MASTER: A Multiple Aspects View on Trajectories

Discussion Paper

Ronaldo dos Santos Mello[1], Vania Bogorny[1], Luis Otavio Alvares[1],
Luiz Henrique Zambom Santana[1], Carlos Andres Ferrero[1,2],
Angelo Augusto Frozza[1,3], Geomar Andre Schreiner[1], and Chiara Renso[4]

[1] Universidade Federal de Santa Catarina - PPGCC, Florianopolis, Brazil
[2] Instituto Federal de Santa Catarina, Lages-SC, Brazil
[3] Instituto Federal Catarinense, Camboriu-SC, Brazil
[4] ISTI-CNR, Pisa, Italy

Abstract. For many years trajectory data have been treated as sequences of space-time points or stops and moves. However, with the explosion of the Internet of Things (IoT) and the flood of Big Data generated on the Internet, like weather channels and social network interactions, which can be used to enrich mobility data, trajectories become more and more complex, with multiple and heterogeneous data dimensions that can be integrated with trajectories. In this paper we introduce multiple aspect trajectories and we propose a robust conceptual and logical data model and a storage solution for efficient multiple aspect trajectory queries. The main strength of our data model is the combination of simplicity and expressive power to represent heterogeneous aspects, ranging from simple labels to complex objects. We evaluate the proposed model in a tourism scenario.

1 Introduction

With the explosion of the *Internet of Things (IoT)* and the flood of *Big Data* generated on the Internet, like weather channels and social network interactions (*e.g., Flickr, Facebook, Twitter, Foursquare*), it is now possible to collect huge volumes of movement data about people, animals, and objects like cars, buses, drones, etc. Sensors installed either indoor (*e.g., smart homes*) or outdoor allow the collection of data about the *place*, like temperature, air pollution, noise, luminosity, etc, or about the *object* that is moving around or inside this place, like the heart rate (with a smart watch), the emotional status (with a microphone that analyses the voice intonation), blood pressure, sleeping stages, etc. By collecting all these information we have a new type of movement data, *i.e.*, a trajectory enriched with different semantic aspects.

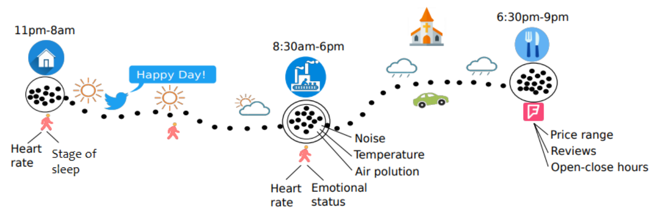

Fig. 1. An example of a multiple aspect trajectory

We can observe from Figure 1 that a trajectory became a complex object with numerous data dimensions that are contextual to the movement and heterogeneous in the form, which we define in this paper as *aspects*. The more aspects we have, the more complete is the representation of the real movement of an object, and more useful and interesting information we can infer about objects and places. The challenge is how to integrate all these heterogeneous information in a single trajectory representation, and the main questions we want to answer in this paper are: *(i) Is it possible to define a data model that is simple in structure, but generic enough to represent any aspect related to the movement, and covers a large number of applications? (ii) Is there a way to efficiently query and extract patterns from data represented in this model?*

We claim that multiple aspects represent a new view over trajectories, and a new paradigm concerning mobility data. These aspects are not only simple semantic labels, but may be complex objects and/or heterogeneous information intrinsically associated to the physical traces of the moving objects.

In this discussion paper we highlight the results of a published paper [4]. In that paper we introduced the concept of multiple aspect trajectory and propose a novel approach for modeling this kind of trajectories called *MASTER*. MASTER comprises a conceptual and a logical data model for multiple aspect trajectories, as well as a storage solution that is very appropriate for multiple aspect trajectory queries. In the full paper we also compared the MASTER model with a competitor DB called SECONDO[5]. Due to the lack of space, we omit here this comparison and we refer the reader to the full paper for details. The main novelties of MASTER compared to other state of the art works (detailed described in the full paper [4]) is that MASTER is more generic since we introduce the notion of "aspect" and how these aspects can enrich the location data. Other approaches do not support relationships between moving objects and do not propose a solutions for storing and querying huge volumes of trajectories and aspects or moving objects.

2 The MASTER Model

The main strength of our conceptual model is the combination of simplicity and expressive power for representing aspects. An aspect may be related to a moving object, to the entire trajectory or any trajectory point, and may hold any type of data, ranging from simple labels to complex objects.

For the logical model, we consider a graph-based representation (the RDF standard [6]) that is generic enough to model trajectories and aspects extracted from heterogeneous data sources, like geolocated structured record files and geolocated social media posts (*e.g.*, tweets). Finally,

we consider NoSQL databases for efficient storage and retrieval of large amounts of trajectory data. Our inspiration comes from the *polyglot persistence* approach [7], which states that a conceptual data model can be split and mapped to several database models for maximizing query performance.

We introduce a conceptual data model for multiple aspect trajectories, which is shown in Figure 2. We start the description of the model with the new concept of *aspect*.

An aspect is a real world fact that is relevant for trajectory data analysis, and it is characterized by an *aspect type*. For instance, the aspect *train* belongs to an aspect type *transportation mode*, and an aspect *rainy* belongs to an aspect type *weather condition*. An aspect type has a set of attributes and it may also be a subtype of a more general aspect type, allowing the modeling of an aspect type *subtypeOf* hierarchy, like *POI*←*accommodation*←*hotel*. More formally, an aspect type is defined as follows.

Definition 1. *An Aspect Type* asp_{type} = $(desc, ATT, asp_{supertype})$ *is a categorization of a real-world fact with a description desc, a set of attributes* $ATT = \{a_1, a_2, \ldots, a_z\}$ *that hold its properties, and a (possibly empty) supertype aspect* $asp_{supertype}$*.*

An aspect type and its attributes act as a metadata definition for an aspect. As a consequence, an aspect is always related to at least one aspect type and its attributes. For example, given an aspect type *weather condition*, some of its attributes could be *temperature*, *wind speed* and *climate*. In the following, we define an aspect.

Definition 2. *An Aspect* $asp = (desc, SAT)$ *is a relevant real-world fact, where desc is the aspect description, and* $SAT = \{at_1, at_2, ..., at_x\}$ *is a non-empty set of aspect types, with their attributes and respective values, which the aspect may hold, being* $at_i = (asp_{type\_k}, ATV_k)$, $at_i \in SAT$, *a tuple with an aspect type* asp_{type_k} *and a non-empty set* $ATV_k = \{a_1 : v_1, a_2 : v_2, \ldots, a_n : v_n\}$ *of attribute-value pairs so that each pair* $(a_i : v_i) \in ATV_k$ *is an instantiation of a property* a_i *of* asp_{type_k} *with a (atomic or multivalued) value* v_i*.*

An aspect definition supports numbers, ranges, text, geometries (when an aspect describes, for example, the shape of a hurricane at a specific time instant), or any type of complex object.

Definition 3. *A Semantic Meaning* $SM = (asp, asp_{type})$ *is an association between an aspect asp and an aspect type* asp_{type} *that gives the context of the aspect, so that* asp_{type} *belongs to the aspect types of the aspect asp.*

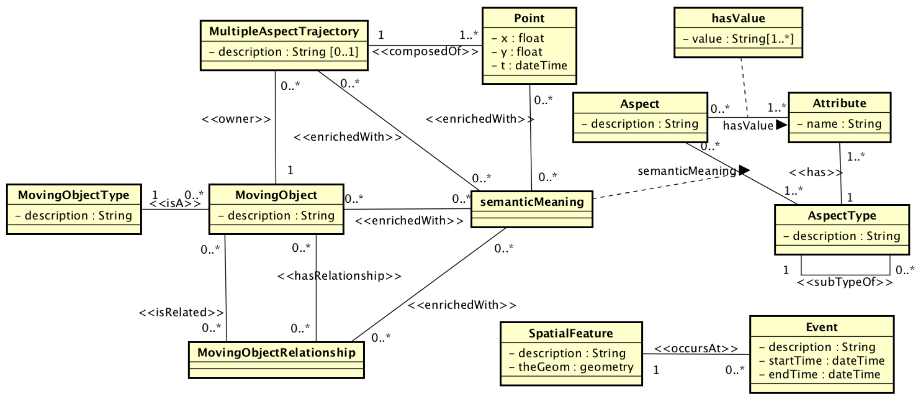

Fig. 2. The Conceptual Model for Multiple Aspect Trajectories

An aspect with a semantic meaning can be associated to a multiple aspect trajectory, a trajectory point, a moving object, or a relationship between moving objects in our conceptual model (see Figure 2). When an aspect varies frequently during the object movement, the aspect with its semantic meaning is associated to each trajectory point and it is called *volatile aspect (VA)*. Some examples are the *visited places* (or stops) and the heart rate. An aspect is also associated to a point when it represents a sparse and instant happening, like a social media post or check-in. When an aspect does not change during an entire trajectory, it is called a *long term aspect (LTA)* and is associated to the multiple aspect trajectory. Examples of this kind of aspect are the *town* on which the trajectory occurs or the person *occupation*. When an aspect holds during the entire life of an object, it is called a *permanent aspect (PA)* and is associated to the object and not to the trajectory. One example is the person *birthplace*. These aspect categories are directly related to the query performance. Queries on volatile aspects, *i.e.*, queries related to trajectory points, will be more time consuming, while long term and permanent aspects will be retrieved more quickly.

Based on these foundations, we now define a multiple aspect trajectory.

Definition 4. *A Multiple Aspect Trajectory mat = (P, S_LTA, mo, desc) is a sequence of points* $P = \langle p_1, p_2, \ldots, p_n \rangle$ *of a moving object mo, a (possible empty) set of long term aspects S_LTA, being* $S\_LTA = \{sm_1, sm_2, ..., sm_p\}$ *a set of semantic meanings, and a description desc, with* $p_i = (x_i, y_i, t_i, S\_VA)$*,* $p_i \in P$*, being x and y the spatial position of mo at the time instant t, and S_VA the set of volatile aspects related to* p_i*, where* $S\_VA = \{sm_1, sm_2, ..., sm_q\}$ *is a set of (possible empty) semantic meanings.*

A multiple aspect trajectory belongs to a moving object. A *moving object* is any entity that moves along space and time. This object is always associated to a type, which can be a person, a drone, an animal, a car, or even a natural phenomenon, like a hurricane. We formally define it in the following.

Definition 5. *A Moving Object mo = (mo_{type}, desc, S_PA) is an entity that can physically move in space and time, having a description desc, a set of (possible empty) permanent aspects S_PA, being S_PA* $= \{sm_1, sm_2, ..., sm_r\}$ *a set of semantic meanings, and a type* mo_{type} *that categorizes it.*

A new feature in MASTER when compared to the state-of-the-art data models for trajectories is the *moving object relationship*. A moving object may hold any type of relationship with other objects, and these relationships may also be characterized by different aspects such as the type of relationship (*e.g.*, *friendship*, *professional*, *family*). We define a moving object relationship in Definition 6.

Definition 6. *A Moving Object Relationship mor = (mo_1, mo_2, S_RA) is a relevant association between two moving objects* $mo1$ *and* $mo2$ *that holds a (possible empty) set of relationship aspects S_RA, being S_RA* $= \{sm_1, sm_2, ..., sm_s\}$ *a set of semantic meanings.*

Finally, we model *spatial features* and *events*. The first one denotes any relevant POI that is not spatially related to trajectory points, so it is not an aspect. Instead, it means any POI located in the trajectory neighborhood, like a nearby restaurant. In Figure 1, an example of spatial feature is the church located between the POIs *work* and *restaurant*. So, when a trajectory point intersects a relevant POI, it is modeled as an aspect. Otherwise, it is a spatial feature. Spatial features are useful for answering spatial queries like *which are the restaurants located at a distance less than α from the trajectory of object A?* or *which trajectories have an envelope whose area is higher than avenue B?* Similarly, an event denotes a happening that does not have a relationship with trajectories, but it is relevant for queries that investigate events in the trajectory neighborhood. An event occurs at a spatial feature and is valid for the period that it happened.

For defining the MASTER logical model we adopt the *Resource Description Framework (RDF)* [9] as our logical data model because RDF data can be modeled as a graph, which is a flexible data structure to represent the high heterogeneity of possible aspects, as well as the great number of aspect relationships with trajectories, points and moving objects. Besides, on using RDF we are consonant with the *Semantic Web* standards of *WWW Consortium (W3C)* for publishing and manipulating data on the Web [10].

Figure 3 shows the proposed logical model, where dotted arrows represent an entity-attribute relationship, continuous arrows represent relationships between entities, and the ellipsis represent entities or attributes. A predicate label followed by a cardinality pair denotes a multivalued relationship. One example is a *point* that may be enriched with zero to several *semantic meanings*. An RDF triple schema in such a modeling is represented by two ellipsis connected by an arrow. One example is a *moving object* (subject) that is the *owner* (predicate) of a *multiple aspect trajectory* (object).

The conversion of the conceptual model to a logical schema in RDF was inspired by several related approaches [2, 1, 3], which propose the following mapping rules:

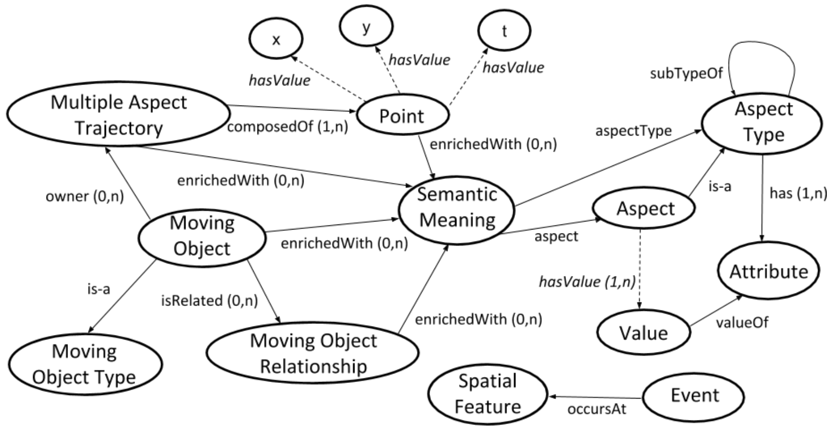

Fig. 3. The Logical Model for Multiple Aspect Trajectories

- an *entity* is converted to a *node*;
- an *attribute* of an entity (or relationship) er_m is converted to a *node* n_t, and an *edge* is defined from er_m to n_t in order to connect them;
- a *relationship* between entities is converted to an *edge* that connects the entities, or an *intermediate node* between the entities.

There is only one rule for the mapping of entities and attributes, so their conversion is straightforward. Even entities without attributes, like *MovingObjectRelationship* (see Figure 2), became nodes in the logical model because they have relationships with other entities.

We decided to consider the conversion to an edge if the relationship has no properties related to it, and the conversion to a node otherwise. On doing so, we avoid the generation of too many nodes in the RDF schema, and only the relationships that hold *semantic meaning* and *hasValue* as properties (see Figure 2) were mapped to nodes, and the second one was renamed to *Value* for sake of understanding. We also decide to maintain only the connections of *Aspect* and *Attribute* with the *Value* node to avoid a redundant edge between *Aspect* and *Attribute*.

The adopted storage solution for maintaining data represented in the MASTER logical model. This solution is called *Rendezvous* [8]. *Rendezvous* is a triplestore based on NoSQL databases for querying large RDF datasets. NoSQL databases have been proposed for managing big data efficiently [7]. Therefore, as multiple aspect trajectories are highly heterogeneous and multidimensional data, NoSQL databases are a suitable storage resource for this new type of trajectory. Compared to related work, *Rendezvous* was chosen due to its multi-

We define a label *hasValue* for this predicate to identify it as an entity-attribute or relationship-attribute connection.

model NoSQL support for storing RDF data and its efficient processing of typical SPARQL queries. *Rendezvous* manages RDF data in a distributed database.

3 Discussion and future works

The MASTER model has been evaluated over two perspectives: *(i)* a qualitative analysis at the conceptual level by modeling a tourism application, as well as an evaluation at the logical level to attest that an RDF-based storage strategy is suitable to answer the main types of multiple aspect trajectory queries; and *(ii)* a quantitative evaluation at the storage level by comparing the query running time performance of our storage solution with a baseline. With this experiment we show the feasibility of MASTER as a data model that can be efficiently stored and accessed. All details are presented in the full paper [4].

We claim that three main types of queries can be posed to multiple aspect trajectories: *(i)* queries that return *moving objects* (*e.g.*, *which are the moving objects that were born in Florianopolis and are male?*); *(ii)* queries that return *trajectories* (*e.g.*, *which are the trajectories that stayed at an accommodation place?*); and *(iii)* queries that return *aspects* (*e.g.*, *which accommodations were visited in Paris by persons that were born in Florianopolis and are male?*). These three queries are examples of *star-shaped*, *chain-shaped* and *complex* queries, respectively.

Table 1 shows these queries written in SPARQL or GeoSPARQL with an arbitrary complexity depending on the number of entities that must be considered to generate the query result, which allows an efficient processing by our RDF storage solution.

Future works include a performance evaluation over larger data sets of enriched trajectories, as well as the evaluation of other Big Data storage technologies, such as NewSQL databases, for maintaining multiple aspect trajectories. We also intend to extend MASTER to model data analytics information over multiple aspect trajectories, considering, for instance, dependencies among aspects.

Although out of the scope of this paper, it is also very important to consider privacy issues that these kinds of enriched trajectories might pose. When combining the different semantic aspects to the location information, privacy breach might happen. It is therefore crucial to develop privacy preserving methods to guarantee privacy when multiple aspects are involved.

Acknowledgments

This work was supported by the Brazilian agencies CAPES and CNPq, and the MASTER Project which received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklowdoska-Curie agreement N. 777695.

Table 1. Typical SPARQL Queries over Multiple Aspect Trajectories

SELECT mo? WHERE { mo? is-a 'Person' mo? enrichedWith sm? sm? aspect 'Florianopolis' sm? aspectType 'birthplace' mo? enrichedWith sm2? sm2? aspect 'Male' sm2? aspectType 'Gender'}	SELECT t? WHERE { t? composedOf p? p? enrichedWith sm? sm? aspectType at? at? subtypeOf 'Accommodation'}	SELECT a? WHERE { a? *rdfs:type* 'Aspect' a? is-a at? at? subtypeOf 'Accommodation' sm? aspectType a? p? enrichedWith sm? FILTER(geof:sfIntersects(?p, Paris)) t? composedOf p? mo? owner t? mo? is-a 'Person' mo? enrichedWith sm? sm? aspect 'Florianopolis' sm? aspectType 'birthplace' mo? enrichedWith sm2? sm2? aspect 'Male' sm2? aspectType 'Gender'}
(a) Star	(b) Chain	(c) Complex

References

1. S. Bagui and J. Bouressa. Mapping rdf and rdf-schema to the entity relationship model. *Journal of Emerging Trends in Computing and Information Sciences*, 5(12):953–961, 2014.
2. M. Choi, C. Moon, and D. Baik. Transformation of a relational database to RDF/RDFS with er^{2}idm. *IEICE Transactions*, 96-D(7):1478–1488, 2013.
3. G. Daniel, G. Sunyé, and J. Cabot. Umltographdb: Mapping conceptual schemas to graph databases. In *Conceptual Modeling - 35th International Conference, ER 2016, Gifu, Japan, November 14-17, 2016, Proceedings*, pages 430–444, 2016.
4. R. dos Santos Mello, V. Bogorny, L. O. Alvares, L. H. Z. Santana, C. A. Ferrero, A. A. Frozza, G. A. Schreiner, and C. Renso. MASTER: A Multiple Aspect View on Trajectories. *Trans. GIS*, 23(4):805–822, 2019.
5. R. H. Güting, V. T. de Almeida, D. Ansorge, T. Behr, Z. Ding, T. Höse, F. Hoffmann, M. Spiekermann, and U. Telle. SECONDO: an extensible DBMS platform for research prototyping and teaching. In *21st ICDE 2005*, pages 1115–1116, 2005.
6. J. Z. Pan. Resource description framework. In *Handbook on Ontologies*, pages 71–90. Springer, 2009.
7. P. J. Sadalage and M. Fowler. *NoSQL distilled : a brief guide to the emerging world of polyglot persistence*. Addison-Wesley, 2013.
8. L. H. Z. Santana and R. dos Santos Mello. Workload-aware rdf partitioning and sparql query caching for massive rdf graphs stored in nosql databases. In *Brazilian Symposium on Databases (SBBD), 2017*, pages 184–195. SBC, 2017.
9. W3C. RDF - Semantic Web Standards, 2018.
10. W3C. Semantic Web - W3C, 2018.