A Geo-ontology Design Pattern for Semantic Trajectories

Yingjie Hu[1], Krzysztof Janowicz[1], David Carral[2], Simon Scheider[3], Werner Kuhn[3], Gary Berg-Cross[4], Pascal Hitzler[2], Mike Dean[5], and Dave Kolas[5]

[1] Department of Geography, University of California Santa Barbara, USA
{yingjiehu,jano}@geog.ucsb.edu
[2] Kno.e.sis Center, Wright State University, USA
{carral.2,pascal.hitzler}@wright.edu
[3] Institute for Geoinformatics University of Münster, Germany
{simon.scheider,kuhn}@uni-muenster.de
[4] Spatial Ontology Community of Practice (SOCOP), USA
gbergcross@gmail.com
[5] Raytheon BBN Technologies, USA
{mdean,dkolas}@bbn.com

Abstract. Trajectory data have been used in a variety of studies, including human behavior analysis, transportation management, and wildlife tracking. While each study area introduces a different perspective, they share the need to integrate positioning data with domain-specific information. Semantic annotations are necessary to improve discovery, reuse, and integration of trajectory data from different sources. Consequently, it would be beneficial if the common structure encountered in trajectory data could be annotated based on a shared vocabulary, abstracting from domain-specific aspects. Ontology design patterns are an increasingly popular approach to define such flexible and self-contained building blocks of annotations. They appear more suitable for the annotation of interdisciplinary, multi-thematic, and multi-perspective data than the use of foundational and domain ontologies alone. In this paper, we introduce such an ontology design pattern for semantic trajectories. It was developed as a community effort across multiple disciplines and in a data-driven fashion. We discuss the formalization of the pattern using the Web Ontology Language (OWL) and apply the pattern to two different scenarios, personal travel and wildlife monitoring.

1 Introduction

The term *trajectory* is used in many different contexts. It can be defined as a path through space on which a moving object travels over time. For example, the path of a projectile can be described by a mathematical model which returns the idealized position of the projectile at each point in time. In other cases, such as studying animal movement, trajectories are defined by a sparse set of temporally-indexed positions or "fixes", while the exact path between these fixes is unknown and has to be estimated, e.g., by using Brownian Bridges [23].

T. Tenbrink et al. (Eds.): COSIT 2013, LNCS 8116, pp. 438–456, 2013.

In some of these cases, the fixes have no specific meaning and are purely an artifact of the used positioning technology, restrictions imposed by energy requirements, area coverage, and so forth. In other cases, the fixes denote important activities and decision points, and researchers may be interested in labeling and classifying them. We will refer to the latter cases as *semantic trajectories* [1]. An example of such semantic trajectories occurs in location-based social networks (LBSN), where the fixes are user check-ins to places and the labels are the names and types of these places [39,28]. The user's location between check-ins is unknown. The distinction between semantic trajectories and other fixes is not always crisp. For instance, the OCEARCH's Global Shark Tracker[1] can only record *pings* of tagged sharks if they surface for a certain amount of time. One could argue that these fixes do not carry any semantics and just reflect technological limitations of the used positioning technology. However, they reveal some important information, namely the event of surfacing and, thus, can be meaningfully labeled. Summing up, with the fast development of location-enabled mobile devices, it has become technically and economically feasible to record a large number of (semantic) trajectories generated by vehicles, animals, humans, and other moving objects (e.g., from the Internet of Things). While GPS has been widely used to detect the outdoor locations of moving objects, WiFi[11,10], RFID[31], and other sensor-tracking techniques have been employed to extend the geo-locating capability to indoor environments [20,32].

There are multiple ways to publish trajectory data in order to make it accessible to others. During the last few years, Linked Data [4] has become one of the methods of choice. It opens up data silos by providing globally unique identifiers for physical objects and information entities, links between them, and semantic annotations to foster discovery, retrieval, and integration. The semantic annotations are realized using shared vocabularies. In a highly heterogeneous and dynamic environment, such as the Web, arriving at commonly agreed and stable domain ontologies is a difficult task and progress has been slow over the last years. Foundational ontologies, such as DOLCE [16], have been usefully applied as a common ground for geo-ontologies [7]. In a Linked Data context, however, foundational ontologies tend to be too abstract and introduce a hardly comprehensible set of ontological commitments difficult to handle for laypersons. Ontology design patterns [14] have emerged as more flexible, reusable, manageable, and self-contained building blocks that help to model reoccurring tasks and provide common ground for more complex ontologies. To reach a higher degree of formalization and further improve interoperability, these patterns can be combined and ultimately aligned with foundational ontologies that act as glue between patterns. An increasing number of geo-ontology design patterns has been developed as joint community effort by domain experts and ontology engineers during so-called Geo-Vocabulary Camps (GeoVocamps) [9,8].

In this paper, we propose an ontology design pattern for semantic trajectories and demonstrate its applicability. While trajectory ontologies have been

[1] `http://sharks-ocearch.verite.com/`

developed before [37,34], they were confined to specific application areas and were not optimized for querying Linked Data, e.g., via the GeoSPARQL query language [2]. The proposed pattern is developed with two major goals. First, it should be directly applicable to a variety of trajectory datasets and, thus, reduce the initial hurdle for scientists to publish Linked Data [3]. Secondly, it should be easily extensible, e.g., by aligning to or matching with existing trajectory ontologies, foundational ontologies, or other domain specific vocabularies.

The remainder of this paper is structured as follows. First, we introduce some background materials and related work supporting the understanding of the proposed ontology design pattern. Section 3 introduces the conceptual foundation for the pattern. Next, in section 4, we discuss the formalization of the pattern using the Web Ontology Language (OWL). In section 5, we demonstrate how to annotate two trajectory datasets using the proposed pattern, in order to evaluate its applicability. We conclude by summarizing our results and pointing out directions for future work.

2 Background and Related Work

In this section we introduce related research and background materials relevant for the presented geo-ontology design pattern.

2.1 Semantic Trajectories

A trajectory consists of a series of spatiotemporal points generated by the moving object. These points are often represented as $\{x_i, y_i, t_i\}$ (with x_i, y_i denoting a position in the 2D geographic plane, and t_i representing a time point) or $\{x_i, y_i, z_i, t_i\}$ (with z_i denoting the elevation information) if the trajectory should be analyzed in a 3D space. While such spatiotemporal points support an exploration of the mobility pattern of a moving object [13], many applications require an understanding of additional information to interpret the trajectories. For example, a traffic analysis based on car trajectories may not be able to derive meaningful results without incorporating information about road networks. Similarly, studies on bird migration patterns may require an understanding of the features of the particular bird species (e.g., their body sizes, food sources, and competitors) as well as information about the weather conditions during their flight.

Semantic trajectories fill this gap by associating the spatiotemporal points and segments with geographic and domain knowledge, as well as other related information [5,1,33]. These semantically enriched trajectories facilitate the discovery of new knowledge, which otherwise may not be easily found. For example, human trajectories are best understood when the positional fixes can be labeled with activities performed at these places and the places are associated with semantic categories such as *restaurant* or *grocery store*.

2.2 Ontology Design Patterns

Ontology design patterns are derived from the common conceptual patterns that emerge in different domains when solving different tasks. A good example (given by Gangemi) is the *participation pattern*, which can be observed in enterprise models, software management, fishery, and many other domains [14,17,29,15]. Ontology design patterns capture the common conceptualization among knowledge engineers and domain experts, and can serve as building blocks or strategies for the design of future (more complex) ontologies. These flexible and self-contained building blocks appear to fit the needs to model knowledge for domain applications where more complicated and abstract ontologies may be difficult to apply. Lately, ontology design patterns have become popular in the geospatial semantics community [12,9,8]. A series of so-called Geo-Vocabulary Camps (GeoVoCamps[2]) have been held to promote the joint development, documentation, and testing of geo-patterns. These camps try to bring ontology engineers and domain experts together for two to three days to discuss and implement pattern ideas. Usually, patterns are conceptually developed during the camp and implemented and tested later. A follow-up camp (potentially with different participants and also embarking on new patterns) evaluates and refines the results. This paper is the result of such a community process.

There are two major types of ontology design patterns: logical patterns and content patterns, though other types have also been discussed in the literature [9,14]. Logical patterns deal with issues arising from the formal semantics of a knowledge representation languages, and therefore are independent from application domains. Content patterns often focus on domain knowledge and are used to model recurrent domain facts. The ontology design pattern proposed in this work is a content pattern addressing the design of classes and properties found commonly in semantic trajectories across application domains.

2.3 Semantic Trajectory Ontologies

As an ontology design pattern reflects a common conceptualization of domain experts with respect to a modeling problem, it is worthwhile to review the existing semantic trajectory ontologies in order to ensure consistency. A conceptual view on trajectories has been proposed by Spaccapietra et al. [34] who decompose a trajectory into a series of moves and stops. This stop-move conceptualization has been applied in several other trajectory studies, and the stops and moves are often coupled with corresponding geographic information to help interpret them [1,18,30]. Transportation networks are an important type of geographic information which is often utilized to make sense of the trajectories [35,27]. Other geo-data, such as those on Points Of Interest (POI), weather, land use, vegetation, and habitats, have also been employed to improve the understanding of trajectories [34,6,19,38]. While geographic information is the key contributor and commonality, domain knowledge is included in trajectories and their ontologies to help understand domain facts [37].

[2] `http://vocamp.org/wiki/Main_Page`

3 Conceptual Foundation and Motivation for the Pattern

Creating an ontology design pattern requires a generic use case (GUC), general enough to capture the recurring issues in cross-domain projects [14]. *Competency questions* are often utilized to discover and refine the GUC in a particular domain. A competency question is a typical query that a domain expert might want to submit to a knowledge base to complete a particular task [17]. A good ontology design pattern should define all and only the conceptualizations that are necessary to answer the competency questions shared by domain experts.

We conceptualize and motivate the trajectory design pattern using competency questions. For readability, we will use particular examples, e.g., related to animal movement, without restricting the pattern to those application areas. With the spatiotemporal information of the points contained in the trajectories, we can answer queries such as these:

Question 1. *"Show the birds which stop at x and y"*

Question 2. *"Show the birds which move at a ground speed of 0.4 m/s"*

While spatiotemporal points may only provide a basic level of movement understanding, geographic information, such as on points of interests, allows queries like:

Question 3. *"Show the trajectories which cross national parks"*

Domain knowledge is another important information source, enabling queries like:

Question 4. *"Show the trajectories of the birds which are less than one year old".*

In addition, information about the data creator (such as the location-tracking device) is necessary to answer related queries such as:

Question 5. *"Show the trajectories captured by Gamin GPS" or "show the trajectories generated by iPhone users".*

In order to answer these questions, an ontology design pattern needs to distinguish a number of relations. We introduce these abstract relations in Table 1, before formalizing them in Section 4. First, in order to query trajectories by spatiotemporal positions (Question 1), we need to segment the trajectory through fixes, which, in turn, require spatial and temporal reference systems (*hasSegment*, *startsFrom*, *endsAt*, *hasLocation*, *atTime*). Second, in order to answer questions about movement properties (Question 2), we need attributes for fixes and segments (*hasAttribute*). Third, in order to describe the geography of a trajectory (Question 3), it needs to be related to relevant geographic features. In the simplest case, this can be done by relating fixes to geographic features. Fourth, in order to identify and categorize a moving object (such as a bird, Question 4), we need to relate it to segments of the trajectory (*isTraversedBy*).

Table 1. Basic relations needed to answer the competency questions

Name	Type	Explanation
hasSegment	*SemanticTrajectory* $\times$ *Segment*	A segment of a trajectory
startsFrom	*Segment* $\times$ *Fix*	The from fix of a segment
endsAt	*Segment* $\times$ *Fix*	The to fix of a segment
isTraversedBy	*Segment* $\times$ *MovingObject*	A moving object traversing a segment
hasLocation	*Fix* $\times$ *Position*	The spatial position of a fix
atTime	*Fix* $\times$ *TemporalThing*	The temporal position of a fix
hasAttribute	*Segment* $\sqcup$ *Fix* $\times$ *Attribute*	An attribute of a segment or a fix
hasCreator	*Fix* $\times$ *Source*	The creator of a fix

Fifth, in order to query for properties of the trajectory data creator (Question 5), we need a relation between fixes and their source (*hasCreator*).

Figure 1 illustrates the creation of a semantic trajectory by integrating relevant knowledge with a person's daily trajectory. From top to bottom, the trajectory is enriched with a variety of data and acquires the capability to answer more advanced queries. This example demonstrates a general use case which is designed to support as many kinds of queries as possible. In a particular application, it may not be necessary to include some information (e.g., about the data creator). The pattern is designed such that information can be added at different levels and resolutions, e.g., by sub-typing. This idea will be discussed in more detail in the following section.

4 OWL Formalization

In this section, we present our geo-ontology design pattern, based on the previously described conceptual foundations. A schematic view of the pattern is shown in Figure 2. In the following paragraphs, we respectively discuss the classes and properties within the pattern and formally encode them using Web Ontology Language (OWL). We make use of Description Logics (DL) [22] notation, as we believe this improves the readability and understandability of the axioms presented. To encode our pattern, we make use of the logic fragment DLPE as defined in [8], which allows for tractable reasoning. Note that tractable reasoning is important for producing an efficient implementation of the pattern.

Fix. A fix is defined as a spatiotemporal point $\{x_i,y_i,t_i\}$ indicating the position of a moving object at an instant of time. It can be captured by a location measurement device (such as a GPS), but can also involve other types of points, such as check-ins on location-based social networks (LBSN) or centroids of regions passed by a moving object, but not automatically recorded by a device. Fixes are the atoms of the presented ontology design pattern: they not only capture the spatiotemporal information of a trajectory, but also link the

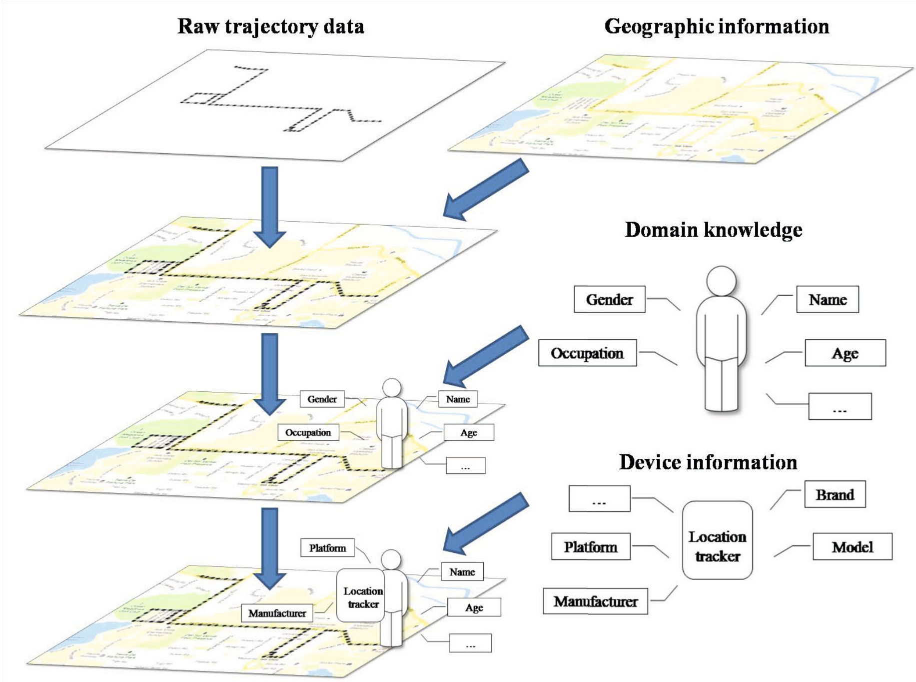

Fig. 1. An example semantic trajectory of an individual's daily activities

segments and provide information on attributes and metadata. By Axiom 1, a fix is enforced to have a timestamp and a position and to belong to a trajectory.

$$\begin{aligned} Fix \sqsubseteq\ & \exists atTime.OWL\text{-}Time{:}Temporal\ Thing \sqcap \exists hasLocation.Position \\ & \sqcap \exists hasFix^{-}.SemanticTrajectory \end{aligned} \tag{1}$$

The number of fixes in a trajectory depends on the requirements of the particular application. Resolution can be as coarse as containing only the important trajectory points (e.g., check-ins on LBSN), but can also be as fine as including points recorded according to a sampling rate of the location-tracking device. This scale-neutral design makes the pattern flexible, allowing users to model trajectories at different scales. Real-world examples of fixes include a stop of a migration flock in a wet land area, an intersection a vehicle has passed, or a restaurant visited.

Segment. A segment is defined by a starting fix $\{x_i,y_i,t_i\}$ and an ending fix $\{x_j,y_j,t_j\}$. $t_i < t_j$, $\{x_i,y_i\}$ is not necessarily different from $\{x_j,y_j\}$, as the moving object may stay at a the same position for a time period. An encoded formalization of a segment is given by Axioms 2–5. Axiom 2 enforces that every segment is connected to some fixes through the properties *startsFrom* and *endsAt*. Axioms 3 and 4 enforce that every segment is connected to at most two fixes, as

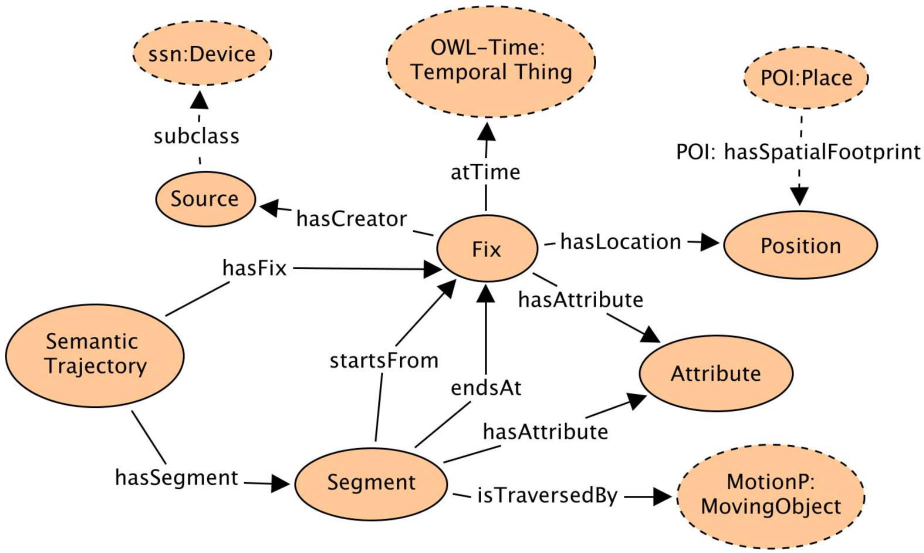

Fig. 2. Schematical description of the pattern

with these two axioms we declare both *startsFrom* and *endsAt* to be functional. Every segment is related to a trajectory as enforced by Axiom 5.

$$Segment \sqsubseteq \exists startsFrom.Fix \sqcap \exists endsAt.Fix \tag{2}$$

$$\top \sqsubseteq \leq 1 startsFrom.\top \tag{3}$$

$$\top \sqsubseteq \leq 1 endsAt.\top \tag{4}$$

$$Segment \sqsubseteq \exists hasSegment^{-}.SemanticTrajectory \tag{5}$$

As a segment is determined by its corresponding *startsFrom* and *endsAt* fixes, it inherits scalability from the fixes. A segment can, thus, be a route connecting two cities in a coarse-scale application, but also a line linking two spatiotemporal points on the same road in an application at a finer scale.

OWL-Time:Temporal Thing. We reuse *OWL-Time* to express the temporal information associated with a fix. As shown in Figure 2, the relation of *atTime* links a fix with an instance of the class *OWL-Time:Temporal Thing*. *OWL-Time* is part of the W3C Semantic Web Activity and has been used in many applications before, e.g., [36]. It can express rich temporal information using relations and classes such as *owl:before* and *owl:timeZone*. Embedding *OWL-Time* in the semantic-trajectory ontology design pattern not only captures the temporal relations among fixes, but also makes the pattern more reusable for those familiar with *OWL-Time*.

Position and Point-of-Interest (POI). As a fix is a spatiotemporal point, it contains a *position*. A *position* is defined as a coordinate tuple $\{x_i, y_i\}$ in a 2D plane, or $\{x_i,y_i,z_i\}$ in 3D space. It acts as an *interface* to integrate geographic information into the ontology design pattern. The concept of *interface* is well known from object-oriented programming language, as an enabler for a class to acquire additional functions. Existing POI ontologies can be integrated with the trajectory pattern to include geographic data. In Figure 2, we show an example of integrating the POI ontology developed in GeoVoCampSB2012[3] (the classes and relations from the POI ontology are shown by dotted lines). A *POI* can be any geographic feature that the user is interested in (e.g., a gas station or a tourist attraction), and can be represented by various vector geometries (e.g., polygons, polylines, and points).

Ordering Fixes within a Trajectory. We automatize the creation of properties *hasNext*, *hasSuccessor*, *hasPrevious*, and *hasPredecessor* making use of DL Axioms 6–10. These properties link fixes in the appropriate order within a given trajectory. The property *hasNext* is automatically instantiated between the two fixes related to the same segment by Axiom 6.[4] Note that the functionality of roles *startsFrom* and *endsAt* prevents the creation of incorrect instances of the *hasNext* property. With Axiom 7 we define *hasNext* to be a subrole of *hasSuccessor*, which is enforced to be transitive due to Axiom 8. Properties *hasPrevious* and *hasPredecesor* are defined to be inverses of *hasNext* and *hasSuccessor* with Axioms 9 and 10.

$$\mathit{startsFrom}^{-} \circ \mathit{endsAt} \sqsubseteq \mathit{hasNext} \tag{6}$$
$$\mathit{hasNext} \sqsubseteq \mathit{hasSuccessor} \tag{7}$$
$$\mathit{hasSuccessor} \circ \mathit{hasSuccessor} \sqsubseteq \mathit{hasSuccessor} \tag{8}$$
$$\mathit{hasNext}^{-} \sqsubseteq \mathit{hasPrevious} \tag{9}$$
$$\mathit{hasSuccessor}^{-} \sqsubseteq \mathit{hasPredecesor} \tag{10}$$

As previously stated, we can enforce an ordering among the fixes within a trajectory, something that may be useful to query data within an application. Also, we can easily verify that the time restrictions for a set of fixes are consistent with respect to the timestamps, now that these are related by the *hasNext* and *hasSuccessor* properties.

StartingFix, EndingFix, and Stop. The concepts of *StartingFix*, *EndingFix*, and *Stop* are important for the queries on trajectory data [37,34]. These concepts are not explicitly defined in the ontology design pattern; instead they are derived from the fixes and segments. The *StartingFix* has the earliest timestamp and links to only one segment through the property of $\mathit{startsFrom}^{-}$. Similarly the *EndingFix* is the one which has the latest timestamp and which links to only one

[3] `http://geog.ucsb.edu/~jano/POIpattern.eps`

[4] Axiom 6 is a role chain. Due to this axiom, *hasNext(a, b)* is entailed if $\mathit{startsFrom}^{-}$ *(a, c)* and *endsAt(c, b)* are the case for any individual *c*.

segment through to^{-}. A stop is a segment whose length (the Euclidean distance between the *startsFrom* fix and the *endsAt* fix) is shorter than a threshold defined by the user, and the time difference between the *startsFrom* fix and the *endsAt* fix indicates the duration of the stop.

We can automatically detect all fixes that are the start and end of a trajectory, but this comes at the price of loosing tractability. In any case, we include the necessary axioms to automatize this classification of fixes and leave it to the user to choose whether to utilize these axioms to the pattern. Fixes where a trajectory starts or ends are appropriately placed in classes *StartingFix* and *EndingFix* with Axioms 11 and 12. We also extend the starting and ending classification to segments and automatically classify these into *StartingSegment* and *EndingSegment* with Axioms 13 and 14.

$$Fix \sqcap \neg\exists endsAt.Segment \sqsubseteq StartingFix \tag{11}$$
$$Fix \sqcap \neg\exists startsFrom.Segment \sqsubseteq EndingFix \tag{12}$$
$$Segment \sqcap \exists startsFrom.StartingFix \sqsubseteq StartingSegment \tag{13}$$
$$Segment \sqcap \exists endsAt.EndingFix \sqsubseteq EndingSegment \tag{14}$$

Attribute and hasAttribute. The class *Attribute* and the corresponding relation *hasAttribute* have been defined as the generic class and relation to connect fixes and segments to their attribute values, such as the speed at a particular fix or the bearing of a segment. Users of the pattern can either remain on this level or define their own subclasses and subroles, e.g., hasSpeed.Speed, based on the requirements of the particular applications. This strategy is a well-established practice, and has been used in many applications and patterns [36]. Both are key for the development of a successful and reusable pattern, i.e., sub-typing them gives the pattern the required flexibility without introducing domain knowledge. *Attribute* and *hasAttribute* can also be used to store the pre-calculated spatial distance or time duration of a segment so that such values do not need to be dynamically calculated for each query.

Source. The *Source* class captures the knowledge about the device or the subject which has collected the trajectory data. Potential device information may include the device's manufacturer, produced year, accuracy in terms of location and time, product model, and so forth. Such information has important meaning since even for the same moving object in the same trajectory, different devices or subjects may generate different degrees of uncertain data. Similar to the *Position* class, this class also serves as an interface that allows the ontology design pattern to acquire additional information to support more complex queries. To give a concrete example, Figure 2 shows the integration of the W3C SSN-XG semantic sensor network ontology developed in [12].[5]

[5] The fact that the W3C SSN-XG ontology was developed around an ontology design pattern as its skeleton is further evidence for the effectiveness of patterns.

isTraversedBy. This relation links a *Segment* with the corresponding moving object. The *MotionP:MovingObject* class is borrowed from the *Motion Pattern* developed in a previous GeoVoCamp. It can be used as a hook for the integration of domain knowledge about the moving object, such as the name of a person, the species of a bird, the manufacturer of a car, and many other types of information that are necessary to answer the user's queries. Users can also utilize other ontologies, such as FOAF (which is used to model information about people and the relations with their friends) or the bird ontology ONKI[6], to capture related knowledge.

Semantic Trajectory. This class serves as the access point for the ontology design pattern. A semantic trajectory conveys fixes, segments, and related knowledge into a meaningful path connecting the origin and destination. We encode some features over individuals in class *SemanticTrajectory* with Axioms 15–17. Axiom 15 enforces that every trajectory is linked to at least one segment through the *hasSegment* property. Axioms 15 and 17 automatize the *hasFix* relationship from every trajectory to every related segment within this trajectory.

$$\mathit{SemanticTrajectory} \sqsubseteq \exists \mathit{hasSegment}.\mathit{Segment} \tag{15}$$

$$\mathit{hasSegment} \circ \mathit{startsFrom} \sqsubseteq \mathit{hasFix} \tag{16}$$

$$\mathit{hasSegment} \circ \mathit{endsAt} \sqsubseteq \mathit{hasFix} \tag{17}$$

Domain and Ranges and Class Disjointness. We declare all classes defined for the pattern to be disjoint (not shown here for lack of space and to improve readability). This is not only considered to be a good practice while modeling with OWL, as it allows for further inference, but also a necessary condition for the pattern to be expressed in DLPE.

We also recommend the definition of domains and ranges for existing classes, as these axioms are useful in order to complete missing information in some scenarios. We include Axioms 18–21 as an example, to show how to enforce some of these restrictions.

$$\exists \mathit{hasSegment}.\mathit{Segment} \sqsubseteq \mathit{SemanticTrajectory} \tag{18}$$

$$\exists \mathit{hasSegment}^{-}.\mathit{SemanticTrajectory} \sqsubseteq \mathit{Segment} \tag{19}$$

$$\exists \mathit{hasFix}.\mathit{Segment} \sqsubseteq \mathit{SemanticTrajectory} \tag{20}$$

$$\exists \mathit{hasFix}^{-}.\mathit{SemanticTrajectory} \sqsubseteq \mathit{Fix} \tag{21}$$

Note that we do **not** include strict domain and range declarations such as $\exists hasSegment.\top \sqsubseteq SemanticTrajectory$. Defining strict domain and ranges over the properties in an ontology have proven to reduce interoperability instead of

[6] `http://onki.fi/en/browser/overview/avio`

fostering it. Defining domains and ranges over existing classes is less intrusive, and we believe it will be more useful in practice. It is easy to see how these axioms enforcing domain and range could be extended to the rest of classes and relationships presented in Figure 2.

Since pair class disjointness is enforced across all classes presented in the ontology, the pattern satisfies all conditions and can be expressed within the DL fragment DLPE as described in [8]. As previously commented, this is only the case if we have that Axioms 11–14 are not part of the pattern. These axioms were depicted in this section as we believe they may become useful in some particular situation. Nonetheless, we do not recommend to include them a priori since the addition of these axioms makes the reasoning process exponential with respect to the ontology size.

In summary, the geo-ontology design pattern uses *fixes* and *segments* to capture the trajectory data, and defines a number of interfaces to integrate related geographic information, domain knowledge, and device data.

5 Applications to Trajectory Data

A successful ontology design pattern should have the usability that allows it to be applied to a wide range of datasets, solving problems of discovery and integration. It should not be too specific, nor introduce particular application perspectives. In this section, we use our semantic trajectory pattern to annotate datasets of two kinds: trajectories generated by human travelers and by animals. We also show how existing ontologies (such as a POI ontology) can be combined with the design pattern to capture related knowledge.

5.1 Modeling Human Trajectories

Human trajectories have been investigated by psychologists, anthropologists, geographers, and traffic planners to understand human behavior. In recent years, trajectory data from individuals have also been used to improve personal information management by providing information which is related to the user's current activities [24]. In the following paragraphs, we apply our ontology design pattern to an individual's trajectory data recorded by a handheld GPS receiver. During the trip, the user switched the transportation mode from walking to driving a car, so that the moving object changed between different segments of the trajectory. Graphic notations are employed to visualize the integration of our ontology design pattern with existing ontologies.

Figure 3 shows part of Mike's trajectory annotation, using the proposed design pattern, for his trip to the GeoVoCamp Dayton 2012, integrating location data, GPS positions, personal data, vehicle information, and so forth. We extracted two representative segments and four fixes from the entire dataset to illustrate the application. The two segments are *traversedBy* the person and his car respectively, and the information about the moving objects is included.

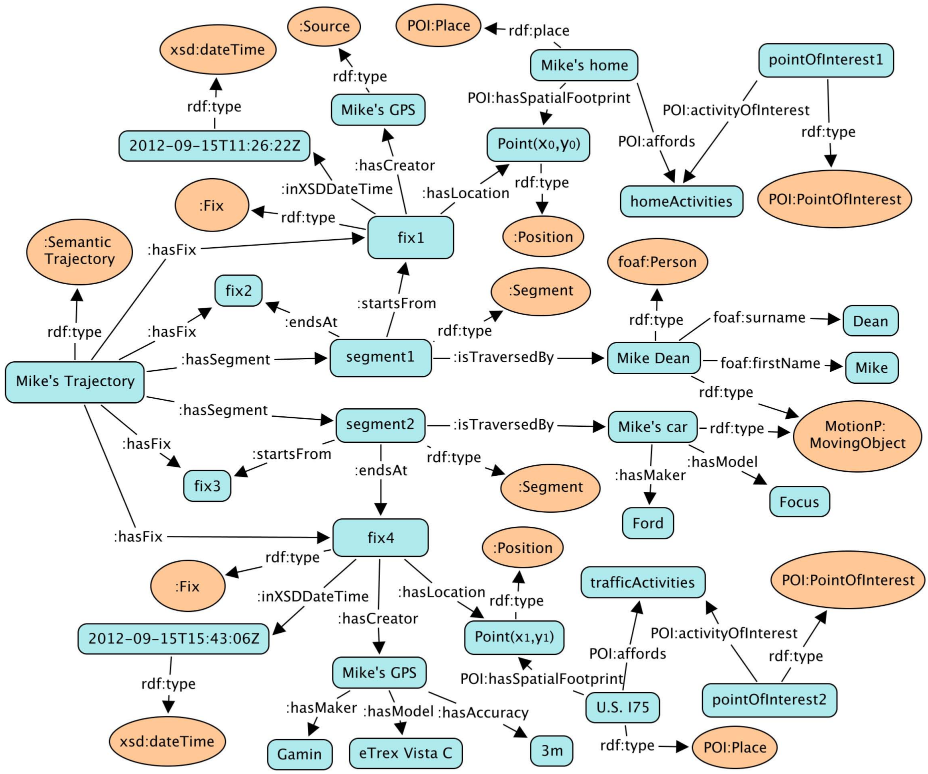

Fig. 3. Graphic notation for part of a person's trajectory annotation (blue rectangles represent entities and orange circles represent classes)

This integration enables queries such as "show the segments traversed by Mike without his car". While each segment has two fixes, we only display the relations of one fix in the figure, as the others have similar relations. A *fix* is linked to a timestamp represented by *xsd:dateTime* and a position[7]. To demonstrate the trajectory pattern's capability of integration with exiting ontologies, we combine it with the POI pattern developed in a previous GeoVoCamp through the POI interface. A *fix* is also linked with a timestamp, and a *source* (the Gamin GPS in this case). Information about the GPS device (such as the maker and mode) is also integrated into the example. We formally encode this ontology using the N3 syntax. Fragments of the code are shown in Table 2.

5.2 Application to Wildlife Monitoring

Here, we apply the design pattern to wildlife monitoring, using the trajectory data for a bird as example. The dataset is from the MoveBank, an online database

[7] For privacy considerations, we replace the real coordinates with x_i and y_i

Table 2. Part of the code for the individual trajectory using N3

:mikesTrajectory	a	:SemanticTrajectory;
	:hasSegment	:segment1, :segment2, ...;
	:hasFix	:fix1, :fix2, :fix3, :fix4, ...;
:mike	a	foaf:Person;
:mikesCar	a	MotionP:MovingObject;
:mikesGPS	a	:Source;
:mikesHome	a	POI:Place;
	POI:hasSpatialFootprint	:pos1;
:segment1	a	:Segment;
	:startsFrom	:fix1;
	:endsAt	:fix2;
	:isTraversedBy	:fordFocus;
:fix1	a	:Fix;
	:hasCreator	:mikesGPS;
	:inXSDDataTime	:2012-09-15T11:26:22Z;
	:hasLocation	:pos1;
:pos1	a	:Position;
	:geo:astWKT	Point(x_0,y_0);

providing animal track data openly to researchers[8]. The moving object is a toucan (a type of ramphastos sulfuratus), and the trajectory data contains the information about its positions, timestamps, temperatures of the environment, speeds, accuracy, directions, as well as some tracking device information, such as battery voltage [25,26]. Figure 4 shows the semantic annotation for part of the toucan's trajectory.

For reason of space limitation and readability, only one segment and two fixes are shown in the figure; the full data can be stored in any RDF triple store. The segment is *traversedBy* the toucan, and the application-specific information of this bird, such as its taxonomic name and local identifier, is also included in this example. Each *fix* has its corresponding time and position information. Unlike the human example, each *fix* of the bird's trajectory has several additional attributes, such as the temperature, its speed, and heading direction, which describe the status of the toucan and the environment at that *fix*. These attributes are expressed by sub-typing the *Attribute* class and the *hasAttribute* relation in the ontology design pattern. Such attributes enable queries such as "show the

[8] `https://www.movebank.org/`

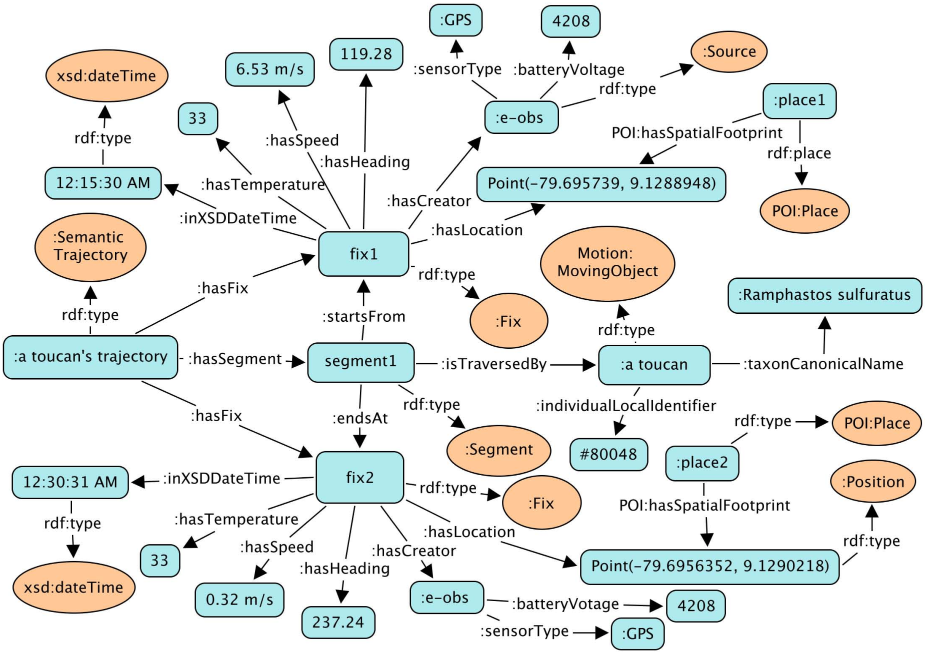

Fig. 4. Graphical notation for part of the annotation of the toucan's trajectory (blue rectangles for entities and orange circles for classes).

fixes where the toucan is moving at a speed higher than 6 m/s". The fixes are generated by the *e-obs*, a tracking device whose *sensorType* and *batteryVotage* are also captured by this trajectory ontology.

6 Conclusions

In this paper we presented a geo-ontology design pattern for semantic trajectories and demonstrated its applicability. The pattern resulted from a joint effort of domain experts and knowledge engineers. It can be used to semantically annotate trajectory data from a range of different domains such as navigation and wildlife monitoring. The major advantages of using the proposed pattern (also in comparison to existing work) are:

– **Expressiveness**. The design pattern can express a trajectory's spatiotemporal properties, geographic knowledge, domain knowledge, as well as relations among them. The pattern's formalization goes far beyond the typical surface semantics that reduces ontologies to mere subsumption hierarchies. Instead, it uses the expressive power of OWL to support a wide range of inferences.

This makes the pattern suitable for semantic annotation (of Linked Data), reasoning, and support for retrieval of scientific data (e.g., in semantics-enabled cyber-infrastructures).

- **Simplicity**. Only a minimal number of classes and relations are defined, which makes the design pattern easy to understand, reuse, and extend. The pattern can be used as a skeleton for more complex ontologies by sub-typing. This is in line with the approaches taken previously by other researchers, e.g. in the Simple Event Model (SEM) [21]. We do not restrict the domains and ranges of the used relations on a global level, in order to avoid unintended inferences. Misunderstandings of the formal semantics underlying such restrictions have been identified as a common source of errors (especially made by those new to ontology engineering).
- **Flexibility**. The provided interfaces (generic classes such as *Source*) allow the user to integrate related knowledge according to the specific needs of the application (users can also leave interfaces open and use the pattern directly without sub-typing). The pattern can model not only trajectories that have already been recorded, but also hypothesized or planned trajectories (e.g., in the context of navigation).
- **Scalability**. Depending on the required granularity of a particular application, the ontology design pattern can model trajectories at different scales. For instance, the physical movement path can be resolved to any degree based on the sample interval for fixes.

In a broader context, the pattern also contributes to a data-driven geo-ontology engineering paradigm. The success of a pattern and the methodology as such can only be evaluated over the years based on its usage *in the wild*. Nonetheless, the ability to develop such patterns (see also [9,8]) in a community process, agree on their ontological commitments, implement them in OWL, document them using real data from different domains, and publish the results, provides insights into the significant potential of the use of patterns, and their appeal to domain scientists, e.g., in a setting such as NSF's EarthCube [3].

While the presented work also demonstrates how to combine different patterns, so far they have mostly been developed independently and without an overarching structure. In fact, and in contrast to other domains, there is no common platform for geo-ontology design patterns, documentations, best practice, examples, and so forth that would significantly lower the initial hurdle for domain scientists interested in semantically annotating their data. This will be one of the main goals for the next years of geospatial semantics research. To address this challenge, we recently propose *Descartes-Core* as a community-wide collection of vocabularies, (geo-)ontology design patterns, best practice guides, examples, software, and services, with the aim to foster semantic interoperability between different sources without restricting semantic heterogeneity.

Finally, we plan to develop an optional alignment layer between the trajectory pattern and the DOLCE foundational ontology in a similar way as done for the W3C SSN-XG ontology before [12]. We expect that this will further foster interoperability and reuse of the pattern.

Acknowledgements. This work is a collaborative outcome of the GeoVoCampDayton2012[9]. Some of the authors credit funding from European Commission (ICT-FP7-249120 ENVISION project), as well as the German Research Foundation (Research fellowship grant DFG SCHE 1796/1-1). Authors from Wright State University acknowledge funding from by the National Science Foundation under award 1017225 III: Small: TROn – Tractable Reasoning with Ontologies. Mike Dean, Gary Berg-Cross, and Dave Kolas acknowledge funding from the NSF grant 0955816, INTEROP–Spatial Ontology Community of Practice.

References

1. Alvares, L.O., Bogorny, V., Kuijpers, B., De Macedo, J.A.F., Moelans, B., Vaisman, A.: A model for enriching trajectories with semantic geographical information. In: Samet, H., Shahabi, C., Schneider, M. (eds.) Proceedings of the 15th ACM International Symposium on Geographic Information Systems, ACM-GIS 2007, Seattle, Washington, USA, November 7-9. ACM Press (2007)
2. Battle, R., Kolas, D.: Enabling the geospatial Semantic Web with Parliament and GeoSPARQL. Semantic Web 3(4), 355–370 (2012)
3. Berg-Cross, G., Cruz, I., Dean, M., Finin, T., Gahegan, M., Hitzler, P., Hua, H., Janowicz, K., Li, N., Murphy, P., Nordgren, B., Obrst, L., Schildhauer, M., Sheth, A., Sinha, K., Thessen, A., Wiegand, N., Zaslavsky, I.: Semantics and Ontologies for EarthCube. In: Proceedings of the 2012 Workshop on GIScience in the Big Data Age, In Conjunction with the Seventh International Conference on Geographic Information Science 2012 (GIScience 2012), Columbus, Ohio, USA, September 18 (2012)
4. Bizer, C., Heath, T., Berners-Lee, T.: Linked Data – The Story So Far. International Journal on Semantic Web and Information Systems 5(3), 1–22 (2009)
5. Bogorny, V., Kuijpers, B., Alvares, L.O.: ST-DMQL: A semantic trajectory data mining query language. International Journal of Geographical Information Science 23(10), 1245–1276 (2009)
6. Brakatsoulas, S., Pfoser, D., Tryfona, N.: Modeling, storing, and mining moving object databases. In: 8th International Database Engineering and Applications Symposium (IDEAS 2004), Coimbra, Portugal, July 7-9, pp. 68–77. IEEE Computer Society (2004)
7. Brodaric, B., Probst, F.: Enabling cross-disciplinary e-science by integrating geoscience ontologies with Dolce. IEEE Intelligent Systems 24(1), 66–77 (2009)
8. Carral, D., Scheider, S., Janowicz, K., Vardeman, C., Krisnadhi, A.A., Hitzler, P.: An ontology design pattern for cartographic map scaling. In: Cimiano, P., Corcho, O., Presutti, V., Hollink, L., Rudolph, S. (eds.) ESWC 2013. LNCS, vol. 7882, pp. 76–93. Springer, Heidelberg (2013)
9. Carral, D., Janowicz, K., Hitzler, P.: A logical geo-ontology design pattern for quantifying over types. In: Cruz, I.F., Knoblock, C., Kröger, P., Tanin, E., Widmayer, P. (eds.) SIGSPATIAL 2012 International Conference on Advances in Geographic Information Systems (Formerly Known as GIS), SIGSPATIAL 2012, Redondo Beach, CA, USA, November 7-9, pp. 239–248. ACM (2012)

[9] `http://vocamp.org/wiki/GeoVoCampDayton2012`

10. Chan, L.-W., Chiang, J.-R., Chen, Y.-C., Ke, C.-N., Hsu, J., Chu, H.-H.: Collaborative localization: Enhancing WiFi-based position estimation with neighborhood links in clusters. In: Fishkin, K.P., Schiele, B., Nixon, P., Quigley, A. (eds.) PERVASIVE 2006. LNCS, vol. 3968, pp. 50–66. Springer, Heidelberg (2006)
11. Chiou, Y., Wang, C., Yeh, S., Su, M.: Design of an adaptive positioning system based on WiFi radio signals. Computer Communications 32(7), 1245–1254 (2009)
12. Compton, M., Barnaghi, P.M., Bermudez, L., Garcia-Castro, R., Corcho, Ó., Cox, S., Graybeal, J., Hauswirth, M., Henson, C.A., Herzog, A., Huang, V.A., Janowicz, K., Kelsey, W.D., Phuoc, D.L., Lefort, L., Leggieri, M., Neuhaus, H., Nikolov, A., Page, K.R., Passant, A., Sheth, A.P., Taylor, K.: The SSN ontology of the W3C semantic sensor network incubator group. Journal on Web Semantics 17, 25–32 (2012)
13. Dodge, S., Weibel, R., Lautenschütz, A.K.: Towards a taxonomy of movement patterns. Information Visualization 7(3), 240–252 (2008)
14. Gangemi, A.: Ontology design patterns for semantic web content. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 262–276. Springer, Heidelberg (2005)
15. Gangemi, A., Fisseha, F., Keizer, J., Lehmann, J., Liang, A., Pettman, I., Sini, M., Taconet, M.: A core ontology of fishery and its use in the fishery ontology service project. In: First International Workshop on Core Ontologies, EKAW Conference. CEUR-WS, vol. 118 (2004)
16. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.: Sweetening ontologies with DOLCE. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) EKAW 2002. LNCS (LNAI), vol. 2473, pp. 166–181. Springer, Heidelberg (2002)
17. Grüninger, M., Fox, M.S.: The role of competency questions in enterprise engineering. In: Proceedings of the IFIP WG5, vol. 7, pp. 212–221 (1994)
18. Güting, R.H., Böhlen, M.H., Erwig, M., Jensen, C.S., Lorentzos, N.A., Schneider, M., Vazirgiannis, M.: A foundation for representing and querying moving objects. ACM Transactions on Database Systems (TODS) 25(1), 1–42 (2000)
19. Güting, R., De Almeida, V., Ding, Z.: Modeling and querying moving objects in networks. The VLDB Journal 15(2), 165–190 (2006)
20. Gwon, Y., Jain, R., Kawahara, T.: Robust indoor location estimation of stationary and mobile users. In: Proceedings IEEE INFOCOM 2004, the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies, Hong Kong, China, March 7-11, pp. 1032–1043. IEEE (2004)
21. van Hage, W.R., Malaise, V., Segers, R.H., Hollink, L., Schreiber, G.: Design and use of the Simple Event Model (SEM). Journal on Web Semantics 9(2), 128–136 (2011)
22. Hitzler, P., Krötzsch, M., Rudolph, S.: Foundations of Semantic Web Technologies. CRC Press (2010)
23. Horne, J.S., Garton, E.O., Krone, S.M., Lewis, J.S.: Analyzing animal movements using Brownian bridges. Ecology 88, 2354–2363 (2007)
24. Hu, Y., Janowicz, K.: Improving personal information management by integrating activities in the physical world with the semantic desktop. In: Cruz, I.F., Knoblock, C., Kröger, P., Tanin, E., Widmayer, P. (eds.) SIGSPATIAL 2012 International Conference on Advances in Geographic Information Systems (formerly known as GIS), SIGSPATIAL 2012, Redondo Beach, CA, USA, November 7-9, pp. 578–581. ACM (2012)
25. Kays, R., Jansen, P.A., Knecht, E.M., Vohwinkel, R., Wikelski, M.: The effect of feeding time on dispersal of virola seeds by toucans determined from gps tracking and accelerometers. Acta Oecologica 37(6), 625–631 (2011)

26. Kays, R., Jansen, P.A., Knecht, E.M., Vohwinkel, R., Wikelski, M.: Data from: The effect of feeding time on dispersal of virola seeds by toucans determined from gps tracking and accelerometers. Movebank Data Repository (2012)
27. Li, X., Claramunt, C., Ray, C., Lin, H.: A semantic-based approach to the representation of network-constrained trajectory data. In: Riedl, A., Kainz, W., Elmes, G.A. (eds.) Progress in Spatial Data Handling, pp. 451–464. Springer (2006)
28. McKenzie, G., Adams, B., Janowicz, K.: A thematic approach to user similarity built on geosocial check-ins. In: Proceedings of the 2013 AGILE Conference (to appear, 2013)
29. Mika, P., Oberle, D., Gangemi, A., Sabou, M.: Foundations for service ontologies: aligning OWL-S to Dolce. In: Proceedings of the 13th World Wide Web Conference, pp. 563–572. ACM (2004)
30. Mouza, C., Rigaux, P.: Mobility patterns. GeoInformatica 9(4), 297–319 (2005)
31. Ni, L., Liu, Y., Lau, Y., Patil, A.: LANDMARC: indoor location sensing using active RFID. Wireless Networks 10(6), 701–710 (2004)
32. Priyantha, N.: The cricket indoor location system. Ph.D. thesis, Massachusetts Institute of Technology (2005)
33. Schmid, F., Richter, K.-F., Laube, P.: Semantic trajectory compression. In: Mamoulis, N., Seidl, T., Pedersen, T.B., Torp, K., Assent, I. (eds.) SSTD 2009. LNCS, vol. 5644, pp. 411–416. Springer, Heidelberg (2009)
34. Spaccapietra, S., Parent, C., Damiani, M., De Macedo, J., Porto, F., Vangenot, C.: A conceptual view on trajectories. Data and Knowledge Engineering 65(1), 126–146 (2008)
35. Vazirgiannis, M., Wolfson, O.: A spatiotemporal model and language for moving objects on road networks. In: Jensen, C.S., Schneider, M., Seeger, B., Tsotras, V.J. (eds.) SSTD 2001. LNCS, vol. 2121, pp. 20–35. Springer, Heidelberg (2001)
36. Willems, N., van Hage, W.R., de Vries, G., Janssens, J.H.M., Malaise, V.: An integrated approach for visual analysis of a multisource moving objects knowledge base. International Journal of Geographical Information Science 24(10), 1543–1558 (2010)
37. Yan, Z., Macedo, J., Parent, C., Spaccapietra, S.: Trajectory ontologies and queries. Transactions in GIS 12, 75–91 (2008)
38. Yan, Z.: Towards semantic trajectory data analysis: A conceptual and computational approach. In: Rigaux, P., Senellart, P. (eds.) Proceedings of the VLDB 2009 PhD Workshop. Co-located with the 35th International Conference on Very Large Data Bases (VLDB 2009), Lyon, France, August 24. VLDB Endowment (2009)
39. Ying, J.J.C., Lu, E.H.C., Lee, W.C., Weng, T.C., Tseng, V.S.: Mining user similarity from semantic trajectories. In: Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks, pp. 19–26. ACM (2010)