Representing Temporal Knowledge in the Semantic Web: The Extended 4D Fluents Approach

Sotiris Batsakis, Euripides G.M. Petrakis
Department of Electronic and Computer Engineering
Technical University of Crete (TUC)
Chania, Crete, GR-73100, Greece
batsakis@softnet.tuc.gr, petrakis@intelligence.tuc.gr

***Abstract*—Representing information that evolves in time in ontologies, as well as reasoning over static and dynamic ontologies are the areas of interest in this work. Building upon well established standards of the semantic Web and the 4D-fluents approach for representing the evolution of temporal information in ontologies, this work demonstrates how qualitative temporal relations that are common in natural language expressions (i.e., relations between time intervals like "before", "after", etc.) are represented in ontologies. Existing approaches allow for representations of temporal information, but do not support representation of qualitative relations and reasoning.**

I. Introduction

Ontologies offer the means for representing high level concepts, their properties and their interrelationships. Dynamic ontologies will in addition enable representation of information evolving in time. In particular, dynamic ontologies are not only suitable for describing static scenes with static objects (e.g., objects in photographs) but also enable representation of events with objects and properties changing in time (e.g., moving objects in a video). Representation of both static and dynamic information in ontologies, as well as reasoning over static and dynamic ontologies are exactly the problems this work is dealing with.

Representation of dynamic features calls for mechanisms allowing representation of the notion of time (and of properties varying in time) [1]. Methods for achieving this goal include (among others), temporal description logics [11], temporal RDF [13], versioning [6], named graphs [18], reification, N-ary relations [2] and the 4D-fluent (perdurantist) approach [9] with the last being the most efficient. All approaches suffer from data redundancy as several objects are created for each binary relationship changing in time (i.e., for each new event, a new temporal object and an additional binary relationship for each temporal property of this object is created and associated with existing classes) thus complicating the ontology. Also, adding a time argument to binary relationships may (as in reification and named graphs) complicate application of OWL language constructs (e.g., cardinality constraints, inverse, transitive relations are no longer applicable) thus limiting OWL expressivity and obstructing reasoning. The 4D fluents approach, still suffers from data redundancy but maintains OWL expressiveness and reasoning support (i.e., an OWL reasoner such as Pellet can still be applied to fully exploit OWL semantics over the 4D fluent representation). However, time and temporal constructs representing the evolution of binary relationships in time, still offer additional semantics which can be exploited by applying additional rules (e.g., rules on Allen relationships). This is also a problem this work is dealing with.

Reasoning on spatio-temporal knowledge is still an active research area and has been investigated previously in other domains (temporal logics [11], temporal data bases [10]). To the best of our knowledge this is the first work to address this problem within the context of ontologies. More specifically, we show how results from previous research efforts [17], [28], [25] can be ported into ontological representations such as the extended 4D fluents representation proposed in this work.

In our earlier work [4] we showed how temporal information (also the evolution of temporal concepts) can be represented effectively in OWL. Concepts varying in time are represented as 4-D dimensional objects, with the 4-th dimension being the time. This work extends this approach in certain ways: The 4-D fluents mechanism is enhanced with qualitative (in addition to quantitative) temporal expressions allowing for the representation of temporal intervals with unknown starting and ending points by means of their relation (e.g., "before", "after") to other time intervals. Adding reasoning support to the above representation is also a contribution of the present work: A set of inference rules is proposed whose purpose is to assert additional implied facts into the knowledge base (i.e., determine the temporal relation between two events given their relations with a third one). Reasoning becomes feasible by using a tractable subset of the set of Allen's relationships [17]. Specifically, the reasoning mechanism incorporates rules for inferring certain temporal relations from existing ones using additional axioms based on compositions of Allen relations and by checking temporal assertions for consistency (i.e., path consistency checking is implemented).

Adding query support to the extended 4D fluent representation is an additional contribution of this work. More specifically, we extend the TOQL query language [4] to handle qualitative temporal relationships and the extended 4D fluent representation.

Related work in the field of knowledge representation is discussed in Section II. This includes issues related to representing and reasoning over information evolving in time.

The temporal representation model is presented in Section III and the corresponding reasoning mechanism in Section III-A, followed by evaluation in Section IV and conclusions and issues for future work in Section V.

II. Background and Related Work

Several representation languages are defined for the Semantic Web, the most important of them are referred to as the OWL-family [7], [22] of ontology languages for ontology building and knowledge representation. Representation languages such as RDF, OWL (which is based on description logics), the same as frame-based and object-oriented languages (F-logic) are all based on binary relations. Binary relations simply connect two instances (e.g., an employee with a company) without any temporal information. Nevertheless, representation of time using OWL is feasible, although complicated [2], [9].

The OWL-Time temporal ontology [5] describes the temporal content of Web pages and the temporal properties of Web services. Apart from language constructs for the representation of time in ontologies, there is still a need for mechanisms for the representation of the evolution of concepts (e.g., events) in time. This is related to the problem of the representation of time in temporal (relational and object oriented) databases. Existing methods are relying mostly on temporal Entity Relation (ER) models [10] taking into account valid time (i.e., time interval during which a relation holds), transaction time (i.e., time at which a database entry is updated) or both. Also time is represented by time instants, intervals or finite sets of intervals. However, representation of time in OWL differs because (a) OWL semantics are not equivalent to ER model semantics (e.g., OWL adopts the *Open World Assumption* while ER model adopts the *Closed World Assumption*) and (b) relations in OWL are restricted to binary ones. Representation of time in the Semantic Web can be achieved using *Temporal Description logics (TDLs)* [11], [12], *Reification*, *N-ary relations* [2], *temporal RDF* [13], *Versioning* [6], *named graphs* [18] or *4D-fluents* [9].

Temporal Description Logics (TDLs) extend standard description logics (DLs) that form the basis for semantic Web standards with additional constructs such as “always in the past”, “sometime in the future”. TDLs offer additional expressive capabilities over non temporal DLs and retain decidability (with an appropriate selection of allowable constructs) but they require extending OWL syntax and semantics with additional temporal constructs. Representing information regarding specific time points requires support for concrete domains, resulting to the proliferation of objects [11].

Temporal RDF [13] proposes extending RDF by labeling properties with the time interval they hold. This approach also requires extending the syntax and semantics of the standard RDF, although representation over RDF (e.g., using reification) can be achieved. Note that Temporal-RDF cannot express incomplete information , by means of qualitative relations.

Reification is a general purpose technique for representing *n*-ary relations using a language such as OWL that permits only binary relations. Specifically, an *n*-ary relation is represented as a new object that has all the arguments of the *n*-ary relation as objects of properties. For example if the relation *R* holds between objects A and B at time t, this is expressed as *R(A,B,t)*. Furthermore, in OWL using reification this is expressed as a new object with R, A, B and t being objects of properties. Fig. 1 illustrates the relation *WorksFor(Employee, Company, TimeInterval)* representing the fact that an employee works for a company during a time interval. Reification suffers mainly from two disadvantages: (a) data redundancy, because a new object is created whenever a temporal relation has to be represented (this problem is common to all approaches based on non temporal Description Logics such as OWL-DL) and (b) offers limited OWL reasoning capabilities [9] since relation R is represented as the object of a property thus OWL semantics over properties are no longer applicable (i.e., the properties of a relation are no longer associated directly with the relation itself).

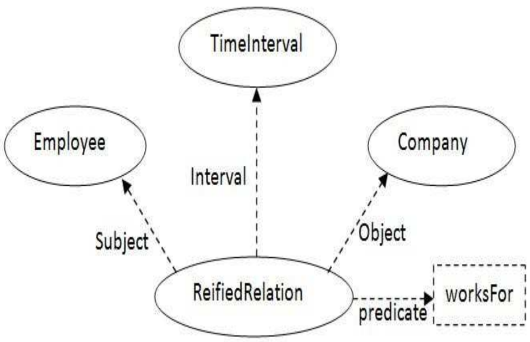


Fig. 1. Example of Reification

N-ary relations is also a general purpose technique that represents an n-ary relation using an additional object. In contrast to reification, the n-ary relation is not represented as the object of a property but as two properties each related with the new object. These two objects are related to each other with an n-ary relation. This is also illustrated in Fig.2. This approach requires only one additional object for every temporal interval, maintains property semantics but suffers from data redundancy in the case of inverse and symmetric properties [2] (e.g., the inverse of a relation is added explicitly twice instead of once as in 4D-fluents).

Versioning [6] suggests that the ontology has different versions (one per instance of time). When a change takes place, a new version is created. Versioning suffers from several disadvantages: (a) changes even on single attributes require that a new version of the ontology be created leading to information redundancy (b) searching for events occurred at time instances or during time intervals requires exhaustive searches in multiple versions of the ontology, (c) it is not clear how the relation between evolving classes is represented. Furthermore, ontology languages such as OWL [7] are based on binary relations (relations connecting two instances) with no time dimension regarding ontology versions.

Named Graphs [18] represent the temporal context of a

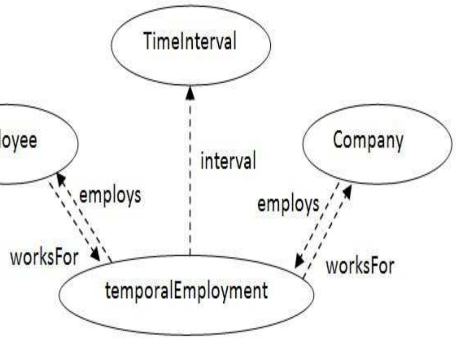


Fig. 2. Example of N-ary Relations

property by inclusion of a triple representing the property in a named graph (i.e., a subgraph into the RDF graph of the ontology specified by a distinct name). The default (i.e., main) RDF graph contains definitions of interval start and end points for each named graph, thus a property is stored in a named graph with start and end points corresponding to the time interval that the property holds. Named graphs are not part of the OWL specification [24] (i.e., there are not OWL constructs translated into named graphs) and they are not supported by OWL reasoners.

The *4D-fluent* (perdurantist) approach [9] shows how temporal information and the evolution of temporal concepts can be represented effectively in OWL. Concepts in time are represented as 4-dimensional objects with the 4th dimension being the time. Time instances and time intervals are represented as instances of a *time interval* class which in turn is related with time concepts varying in time. Changes occur on the properties of the temporal part of the ontology keeping the entities of the static part unchanged. The 4D-fluent approach still suffers from data redundancy but in contrast to other approaches it maintain full OWL expressiveness and reasoning support. N-ary relations[2] is considered to be an alternative to the 4-D fluents approach, although the 4-D fluents representation where the property is holding among two timeslices of objects and not between the two objects and the intermediate object representing their relation may seems more natural to users. TOWL [23] is a temporal representation approach based on 4-D fluents that extends OWL syntax with temporal concepts and supports quantitative time intervals.

III. Extended 4D Fluents Approach

Following the approach by Welty and Fikes [9], to add time dimension to an ontology, classes *TimeSlice* and *TimeInterval* with properties *tsTimeSliceOf* and *tsTimeInterval* are introduced. Class *TimeSlice* is the domain class for entities representing temporal parts (i.e., “time slices”) and class *TimeInterval* is the domain class of time intervals. A time interval holds the temporal information of a time slice. Property *tsTimeSliceOf* connects an instance of class *TimeSlice* with an entity, and property *tsTimeInterval* connects an instance of class *TimeSlice* with an instance of class *TimeInterval*. Properties having a time dimension are called fluent properties and connect instances of class *TimeSlice*.

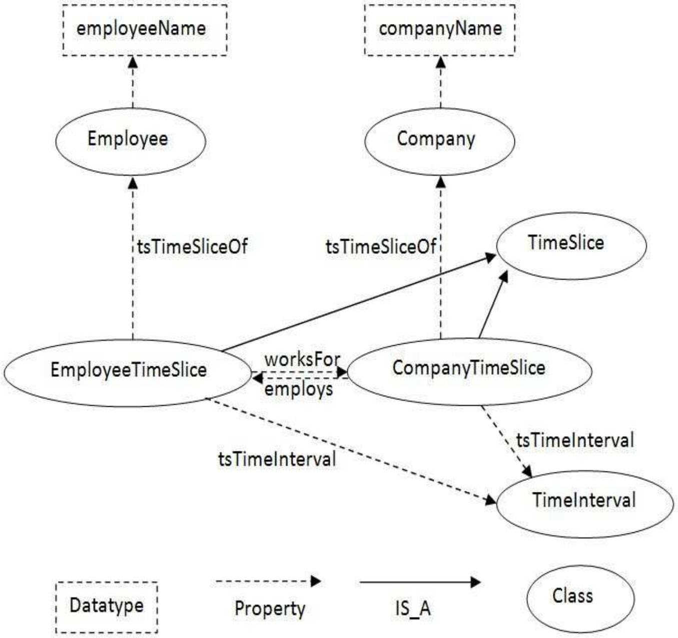


Fig. 3. Dynamic Enterprise Ontology

Fig. 3 illustrates a temporal ontology with classes *Company* with datatype property *companyName* and *Employee* with datatype property *employeeName*. In this example, *CompanyName* and *EmployeeName* are static properties (their value do not change in time), while properties *employs* and *worksFor* (i.e., inverse of employs) are dynamic (fluent) properties whose values may change in time. Because they are fluent properties, their domain (and range) is of class *TimeSlice*. *CompanyTimeSlice* and *EmployeeTimeslice* are instances of class *TimeSlice* and are provided to denote that the domain of properties *worksFor* and *employs*, are time slices restricted to be slices of a specific class. For example, the domain of property *employs* is not class *TimeSlice* but it is restricted to instances that are time slices of class *Company*.

The 4-D fluent mechanism forms the basis of the proposed temporal ontology representation. In this work, the 4D-fluent representation is enhanced with qualitative temporal relations holding between time intervals whose starting and ending points are not specified. This is implemented by introducing temporal relationships as object relations between time intervals. This can be one of the 13 pairwise disjoint Allen’s relations [17] of Fig. 4.

By allowing for qualitative relations the expressive power of the representation increases. Temporal RDF and 4-D fluents both require closed temporal intervals for the representation of temporal information, while semiclosed and open intervals can’t be represented effectively in a formal way. If their endpoints are unknown, ad-hoc approaches [18] that handle open intervals by extending their start or end point infinitely are not appropriate, since lack of knowledge (about their

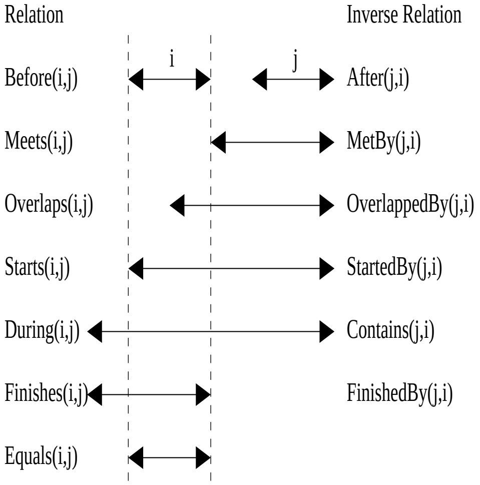


Fig. 4. Allen's Temporal Relations

endpoints) is interpreted as if a property always holds in the past or future. In this work, this is handled by Allen relations: for example, if interval $t1$ is known and $t2$ is unknown but we know that $t2$ starts when $t1$ ends, then we can assert that $t2$ is $met\ by$ $t1$. Likewise, if an interval $t3$ with unknown endpoints is introduced and $t3$ is $before$ $t1$ then, using compositions of Allen relations [17], we infer that $t3$ is $before$ $t2$ although both interval's endpoints are unknown and their relation is not represented explicitly in the ontology. Semiclosed intervals can be handled in a similar way. For example, if $t1$ starts at time point 1, still holds at time point 2, but it's endpoint is unknown, we assert that $t1$ is $started\ by$ interval $t2$:$[1, 2]$. Fig.5 illustrates the dynamic ontology schema representing the scenario "George lived in Crete from 2004 to 2010 and then he moved to Athens". In this example, we don't know whether George still lives in Athens.

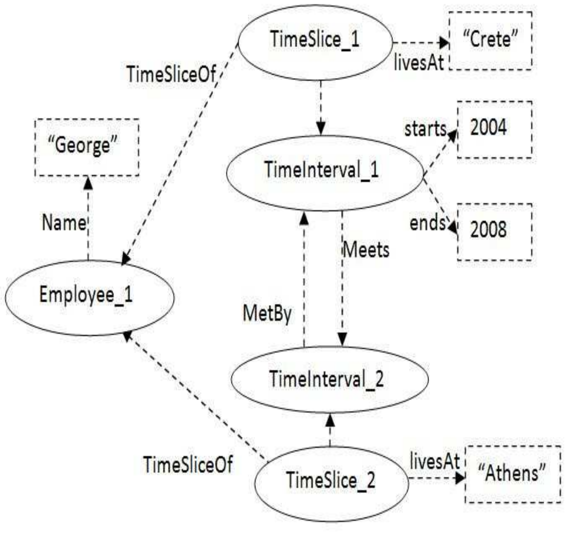


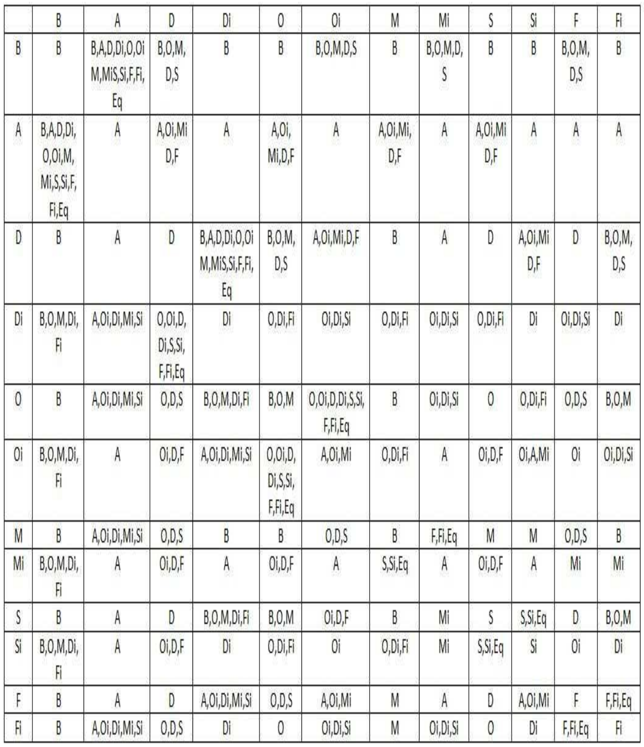
Fig. 5. Instantiation example.

Overall, the model demonstrates enhanced expressivity compared to previous approaches [18], [19], [23], [15] by combining 4D-fluents [9] with Allen's temporal relations, their formal semantics and composition rules as defined in [17].

A. Temporal Reasoning

Reasoning is realized by introducing a set of SWRL [27] rules operating on temporal intervals. Reasoners that support DL-safe rules such as Pellet [16] can be used for inference and consistency checking over temporal relations. In addition to reasoning applying on temporal relations, the Pellet reasoner is applied on the ontology schema to infer additional facts using OWL semantics (e.g., facts due symmetric relationships and class-subclass relationships).

The temporal reasoning rules are based on composing pairs of basic Allen's relations of Fig. 4 as defined in [17]. The composition table of basic Allen's relations is presented in Table I. Relations BEFORE, AFTER, MEETS, METBY, OVERLAPS, OVERLAPPEDBY, DURING, CONTAINS, STARTS, STARTEDBY, ENDS, ENDEDBY and EQUALS are represented using symbols B, A, M, Mi, O, Oi, D, Di, S, Si, F, Fi and = respecively. Compositions with EQUALS are not presented since these compositions keep the initial relations unchanged. The composition table represents the result of the composition of two Allen relations. For example, if relation $R1$ holds between $interval1$ and $interval2$ and relation $R2$ holds between $interval2$ and $interval3$ then the entry of the Table I corresponding to line $R1$ and column $R2$



	B	A	D	Di	O	Oi	M	Mi	S	Si	F	Fi
B	B	B,A,D,Di,O,Oi M,MiS,Si,F,Fi, Eq	B,O,M, D,S	B	B	B,O,M,D,S	B	B,O,M,D, S	B	B	B,O,M, D,S	B
A	B,A,D,Di, O,Oi,M, Mi,S,Si,F, Fi,Eq	A	A,Oi,Mi D,F	A	A,Oi, Mi,D,F	A	A,Oi,Mi, D,F	A	A,Oi,Mi D,F	A	A	A
D	B	A	D	B,A,D,Di,O,Oi M,MiS,Si,F,Fi, Eq	B,O,M, D,S	A,Oi,Mi,D,F	B	A	D	A,Oi,Mi D,F	D	B,O,M, D,S
Di	B,O,M,Di, Fi	A,Oi,Di,Mi,Si	O,Oi,D, Di,S,Si, F,Fi,Eq	Di	O,Di,Fi	Oi,Di,Si	O,Di,Fi	Oi,Di,Si	O,Di,Fi	Di	Oi,Di,Si	Di
O	B	A,Oi,Di,Mi,Si	O,D,S	B,O,M,Di,Fi	B,O,M	O,Oi,D,Di,S,Si, F,Fi,Eq	B	Oi,Di,Si	O	O,Di,Fi	O,D,S	B,O,M
Oi	B,O,M,Di, Fi	A	Oi,D,F	A,Oi,Di,Mi,Si	O,Oi,D, Di,S,Si, F,Fi,Eq	A,Oi,Mi	O,Di,Fi	A	Oi,D,F	Oi,A,Mi	Oi	Oi,Di,Si
M	B	A,Oi,Di,Mi,Si	O,D,S	B	B	O,D,S	B	F,Fi,Eq	M	M	O,D,S	B
Mi	B,O,M,Di, Fi	A	Oi,D,F	A	Oi,D,F	A	S,Si,Eq	A	Oi,D,F	A	Mi	Mi
S	B	A	D	B,O,M,Di,Fi	B,O,M	Oi,D,F	B	Mi	S	S,Si,Eq	D	B,O,M
Si	B,O,M,Di, Fi	A	Oi,D,F	Di	O,Di,Fi	Oi	O,Di,Fi	Mi	S,Si,Eq	Si	Oi	Di
F	B	A	D	A,Oi,Di,Mi,Si	O,D,S	A,Oi,Mi	M	A	D	A,Oi,Mi	F	F,Fi,Eq
Fi	B	A,Oi,Di,Mi,Si	O,D,S	Di	O	Oi,Di,Si	M	Oi,Di,Si	O	Di	F,Fi,Eq	Fi

TABLE I
Composition Table for Allen's temporal relations.

denotes the possible relation(s) holding between $interval1$ and $interval3$. Not all compositions yield a unique relation as a result. For example the composition of relations $During$ and $Meets$ yields the relation $Before$ as result while the composition of relations $Overlaps$ and $During$ yields three possible relations *Starts, Overlaps and During*. Rules corresponding to compositions of relations *R1, R2* yielding unique relations $R3$ as a result can be represented using SWRL as follows:

$$R1(x,y) \wedge R2(y,z) \rightarrow R3(x,z)$$

An example of temporal inference rule is the following:

$$DURING(x,y) \wedge MEETS(y,z) \rightarrow BEFORE(x,z)$$

Rules yielding a set of possible relations as a result can't be represented in SWRL since disjunctions of atomic formulas are not permitted as a rule head. Instead, disjunctions of relations are represented using new relations whose compositions must also be defined and asserted into the knowledge base. For example, if the relation DOS represents the disjunction of relations *During, Overlaps* and *Starts*, then the composition of $Overlaps$ and $During$ can be represented as follows:

$$OVERLAPS(x,y) \wedge DURING(y,z) \rightarrow DOS(x,z)$$

Note that the set of possible disjunctions over all basic Allen's relations is 2^{13} but subsets of this set that are closed under composition (i.e., compositions of relation pairs from this subset yield also a relation in this subset) do exist [25], [28]. In this work we use the tractable subset introduced in [28].

In addition to the above, the following axioms are also asserted into the knowledge base:

- Four transitivity axioms (for the relations BEFORE, FINISHEDBY, CONTAINS, STARTEDBY).
- Six inverse axioms (relations AFTER, METBY, OVERLAPPEDBY, STARTEDBY, CONTAINS and FINISHEDBY are the inverses of BEFORE, MEETS, OVERLAPS, STARTS, DURING and FINISHES respectively).
- One equality axiom (relation EQUALS).
- Rules defining the relation holding between two intervals with known starting and ending points (e.g., if ending of $interval1$ is smaller than the start of *interval2* the *interval1* is *before interval2*) are part of the ontology as well.

Notice that, starting and ending points of intervals are represented using concrete datatypes such as *xsd:date* that support ordering relations. Axioms concerning relations that represent disjunctions of basic relations are defined using the corresponding axioms for these basic relations. Specifically, compositions of disjunctions of basic relations are defined as the disjunction of the compositions of these basic relations. For example the composition of relation DOS (representing the disjunction of *During, Overlaps and Starts*), and the relation $During$ yields the relation DOS as a result as follows:

$$DOS \circ During \rightarrow (During \vee Overlaps \vee Starts) \circ During \rightarrow$$

$$(During \circ During) \vee (Overlaps \circ During) \vee (Starts \circ During)$$

$$\rightarrow (During) \vee (During \vee Overlaps \vee Starts) \vee (During)$$

$$\rightarrow During \vee Starts \vee Overlaps \rightarrow DOS$$

The symbol $\circ$ denotes composition of relations, and compositions of basic (non-disjunctive) relations are defined using Table I. Similarly, the inverse of a disjunction of basic relations is the disjunction of the inverses of these basic relations as presented in Fig. 4. For example the inverse of the disjunction of relations $Before$ and $Meets$ is the disjunction of the inverse relations of $Before$ and $Meets$ ($After$ and $MetBy$ respectively).

By applying compositions of relations the implied relations may be inconsistent. Consistency checking is achieved using path consistency [14], [25], [28]. Path consistency is implemented by consecutive applications of the following formula:

$$\forall x,y,k\; R_s(x,y) \leftarrow R_i(x,y) \cap (R_j(x,k) \circ R_k(k,y))$$

representing intersection of compositions of relations with existing relations (the symbol $\cap$ denotes intersection and the symbol $\circ$ denotes composition and symbols R_i, R_j, R_k, R_s denote Allen relations). The formula is applied until a fixed point is reached (i.e., application of rules doesn't yield new inferences) or until the empty set is reached, implying that the ontology is inconsistent.

An additional set of rules defining the result of intersection of relations holding between two intervals are also introduced. These rules have the form:

$$R1(x,y) \wedge R2(x,y) \rightarrow R3(x,y)$$

where $R3$ can be the empty relation. For example the intersection of relation DOS (represents the disjunction of *During, Overlaps and Starts*), and the relation $During$ yields the relation $During$ as result:

$$DOS(x,y) \wedge During(x,y) \rightarrow During(x,y)$$

Intersection of relations $During$ and $Starts$ yields the empty relation, and an inconsistency is detected:

$$Starts(x,y) \wedge During(x,y) \rightarrow \bot$$

Notice that, using the full set of 2^{13} relations leads to intractability [29]. Tractable subsets of relations that polynomial time algorithms such as path-consistency are sound and complete (while these algorithms are approximation algorithms in the case of the full Allen algebra) do exist [25], [28], [30]. The largest such set (corresponding to the maximal tractable subset of Allen relations containing all basic relations when applying the path consistency method) comprises of 868 relations [25]. Tractable subsets of Allen relations containing 83 or 188 relations [28] can be used for reasoning as well, offering reduced expressivity but increased efficiency over the maximal subset of [25].

An ontology based on a set containing 83 relations (i.e., the *continuous endpoint subclass* presented in [28]) has been implemented in this work. Other relations corresponding to disjunctions of basic relations that are not supported (i.e., they don't belong to the subset referred to above) can't be asserted into the ontology. In [28] reasoning regarding time instants in addition to intervals is presented as well. Specifically qualitative relations regarding instants form a tractable set if the relation $\neq$ (i.e., a temporal instant is before *or* after another instant) is excluded. Reasoning regarding relations between interval and instants is achieved by translating interval relations to relations regarding their endpoints as specified in [17].

B. Querying Temporal Information

Querying temporal information over the semantic Web using general purpose languages such as [8] and SeRQL [3] is a tedious task. Recent work on query languages for temporal ontologies include TOQL [4] (extended with spatial operators at [33]) and t-SPARQL [18] using 4-D fluents and named graphs respectively for the representation of temporal information. Notice that, t-SPARQL suggests using named graphs as the underlying representation mechanism of temporal information and therefore, does not preserve OWL expressiveness, has no reasoning support and does not support representation of qualitative temporal expressions. TOQL handles all these issues. In this work TOQL is used for querying the temporal ontology.

TOQL is a query language that treats classes and properties of an ontology almost like tables and columns of a database. The language is enhanced with a set of temporal operators (i.e., the AT and Allen operators). TOQL follows an SQL-like syntax (SELECT-FROM-WHERE) and supports SQL operators and constructs such as LIMIT, OFFSET, AND, OR,MINUS, UNION, UNION ALL, INTERSECT, EXISTS, ALL, ANY, IN.

TOQL also introduces clause "AT" which compares a fluent property (i.e., the time interval in which the property is true) with a time period (time interval) or time point and returns fluents holding true at the specified time interval, thus enabling temporal queries without requiring familiarity with the underlying representation mechanism for the end user. For example the following TOQL query retrieves the name of the company employee "x" was working for, from time=3 to time=5:

```
SELECT Company.companyName
FROM Company, Employee
WHERE Company.hasEmployee:Employee AT(3,5)
AND Employee.employeeName LIKE "x"
```

The following Allen operators are also supported: BEFORE, AFTER, MEETS, METBY, OVERLAPS, OVERLAPPEDBY, DURING, CONTAINS, STARTS, STARTEDBY, ENDS, ENDEDBY and EQUALS, representing the corresponding relations holding between two time intervals specified either using quantitative (i.e., interval with specified end points) description or qualitative Allen relations. The following query retrieves the name of the company that hired employee "x" and *then* employee "y":

```
SELECT Company.companyName
FROM Company, Employee AS E1, Employee AS E2
WHERE Company.hasEmployee:E1
BEFORE Company.hasEmployee:E2
AND E1.employeeName like "x"
AND E1.employeeName LIKE "y"
```

In this work, extending TOQL to support queries over qualitative relations required certain modifications to the language. The basic SQL syntax remains the same, however, Allen operators aren't translated to comparisons of interval endpoints as in [4] but to Allen relations holding between intervals after reasoning is applied. The AT operator in [4] requires that interval endpoints are defined. Here, we introduce two additional operators namely *ALWAYS_AT* and *SOMETIME_AT* querying for fluents holding always during the interval in question and some time in the interval in question respecively. The AT operator in [4] corresponds to the proposed ALWAYS_AT operator. Specifically, the ALWAYS_AT operator returns fluents holding at intervals that EQUALS, CONTAINS, STARTEDBY or ENDEDBY the interval in question. The SOMETIME_AT operators returns fluents holding at intervals that OVERLAP, OVERLAPPEDBY, START, STARTEDBY, END, ENDEDBY, EQUAL, CONTAIN or DURING the interval in question. These semantics in conjunction with the reasoning mechanism will allow for application of the operators on qualitative intervals in addition to quantitative ones that are supported by the AT operator.

IV. Evaluation

The resulting OWL ontology is characterized by *SHRIF(D)* DL expressivity and it is decidable since it doesn't contain role inclusion axioms with cyclic dependences [21] (role axioms in the ontology are restricted to disjointness, transitivity and inverse axioms). Adding the set of temporal qualitative rules of Sec. III-A retains decidability since rules are DL-safe rules as defined at [26], [31] and they apply only on named individuals of the ontology Abox using Pellet (which support DL-safe rules [32]). Furthermore, computing the rules has polynomial time complexity since a tractable subset of Allen's relations is used.

As shown in [14], [25], [28], by restricting the supported relations set to a tractable subset of Allen's algebra, path consistency has $O(n^5)$ time complexity (with n being the number of intervals). Also, any time interval can be related with every other interval by at most k relations, where k is the size of the set of supported relations. Therefore, for n intervals, using $O(k^2)$ rules, at most $O(kn^2)$ relations can be asserted into the knowledge base. Note that, extending the model for the full set of relations would result into an intractable reasoning procedure.

An alternative approach towards implementing a temporal reasoner would be to extend Pellet to handle a (tractable) relations set, along with the supported axioms and path consistency checking, similarly to the way PelletSpatial [20] implements reasoning over RCC-8 topologic relations. This approach has the following advantages: (a) The underlying representation is more simple since only the 13 Basic Allen relations have to be defined and (b) certain improvements regarding efficiency and scalability can be added. On the other hand, this approach requires additional software to handle the ontology, while our approach requires only standard semantic Web tools such as Pellet and SWRL. Because reasoning is part of the ontology model, maintenance of the ontology requires that changes are applied to the ontology only and not to the reasoner (other approaches such as [20] require modifying both the ontology and the reasoner).

V. Conclusions and Future Work

We introduce an ontology model capable of handling temporal information in ontologies. The proposed model extends the 4D fluent representation of [4] to handle both quantitative and qualitative temporal information. The representation mechanism incorporates reasoning rules for inferring certain temporal relations from existing ones and for checking temporal assertions for consistency. Extending the model to support spatial relations and addressing scalability issues using appropriate indexing mechanisms are directions for further research.

Extending TOQL [4] to handle the proposed 4D fluent representation is another contribution of this work. A desirable feature of TOQL is that it does not require that the user be familiar with the peculiarities of the underlying 4D fluent representation mechanism (which may be complicated leading to complicated query expressions in other query languages such as SPARQL [8]). Extending SPARQL, the current W3C standard to support 4D fluents and similar operators is an important issue for future research. t-SPARQL [18] is an example of work along these lines. Notice though that t-SPARQL suggest using named graphs as the underlying temporal representation (does not support 4D fluents) and therefore, does not maintain full OWL expressiveness and has no reasoning support.

References

[1] P. Grenon and B. Smith. “SNAP and SPAN: Towards Dynamic Spatial Ontology”. Spatial Cognition and Computation, Vol 4, No. 1, pp 69–104, 2004.

[2] N. Noy and B. Rector. “Defining N-ary Relations on the Semantic Web”. W3C Working Group Note 12, April 2006, http://www.w3.org/TR/swbp-n-aryRelations/

[3] B. V. Aduna. “The SeRQL query language”. User Guide for Sesame 2.1, Chapter 9, 2002–2008. http://www.openrdf.org/doc/sesame2/2.1.2/users/ch09.html.

[4] E. Baratis, E.G. Petrakis, S. Batsakis, N. Maris and N. Papadakis. “TOQL:Temporal Ontology Querying Language” Int. Symposium on Spatial and Temporal Databases (SSTD 2009), pp. 338-354, 2009.

[5] J.R. Hobbs and P. Fang. “Time Ontology in OWL”. W3C Recommendation, September 2006. http://www.w3.org/TR/owl-time/.

[6] M. Klein and D. Fensel. “Ontology Versioning for the Semantic Web”. In *International Semantic Web Working Symposium (SWWS’01)*, pages 75–92, California, USA, July–August 2001.

[7] D. L. McGuinness and F. VanHarmelen. “OWL Web Ontology Language Overview”. W3C Recommendation, February 2004. http://www.w3.org/TR/owl-features.

[8] E. Prud’hommeaux and A. Seaborne. “SPARQL Query Language for RDF”. W3C Recommendation, January 2008. http://www.w3.org/TR/rdf-sparql-query.

[9] C. Welty and R. Fikes. “A Reusable Ontology for Fluents in OWL”. *Frontiers in Artificial Intelligence and Applications*, 150:226–236, 2006.

[10] H. Gregersen, and C.S. Jensen. “Temporal Entity Relationship Models - A Survey”. IEEE Transactions on Knowledge and Data Engineering 3, pp. 464-497, 1999.

[11] A. Artale, and E. Franconi. “A Survey of Temporal Extensions of Description Logics”. Annals of Mathematics and Artificial Intelligence, 30(1-4), 2001.

[12] C. Lutz, F. Wolter, and M. Zakharyaschev. “Temporal Description Logics: A Survey”. In Proc. TIME08, IEEE Press, 2008.

[13] C. Gutierrez, C. Hurtado, and A. Vaisman. “Introducing Time into RDF”. In IEEE Transactions on Knowledge and Data Engineering, 19(2) , pp. 207-218, 2007.

[14] J. Renz, and B. Nebel. “Qualitative Spatial Reasoning using Constraint Calculi”. In Handbook of Spatial Logics, Springer, Netherlands , pp. 161-215, 2007.

[15] A. Sheth, I. Arpinar, M. Perry, and F. Hakimpour. “Geospatial and Temporal Semantic Analytics”. In Handbook of Research in Geoinformatics, Hassan A. Karimi (eds), Chapter XXI ,2009.

[16] B. Parsia and E. Sirin. “Pellet: An OWL DL reasoner.” Third International Semantic Web Conference (ISWC 2004), 2004.

[17] J. F. Allen. “Maintaining Knowledge About Temporal Intervals”. *Communications of the ACM.* 26:832-843, 1983.

[18] J. Tappolet, and A. Bernstein “Applied Temporal RDF: Efficient Temporal Querying of RDF Data with SPARQL.” In: Proceedings of the European Semantic Web Conference, LNCS 5554, 308-322, 2009.

[19] H. Chen, F. Perich, T. Finin, and A. Joshi “SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications” Int. Conference on Mobile and Ubiquitous Systems: Networking and Services, pp. 258–267, 2004.

[20] M. Stocker, and E. Sirin “PelletSpatial: A Hybrid RCC-8 and RDF/OWL Reasoning and Query Engine” OWLED 2009.

[21] I. Horrocks, O. Kutz and U. Sattler “The Even More Irresistible SROIQ” In: Proc. KR 2006, Lake District, UK, 2006.

[22] B.C. Grau, I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider and U. Sattler. “OWL 2: The Next Step for OWL.” In: Web Semantics: Science, Services and Agents on the World Wide Web, vol. 6, pp. 309-322, 2008.

[23] V. Milea, F. Frasincar, and U. Kaymak “Knowledge Engineering in a Temporal Semantic Web Context” In The Eighth International Conference on Web Engineering (ICWE 2008), IEEE Computer Society Press,2008.

[24] B. Motik, P.F. Patel-Schneider, and I. Horrocks “OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax” W3C Recommendation, 2009. http://www.w3.org/TR/owl2-syntax/

[25] B. Nebel, and H.J. Burckert “Reasoning about Temporal Relations: A Maximal Tractable Subclass of Allen’s Interval Algebra” Journal of the ACM (JACM), Vol.42(1), pages:43–66, 1995.

[26] J. de Bruijn “RIF RDF and OWL Compatibility.” W3C Working Draft (July 2009). http://www.w3.org/TR/rif-rdf-owl/

[27] 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, 2004. http://www.w3.org/Submission/SWRL/

[28] P. van Beek, and R. Cohen “Exact and approximate reasoning about temporal relations.” Computational intelligence, Vol 6(3), pp. 132–147, 1990.

[29] M. Vilain, H. Kautz, and P. van Beek. “Constraint propagation algorithms for temporal reasoning: a revised report.” In Readings in qualitative reasoning about physical systems, edited by D. S. Weld and J. de Kleer, pp. 373-381, 1989.

[30] A. Krokhin, P. Jeavons, and P. Jonsson. “Reasoning about temporal relations: The tractable subalgebras of Allen’s interval algebra.” Journal of the ACM 50(5), pp. 591-640, 2003.

[31] B. Motik, U. Sattler, and R. Studer “ Query Answering for OWL-DL with rules.” Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 3(1), Rules Systems, pp. 41-60, July 2005.

[32] V. Kolovski, B. Parsia, and E. Sirin “Extending the SHOIQ (D) tableaux with dl-safe rules: First results” Proceedings International Workshop on Description Logic, DL-2006, 2006.

[33] S. Batsakis, and E. Petrakis “SOWL:Spatio-temporal Representation, Reasoning and Querying over the Semantic Web” 6th International Conference on Semantic Systems (I-SEMANTICS 2010), Graz, Austria, September 1-3, 2010.