Ontology Engineering Group
Laboratorio de Inteligencia Artificial
Dept. Inteligencia Artificial, Facultad de Informática – UPM

OEG Publication

Fernández-López M, Gómez-Pérez A

Searching for a Time Ontology for Semantic Web Applications

Formal Ontology in Information Systems (FOIS-2004)
Proceedings of the Formal Ontology in Information Systems. Third Interntacional Conference (FOIS-2004)
Frontiers in artificial Intelligence and Applications vol. 114
Editorial: IOS Press
4-6 de noviembre de 2004.
Turín, Italia.
Pages: 331-341

Campus de Montegancedo, s/n – Boadilla del Monte – 28660 Madrid
Phone: +34 91 336 74 39 – Fax: +34 91 352 48 19

Searching for a Time Ontology for Semantic Web Applications

Mariano FERNÁNDEZ-LÓPEZ, Asunción GÓMEZ-PÉREZ

Facultad de Informática . Universidad Politécnica de Madrid
Campus de Montegancedo, s/n. 28660 Boadilla del Monte. Madrid. Spain
{mfernandez@fi.upm.es, asun@fi.upm.es}

Abstract. We present our experience on reusing time ontologies in Fund Finder, a semantic web application of the EU Esperonto project. On the one hand, we show a set of time ontologies implemented in different machine readable languages. Such ontologies are analyzed considering a series of features typical of time models (e.g., if they consider different granularities or different time zones). On the other hand, we present the specification of time modeling necessities for Fund Finder. Finally, we choose the ontology which fits best with the specifications.

Introduction

Time modeling has been and is an important topic of the Artificial Intelligence (AI) and on the database (DB) fields. Regarding temporal DBs, the works of Aranha and colleagues [3], De Castro and colleagues [10], and Bair and colleagues [4] deserve special mention. Concerning AI models about time, some of the classical approaches that we consider very important in the area are McDermott's work [25], who proposed a temporal logic for plans; and Allen's work [1], [2], who presented a calculus of time intervals that have influenced enormously on time modeling in the last years. More recently, there have been outstanding contributions on temporal modeling, for instance, Bettini and colleagues' [6] papers on time granularities (e.g., transformation of time expressed in natural days into working days); Kitamura and colleagues [22] and Norwig and Russell's [28] formal time representations for general use; Varzi and colleagues' study of time modeling from the Formal Ontology point of view [34], [35]; etc. We also consider praiseworthy Haye's work [19], which presents different ways to approach time modeling; and Schreiber's overview [31] on time ontologies in Computer Science. With regard to the mailing lists that have dealt with this issue, we can mention OntoWeb SIG1 mailing list1, where Leo Brost provided an extract of references related to time modeling, and the DAML mailing list2.

According to Studer and colleagues' definition of *ontology* ([32], page 185, based on Gruber's definition [17]), *an ontology is a formal, explicit specification of a shared conceptualization*. When explaining such a definition, the authors interpret the word *shared* as agreed, and the word *formal* as *machine readable*. Therefore, if we assume this definition, a time ontology is an agreed time model implemented in a machine-readable language. A representative set of time ontologies (not necessarily an exhaustive set) is made up of:

- the time ontology in Upper Cyc Ontology[3], which is included in the Cyc knowledge base [23], and is implemented in KIF [16] and XML [7];

[1] http://ext4-www.ics.forth.gr/mail/ontoweb-sig1/0012.html
[2] http://www.daml.org/listarchive/daml-time/
[3] http://www.cyc.com/cyc-2-1/cover.html

- the Unrestricted Time ontology[4], developed at the Italian National Research Council (CNR) and codified in Ontolingua language [10];
- the Simple Time ontology[5], which has been considered for developing further time ontologies, like the Reusable Time and the Kestrel Institute Time, and is implemented in Ontolingua language;
- the Reusable Time [36], a very detailed ontology time developed at Stanford University and implemented in the Ontolingua language;
- the Kestrel Institute Time[6] ontology, implemented in DAML+OIL [10];
- the SRI Time ontology[7]; implemented in DAML+OIL by the SRI's Artificial Intelligence Center;
- the modeling of time in SUMO, developed by the IEEE Standard Upper Ontology working group, and implemented in different languages (e.g. OWL [8], KIF [16], LOOM [24] and Protégé format [29]);
- the DAML time ontology, implemented inside the DAML group in KIF and OWL; and
- the AKT Time ontology, developed inside the AKT initiative and implemented in OCML [27].

Since there are several time ontologies, our goal is not to implement a new theory of time, but to reuse the most appropriate ontology for the Fund Finder application developed on the framework of the Esperonto project. In this paper we report about the process followed to select the most appropriate ontology. The paper is organized as follows. Section 1 presents a state of the art of the time ontologies that are implemented in an ontology language. Section 2 presents how we have selected the most appropriate time ontology according to the time modeling requirements of the Esperonto Fund Finder application. Finally, section 3 presents the final conclusions.

1 State of the art of the time ontologies

In order to easily understand this paper, we present some notions of time in an intuitive manner which will facilitate the selection of time ontologies to non-experts on time theories.

- *Modeling of time points*. An example of time point is 26th April 2004: 16:00:00 GMT.
- *Modeling of time interval*. An example of time interval is from 1st April 2004 to 30th April 2004.
- *Absolute and relative time*. On the one hand, we can say that a time is represented in an absolute way when it is related to a fact[8]. For example, we can associate 1789 with the beginning of the French Revolution. On the other hand, we say that the valid time of a fact is represented in a relative way when it is related to the valid time of another fact (e.g. the Russian Revolution was after the French Revolution).
- *Modeling of relations between time intervals*. The most well-known relations between intervals are the ones identified by Allen [1] [2]. Some of them are shown in figure 1.

[4] http://ontology.ip.rm.cnr.it/onto/ON9.3-OL-HTML/unrestricted-time/
[5] http://www-ksl-svc.stanford.edu/
[6] http://www.kestrel.edu/DAML/2000/12/TIME.daml
[7] http://www.ai.sri.com/daml/ontologies/sri-basic/1-0/Time.daml
[8] http://www.cs.auc.dk/~csj/Glossary/

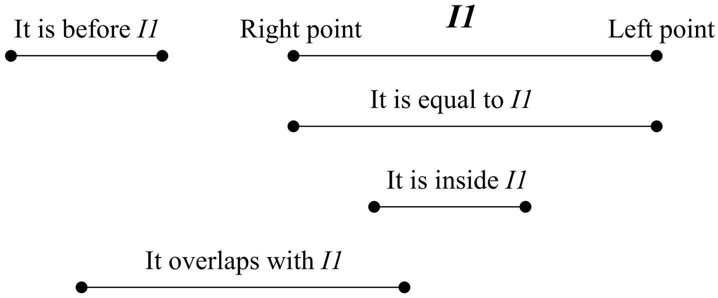


Figure 1: Relations between intervals

- *Modeling of convex and non convex intervals*. These notions allow identifying periodic intervals with gaps between them (e.g. "every Wednesday"), which are called non convex intervals. Conversely, convex intervals are those that are not composed of "separate pieces" (e.g., from 1st April 2004 to 30th April).
- *Distinction between open and closed intervals*. Sometimes, the interval end points might be or not included in the interval. For example, [1985, 1986) is an interval left closed and right open.
- *Explicit modeling of proper intervals*. An interval whose extremes are different is called proper (see DAML ontology documentation[9] and [10]). Thus, for example, [1985, 1986] is a proper interval, however, [1985, 1985] is not.
- *Modeling of concatenation of intervals*, which is used to obtain bigger intervals from the aggregation of others that are smaller.
- *Linking other type of concepts with concepts of time*. An example is to relate processes or events with the period of time in which they happen. A representative work on this topic appears in [34], [35].
- *Modeling of calendar and clock standards*, which are used to express the time (e.g. *hour*, *day*, *Monday*, *January*, *am* and *pm* time*, leap years, time zones*, etc.). In our analysis we will focus only on:

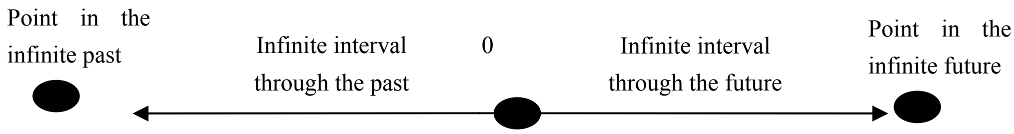
  – The *modeling of years, months, days, hours and minutes* in, at least, their numerical representation.
  – The *modeling of time zones*.

- *Modeling of different temporal granularities.* For example, the duration of an event in days can be transformed into minutes. This operation is usually easy to carry out. However, there are cases where the transformation can be more difficult. Let's think, for instance, in the transformation of an interval between two different natural days into working days. (See [6] and [26]).
- *Total ordering*. Total ordering means that, for every pair of temporal points t_1 and t_2, necessarily $t_1 < t_2$ or $t_2 < t_1$. It is useful to make this assumption in many applications. However, and in distributed systems mainly, this assumption may not be so appropriate. For example the time point t_1 when the process P_1 is started in the machine M_1 and the point t_2 when the process P_2 is started in the machine M_2 may not be linked through the relations $t_1 < t_2$ nor through $t_2 < t_1$.
- *Infinity*. An infinite interval is that which is not limited in the past or in the future. There are two common ways of allowing infinitely long intervals (see note 9). In the first approach (see figure 2), an infinite interval in the past and in the future is modeled through

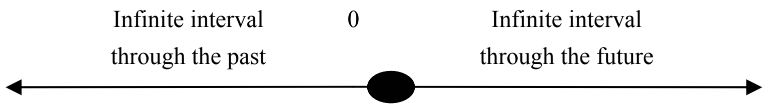
[9] http://www.cs.rochester.edu/~ferguson/daml/
[10] http://vacuumcleaner.cs.kun.nl/c-corn/documentation/CoRN.ftc.MoreIntervals.html

a point in the negative infinite and another one in the positive infinite. In the second approach, an infinite interval in the past and in the future is modeled through one interval with no beginning and one interval with no ending. Given that the second approach does not allow considering points in the infinite, both approaches are incompatible with the same ontology. This is the reason why some ontologies leave this part of the modeling open.

- *Density*, which is used to represent that between any two distinct points there is a third distinct point.



a) Interval with a point in the infinite past and a point in the infinite future



b) Interval with a point neither in the infinite past nor in the infinite future

Figure 2: The two usual representations of infinite temporal intervals (see note 9)

- *Isomorphism to the real numbers*. The set of real numbers is very often the model of the time theory. We can observe that if we take this model in an exact way, we assume the properties of density, convexity, total ordering; we also assume that there are no points in the infinite.
- *If they provide axioms*. The ontology community distinguishes ontologies that are mainly taxonomies from ontologies that model the domain in a deeper way and provide *more restrictions* on domain semantics. The community calls them *lightweight* and *heavyweight ontologies* respectively. Lightweight ontologies include concepts, concept taxonomies, relationships between concepts, and properties that describe concepts, whereas heavyweight ontologies add axioms to lightweight ontologies. Axioms clarify the intended meaning of the terms gathered on the ontology, and allow complex reasoning.

1.1. The time modeling in Upper Cyc

Cyc's Upper Ontology (see note 3) is contained in the Cyc Knowledge Base [23], which holds a huge amount of common sense knowledge. The Cyc KB is being built upon a core of over 1,000,000 assertions hand-entered and designed to gather a large portion of what people normally consider consensus knowledge of the world. It is divided into hundreds of microtheories (bundles of assertions in the same domain) and is available in KIF[11] [16] and XML [7].

The Cyc's Upper Ontology about time models temporal points and time intervals. The key concept in this time ontology is *temporal thing*, which is described by the properties *duration, ending point* and *ending date.* Subclasses of *Temporal thing* are: *event*, *time in-*

---

[11] http://suo.ieee.org/SUO/ontologies/Cyc-upper.txt

terval and *something existing*. The ontology allows modeling a *Temporal thing* that is contiguous or cotemporal with another; it also allows modeling that a temporal thing ends after or before another. Besides, similar relations to the ones aforementioned can also be modeled. Upper Cyc models absolute time, years, months, days, hours, minutes, time zones and different granularities. However, it does not distinguish convex and non convex intervals, open and closed intervals, concatenation of intervals. It does not assume total ordering, infinity, density, and isomorphism to the real numbers.

This ontology also models individual *time points*, although they are seldom mentioned in other definitions. The ontology includes definitions for modeling that a time point is simultaneous with other time point or that a point happens after another.

Cyc Upper ontology allows relating formulas and temporal things to express that a proposition is true during the temporal extent of a temporal thing by means of the relation *holds*. For example, *holds* allows associating the fact "Columbus is in America" with the date 14th October 1492.

The public version of the Upper Cyc ontology provides classes and relations only, but not axioms.

1.2. The Unrestricted Time ontology

The Unrestricted Time ontology (see note 4) was developed by the Italian CNR, and was built to be reused by the ON9[12] library of medical ontologies [12]. This ontology is based on a top level ontology with very abstract concepts (*entity*, *process*, etc.), and is implemented in Ontolingua language.

The key concept in this time ontology is *temporalized entity*. Subclasses of *temporalized entity* are: *situation, process* and *time span*. The ontology allows modeling that a *temporalized entity* co-occurs, overlaps, meets, precedes, etc. with other *temporalized entity.* This ontology does not distinguish explicitly between temporal points and time intervals; time zones, granularities, calendar and clock standards are not considered in the ontology. On the other hand, proper intervals, open intervals and convex intervals are not distinguished. However, total ordering, infinity and density are assumed, whereas the isomorphism to real numbers is not explicitly assumed. This ontology provides axioms for most of its definitions.

1.3. The Simple Time ontology

The Simple Time ontology (see note 5) was developed at the Knowledge System Laboratory (KSL) at Stanford University. It has been taken into account to develop other ontologies such as the Kestrel Time Ontology and the Reusable Time.

The main concepts in Simple Time are *time point* and *time range*. *Time point* has the properties of: *second*, *minute*, *hour*, *day*, *month* and *year*. *Time range* has the properties of *duration*, *start point* and *end point*, and has the relations of *before*, *before or equal*, *meets*, etc.

Proper intervals, open intervals, and convex intervals are not distinguished. Simple Time models absolute time. This ontology does not assume an isomorphism to the real numbers nor does it assume infinity. However, according to its documentation, it assumes total ordering and density and provides axioms for most of its definitions.

1.4. The Reusable Time Ontology

[12] http://saussure.irmkant.rm.cnr.it/ON9/index.html

The Reusable Time ontology[13] was also developed at the KSL at Stanford University [36]. The ontology is compatible with the HPKB-Upper-Level-Kernel ontology in the Ontolingua library, which was developed in DARPAs High Performance Knowledge Base (HPKB) program [8].

The main concepts of Reusable Time are *time point* and *time interval*. *Time point* has the properties of *granularity*, *minute*, *hour*, *day*, *month*, *year*, etc. Therefore, it allows representing absolute time. *Time interval* has the properties of *duration*, *starting point*, *ending point*, etc., and has the relations of *before*, *precedes*, *meets*, *starts after ending of*, etc.

This ontology distinguishes convex and non convex intervals, and open and closed intervals, and assumes density and infinity through the definition of the points *infinite past* and *infinite future*. The Reusable Time ontology does not explicitly assume an isomorphism to the real numbers. According to its documentation, it assumes total ordering and density. In addition, it provides axioms for most of its definitions.

1.5. The Kestrel Time Ontology

The Kestrel Time Ontology (see note 6) has been developed by the Kestrel Institute. The main concepts in this ontology are: *temporal point* and *temporal interval*. *Temporal point* has the properties of *time* and *time unit*. *Temporal interval* has the properties of *duration*, *start time point* and *end time point*, and has the relations of *before*, *same start*, etc. with itself. Proper intervals, open intervals and convex intervals are not distinguished.

The Kestrel Time Ontology does not assume an isomorphism to the real numbers nor does it assume infinity. However, according to its documentation, it assumes total ordering and density. This ontology does not define absolute nor relative time. Besides, it does not provide axioms; consequently, it can be included in the set of light ontology.

1.6. The SRI's Time ontology

The SRI Time ontology (see 7) has been developed at the SRI's Artificial Intelligence Center. The main concepts in this ontology are *instant* and *interval*, which are subclasses of *temporal entity*. *Instant* is linked to interval through the relations *inside*, *start of* and *end of*. *Temporal entity* is linked to itself via the relation *before*. Proper intervals, open intervals and convex intervals are not distinguished.

This ontology does not define absolute nor relative time. It defines different ways to measure the time (year, month, etc.) but not time zones. SRI Time does not assume explicitly the following features: isomorphism to the real numbers, total ordering, points at infinity and density. The SRI Time ontology does not provide axioms, hence, it is a light ontology.

1.7. The modeling of time in SUMO

The Standard Upper Ontology[14] is the result of a joint effort to create a large, general-purpose, formal ontology [30]. It is promoted by the IEEE Standard Upper Ontology working group, and its development began in May 2000. The participants were representatives of government, academia, and industry from several countries. The effort was officially approved as an IEEE standard project in December 2000.

There are currently two "starter documents". One of them is known as SUMO (Suggested Upper Merged Ontology). The goal of this document is to create a comprehensive and consistent top-level ontology from some of the best public sources. Therefore, SUMO

---

[13] http://ksl.stanford.edu/ontologies/time
[14] http://suo.ieee.org/

takes into account some high level distinctions, and contains temporal concepts and processes.

The main concepts of this ontology are *time point* and *time interval*, which are subclasses of *time position*. A time position can be a *temporal part* of a time position. *Time point* is linked to itself through the relations *before* and *earlier*. *Time interval* has the properties of *duration*, *begin point* (called *beginFn* in the ontology) and *end point* (called *EndFn*), and has the relations of *meets temporally*, *meets*, etc. with itself (in fact, SUMO implements the Allen calculus [1] [2]).

This ontology allows modeling years, months, days, years and minutes. Therefore, it permits modeling relative time. Proper intervals, open intervals and convex intervals are not distinguished.

SUMO time ontology links time and events through the relations *time* and *whenFn*. As in the case of Cyc, it also links time to formulas through the relation *holds during*. Such a relation means that the formula is true during a given time.

According to the documentation, SUMO seems to assume an isomorphism to the real numbers; therefore, it also assumes total ordering, no points at infinity, density and convexity. It provides axioms for most of its definitions.

1.8. The DAML Time ontology

The DAML Time ontology is being developed by Jerry R. Hobbs with contributions from George Ferguson, James Allen, Richard Fikes, Pat Hayes, Drew McDermott, Ian Niles, Adam Pease, Austin Tate, Mabry Tyson, and Richard Waldinger (see note 9). These are authors known by their contributions to time modeling, and time ontologies.

The main concepts of this ontology are *instant* and *interval*, which are subclasses of *temporal entity*. *Instant* is linked to interval through the properties of *begins*, *ends*, *inside* and *begins or in*. *Instant* is also linked to an instant temporal description, which is a concept with the properties of *second*, *minute*, *hour*, *day*, *month*, *year* and *time zone*. *Interval* has the subclass *proper interval*, which is related with itself through the relations *interval equals*, *interval before*, *interval starts or finishes*, etc. Proper intervals can be concatenated through the relation *concatenation*.

DAML Time distinguishes between convex and non-convex intervals, while open intervals are not explicitly considered in it. DAML Time does not assume the following features (although it can be extended with them): isomorphism to the real numbers, total ordering, points at infinity and density. It provides axioms for most of its definitions.

The ontology is available in KIF [16] and OWL [8].

1.9. The AKT Time ontology

The AKT Time ontology[15] was created by the Advanced Knowledge Technologies (AKT) project.

The main concepts in this ontology are *time point* and *time interval*, both subclasses of *time position*. *Time point* has the properties of *second*, *minute*, *hour*, *day*, *month*, *year* and *time zone*. Therefore, the AKT time ontology permits modeling absolute time. *Temporal interval* has the property of *duration*, and is linked to *time point* through the properties of *begins at*, and *ends at*. The AKT Time Ontology does not model relations between intervals; nor does it distinguish proper intervals, open intervals and convex intervals. The AKT Time ontology does not provide axioms for its definitions, hence, it is light ontology.

2. Reusing an ontology time in Esperonto applications

When we build or reuse ontologies, we follow the ontology development methodology METHONTOLOGY [12]. The development process proposed by this methodology [13] identifies the [re]use of an ontology as a post-development activity of such ontology. The process of reuse consists in:

1) *Specifying the necessities that the ontology has to satisfy in the new application*. The specification document of the ontology to be reused should include:
   - *Purpose*. Description of the application where it will be reused.
   - *Language in which it is needed*. It is the language (or languages) in which the ontology has to be implemented.
   - *Competency questions*. These are a set of natural language questions used to determine the scope of the ontology (see [18]). These questions and their answers are used to extract the main concepts and their properties, the relations and the formal axioms of the ontology.
   - *Features*. Here it is explained the point of view from which we should approach the domain, the key aspects of the domain that should be modeled, etc.
   - *Scope*. It includes the set of terms to be represented and the detail of their definitions. Such terms are obtained from the competency questions.

2) *Searching an ontology that covers most of the identified necessities*.

3) *Adapting the ontology so that it satisfies the necessities completely*.

4) *Integrating the ontology in the system*. The adaptation of the ontology can involve a translation process, since the ontology can be needed in a different language to that in which it is available.

In the case of the Esperonto project, what we need is a time ontology to be integrated in the semantic web application Fund Finder.

The main objectives of Fund Finder are to allow web users to ask for funding resources according to some parameters (e.g. the geographical area or the time interval of the calls for funding, both of which interest the user), to look for funding complementary resources, funding compatibilities, and incompatibilities, etc. [5].

[15] http://plainmoor.open.ac.uk/ocml/domains/akt-support-ontology/time.html

Consequently, the time ontology has to answer competency questions like *when is the deadline of a call (e.g. GMT time)?*, *when is a call open?*, *which calls are open this month?*, etc.

To answer these questions, the time ontology needs to model: time points, time intervals, relations between intervals (*before*, *overlaps*, etc.), years, months, days, hours, minutes, time zones, and different granularities. Besides, it is important that the terms of the ontology are defined by means of axioms, so that complex reasoning can be applied.

The ontologies that fits best the requirements are SUMO and DAML. On the one hand, the first ontology has the advantage of providing a top level ontology. On the other hand, during the DAML development most of the effort has been exclusively devoted to time modeling. Therefore, this ontology provides a very precise time modeling and an extended documentation.

Given that we have already decided to use the CNR top level ontology of particulars [10], we are not interested in reusing the whole SUMO ontology, so we have decided to reuse the DAML ontology instead.

As we have said in the state of the art (section 1), the DAML time ontology is currently available in OWL and KIF. However, only the KIF version has axioms. Therefore, to reuse it with axioms we need to import the KIF code from our ontology development tool. Once the ontology is imported, we have to adapt it to our necessities. This adaptation should include a study on the automatic reasoning using the ontology. Axioms represent the meaning of terms in a precise way, but their processing can be difficult. For example, in the DAML time ontology, we can find this axiom:

```
after(T1,T2) ↔ before(T2,T1)
```

which defines the term *after* starting from the term *before*. It is a precise definition, but we should choose the direction of the reasoning so that a computer can use it. That is, we have to decide if we will use *after* to obtain *before*, or *before* to obtain *after*. The modeling of both directions could provoke an infinite loop in the execution.

3. Conclusions

In this paper, we have presented the study of different time ontologies as candidate to model time in the Esperonto project applications. First of all, we have presented a state of the art on time ontologies. This state of the art has included a catalog of features useful to analyze temporal models. It has also included a short analysis of a set of ontologies, that is, agreed temporal models implemented in a machine readable language. Some of these ontologies provide a set of reusable temporal terms defined through axioms.

Once we have studied the temporal ontologies, we have specified the time modeling necessities in the Esperonto applications. We have seen that we have to consider temporal intervals as well as temporal points, different granularities, relations between points and intervals, and different time zones.

After analyzing the candidate time ontologies, we have chosen the DAML ontology to model the time in the Esperonto applications because DAML provides a detailed and documented time model.

Given that the only version with axioms available is the KIF version, we need to import it from our ontology development tool. Then, we have to adapt the ontology to our specifications. In this adaptation we have to pay special attention to the automatic computation of the resulting definitions.

An important conclusion of the work that we have carried out is that we can have time ontologies to be reused in new applications. Such a reuse can save time in the development of the whole system. However, a conceptual model using intermediate representations (e.g.

tables and graphs) is not available for any of these ontologies, and this fact has made more difficult the work of analysis. However, this analysis problem has been specially attenuated in the case of the Reusable Time and the DAML time ontology, since they have extensive documentation. In addition, the DAML time ontology documentation presents the definitions in first order logic, which is a formalization way well known by most of the Computer Science people.

7. Acknowledgement

This work has been partially supported by the IST European Project Esperonto (IST-2001-34373). We are also grateful of Rosario Plaza for her help in checking and reviewing the grammar and spelling of the paper and improving clarity and style.

References

[1] Allen JF (1984) *Towards a General Theory of Actions and Time*. Artificial Intelligence 23:123-154

[2] Allen JF (1983) *Maintaining Knowledge about Temporal Intervals*. Communications of the ACM. 26(11):832-843

[3] Aranha RFM, Ganti V, Narayanan S, Muthukrishnan CR, Prasad STS, Ramamritham K (1996) *Implementation of a real-time database system*. Information Systems, 21(1):55-74

[4] Bair J, Böhlen M, Jensen CS, Snodgrass RT (1997) *Notions of upward compatibility of temporal query languages*. Wirtschaftinformatik, 39(1):25-34

[5] Barrasa J, Corcho O, Blanco R, Gómara C (2003) *"Test Case System Specification Fund Finder"*. D9.1 Esperonto deliverable

[6] Bettini C, Wang XS, Jajodia S (1996) *A general framework and reasoning models for time granularity*. In Chittaro L, Goodwin SD, Hamilton HJ, Montanari A (eds) Proceedings of the Third International Workshop on Temporal Representation and Reasoning (TIME'96), Key West, Florida

[7] Bray T, Paoli J, Sperberg-McQueen CM, Maler E (2000) *Extensible Markup Language (XML) 1.0*. W3C Recommendation. *http://www.w3.org/TR/REC-xml*

[8] Cohen P, Schrag R, Jones E, Pease A, Lin A, Starr B, Gunning D, Burke M (1998) *The DARPA High-Performance Knowledge Bases Project*. AI Magazine, 19(4):25-50

[9] Dean M, Connolly D, van Harmelen F, Hendler J, Horrocks I, McGuinness DL, Patel-Schneider PF, Stein LA (2002) *OWL Web Ontology Language 1.0 Reference*. W3C Working Draft. *http://www.w3.org/TR/owl-ref/*

[10] De Castro C, Grandi F, Scalas MR (1997) *Schema versioning for multitemporal relational databases*. Information Systems, 22(5):249-290

[11] Farquhar A, Fikes R, Rice J (1997) *The Ontolingua Server: A Tool for Collaborative Ontology Construction*. International Journal of Human Computer Studies 46(6):707–727

[12] Fernández-López M, Gómez-Pérez A, Pazos A, Pazos J (1999) *Building a Chemical Ontology Using Methontology and the Ontology Design Environment*. IEEE Intelligent Systems & their applications 4(1):37–46

[13] Fernández-López M, Gómez-Pérez A, Juristo N (1997) *METHONTOLOGY: From Ontological Art Towards Ontological Engineering*. Spring Symposium on Ontological Engineering of AAAI. Stanford University, California, pp 33–40

[14] Gangemi A, Guarino N, Oltramari A (2001) *Conceptual analysis of lexical taxonomies: the case of Wordnet top-level*. In: Smith B, Welty C (eds) International Conference on Formal Ontology in Information Systems (FOIS'01). Ogunquit, Maine. ACM Press, New York, pp 3–15

[15] Gangemi A, Pisanelli DM, Steve G (1998) *Some Requirements and Experiences in Engineering Terminological Ontologies over the WWW*. In: Gaines BR, Musen MA (eds) 11th International Workshop on Knowledge Acquisition, Modeling and Management (KAW'98). Banff, Canada, SHARE10:1–20

[16] Genesereth MR, Fikes RE (1992) *Knowledge Interchange Format. Version 3.0. Reference Manual*. Technical Report Logic-92-1. Computer Science Department. Stanford University, California. *http://meta2.stanford.edu/kif/Hypertext/kif-manual.html*

[17] Gruber TR (1993) *Toward principles for the design of ontologies used for knowledge sharing*. In: Guarino N, Poli R (eds) International Workshop on Formal Ontology in Conceptual Analysis and Knowledge Representation. Padova, Italy. (Formal Ontology in Conceptual Analysis and Knowledge Representation) Kluwer Academic Publishers, Deventer, The Netherlands. *http://citeseer.nj.nec.com/gruber93toward.html*

[18] Grüninger M, Fox MS (1995) *Methodology for the design and evaluation of ontologies* In Skuce D (ed) IJCAI95 Workshop on Basic Ontological Issues in Knowledge Sharing, pp 6.1–6.10

[19] Hayes PJ (1995) *A catalog of temporal theories*. Technical Report UIUC-BI-AI-96-01 at the Beckman Institute and Departments of Philosophy and Computer Science University of Illinois. *http://www.coginst.uwf.edu/users/phayes/TimeCatalog1.ps* and *http://www.coginst.uwf.edu/users/phayes/TimeCatalog2.ps*

[20] Horrocks I, van Harmelen F (eds) (2001) *Reference Description of the DAML+OIL (March 2001) Ontology Markup Language*. Technical report. *http://www.daml.org/2001/03/reference.html*

[21] IEEE (1996) *IEEE Standard for Developing Software Life Cycle Processes*. IEEE Computer Society. New York. IEEE Std 1074-1995

[22] Kitamura Y, Ikeda M, Mizoguchi R (1997) *A Causal Time Ontology for Qualitative Reasoning*. In: Georgeff MP, Pollack ME (eds) 15th International Joint Conference on Artificial Intelligence (IJCAI'97), Nagoya, Japan. Morgan Kaufmann Publishers, San Francisco, California, pp 501-506

[23] Lenat DB, Guha RV (1990) *Building Large Knowledge-based Systems: Representation and Inference in the Cyc Project*. Addison-Wesley, Boston, Massachusetts

[24] MacGregor R (1991) *Inside the LOOM clasifier*. SIGART bulletin 2(3):70–76

[25] McDermott D (1982) *A Temporal Logic for Reasoning about Processes and Plans*. Cognitive Science 6:101-155

[26] Montanari A, Peron A, Policriti A (2002) *Extending Kamp's Theorem to Model Time Granularity*. Journal of Logic and Computation, 12(4):641–678

[27] Motta E (1999) *Reusable Components for Knowledge Modelling: Principles and Case Studies in Parametric Design*. IOS Press, Amsterdam, The Netherlands

[28] Norvig S, Russell P (2003) *Artificial Intelligence: A Modern Approach*, 2nd edn, section 10.1. Series in Artificial Intelligence. Prentice Hall, New Jersey, State of New Jersey

[29] Noy NF, Fergerson RW, Musen MA (2000) *The knowledge model of Protege-2000: Combining interoperability and flexibility*. In: Dieng R, Corby O (eds) 12th International Conference in Knowledge Engineering and Knowledge Management (EKAW'00). Juan-Les-Pins, France. (Lecture Notes in Artificial Intelligence LNAI 1937) Springer-Verlag, Berlin, Germany, pp 17–32

[30] Pease RA, Niles I (2002) *IEEE Standard Upper Ontology: A Progress Report*. The Knowledge Engineering Review 17(1):65–70

[31] Schreiber FA (1994) *Is Time a Real Time? An Overview of Time Ontology in Informatics*. In Haland WA, Stoyenko AD (eds). Real Time Computing, Springer Verlag NATO-ASI, Berlin, Germany F127:283-307

[32] Studer R, Benjamins VR, Fensel D (1998) *Knowledge Engineering: Principles and Methods*. IEEE Transactions on Data and Knowledge Engineering 25(1-2):161–197

[33] Uschold M, Grüninger M (1996) *Ontologies: Principles, Methods and Applications*. Knowledge Engineering Review 11(2):93–155

[34] Varzi A, Pianesi F (1996) "Events, Topology, and Temporal Relations". The Monist, 78(1):89-116. *http://www.columbia.edu/~av72/papers/Monist_1996.pdf*

[35] Varzi A, Pianesi F (1996) *Refining Temporal Reference in Event Structures*. Notre Dame Journal of Formal Logic, 37(1):71-83. *http://www.columbia.edu/~av72/papers/Ndjfl_1996.pdf*

[36] Zhou Q, Fikes R (2000) *A Reusable Time Ontology*. Technical Report KSL-00-01. Knowledge Systems Laboratory, Stanford, California