Context ontology for Event-Driven Information Systems

Ana Šaša Bastinos, Marjan Krisper
Information Systems Laboratory
Faculty of Computer and Information Science, University of Ljubljana
Ljubljana, Slovenia
{ana.sasa, marjan.krisper}@fri.uni-lj.si

***Abstract*—Event-driven architecture and complex event processing have become important topics in achieving business reactivity and proactivity. However, the use of these approaches in real-world solutions remains limited. One of the reasons for this is insufficient support for semantics in capturing and defining of complex events. In this paper, we address this problem. We present a framework for ontology-based support for complex events, which allows for semantically enriched event definitions and automatic recognition. In order to automatically recognize complex events, an appropriate event definition basis has to be defined. The paper represents a continuation of our previous work by enhancing this basis and applying the aspectual model from the field of linguistics to our base event ontology. We believe that this framework will provide a generic approach that will allow for complex event recognition for events of various complexities and from different domains.**

Keywords-event; complex event; ontology; aspectual model.

I. INTRODUCTION

With increasing demands for agility and reactivity of business processes, scientific circles and leading information technology companies have paid a lot of attention to EDA (event-driven architecture) and CEP (complex event processing). EDA and CEP enable event-driven systems – systems in which actions result from business events [1]. Despite the recognized need for support of events to improve automation, signal processing, data acquisition, manufacturing, computer aided simulations, and business activity monitoring [1][2], the use of EDA and CEP approaches in real-world solutions remains limited. One of the reasons for this is insufficient support for semantics in capturing and defining of complex events (CE) [3]. Existing EDA and CEP approaches do not take into consideration different expressivity requirements that are needed in definition of a large number of diverse CE. Because of this, detection of CE that require semantically expressive descriptions is not automated and experts are required to monitor business operation in order to determine if a complex event has occurred. Such an approach can decrease reactiveness and proactiveness of an organization [3]. We addressed some of these issues in our previous work presented in [3], where we discussed ontology-based framework for complex events. In this paper, we propose an improved ontology basis for this framework. We improve the semantic event definition and recognition by development of a complex event ontology based on Tremblay's adjustment of Pustejovsky's aspectual model [4]. The purpose of applying this cognitive approach to the complex event ontology is to enable richer semantic complex event definitions that better reflect real world events thus providing an improved mechanism for complex event detection.

The paper is structured as follows. In the next section, we introduce the main concepts of event-driven architectures. In section III., we describe our ontology-based framework for complex events. In section IV, we present Pustejovsky's aspectual model and Tremblay's adjustment of this model. In section V, we present our application of this model to our ontology-based framework for complex events. In section VI, we provide final conclusions and discuss our further work.

II. EVENT-DRIVEN ARCHITECTURE

An entity is considered event-driven when it acts in response to an event. EDA is a paradigm that describes an approach to information systems development with a focus on developing an architecture that has the ability to detect events and react intelligently to them [5]. EDA represents a complement to the service-oriented architecture (SOA), which has become one of the most recognized paradigms in information systems development in the recent years [6]. By enhancing the paradigm of SOA, enterprises can improve their ability for business transformation by implementing event-driven architectures that automatically detect and react to significant business events [5]. An important part of every EDA that enables and predetermines to what level a system is able to detect and respond to complex events is complex event processing (CEP). CEP is computing that performs operations on complex events, including reading, creating, transforming, or abstracting them [7]. CEP systems can be classified as advanced decision support systems [3]. In comparison with other types of decision support systems that are not event driven, CEP systems focus on increasing system reactivity and proactivity based on information carried by member events. Events can be simple or complex. A simple event is as an event that is not an abstraction or composition of other events. A complex event is an event that is an abstraction of other events called its members [7].

III. ONTOLOGY-BASED FRAMEWORK FOR COMPLEX EVENTS

In this section, we discuss results of our previous work [3], which are the basis for the research presented of this

 ISBN: 978-1-61208-184-7

paper. In [3], we observed that semantic descriptions providing effective and expressive models for understanding of CE structures and their processing could be very helpful for definition and automation detection of CEs. We presented a framework that enables highly expressive event models and is based on ontologies and the Web Ontology language (OWL) [8]. We see ontologies and ontology representation languages as an opportunity to effectively deal with CEs in EDAs. Ontologies intrinsically provide a means for highly expressive semantic descriptions. In our framework, they are used to semantically describe CEs through conceptualizations of member events they are composed of. We use OWL (Web Ontology Language) as the ontology representation language, because it provides a very appropriate foundation for CE definitions and has a wide support for reasoning. We developed a service that makes part of an EDA and enables translation of event data to OWL, detection of CEs and their triggering. It can act as an event source and as an event sink, which makes our approach complementary to existing approaches to support CEs that require higher expressivity and semantic descriptions. Our framework has been used in a case study project for electrical distribution domain, where it has been shown to be very useful and has improved the overall system flexibility and reactivity.

Our event-driven architectural framework defines a Complex Event Service (CES) that makes part of the overall event-driven service oriented architecture. Fig. 1 illustrates a high-level structure of the CES.

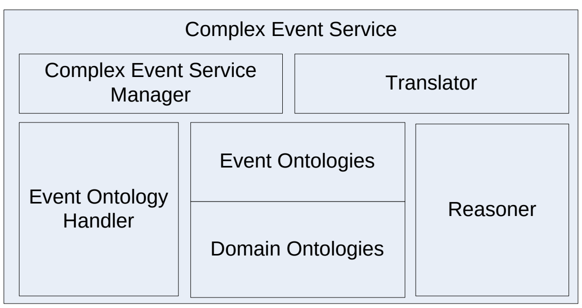


Figure 1. Components of the Complex Event Service

The CES is responsible for detection of CEs based on their complex event definitions and member event occurrences. It acts as an event sink for member events of the CEs it is responsible for, and it acts as an event source for the detected CEs. It catches events and triggers CEs when it detects them based on their definition and member events occurrences. The CES consists of five components (Fig. 1): Complex Event Service Manager, Translator, Ontology, Event ontology handler, and Reasoner. Ontology comprises event and domain ontologies. It is a passive entity that is handled by the Event Ontology Handler and used by the Reasoner to infer new information based on the existing information in the ontology. It comprises information about the domain, about events and event types. An event ontology supports one CE type or several related CE types. We say that an event ontology is subscribed to all events that are members of the CEs that the ontology defines. CES Manager is the component that links the CES with its environment by receiving member events and triggering detected CEs. It also orchestrates the other active components of the CES into a complete process. In the remainder of this section we describe the event ontologies and domain ontologies. For more details about other components of the framework please refer to [3].

Domain ontologies comprise information about the domain where the system is used, for example the electrical distribution domain. For creation and maintenance activities of the domain ontologies, different generic ontology development methodologies that are available can be used. The basic structure of an event ontology is defined by the base event ontology. Fig. 2 illustrates the base event ontology from our previous work.

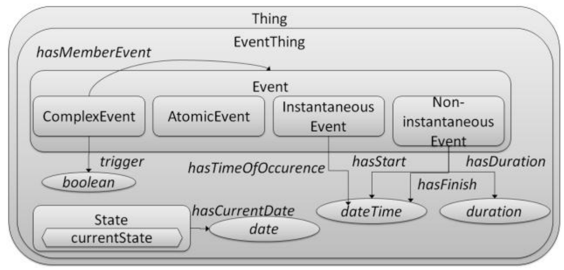


Figure 2. Base event ontology v1.0

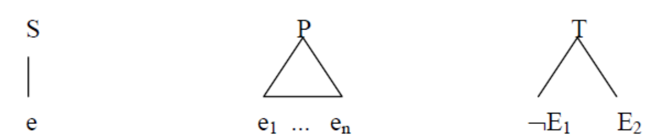
In this paper, we propose an improvement of the base event ontology by applying to it the aspectual model from the field of linguistics. In the next section, we thus introduce this model and in Section V, we present the resulting base event ontology.

IV. Event Types In the Literature on Aspect

In 1959, Vendler defined four types of event classes: state, accomplishment, activity, and achievement [10]. Later, Pustejovsky developed an aspectual model in which he only distinguished three main event types [11]:

- States (S), which are single events evaluated relative to no other event (e.g., be sick, love, know),
- Activities or processes (P), which are sequences of events identifying the same semantic expression (e.g., run, push, drag), and
- Transitions (T), which are events identifying a semantic expression evaluated relative to its opposition (e.g., give, build, open, destroy).

If we define ET as the event type domain, and E as the event domain, then we can represent the Pustejovsky's structural representations of states, processes and transitions as illustrated in the following figure:

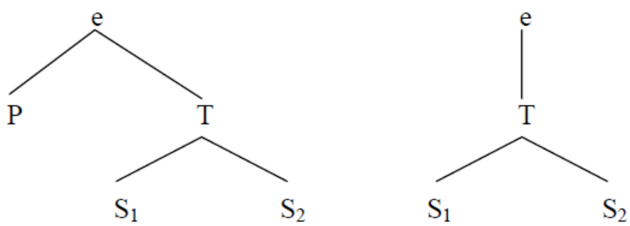


(a) State (S) (b) Process (P) (c) Transition (T)

Figure 3. Structural representations of states, processes and transitions

where e, e1,…en ∈E, and E1,E2∈ET.

Pustejovsky also defined accomplishments and achievements as composite event types and defined their structure. Our work is based on Tremblay's adjustment of Pustejovsky's model:



(a) Accomplishment (b) Achievement

Figure 4. Structural representations of accomplishments and achievements

In his model, an accomplishment is an event composed of a transition from an initial state into the target state, and a process that causes the transition. An achievement is an event composed of a transition from an initial state into the target state (also called achievement without preliminaries).

V. BASE EVENT ONTOLOGY

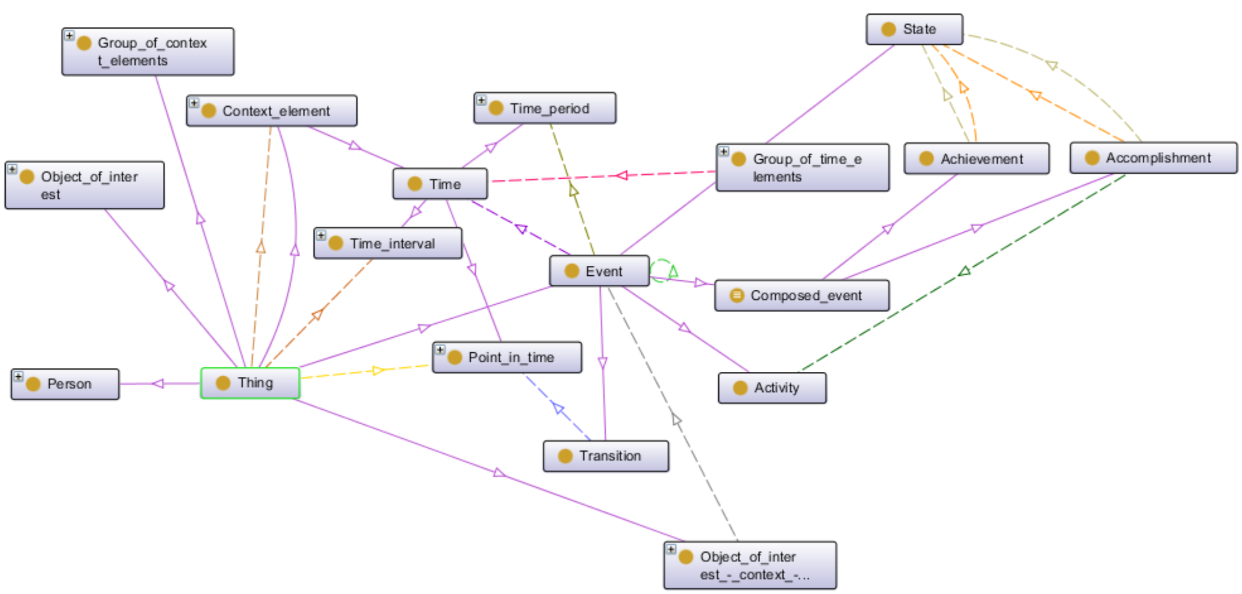
In order to establish an appropriate base event ontology that will satisfy the goal of establishing an expressive base for semantic definitions of complex events, we applied the Tremblay's adjustment of Pustejovsky's aspectual model to the base event ontology. In order to define events we used the concept of a context. In human minds, real-world situations are entertained as contexts [10]. A context is the situation within which something exists or happens, and that can help explain it [9].

The base event ontology is illustrated in Fig. 5. We distinguish between the four categories of events and define object properties between the corresponding classes in order to refleck the structural relations between different event types (Fig. 4). We also define a context element class to represent a general element of an ontology which can represent different concepts important to describe a situation, for example an event context. Object of interest is a class that is used as a basis for decision making, for example when occurrence of a complex event depends on presence or certain state of a specific object. Fig. 4 presents the most important classes and properties of the base event ontology.

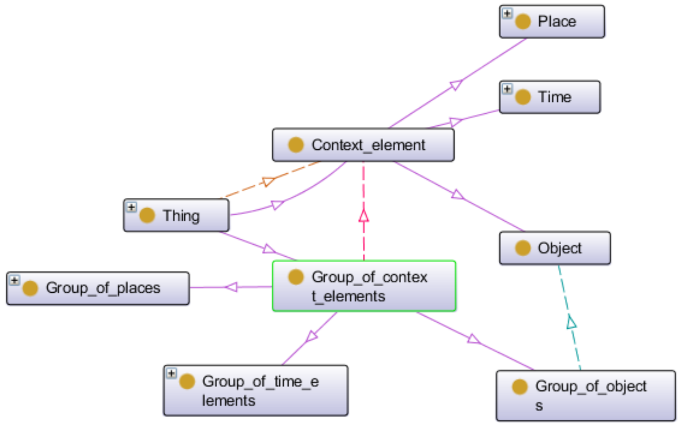
Every event ontology imports at least one domain ontology, the base event ontology and can import one or more other event ontologies. Event ontologies define subclasses of the Event class and its subclasses. Necessary and sufficient membership conditions have to be defined for every complex event, which is represented by the ComposedEvent class. Domain and event ontology concepts can be used to define these conditions. Thus, the necessary and sufficient membership conditions represent the join between the domain and the event concepts. In the event ontology classes and properties are defined that relate an event type to its context.

VI. CONCLUSIONS AND FURTHER WORK

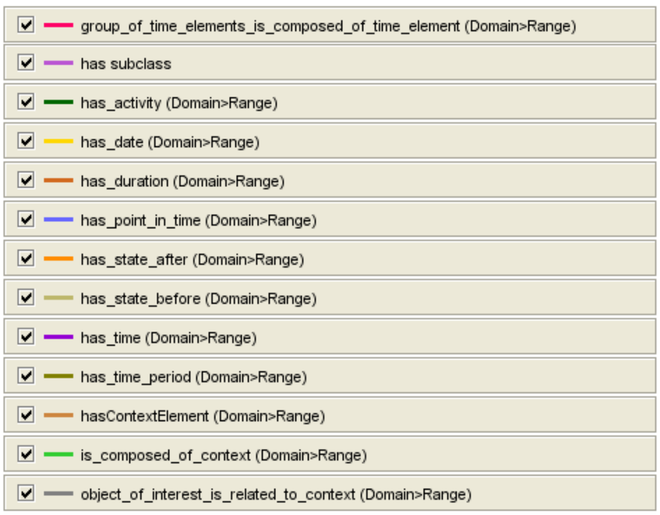
The paper presents an event-driven framework that is based on ontologies. We focused on the base event ontology, which is an important component of the framework. It determines the overall structure for every event ontology and the way the events will be defined. In this paper, we have developed the base event ontology by applying the aspectual model from the field of linguistics, which has studied the event types for several decades.



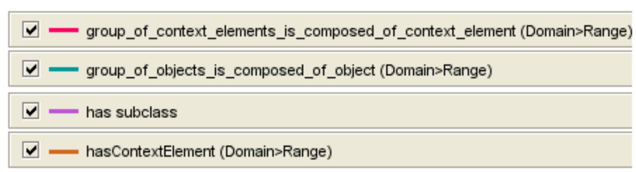
(a) Base event ontology v2.0 visual representation



(b) Base event ontology v2.0 visual representation of context elements



(c) Legend for fig. a



(d) Legend for fig. b

Figure 5. Base event ontology v2.0

The presented base event ontology is work in progress. We are currently preparing a case study for sales domain. In our further work, we will perform several other case studies for different business domains to demonstrate the usefulness of this ontology in CE definition and, more importantly, automatic recognition of different types of CE.

REFERENCES

[1] D. Bugaite D. and O. Vasilecas, "Events Propagation from the Business System Level into the Information System Level". In Barry C. et al. (eds.). Information Systems Development. Springer US, p. 252, 2009.

[2] R. Adaikkalavan and S. Chakravarthy, "Event Specification and Processing for Advanced Applications: Generalization and Formalization," Proc. International Conference on Database and Expert Systems Applications, Regensburg, Germany, 2007, pp. 369-379.

[3] A. Sasa, O. Vasilecas, "Ontology-Based Support for Complex Events," Electronics and electrical engineering, vol. 113, no. 7, 2011.

[4] A. Tremblay, "A Cognitive Approach to Accomplishments, Fundamentally Imperfective Accomplishments, and Achievements with and without Preliminaries; A Cognitive Approach to Aspectual Categories," University of Alberta, 2007.

[5] H. Taylor, F. Martinez, A. Yochem, L. Phillips, Event-Driven Architecture: How SOA Enables the Real-Time Enterprise. Addison-Wesley, 2009.

[6] K. Chandy and W. Schulte, Event Processing: Designing IT Systems for Agile Companies, 1st ed. McGraw-Hill Osborne Media, 2009.

[7] D.C. Luckham, R. Schulte, "Event Processing Glossary – Version 1.1," July 2008. Available on: http://complexevents.com.

[8] W3C, OWL 2 Web Ontology Language, W3C, 2009. Available on: http://www.w3.org/TR/owl2-overview/.

[9] Cambridge Dictionary of British English, Cambridge Dictionaries Online. Available on: http://dictionary.cambridge.org/dictionary/british/.

[10] Z. Vendler, Facts and Laws, Doctoral disertation, Harvard University, 1959.

[11] J. Pustejovsky, "The syntax of event structure.," Cognition, vol. 41, no. 1-3, pp. 47-81, 1991.