The spatio-temporal semantics from a perdurantism perspective

Benjamin Harbelot*, Helbert Arenas†, Christophe Cruz‡
Laboratoire Le2i, UMR-6302 CNRS,Departement Informatique
University of Burgundy
Dijon, France
*benjamin.harbelot@checksem.fr, †helbert.arenas@checksem.fr, ‡christophe.cruz@u-bourgogne.fr

***Abstract*—In this paper we present the "continuum model". Our work follows a "perdurantism" approach and is designed to handle dynamic phenomena extending the 4D-fluent with the use of semantic web technologies. In our approach we represent dynamic entities as constituted by time slices each with semantic, geometric and temporal components. Our model is able to link the diverse representations of an entity and allows the inference of qualitative information from quantitative one. The inference results are later added to the ontology in order to enhance the knowledge base. The model has been implemented using OWL and SWRL. Our preliminary results are promising and we plan to further develop the model in the near future to increase the suitable data sources.**

***Keywords*-spatio-temporal; semantics; GIS; perdurantism.**

I. INTRODUCTION

For the design of a spatio-temporal knowledge system, it is necessary to consider the three components of an entity representation: 1) Spatial: consisting in the geometry, 2) Temporal: which defines the interval of existence of the geometries and finally 3) Semantic: which defines a meaning for the entity beyond the purely geographic one [1]. Currently available GIS tools lack the capacity to perform inference or reasoning on information from spatio-temporal dynamic phenomena. An alternative to classic GIS tools are Semantic Web technologies, tools specifically designed to perform reasoning and inference. In this research we use Semantic Web technologies to develop the "continuum model", an ontology that allows us to represent diverse dynamic entities and analyse their relationships along time. Traditionally ontologies are static in the sense that the information represented in them does not change in time or space. In this paper we introduce the continuum model, an ontology that extends the 4D-fluent providing it with the required capabilities to keep track of spatial and semantic evolution of entities along time.

In Section II we discuss related work in the field of spatio-temporal knowledge representation. In Section III we introduce the continuum model, we present the model specification using description logics, in Section IV we describe how the model operates using an example and later we indicate our conclusions and future work.

II. RELATED WORK

The development of a spatial-temporal knowledge system involves two aspects, first the representation of the knowledge and second, the necessary mechanisms to perform analysis and querying.

A. Representing temporal data

The two main philosophical theories concerning the representation of object persistence over time are: *endurantism* and *perdurantism*. The first one, *endurantism*, considers objects as three dimensional entities that exist wholly at any given point of their life. On the other hand, *perdurantism*, also known as the four dimensional view, considers that entities have temporal parts, "time slices" [2]. From a perdurantism point of view the temporal dimension of an entity is composed by all its time slices. It therefore represents the different properties of an entity over time as *fluent*. A *fluent* is a property valid only during certain intervals or moments in time. From a designer point of view, the *perdurantism* approach offers advantages over the *endurantism* allowing richer representations of real world phenomena [3].

The implementation of a *perdurantism* approach within an ontology, requires the conversion of static properties into dynamic ones. The two primary Semantic Web languages are OWL and RDF, unfortunately both of them provide limited support for temporal dynamics [4]. The OWL-Time ontology describes the temporal content of web pages and temporal properties of web services. Moreover, this ontology provides good support for expressing topological relationships between times or time intervals, as well as times or dates [5]. However OWL allows only binary relations between individuals. In order to overcome this limitation several methodologies have been proposed for the representation of dynamic objects and their properties. Among the most well known are the temporal description logic, temporal RDF, versioning, reification, N-ary relationships and the 4D-fluent approach.

Temporal RDF [6] proposes an extension of the standard RDF for naming properties with the corresponding time interval. This allows an explicit management of time in RDF. However Temporal RDF uses only RDF triples, therefore it does not have all the expressiveness of OWL for instance, it is not possible to employ qualitative relations. Reification is a technique used to represent n-ary relations, extending languages such as OWL that allow only binary relations [7]. In [4], the authors developed a lightweight model using Reification. The model is designed to be deployed on top of existing OWL ontologies extending their temporal capabilities. The model also implements a set of SWRL (Semantic Web Rule Language) operators to query the ontology. Reification allows the use of a triple as object or subject of a property. But this method has also its limitations, for instance the transformation from a static property into a dynamic one increases substantially the complexity of the ontology, reducing the querying and inference capabilities. Additionally reification is prone to redundant objects which reduces its effectiveness. Versioning is described as the ability to handle changes in ontologies by creating and managing multiple variants of them [8]. However, the major drawback of Versioning, is the redundancy generated by the slightest change of an attribute. In addition, any information requests must be performed on multiple versions of the ontology affecting its performance.

The 4D-fluent approach is based on the *perdurantism* philosophical approach.It considers that the existence of an entity can be expressed with multiple representations, each corresponding to a defined time interval. In the literature 4D-fluent is the most well known method to handle dynamic properties in an ontology. It has a simple structure allowing to easily transform a static ontology into a dynamic one although it has some limitations [9]. The 4D fluent approach allows the recording of frequent time slices but it can not handle explicit semantics. This fact causes two problems: 1) It is difficult to maintain a close relationship between geometry and semantics; and 2) It increases the complexity for querying the temporal dynamics and understanding the modelled knowledge. Furthermore, this approach does not define qualitative relations to describe the type of change that has occurred or to describe the temporal relationships between objects. We cannot then know which entities have undergone a change and what entities might be the result of that change. Regardless of its limitations the 4D-fluent approach offers a solid starting point for the representation of temporal information in OWL. A work based on 4D-fluent is SOWL, which extends the ontology OWL-time making it able to handle qualitative relations between intervals, such as “before” or “after” even with intervals with vague ending points [10].

B. Querying the ontology

Traditionally SPARQL has been the most common language to query an ontology. SPARQL is a W3C recommendation that operates at the level of RDF graphs. However, the queries become relatively complex in a space-temporal system. An extension of this language, st-SPARQL [11], defines new functions that allow it to handle geometries but not temporal data. St-SPARQL is based on an extension of RDF called st-RDF that integrates contact geometries and incorporates time in RDF. St-SPARQL and SPARQL are both based on RDF graphs, therefore it is impossible to draw any inference with them.

In [12] the authors introduce a model in which spatial-temporal information contained in a database and a spatial-temporal inference system work together. However, no information is given on the Semantic Web technologies, only the Java language is quoted as a component of the inference engine, therefore the universality and effectiveness of the inference system can be questioned. Another work is [13] in which the authors propose a reasoning system that combines the topological calculus capabilities of a GIS and the inference capabilities of the semantic web field. However the notion of time is not incorporated in this model.

The capability of switching from quantitative to qualitative data is only possible with a reasoning system. In the case of SOWL this is possible thanks to the implementation of SWRL built-ins. In SOWL the built-ins allow the system to infer topological, directional and metric relations between entities. Qualitative information can be inferred from quantitative one and can be used as an alternative in the case of missing quantitative data. In order to query the ontology the developers of SOWL implemented a language similar in syntax to SQL. This language performs simple spatial-temporal querying for both static and dynamic data [10].

Our literature review suggest us that the most suitable approach to develop a spatial-temporal knowledge system should follow a 4D-fluent approach using SWRL built-ins to perform complex queries and reasoning. In the next section we will describe how we implemented this approach in the continuum model.

III. THE CONTINUUM MODEL

The 4D-fluent approach does not allow an entity to change its nature, only allows the change of the value of some of its properties. However the semantics associated with a geometry may change. For example a land parcel may change from being forest into being urban. In this example the geometry has not changed, however there is a semantic change (See figure 1A). It is equally possible that the semantics might not change while the geometry evolves. For instance, a given urban land parcel might expand by purchasing neighbouring parcels (see figure 1B).

In order to represent a dynamic entity in the continuum model we create a set of object time slices, each constituted by three components as depicted in figure 2A: 1)Semantic: To describe the knowledge associated with the entity. 2) Spatial: It is the graphical representation. 3) Temporal: It represents the interval or time instants that describe the temporal existence. The goal of the continuum model is to follow the evolution of entities though time. To achieve this goal the model records the changes that entities might

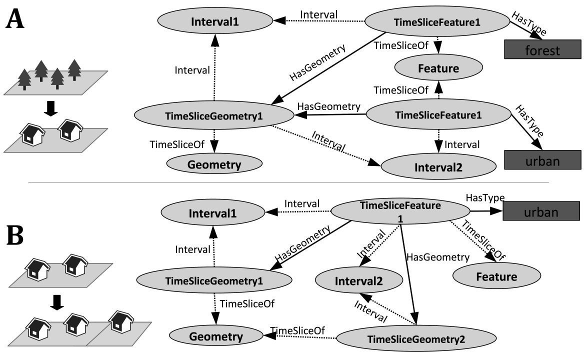

Figure 1. Examples of the evolution A) Two different semantic objects for the same geometry. B) Two related geometries for the same semantic object.

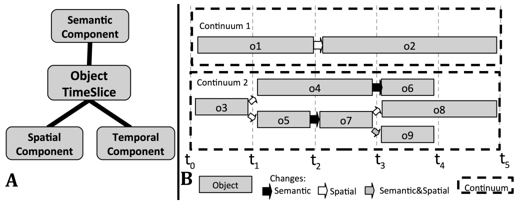

Figure 2. A) The three components of an entity within the continuum model. B) Using the continuum model to represent the evolution of an entity.

go through in their semantic or spatial components along time. For this purpose the model creates a new component representation every time a change occurs (spatial or semantic). The resulting child object retains all the remaining characteristics from the original parent object. Each change adds to the genealogy of the spatio-temporal objects. The parent-child relation is recorded in the system, allowing the analysis and querying of the information. The model enforces a coherency between the time intervals of objects contained in the system.

Figure 2B depicts an example of objects genealogy. In this example objects “o4” and “o5” are children of object “o3”, and are the result of an spatial change in the parent object. The system enforces temporal coherency, children objects can not occur before the parent interval. It is possible to characterize the evolution of each object in the model according to the conceptual hierarchy depicted in figure 3.

The continuum groups related objects, which have a valid time interval of existence.The model links individual objects to their context. For instance an object can belong to more than one continuum, therefore continuums can intersect. Our system allows the definition of qualitative relations between spatial-temporal objects, even when this object belong to different continuums. Figure 2B depicts the evolution of an entity and how the continuum concept is used to study it.

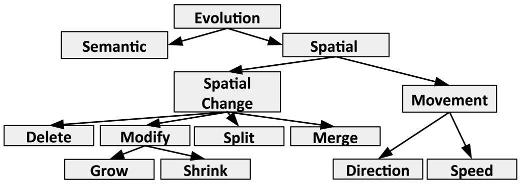

Figure 3. Qualification of transition in the spatial graph.

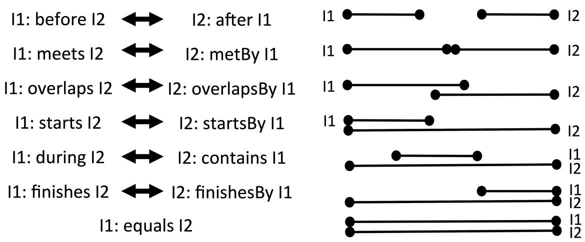

Figure 4. Allen temporal relations.

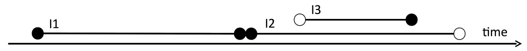

Figure 5. Using Allen temporal relations to infer new knowledge.

In our model we have implemented qualitative temporal relations based on binary and mutually exclusive relations as proposed by Allen [14] (see figure 4). The addition of Allen relations increase the expressive power of the system by adding qualitative information in addition to the quantitative one. By using defined Allen relations between intervals we can obtain qualitative information even from intervals with vague endpoints in a similar fashion to [9]. For example, figure 5 depicts intervals “I1”, “I2” and “I3”. While we know the start and ending points of “I1”, we do not know the ending point of “I2”, and we do not know the starting point of “I3’. However we know that “I1” meets “I2” and that “I2” contains “I3”. Then we can infer that because “I2” contains “I3”,then “I3” must be after “I1”, even if the information about start and ending points is incomplete. Lack of knowledge caused by semi closed intervals is largely filled by the integration of Allen relations to the model.

In GIS, objects or regions are represented by points, lines, polygons or other more complex figures based on these geometries. All these geometries are defined using the coordinates of points which are quantitative information. There are mainly three types of relationships between geometries: directional, metric, and topological relationships. The topological analysis between two objects is done using the models: Dimensionally Extended Nine-Intersection Model (DE-9IM) or RCC8 [15]. In both cases, we obtain an equivalent set of topological relationships for specific regions. To calculate the spatial relationships between two geometries the DE-9IM model takes into account the inside, the outside, and the contour of the geometries leading to the analysis of nine intersections as described in [15] .

There are eight possible spatial relationships of the result-

Table I
TOPOLOGICAL PREDICATES AND THEIR CORRESPONDING MEANINGS.

Topological	Predicate Meaning
Equals	The Geometries are topologically equal.
Disjoint	The Geometries have no point in common.
Intersects	The Geometries have at least one point in common (the inverse of Disjoint).
Touches	The Geometries have at least one boundary point in common, but no interior points.
Crosses	The Geometries share some but not all interior points, and the dimension of the intersection is less than that of at least one of the Geometries.
Overlaps	The Geometries share some but not all points in common, and the intersection has the same dimension as the Geometries themselves.
Within	Geometry A lies in the interior of Geometry B
Contains	Geometry B lies in the interior of Geometry A (the inverse of Within)

ing analysis-9IM (see table I).

The relationships based on quantitative information can be translated later into qualitative data [16], in a similar fashion as we have described for the temporal aspect. By analysing a set of moments and time intervals it is possible to deduce qualitative topological relationships between objects.

In this section we use a Tarski-style specification to describe the model main components.

To represent time intervals we follow the semantics suggested by Artale and Franconi (1998). We can think of the temporal domain as a linear structure $\mathcal{T}$ composed by a set of temporal points $\mathcal{P}$. The components of $\mathcal{P}$ follow a strict order $<$, which forces all points between two temporal points t_1 and t_2 to be ordered. By selecting a pair $[t_1, t_2]$ we can limit a closed interval of ordered points. The set of interval structures in $\mathcal{T}$ is represented by $\mathcal{T}^\star_<$ [17].

Temporal Points:

$$\mathcal{P} \qquad \mathcal{P}^\mathcal{I} \subseteq \Delta^\mathcal{I}$$

Time Intervals:

$$\mathcal{T}^\star_< \qquad [to, tf] \doteq \{x \in \mathcal{P} | to \leq x \leq tf, to \neq tf\} in \mathcal{T}$$

To define the relations identified by Allen [14] (See figure 4) we first define two intervals $i1$ and $i2$: $\mathcal{T}^\star_<(i1)$, $\mathcal{T}^\star_<(i2)$, being i_{to} the starting point and i_{tf} the ending point of the intervals.

$$\begin{array}{ll}
Before(i1, i2) & (i1_{tf} < i2_{to}) \\
Meets(i1, i2) & (i1_{tf} = i2_{to}) \\
Overlaps(i1, i2) & (i1_{tf} > i2_{to}) \land (i1_{tf} < i2_{tf}) \\
Starts(i1, i2) & (i1_{to} = i2_{to}) \land (i1_{tf} < i2_{tf}) \\
During(i1, i2) & (i1_{to} > i2_{to}) \land (i1_{tf} < i2_{tf}) \\
Finishes(i1, i2) & (i1_{to} > i2_{to}) \land (i1_{tf} = i2_{tf}) \\
Equasl(i1, i2) & (i1_{to} = i2_{to}) \land (i1_{tf} = i2_{tf})
\end{array}$$

The Spatial representation of an object ($\mathcal{SR}$) is composed by a spatial reference system (SRS) and a geometry ($\mathcal{G}$) (A more complex definition is possible, however for the sake of simplicity we will refer only to the essential components of a geographic feature definition).

Spatial Reference System: As defined by the European Petroleum Standards Group (EPSG) [18]

$$SRS^\mathcal{I} \subseteq \Delta^\mathcal{I}$$

Geometries: A set of coordinates that define points, lines, curves, surfaces and polygons.

$$\mathcal{G}^\mathcal{I} \subseteq \Delta^\mathcal{I}$$

$$\forall HasSRS.SRS \equiv \{x \in \Delta^\mathcal{I} | \forall s.(x, s) \in HasSRS^\mathcal{I} \rightarrow s \in SRS^\mathcal{I}\}$$

$$\forall HasGeom.\mathcal{G} \equiv \{x \in \Delta^\mathcal{I} | \forall g.(x, g) \in HasGeo^\mathcal{I} \rightarrow g \in \mathcal{G}^\mathcal{I}\}$$

Then the spatial representation can be defined as:

$$\mathcal{SR} \equiv \forall HasSRS.SRS \sqcap \forall HasGeom.\mathcal{G}$$

The spatial relations between geometries are defined by the Extended Nine-Intersection model (DE-9IM) [15].

The semantic component of the objects is represented by $\mathcal{S}$. It describes the nature of the entities and can be composed by one or more alphanumeric properties.

Each object time slice ($\mathcal{O}$) in the continuum model has three components: 1) a time interval ($\mathcal{T}^\star_<$), 2) a spatial representation ($\mathcal{SR}$) and 3)a semantic component ($\mathcal{S}$).

$$\mathcal{O} \equiv \quad \forall HasSR.\mathcal{SR} \quad \sqcap \quad \forall HasInterval.\mathcal{T}^\star_< \quad \sqcap \quad \forall HasSemDef.\mathcal{S}$$

In the continuum model a change on the spatial representation or on the semantic component generates a new object which has a *child* - *parent* relationship with the original object, additionally we know that the time interval of the parent object *meets* the time interval of the child object (see figure 4). The parent child relationship between object $o1$ and $o2$ is defined by the relationships between their spatial representations ($o1_{sr}$ and $o2_{sr}$), their semantic definitions ($o1_s$ and $o2_s$) and their time intervals ($o1_i$ and $o2_i$)

$$\begin{array}{ll}
\forall HasChild.\mathcal{O} & \{o_1 \in \mathcal{O}^\mathcal{I} | \forall o2.(o1, o2) \in HasChild^\mathcal{I} \rightarrow \\
& o2 \in \mathcal{O}^\mathcal{I} \land \\
& \exists((o1_{sr} \neq o2_{sr}) \lor (o1_s \neq o2_s)) \land \\
& (meets(o1_i, o2_i))\}
\end{array}$$

where: $\{o1, o2\} \in \mathcal{O}$, $\{o1_{sr}, o2_{sr}\} \in \mathcal{SR}$ and $\{o1_s, o2_s\} \in \mathcal{S}$

The spatial transitions in the model are a subset of the $HasChild$ relationship: $SpatialEvolution \sqsubseteq HasChild$ We have implemented the following spatial transitions: (see table I for a definiton of topological relations)

$$\begin{array}{l}
Merge(input, output) \\
input = \{a_1, a_2..a_n\} | \forall x \in input \rightarrow \mathcal{SR}(x) \\
\mathcal{SR}(output) \land output = (a_1 \cup a_2 \cup \ldots a_n)
\end{array}$$

$$\begin{array}{l}
Split(input, output) \\
\mathcal{SR}(input) \land output = \{a_1, a_2..a_n\} | \forall x \in output \rightarrow SR(x)
\end{array}$$

$$Equals(input, output)$$

$$\begin{array}{l}
Delete(input, output) \\
\mathcal{SR}(input) \land Equals(output, \emptyset)
\end{array}$$

$$\begin{array}{l}
Grow(input, output) \\
\{input, output\} \in \mathcal{SR} \land Within(input, output)
\end{array}$$

$$\begin{array}{l}
Shrink(input, output) \\
\{input, output\} \in \mathcal{SR} \land Contains(input, output)
\end{array}$$

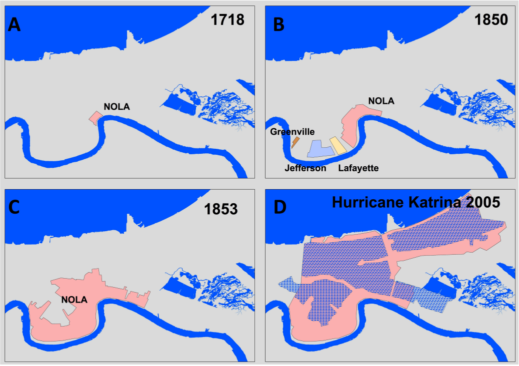

Figure 6. City of New Orleans along time.

IV. EXAMPLE CONTINUUM

The continuum model is flexible enough to be adapted in multiple fields. For this example we will use it to study the urban evolution of the city of New Orleans, Louisiana. Figures 6 and 7 represent the urban evolution of the entity *"city of New Orleans"*. Each one of its multiple representations along its history is a *time slice*. Figure 6 depicts the urban evolution of the city, we can see the historic French Quarter, founded in 1718 (Figure 6A), and how the city grew until around 1853 (Figure 6C) when it went trough a conurbation process with the cities of Greenville, Jefferson and Lafayette (Figure 6B). The city continued its growth and by 1949 it reached its approximately modern size. In August of 2005 Hurricane Katrina landed near the city causing a major flood, also depicted in figure 6D. First we define the class Human Settlement ($\mathcal{HS}$) as a subclass of the objects ($\mathcal{O}$), $\mathcal{HS} \sqsubseteq \mathcal{O}$ therefore it has all three components, spatial ($\mathcal{SR}$), temporal ($\mathcal{T}$) and semantic ($\mathcal{S}$).

The conurbation process involves two cities merging. Using the model we can represent the process as:

$$\{a1, b\} \in \mathcal{HS} | Contains(a1_{sr}, b_{sr}) \wedge Meets(a1_i, b_i)$$
$$\rightarrow ConUrbation(a1, b)$$

Figure 9 depicts the form how the model will be used in the New Orleans example. The spatial representation of the time slice *Nola2* contains the spatial representation of *Jefferson*, therefore there is a conurbation process by the year 1853.

Figure 6D depicts the area flooded by Hurricane Katrina in 2005.We can create a new class *risk areas* as $\mathcal{RA}$ ($\mathcal{RA} \sqsubseteq \mathcal{O}$), representing the flooded area. Then we can identify the process *growth in risk area* as:

$$\{a, b\} \in \mathcal{HS} \wedge r \in \mathcal{RA} | Grow(a, b) \wedge$$
$$(Overlaps(a_{sr}, r_{sr}) = \emptyset) \wedge (Overlaps(b_{sr}, r_{sr}) \neq \emptyset)$$
$$\rightarrow GrowthInRiskArea(a, b)$$

V. CONCLUSION

Figures 8 depicts the representation of the urban growth using the classic 4D-fluent approach, while figure 9 depicts the continuum model. In the later, following the approach by Welty and Fikes, classes *TimeSlice* and *TimeInterval* with properties *TimeSliceOf* and *HasInterval* are introduced to allow the ontology to handle temporal entities. Class *TimeSlice* is the domain class for entities representing temporal parts and class *Interval* is the domain class of intervals. A time interval holds the temporal information of a timeslice. Property *TimeSliceOf* connects an instance of class *TimeSlice* with an entity, and property *HasInterval* connects an instance of class *TimeSlice* with an instance of class *Interval*. Our model enhances the understanding of the data represented in the ontology. First, we removed the notion of *TimeSlice* which does not refer to any object in the real world. *TimeSlice* are replaced by instance from explicit Class providing an explicit semantic. Moreover, the 4D-fluent approach is enhanced by adding several types of qualitative relations.Temporal Allen relations and spatial relationships resulting from analysis-9IM. Understanding data semantics is at the core of our work providing an easier way to manage data and reduces queries complexity. When using, reasoning capabilities specific to the web semantic, the system may enrich itself the knowledge store in the ontology. Our model offers explicit semantic and flexibility for semantics interoperability between information systems

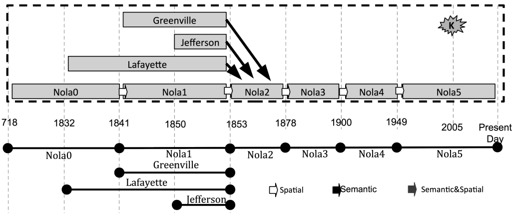

Figure 7. Time frame or urban evolution

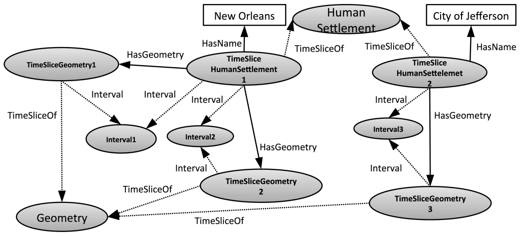

Figure 8. Representation using the 4D fluent

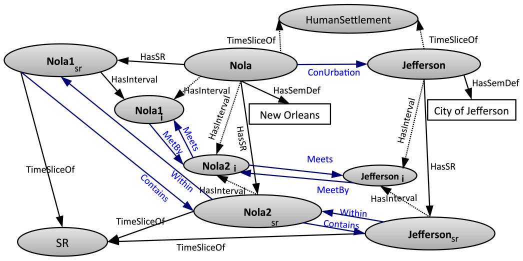

Figure 9. Representation using the continuum model

and data sharing.

The rules are executed via a graphical interface using the Jena API to connect to the ontology and JDBC to access the database. The application automatically detects the presence of spatial built-ins in SWRL rule and automatically starts the calculation in the database. However, we note two limitations to this model: 1) the treatment of a query containing a spatial built-ins can be very long depending on the number of geometry involved in spatial analysis, 2) the execution of SWRL rule containing spatial built-ins is currently dependent on our application and cannot be executed, for example, from traditional plugin SQWRL Tab Query of the Protégé tool.

A spatial built-ins uses quantitative data to launch a spatial analysis which establishes qualitative relations between the geometries involved in the calculation. Currently, our system can automatically rewrite SWRL rules containing spatial built-ins. On one hand, this allows not repeating the calculations that have already been performed. On the other hand, it also provides a SWRL rule no longer containing spatial built-ins but rather a qualitative relationship expressed through a property defined in the ontology. For example, this rules asking for people within a restaurant: $feat : restaurant(?x) \wedge feat : people(?y) \wedge spatialswrlb : Within(?x, ?y) \rightarrow sqwrl : select(?x)$

will be rewritten as:

$feat : restaurant(?x) \wedge feat : people(?y) \wedge sa : HasWithin(?x, ?y) \rightarrow sqwrl : select(?x)$

However, the addition of new objects in the ontology as well as in the spatial database can make the result of the rewritten query incomplete. It should be necessary to restart a calculation with a spatial built-in to update the qualitative relationships between geometries.

REFERENCES

[1] M. Yuan, “Use of a three-domain representation to enhance GIS support for complex spatial-temporal queries,” *Transactions in GIS*, vol. 3, pp. 137–159, March 1999.

[2] C. Welty and R. Fikes, “A reusable ontology for fluents in OWL,” in *Proceedings of 2006 conference on Formal Ontology in Information Systems (FOIS 2006)*, 2006, pp. 226–236.

[3] M. Al-Debei, M. Mourhaf Al Asswad, S. de Cesar, and M. Lycett, “Conceptual modelling and the quality of ontologies: Endurantism vs. perdurantism,” *International Journal of Database Management Systems*, vol. 4, no. 3, June 2012.

[4] M. O’Connor and A. Das, “A method for representation and querying temporal information in OWL,” in *Proceedings of Biomedical Engineering Systems and Technologies BIOSTEC 2010*, 2010, pp. 97–110.

[5] J. Hobbs and F. Pan, “Time ontology in OWL,” (Online) http://www.w3.org/TR/owl-time/, (Accessed on November 2012).

[6] C. Gutierrez, A. Hurtado, and A. Vaisman, “Introducing time into RDF,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, pp. 207–218, February 2007.

[7] P. Hayes, “RDF semantics, W3C Recomendation, 10 february 2004,” (Online) http://www.w3.org/TR/rdf-mt/, 2004, (Accessed on November 2012).

[8] M. Klein and D. Fensel, “Ontology versioning on the Semantic Web,” in *Proceedings of the First International Semantic Web Working Symposium SWWS’01*, July 2001, pp. 75–91.

[9] S. Batsakis and E. Petrakis, “SOWL: Spatio-temporal representation reasoning and querying over the semantic web,” in *Proceedings of the 6th. International Conference on Semantic Systems I–SEMANTICS 2010*, September 2010, pp. 15:1–15:9.

[10] ——, “SOWL: a framework for handling spatio-temporal information in OWL2.0,” *Rule Based Reasoning, Programming, and Applications Lecture Notes in Computer Science*, vol. 6826, pp. 242–249, 2011.

[11] M. Koubarakis and K. Kyzirakos, “Modeling and querying metadata in the semantic sensor web: The model stRDF and the query language stSPARQL,” in *The Semantic Web: Research and Applications*, ser. Lecture Notes in Computer Science, L. Aroyo, G. Antoniou, E. Hyvnen, A. Teije, H. Stuckenschmidt, L. Cabral, and T. Tudorache, Eds. Springer Berlin Heidelberg, 2010, vol. 6088, pp. 425–439. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-13486-9_29

[12] K. Ryu and Y. Ahn, “Application of moving objects and spatiotemporal reasoning,” 2001, a TIMECENTER Technical Report.

[13] A. Karmacharya, C. Cruz, F. Boochs, and F. Marzani, “Integration of spatial processing and knowledge processing through the semantic web stack,” in *GeoSpatial Semantics*, ser. Lecture Notes in Computer Science, C. Claramunt, S. Levashkin, and M. Bertolotto, Eds. Springer Berlin Heidelberg, 2011, vol. 6631, pp. 200–216. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-20630-6_13

[14] J. Allen, “Maintaining knowledge about temporal intervals,” *Communications of the ACM*, vol. 26, no. 11, pp. 832–843, Nov. 1983. [Online]. Available: http://doi.acm.org/10.1145/182.358434

[15] C. Strobl, *Encyclopedia of GIS Springer*. Springer, 2008, ch. Dimensionality Extended Nine–Intersection Model (DE–9IM), pp. 240–245.

[16] N. Brisaboa, I. Mirbel, and B. Pernici, “Constraints in spatio-temporal databases: A proposal for classification,” in *Proceedings of the 3th. International Workshop on Evaluation of Modeling Methods in System Analysis and Design*, 1998.

[17] A. Artale and E. Franconi, “A temporal description logic for reasoning about actions and plans,” *Journal of Artificial Intelligence Research*, vol. 9, pp. 463–506, 1998.

[18] R. Lake, “Introduction to GML,” 2004, http://www.w3.org/Mobile/posdep/GMLIntroduction.html.