reasoning over time and space, which is necessary to capture some of the important characteristics of streaming data and events. Since our goal is to monitor certain specific human activities in a domestic environment, we introduce a CEP engine in order to perform the temporal analysis procedure. This engine allows to combine data from multiple sources to infer events or patterns that suggest more complicated circumstances. In fact, the main objective is to recognize significant events. These identifications could be eventually reused to discover further more complex events, through additional uses of CEP engine.

### E. Implementation of the framework

The OWL ontology is developed and tested in Protégé 4.3 [18], together with the Pellet Reasoner Plug-in [19], which permits the creation and population of equivalent classes. Through the Protégé Plug-in OBDA [12], we write down the statements that map the Database to the ontology, in order to enable the possibility of extracting data from the DB, which was written in MySQL. To implement the framework, we use Java as a coding language to combine several techniques. Thereby, we call functionalities of the OBDA Plug-in to establish a connection to the DB and effectively load the records in the ontology. Then, the ontology is managed by means of the OWL API. Thereafter, the Pellet reasoner is invoked through Jena [20] to perform reasoning over the ontology together with the individuals. The SPARQL query is also executed through Jena. Basically, using Jena we load the ontology file created with Protégé into an ontology model (a Java object implementing the `OntModel` interface). We then choose to utilize Esper as CEP tool for several reasons: its open-source Java library for complex event processing, it can be used in different data stream and CEP applications, it has adapters that allow the user to provide different input formats for the representation of events. The whole Java framework is developed using Eclipse IDE [21].

## IV. Conclusion and Future Work

In this paper, we have illustrated an ontology-based framework to retrieve semantic information from a data repository lacking of the original significance. The ontology represents the central element of the presented methodology, and is basically composed of four layers: a top-level ontology followed by a domain-specific one, and data layer which establishes over a final basis-view layer. This last part is thought as a data preprocessing step. It plays the role to organize data according to the desired context views, in order to allow a proper analysis. In the near future work, we intend to focus on the last part of the framework and carry out temporal pattern identifications. A further development of the CEP analysis method is needed to effectively perform recognition of pre-determined human activities. Once detected such behavioural events, they will be evaluated by means of the CEP engine, and compared with the existing recognition techniques, e.g., Bayesian networks, Hidden Markov Models, Learning Machine. Also, a feasible refinement to classify data will be the definition of custom SWRL rules, and their integration with the existing inference classes.

The ensemble of certain specific actions or behaviours can be considered as markers of some of the most common diseases affecting old people. Hence, discovering such behavioural sequences which commonly characterize diagnostic suspects represents the main motivation of the ongoing AAL-ISABETH project.

However, the AAL represents just one of the many possible domains of application for the introduced approach. Finally, another eventual domain of use could be a Smart City. Such modern urban system of devices connected in a common network has the intent to improve the quality of life and a sustainable economic development. A Smart City is an example of real-time monitoring system in a larger scale, and presents similarities to our dynamic and heterogeneous features. Hence, the proposed approach prescind from the size of the domain of application and can be proposed to manage the fine-grained data generated by heterogeneous networks.

## Acknowledgements

The authors would like to thank every partner of AALISABETH project for the great working collaboration done until now. They also acknowledge the financial contribution of the Marche Region administration for supporting the research on the AALISABETH project.

## References

[1] T. R. Gruber. (retrieved: March, 2014) What is an Ontology? [Online]. Available: http://www-ksl.stanford.edu/kst/what-is-an-ontology.html

[2] M. Gagnon, "Ontology-Based Integration of Data Sources," in *10th International Conference on Information Fusion*, Quebec, Canada, July 2007, pp. 1–8.

[3] D. Bonino, E. Castellina, and F. Corno, "The DOG gateway: Enabling Ontology-based Intelligent Domotic Environments," *IEEE Trans. Consumer Electronics*, vol. 54, pp. 1656–1664, November 2008.

[4] T. Gu, X. H. Wang, H. K. Pung, and D. Q. Zhang, "An Ontology-based Context Model in Intelligent Environments," in *Proceedings of Communication Networks and Distributed System Modeling and Simulation Conference*, San Diego, California, USA, January 2004, pp. 270–275.

[5] D. Preuveneers, J. V. den Bergh, D. Wagelaar, A. Georges, P. Rigole, T. Clerckx, Y. Berbers, K. Coninx, V. Jonckers, and K. D. Bosschere, "Towards an Extensible Context Ontology for Ambient Intelligence," in *Second European Symposium on Ambient Intelligence*, ser. LNCS. Eindhoven, Netherlands: Springer, November 2004, pp. 148 – 159.

[6] F.-L. C. J. B. Mocholí, Sala Pidd and N. J. C., "Ontology for Modeling Interaction in Ambient Assisted Living Environments," in *Proc. IFMBE XII Mediterranean Conference on Medical and Biological Engineering and Computing 2010*, Chalkidiki, Greece, May 2010, pp. 566–658.

[7] A. Fleury, M. Vacher, N. Noury, and S. Member, "SVM-Based Multimodal Classification of Activities of Daily Living in Health Smart Homes: Sensors, Algorithms, and First Experimental Results," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, pp. 274–283, March 2010.

[8] S. M. Sedighi and R. Javidan, "Semantic Query in a Relational Database using a Local Ontology Construction," *South African Journal of Science*, vol. 108,, pp. 97–107, Oct 2012.

[9] J. Barrasa, Óscar Corcho, and A. Gómez-Pérez, "R2O, an Extensible and Semantically based Database-to-Ontology Mapping Language," in *in In Proceedings of the 2nd Workshop on Semantic Web and Databases(SWDB2004)*. Toronto, Canada: Springer, August 2004, pp. 1069–1070.

[10] (retrieved: May, 2014) OWL Web Ontology Language Document Overview (Second Edition). [Online]. Available: http://www.w3.org/TR/owl2-overview/

[11] N. Cullot, R. Ghawi, and K. Ytongnon, "DB2OWL:A Tool for Automatic Database-to-Ontology Mapping." in *SEBD'07*, Brindisi, Italy, June 2007, pp. 491–494.

[12] M. Rodriguez-Muro, L. Lubyte, and D. Calvanese, "Realizing ontology based data access: A plug-in for protg," in *In Proc. of the Workshop on Information Integration Methods, Architectures, and Systems (IIMAS 2008)*. Cancun, Mexico: IEEE Computer Society, April 2008, pp. 286–289.

[13] T. Tran Thi Kim and H. Werthner, "An Ontology Based Framework for Enriching Event Log Data," in *SEMAPRO 2011, The Fifth International Conference on Advances in Semantic Processing*, Lisbon, Portugal, November 2011, pp. 110–115.

[14] D. R. Ferreira and L. H. Thom, "A Semantic Approach to the Discovery of Workflow Activity Patterns in Event Logs," *International Journal of Business Process Integration and Management*, vol. 6, pp. 4–17, July 2012.

[15] D. J. Patterson, D. Fox, H. Kautz, and M. Philipose, "Fine-Grained Activity Recognition by Aggregating Abstract Object Usage," in *Proceedings of the Ninth IEEE International Symposium on Wearable Computers*, ser. ISWC '05, 2005, pp. 44–51.

[16] K. Taylor and L. Leidinger, "Ontology-driven complex event processing in heterogeneous sensor networks," in *Proceedings of the 8th Extended Semantic Web Conference on The Semantic Web: Research and Applications*, Heraklion, Crete, Greece, June 2011, pp. 285–299.

[17] D. Anicic, P. Fodor, S. Rudolph, and N. Stojanovic, "Ep-sparql: A unified language for event processing and stream reasoning," in *Proceedings of the 20th International Conference on World Wide Web*. Hyderabad, India: ACM, April 2011, pp. 635–644.

[18] (retrieved: June, 2014) Protege. [Online]. Available: http://protege.stanford.edu/

[19] (retrieved: June, 2014) Pellet. [Online]. Available: http://clarkparsia.com/pellet/protege/

[20] (retrieved: June, 2014) Apache jena. [Online]. Available: http://jena.sourceforge.net/

[21] (retrieved: June, 2014) Eclipse. [Online]. Available: https://www.eclipse.org/

 