T-Warehousing for hazardous materials transportation

Azedine Boulmakoul[1], **Lamia Karim**[1], **Ahmed Lbath**[2]

1. Computer Science Department, Faculty of Sciences and Technology (FSTM), University Hassan II, Mohammedia, Morocco

azedine.boulmakoul@gmail.com
lkarim.lkarim@gmail.com

2. Computer Science Department, Laboratoire LIG, University Joseph Fourier Grenoble, France

ahmed.Lbath@ujf-grenoble.fr

ABSTRACT. *In recent years, a significant portion of material transported is harmful to human and environment. Thus, the transportation of hazardous materials (HazMat) and its potential consequences raise public interest typically when there is a release of hazardous materials due to an accident. In this paper, we introduce HazMat Trajectory Warehouse (T-Warehousing) that can be used for near real time decision making in different applications domain, using MongoDB as a NoSQL database for scalable, fault-tolerant and distributed space time paths big data storage and processing system. The system components are integrated into an interoperable software infrastructure respecting intelligent transport systems architecture. This infrastructure is distributed and based on a service-oriented architecture. It is also scalable by integration of MongoDB with Hadoop for large-scale distributed data processing.*

RÉSUMÉ. *Le transport des matières dangereuses et ses conséquences potentielles suscitent l'intérêt du public surtout quand il y a une libération de matières dangereuses due à un accident. Cet article traite l'entreposage des trajectoires et propose un modèle conceptuel de représentation de données trajectoires adaptable à plusieurs domaines d'application. Le domaine d'application concerné par cet article est celui du transport de matières dangereuses. Les composants système sont intégrés dans une infrastructure interopérable en respectant l'architecture des systèmes de transport intelligents. Cette infrastructure est distribuée et basée sur une architecture orientée services. Elle est également évolutive par l'intégration de MongoDB avec Hadoop pour le traitement de données distribuées à grande échelle.*

KEYWORDS: *T-Warehousing, HazMat transport trajectories.*

MOTS-CLÉS : *entreposage des trajectoires, transport de matière dangereuse.*

DOI:10.3166/ISI.21.1.39-59 © 2016 Lavoisier

1. Introduction

Dangerous goods or hazardous materials (HazMats) include explosives, toxics gases, flammable liquid and solids, oxidizing substances, and hazardous wastes. HazMat events and shipping data are needed in different location services to take near real time decisions like determining a route which minimizes the likelihood that the risk will be greater than a set threshold. Because hazmat accidents are generally being regarded as low probability and high consequence events. This kind of accidents/incidents attracts public attention as the damage to human being's health, deaths, economic and environment losses are high. Nowadays, it becomes more and more important to combine different applications fields with spatial and time related information. "80% of All Information is Geospatially Referenced" (Fitzke *et al.*, 2010) and valuable real-time geo-tagged data are produced by indoor and outdoor location sensing. The analysis of such spatio-temporal big data raises opportunities for many innovative applications and has multiple challenges as short latency, scalability, performance, query processing, high-precision positioning, and privacy preservation.

The number of Location Based Services is growing which calls for a near real time hazardous space time path data warehouse to analyze and make decision from spatio-temporal captured data. In addition, GeoStream data grow so large that it becomes difficult to capture, store, manage, share, analyze and visualize using classical models and databases. Also, visualizing the implicit information of hazardous transportations trajectories data is very important for analyzing human activities and is of great value in the decision making process.

Our objective is to provide a hazmat space time path data warehouse that can be used for near real time decision making in different applications domain. It is based on the unified trajectories meta-model to be very adaptive to various locations based services.

The remainder of the paper is organized as follows: Section 2 will present related work on trajectories data warehouses. Section 3 presents trajectories and their construction methods. Section 4 provides an overview of space time path presentations and presents the proposed HazMat space time path warehousing conceptual schema. Section 5 presents the proposed system for hazmat space time path data warehouse. Finally, Section 5 will provide conclusions of our proposed data warehouse.

2. Related work

Data warehouses were developed for decision support. They include information from various transactional systems of the company. Data warehouses have emerged around 1990 in response to the need to gather all company's information in a unique database for analysts and managers (Doucet *et al.*, 2001). All data, including their history, are used in many fields, such as: data analysis, decision support and in other applications (Benitez *et al.,* 2001). Spatial data warehouses are based on the concepts of data warehouses and additionally provide support to store, index, aggregate, and analyze spatial data (MacEachren, 2001). The research in this field mostly focuses on conceptual models for spatial data warehouses and SOLAP as a client application on spatial data warehouses (Fubédard *et al.,* 2001). Spatial Eye is an example of spatial data warehouse. Spatio-temporal data warehouses (Elzbieta *et al.,* 2008) complete the spatial data warehouse by including both spatial and time components as there is a need to include the temporal aspects as well. The GeoPKDD trajectory data warehouses (Damiani *et al.,* 2007) aim at extracting user-consumable forms of knowledge from large amounts of raw spatio-temporal geographic data. (Salvatore *et al.,* 2007) discussed the problem of storing and aggregating in a trajectories data warehouse, and they contributed a novel way to compute an approximated presence aggregate function, which algebraically combines a bounded amount of measures stored in the base cells of the data cube.

The Unified Moving Object Trajectories' Meta-model (Boulmakoul *et al.,* 2012) describes a general meta-model that could be used by different application domains; it can also use an object approach and integrates previous trajectories models described in (Güting *et al.,* 200; Meng *et al*., 2003; Wolfson *et al*., 1998;Yan *et al*., 2010). Using the space-time event ontology, the meta-model models space according to OGC Spatial Data Model (OGC, 2008), Observation domain of trajectory, according to OGC Sensor Meta Model and OGC Feature Type, physical and virtual activities between the beginning and the end of space time path (Shaw, 2011), sensors used for collecting moving object's traces, and movement patterns using composite region of interest. Simone *et al.* (Simone *et al.,* 2011) provided a solution named St-Toolkit for designing and implementing trajectory data warehouse based on semantic trajectory model, introduced by (Spaccapietra *et al*., 2008), in relational environment.

In (Salvatore *et al.,* 2007), authors discussed the design of a Data Warehouse and how to compute an approximate aggregate function, which algebraically combines a bounded amount of measures stored in the base cells of the data cube. Authors in (Leonardi *et al*., 2014) present a formal framework for modeling a trajectory data warehouse (TDW), it allows to navigate the aggregate measures obtained from OLAP queries. In (Leonardi *et al.,* 2010), they transform the traditional data cube model into a trajectory warehouse. However, there are several limitations in current data warehouse tools to cope with spatio-temporal data as found in the literature (Simone *et al.,* 2011; Güting *et al.,* 2004 and Vaisman *et al*., 2009). Data warehouses provide a decision support system for large stores of historical data, but are not yet adequate to support the hazmat space time path data warehouse from different facets: raw, structured, semantic, based on region of interest, activities and also none of them are interesting about accident / Incident of hazardous material and and none of them uses NoSQL database to store different kinds of trajectories data warehouses.

3. Trajectories construction

A trajectory is a description of the evolution over time of the physical movement of a moving object. In the following basic presentations of trajectories:

(a) Raw trajectory is the recording of the positions of an object in a specific field of space and time. For a moving object and a given time interval, it is presented as a geometrical locus sequence in the 2D space system (x_i, y_i, t_i).

(b) Structured Trajectory (Spaccapietra *et al*., 2008) is defined as a structured gross trajectory segments corresponding to significant steps in the trace of the path (travel).

(c) Semantic trajectory (Spaccapietra *et al.*, 2008) has a semantic related to the field of applications, it uses the four components (stop, move, begin and end). Stop (S), move (M), beginning (B) and end (E) are not space-time positions, but rather the semantic objects related to general geographical knowledge and the application geographic data.

(d) Another approach describes the movement patterns in space and time contexts based on the concept of region of interest (Giannotti *et al*., 2007) by defining the concept of spatial neighborhood and temporal tolerance.

(e) Yu *et al.* (2007) extended the concept of space-time path to represent both the physical activities (walking, driving, etc.) and virtual (sending e-mail, phone call). As every activity has a geographic location and a time interval, the space-time path has been profiled as a container of all activities that occur by a moving object.

The first step before path storage of vehicles transporting HazMat is the construction of trajectories according to different presentation models. Different techniques have been used to build the raw trajectories, structured, semantic, based on regions of interests and space-time paths (Boulmakoul *et al.*, 2014). The transition state diagram in figure 1 summarizes the different states of vehicles transporting HazMat trajectories.

The space-time coordinates collected from positioning devices may contain imperfections such as double points or consecutive points with the spatial shift and / or time exceeds the set threshold.

To clean raw data, we have several techniques, the first approach uses small squares approximations which aims to reduce the overall impact of errors, the second category is based on a smoothing method based on the core, this approach is based on the idea of smoothing the nearest neighbor and locally weighted regression models.

As regards the raw trajectories of moving objects which undergo network constraints, we use the map-matching cleaning technique (Newson *et al*., 2009) that uses Hidden Markov Model (HMM) to find the network corresponding to a sequence of positions.

Construction of structured trajectories requires segmentation of raw trajectory into structured trajectory, by identifying stops. A stop is determined by a movement speed that tends to 0 or less than a configured threshold, if the time interval of the stop exceeds the set threshold, it is considered the end of the trajectory, the corresponding spatio-temporal points at changes of position between two stops, between the start of the path and the first stop or between the last stop and the end of the trajectory correspond to Move points (M).

The construction of semantic paths is made by enriching the four components start, end and consecutive and alternating sequence of movements and stops of the structured trajectory using the application context data to make more sense when analyzing the trajectories' warehouse.

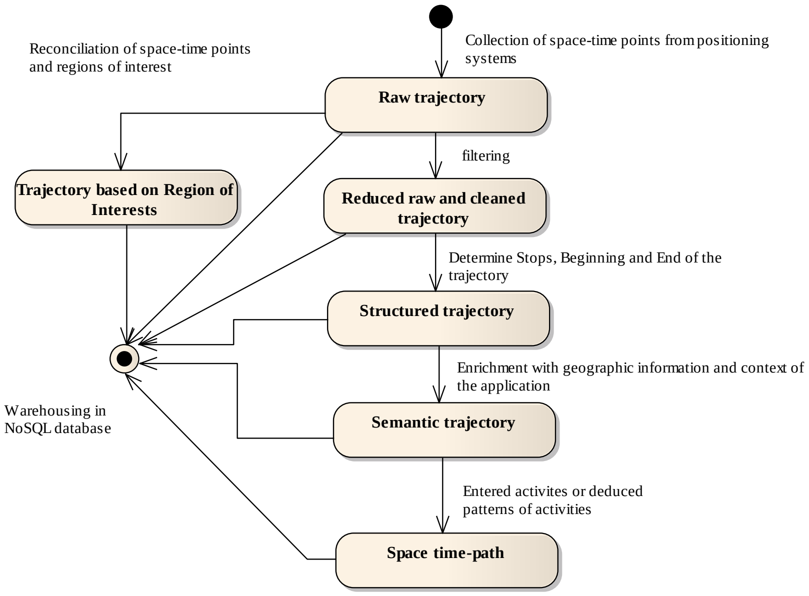
Collection of space-time points from positioning systems
Reconciliation of space-time points and regions of interest
Raw trajectory
filtering
Trajectory based on Region of Interests
Reduced raw and cleaned trajectory
Determine Stops, Beginning and End of the trajectory
Structured trajectory
Enrichment with geographic information and context of the application
Warehousing in NoSQL database
Semantic trajectory
Entered activites or deduced patterns of activities
Space time-path

Figure 1. Transition state diagram of the trajectory object

The construction of trajectories based on regions of interests is made using a rapprochement of raw or structured trajectories with a correspondence table that contains geometric zones and names of regions at a given period.

The construction of space-time paths is made by enriching the four components of the semantic trajectory by moving object activities'.

4. HazMat T-Warehousing

A hazardous material (HazMat) is a substance which, by its physico-chemical characteristics, toxicological, or by the nature of the reactions that may occur. If any leading to a Hazardous Material release, there is risk to humans, property and/or the environment. The risk of transporting hazardous material is the result of a transport accident or incident on health and environment. Transportation accidents of dangerous goods take place in few minutes in an unpredictable place. Given that consequences of an incident are often considerable, location based services purpose is to help against the immediate consequences of the disaster and the random nature of first aid. Several steps must be taken in near real time to avoid the damage. Aside, restrictive regulations based the training of staff (drivers) application of strict driving and traffic rules, an obligation tanks approval for vehicles signaling and hazardous products transported.

4.1. HazMat Transport System modeling

4.1.1. HazMat Transport conceptual class

Several research efforts have been carried out on management of trajectories. Some include modeling and representing trajectories, raw trajectory is the recording of the positions of an object at specific space time domain (GeoStream data), for a given moving object (e.g. person, vehicle) and a given time interval, it is presented as a sequence of geometric position in 2D spatial system [(x_1, y_1, t_1), (x_2, y_2, t_2), . . . (x_m, y_m, t_m)] representing the movement as a sequence of positions at time t1, t2, . . . tm. Structured trajectory (Spaccapietra *et al*., 2008), defined as a raw trajectories structured into segments corresponding to meaningful steps in the trajectory trace (e.g. travel). And in (Spaccapietra *et al.*, 2008) provides a semantic view of trajectory, which enables applications to associate whatever semantics they want with trajectories. However, this approach is only applicable to transactional schema. Indeed, no work has been published using trajectories as semantic objects with activities on multidimensional data modeling. Other recent approach describes trajectories in both spatial and temporal contexts based on Region of Interest (Giannotti *et al.,* 2007) by defining spatial neighborhood and temporal tolerance. The "aquarium" (Hongbo *et al*., 2007) of the relevant time-space unit describes anything having spatial and temporal extent as paths (for instance, people, plants, animal).

Analyzing features of vehicles trajectory's analysis system carrying dangerous materials, has allowed us to bring out all of its business entities. Their modeling is illustrated by the class diagram shown in Figure 2. In fact, we relied on the instantiation of the unified trajectories meta-model for a complete design and analysis (Boulmakoul *et al.,* 2012).

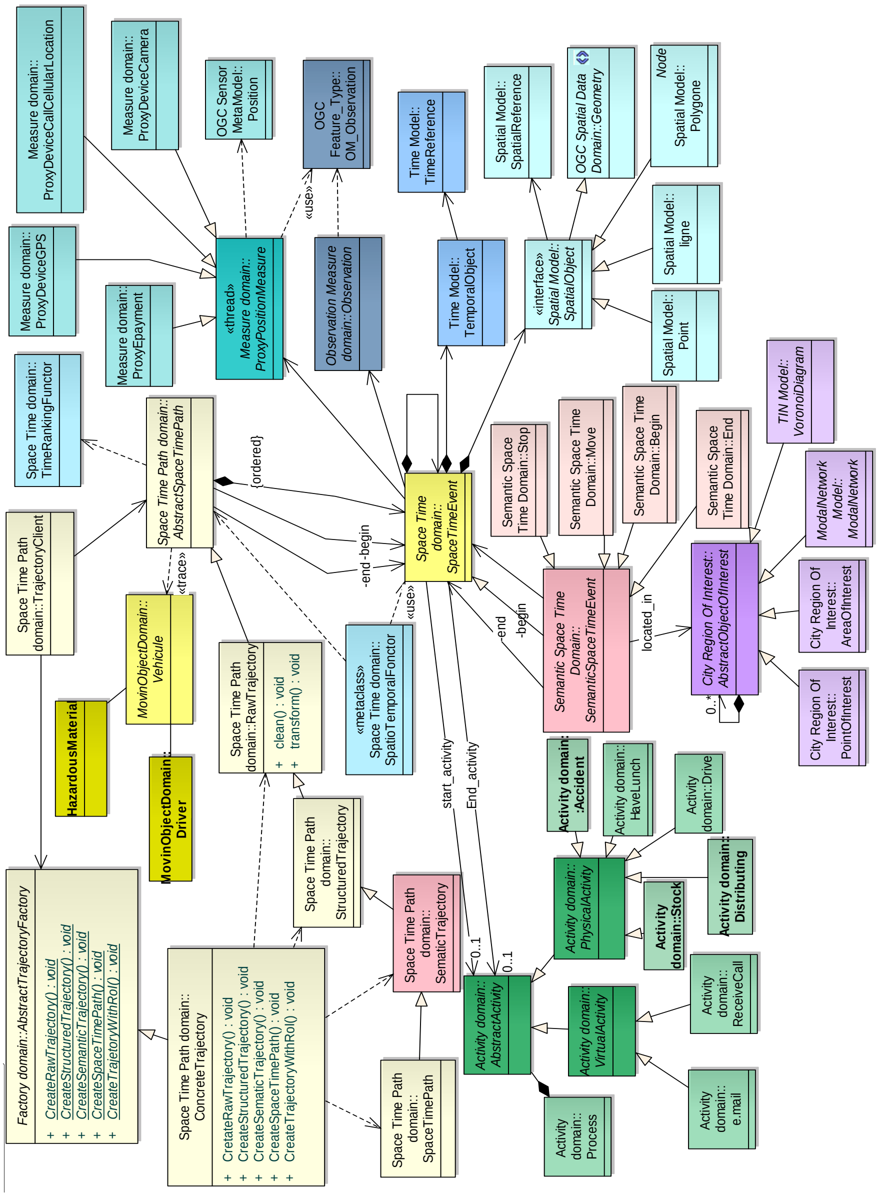
Measure domain::ProxyDeviceCallCellularLocation
Measure domain::ProxyDeviceCamera
OGC Sensor MetaModel::Position
OGC Feature_Type::OM_Observation
Time Model::TimeReference
Spatial Model::SpatialReference
OGC Spatial Data Domain::Geometry
Node Spatial Model::Polygone
Measure domain::ProxyDeviceGPS
«thread» Measure domain::ProxyPositionMeasure
«use»
Observation Measure domain::Observation
Time Model::TemporalObject
«interface» Spatial Model::SpatialObject
Spatial Model::ligne
Spatial Model::Point
Measure domain::ProxyEpayment
Space Time domain::TimeRankingFunctor
Space Time Path domain::AbstractSpaceTimePath
{ordered}
TIN Model::VoronoiDiagram
Semantic Space Time Domain::Stop
Semantic Space Time Domain::Move
Semantic Space Time Domain::Begin
Semantic Space Time Domain::End
ModalNetwork Model::ModalNetwork
Space Time domain::SpaceTimeEvent
-end -begin
Space Time Path domain::TrajectoryClient
«trace»
MovinObjectDomain::Vehicule
«use»
end
-begin
Semantic Space Time Domain::SemanticSpaceTimeEvent
located_in
City Region Of Interest::AbstractObjectOfInterest
City Region Of Interest::AreaOfInterest
City Region Of Interest::PointOfInterest
0..*
Space Time Path domain::RawTrajectory
+ clean() : void
+ transform() : void
«metaclass» Space Time domain::SpatioTemporalFonctor
HazardousMateria
MovinObjectDomain::Driver
start_activity
End_activity
Activity domain::Accident
Activity domain::HaveLunch
Activity domain::Drive
Activity domain::Distributing
Space Time Path domain::StructuredTrajectory
Activity domain::PhysicalActivity
Activity domain::Stock
Factory domain::AbstractTrajectoryFactory
+ CreateRawTrajectory() : void
+ CreateStructuredTrajectory() : void
+ CreateSemanticTrajectory() : void
+ CreateSpaceTimePath() : void
+ CreateTrajetoryWithRoI() : void
Space Time Path domain::ConcreteTrajectory
+ CreateRawTrajectory() : void
+ CreateStructuredTrajectory() : void
+ CreateSematicTrajectory() : void
+ CreateSpaceTimePath() : void
+ CreateTrajectoryWithRoI() : void
Space Time Path domain::SematicTrajectory
0..1
0..1
Activity domain::AbstractActivity
Activity domain::VirtualActivity
Activity domain::ReceiveCall
Space Time Path domain::SpaceTimePath
Activity domain::Process
Activity domain::e.mail

Figure 2. Class diagram of vehicles trajectory's analysis system carrying dangerous materials

4.1.2. HazMat Space Time Path Data warehousing conceptual schema

We propose in the following the use of a data warehouse in near real time to help in case of an eventual risk, by sending alerts to people near to the disaster geographical location, to prohibit some roads and find backup routes.

Multidimensional modeling is the foundation of data warehouses (Song *et al.*, 2001). It is characterized by two primitives which are Facts and Dimensions. Those latter are used to construct the star schema (Freitas *et al.*, 2002), the snowflake schema (Levene *et al.*, 2003) or the constellation schema (Teste, 2001). In the proposed schema, Figure 3, we present the proposed conceptual HazMat space time path warehousing using composite document notation to explicitly show the hierarchical structure of each dimension rather than appearing as an unstructured collection of data items. HazMat Space Time Path Data Warehouse sources and requirements are gathered from different spatio-temporal sensors and also location based service that store accidents and incidents.

In fact, our model is based on the unified trajectories Meta model (Boulmakoul *et al.*, 2012) for designing and implementing hazmat space time path data warehouse. We use the snowflake schema as a multidimensional model. Thanks to redundancy, we can provide horizontally-scalable systems as distribution of data across multiple machines is easy and does not cause problems. Space time path measures are given below.

Hazmat Space Time Path fact table represents the subject orientation and the focus of analysis. It typically contains measures that are attributes representing the specific elements of analysis. A dimension contains attributes that allow exploring measures from different perspectives. Hazmat Space Time Path data warehouse measures are:

– Average_duration: corresponds to the duration of hazardous space time path.

– Total hazmat accident: corresponds to total number of hazardous accidents.

– Total transported hazmat: corresponds to total quantity of hazardous materials transported.

– Frequent hazardous accident: corresponds to hazardous materials that are frequently transported in case of an accident.

– Frequent hazardous causes: corresponds to hazardous materials that are frequently transported in case of an accident.

– Frequent hazardous consequences: corresponds to frequent consequences of hazardous materials transported in case of an accident.

– Frequent accident region: corresponds to frequent regions where hazardous materials accidents occur.

– Frequent time period accident / incident: corresponds to frequent regions where hazardous materials accidents occur.

– Most safety region: corresponds to the most safety regions.

– Minimum_duration; corresponds to the minimum duration of the hazmat space time path.

– Maximum_duration: corresponds to the maximum duration of the hazmat space time path.

– Average_distance: corresponds to the average distance of the hazmat space time path.

– Average_speed: corresponds to the average speed of the hazmat space time path.

– Number_of_stops: contains number of stops in the hazmat space time path.

– Number_of_moves: contains number of moves in the hazmat space time path.

– Most_frequent_ROI: contains the most frequented region of interest in a hazmat space time path.

– Most_frequent_activity: contains the most frequented activity practiced in the hazmat space time path.

– Count_users: contains total number of moving objects taking the same hazmat space time path.

– Count trajectories: contains total number of trajectories of vehicles in a period.

– Shape: corresponds to the interpolated shape of the hazmat space time path.

In the proposed schema, we used hierarchy documents that contain several related levels. Principally, it is used for roll-up and drill-down operations. In the following, we describe dimensions of the proposed space time path conceptual schema:

Vehicle dimension: contains information about the tracked vehicle like reference number, type of vehicle, traveled kilometers, capacity, mobile sensor type and also different signalization. The vehicle can carry several hazardous materials.

Hazardous materials dimension is characterized by ONU number and quantity, and it belongs to a determined class (**class dimension**) according to The European Agreement concerning the International Carriage of Dangerous Goods by Road (ADR). The classification procedure of dangerous goods are based on structure of the list of substances, classes of hazardous goods, nature of transported hazardous goods, physico-chemical and toxicological properties of dangerous goods.

Type of hazmat packaging dimension contains identifiant, brand, volume, reusability, danger and handling labels characteristics for a full Hazardous space time paths analysis.

The vehicle dimension is related to the driver entity with his history of training.

Recipient dimension presents the company that supports the Hazardous materials on arrival.

Carrier dimension presents the company that transports Hazardous materials with or without transport contract.

Filler dimension presents the company that fills dangerous materials in the tracking vehicle.

Charger dimension presents the company that load packaged Hazardous materials in a vehicle.

Accident/Incident dimension contains the estimated amount of lost product, the average retention, the material retention means, and type of failure of retention means and description of the event to minimize damage and facilitate analysis of accidents/incidents during the transport of materials.

Specific weather conditions, Cause of the event, and Consequences of the event dimensions (Intervention of the authorities, estimated amount of material or environmental damage, Product loss, Bodily injury related to dangerous goods) allow a full knowledge in near real time decision when an accident/ incident occurs in a hazardous space time path.

Event ontology has already been proven useful in a wide range of contexts, due to its simplicity and usability. The SHOE General Ontology defines an event as something that happens at a given place and time. In Dublin Core metadata standard, an event is defined as « a non-persistent, time based occurrence ». (Quine, 1985) described events as objects where objects are regions bounded in space and time.

Space Time Event dimension presents an event as an occurrence that happens in a small space and lasts a short time. From spatial point of view, it is a composition of Spatial Object. Each spatial object is characterized by a spatial reference and geometry to model a raw trajectory. A spatial object could be a point, line or a polygon to present raw trajectories. Semantic trajectory (Spaccapietra *et al.*, 2008) expresses the application oriented meaning using four components (stop, move, begin and end).

Stop, move, begin and end are no more spatio-temporal position, but semantic objects linked to general geographic knowledge and application geographic data. From semantic point of view, a semantic object is characterized by a Toponyme, and linked to the semantic begin move end and stop which respectively contains semantic information, time of begin and of the end for a given begin, move stop and end. To analyze the space time path data warehouse, the schema considers the presentation of spatio-temporal data and activities for each event.

Activity dimension contains information about activity’s duration, time of begin and of end activity and also information about the kind of activity (Driving, Distributing, Stocking). As concerning the temporal aspects, it is detailed with others dimensions tables like Hour, Day, Month, Year and Time Period.

Thus, using the star schema combined with structuring dimension in several analysis perspectives, we provided a general HazMat space time path warehousing for different kinds of Space time paths.

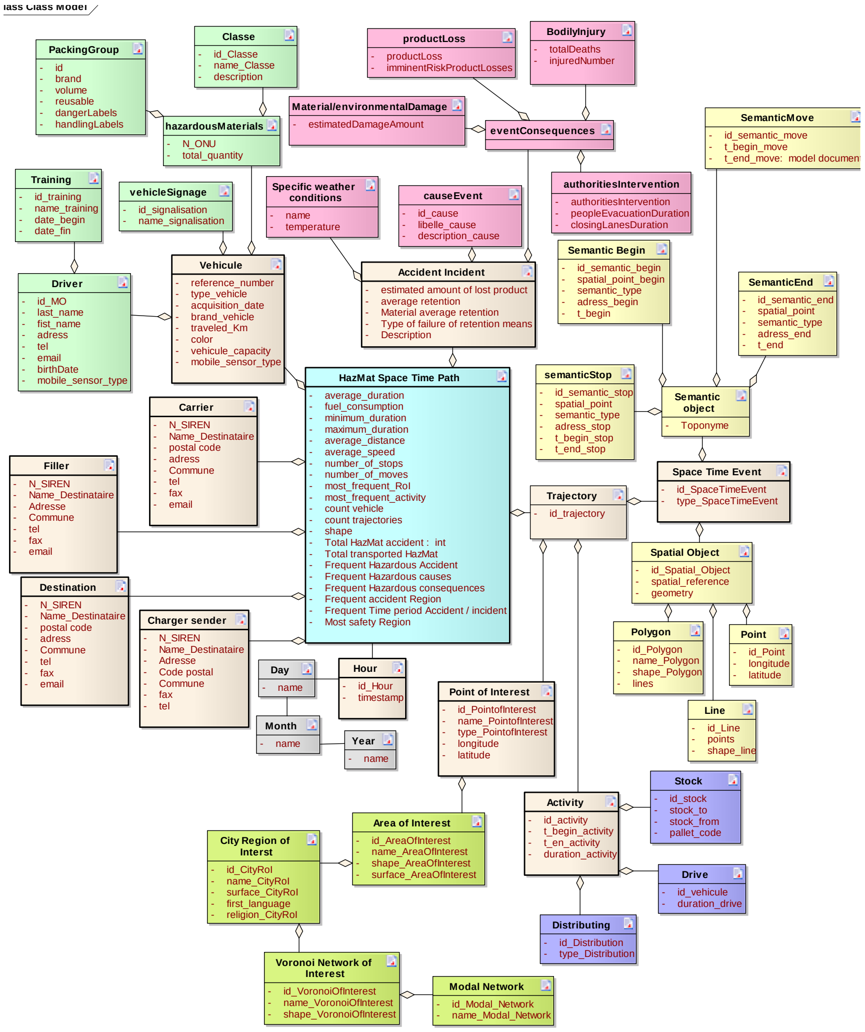

Figure 3. HazMat Space Time Path Data warehousing conceptual schema using composite documents

Additionally, hazmat space time path data warehouse schema presents hazmat trajectories in both spatial and temporal contexts based on Region of Interest and activities. The spatial neighborhood is presented using Point of Interest dimension characterized by longitude, latitude and name. A hierarchy of spatial neighborhood is used for roll-up and drill-down operations. Area of interests has a shape and a surface. City region of interests to present a city as a region of interests, it contains about the city region of interest name, surface, first language and religion. In other applications domain, region of interests could be presented as a Voronoi network of interest or a modal network.

The spatio-temporal data of trajectories are very large; we answer in the next paragraph on the question of storing the trajectories in traditional data warehouses or in big data type solutions.

4.2. Storing trajectories in Big Data solutions

The data collected by positioning devices are Big Data category that is characterized by five “V”:

– Volume: very large amounts of data (Po, Eo, Zo, Yo…);

– Variety: complex data, structured, unstructured...;

– Velocity: data streams (real-time flow rate…);

– Value: extraction of information from data;

– Veracity: precision, accuracy, reliability of data.

These data provide a treasure for location-based services, but the analytical work carried out on these data will have to meet many challenges.

Many challenges that location-based services are faced concern storing trajectories, data from multiple sources are processed in a database to analyse trajectories from different facets of moving objects to deliver for example, a daily report on the number of vehicles carrying hazardous material in a given region.

To deal with these problems, we propose to compare the storage of trajectories in traditional warehouses and Big Data.

Given the very large number of spatial and temporal data to be analyzed, the Big data has become a challenge for location-based services whereas traditional warehouse management tools are not enough.

Spatio-temporal data collected have a fine granularity. Generally, they come from different sources and different types of sensors (GPS, RFID, ATMs, WIFI, etc.), They have a heterogeneous format that requires treatment in the data warehouse, it is necessary to check the cost storage for the expected gain.

Once control of the cost of storage managed, we compare the functional constraints for location-based services in the above Big data and traditional (Relational) decision-making system.

The result of the comparison between storage in traditional and Big Data systems, we conclude that to meet the needs of location-based services, it is better to opt for a decision model in a Big Data platform (Boulmakoul *et al.,* 2013).

The system architecture of data warehouses is traditionally associated with the relational model as operational database and SQL as a query language. Whereas location based services requires more flexibility and agility in analyzing of hazmat space time path data warehouse information. In some key areas of business need and data processing, Document-oriented databases of the categories NoSQL (Not only SQL) offers an alternative powerful and extension to the traditional relational approach. The challenge of analyzing of the massive hazmat space time path data warehouse in near real time requires the use of a fast and scalable solution to bear the burden of voluminous data. MongoDB is chosen for its performances and scalability (Boulmakoul *et al*., 2013).

In this work, we provide a NoSQL database for the storage of hazmat space time path data warehouse. Other supports components are provided for collecting and visualizing of data and spatio-temporal events related to hazmat. All given components are integrated into an interoperable software infrastructure respecting intelligent transport systems architecture. This infrastructure is distributed and based on a service-oriented architecture. It is also scalable by integration of MongoDB with Hadoop for large-scale distributed data processing. In this work, we also give an assessment of the performance, scalability and fault-tolerance of using MongoDB with Hadoop, towards the goal of identifying the right architecture and software environment for HazMat spatio-temporal data analytics.

– NoSQL databases

The NoSQL databases break the limitations of the relational model in terms of scalability and volume. Indeed, a recurrent problem of relational database is the loss of performance when you need to process a large volume of data. In addition, the proliferation of distributed architectures has brought the need for adapting natively solution mechanisms of data replication and load management. The acronym NoSQL signifies “Not Only SQL” (Mike, 2012). It is designed for storing data in a much simpler, flatter, and non-relational manner that allows data repositories to be scaled up. In a NoSQL database, there is no fixed schema so we can store, in the same entity, heterogeneous spatio-temporal data and activities generated by different kinds of locations sensors. NoSQL is a class of database management systems (DBMS) that do not follow all of the rules of a relational DBMS and cannot use traditional SQL to query data. The term is somewhat misleading when interpreted as “No SQL”, and most translate it as “Not Only SQL”, as this type of database is not generally a replacement but, rather, a complementary addition to RDBMSs and SQL.

Relational database scales up by getting faster hardware and adding memories whereas NoSQL, on the other hand, can take advantage of scaling out by spreading the load over many commodity systems. Consequently, NoSQL is an inexpensive database for scaling trajectories space time path data. Companies like (Google, Facebook, Twitter, Amazon, Twitter, Adobe, Viadeo) have left the relational world and all use NoSQL in one way or another because they have seen their needs in terms of load and data volume grow exponentially. Existing NoSQL solutions can be grouped into 4 main families: Key-values Stores, Column Family Stores, Document Databases, and Graph Databases.

– MongoDB

MongoDB (from "humongous") is an open-source document database, and the leading NoSQL database. It is developed by 10gen in 2009 (MongoDB, 2013). It is written in C++, document-oriented storage, full Index, rich document-based queries, and flexible aggregation and data processing. MongoDB may contain several databases. Using JavaScript for its query language, MongoDB supports both single and complex queries. Storing the basis documents format of many modern geospatial applications JSON (JavaScript Object Notation documents) makes it easy to build on top of MongoDB. MongoDB database benefits from ascending, descending, unique and geospatial indexes. To make performance better, JSON is stored by MongoDB in an efficient binary format called BSON. BSON is a binary serialization of JSON documents and stands for Binary JSON. In general, document-oriented (e.g. MongoDB) are most directly relevant to business intelligent because of their more flexible and extensive search and retrieval functionality. To scale its performance on a cluster of servers, MongoDB uses a technique called sharding, which is the process of splitting the data evenly across the cluster to parallelize access.

This is implemented by breaking the MongoDB server into a set of front-end routing servers mongos), that route operations to a set of back-end data servers (mongod).

MongoDB queries examine one record at a time, which means that queries across multiple records must be implemented on the client or use MongoDB's built-in MapReduce (MR). Though MongoDB's MR can be executed in parallel at each shard, there are two major drawbacks: (1) the language for MR scripts is JavaScript, which is slow and has poor analytics libraries, and (2) the SpiderMonkey (Spider, 2013) Javascript implementation used by MongoDB, is not threadsafe, so only one MapReduce program can run at a time.

– Hadoop

Hadoop (Jason, 2009) is the Apache Software Foundation top-level project that provides both distributed storage and computational capabilities. The Hadoop project provides and supports the development of open source software that supplies a framework for the development of highly scalable distributed computing applications. The Hadoop framework handles the processing details, leaving developers free to focus on application logic.

The Hadoop Core project provides the basic services for building a cloud computing environment with commodity hardware, and the APIs for developing software that will run on that cloud. The two fundamental pieces of Hadoop Core are the MapReduce framework, the cloud computing environment, and the Hadoop Distributed File System (HDFS).

Hadoop has been designed to run on multiple servers simultaneously. In practice, the data is spread across different servers, and Hadoop manages a replication system so as to ensure a high availability of data, even when one or more servers are failing. The strength of Hadoop is to benefit from the computational power of multiple servers unmarked cluster. The parallelized processing is managed by MapReduce, whose mission is to distribute the treatments on different servers, and vice versa to aggregate the elementary results in an overall result.

The MapReduce (Alex, 2012) model simplifies parallel processing by abstracting away the complexities involved in working with distributed systems, such as computational parallelization, work distribution, and dealing with unreliable hardware and software. With this abstraction, MapReduce allows the programmer to focus on addressing business needs, rather than getting tangled up in distributed system complications. MapReduce decomposes work submitted by a client into small parallelized map and reduce workers. In traditional applications, the built-in aggregation functionality provided by MongoDB is sufficient for analyzing data. However, storing and analyzing the collected spatio-temporal data of trajectories need more complex data aggregation. This is the reason to use Hadoop as a powerful framework for complex analytics queries in our system architecture.

There are other NoSQL databases that provide Hadoop support. Cassandra is a peer to peer key-value store that has the ability to replace Hadoop's HDFS storage layer with Cassandra (CassandraFS). HBase is an open source distributed column oriented database that provides Bigtable inspired features on top of HDFS. HBase includes Java classes that allow it to be used transparently as a source and/or sink for MapReduce jobs. Our choice of MongoDB is motivated by the need for a document-oriented store for HazMat space time paths visualization on the map.

In Figure 4, we present the proposed scalable architecture for HazMats Space Time Path Data Warehousing respecting intelligent transport systems architecture. The first stage is to collect spatio-temporal data of trajectories, as GPX, OV2, or CSV files from different GPS enabling devices of drivers and vehicles. We use asynchronous .Net sockets for collecting data to data collector server. Then the collected data is processed using Data reducer, Error measures, Reverse Geocoding, and Activity recognition services. The Extract Transform and Load phase from heterogeneous data will be discussed in future article to respect the paper pages limit. After that, data could be stored on MongoDB data base, processed within Hadoop via one or more MapReduce jobs. Output from these MapReduce jobs can then be written back to MongoDB for later querying and *ad-hoc* analysis. Since results format returned from MongoDB are in JSON (JavaScript Object Notation) with no needed conversion, and also JSON is much faster than other XML based technologies. We use JSON format, in the proposed framework, to monitor Dangerous Goods Transport Space Time Path Data in browsers.

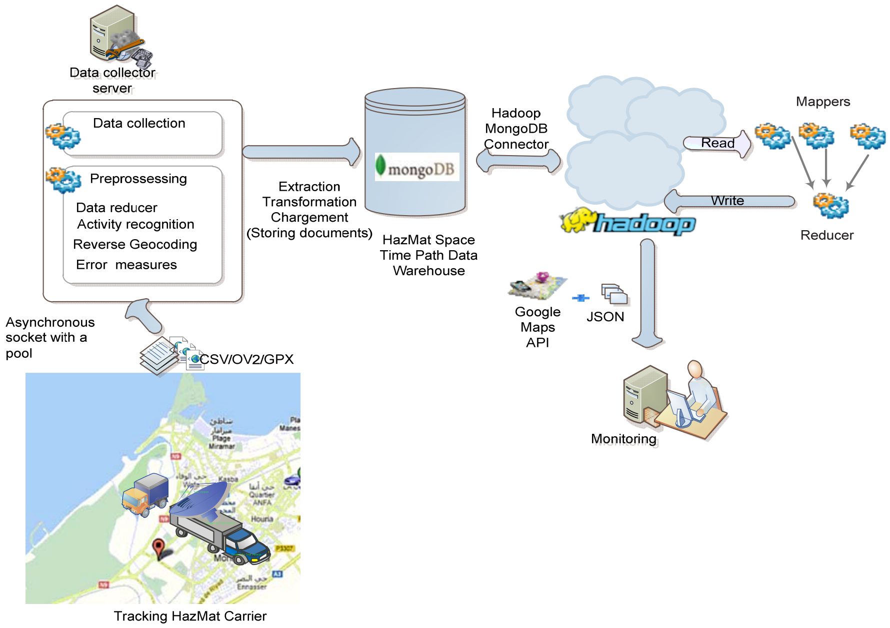

Figure 4. System architecture for HazMats Space Time Path Data Warehousing

Trajectories data volume is huge, storing on a single machine does not allow the treatment in near real time. To process the massive trajectories data, the deployment of the Hadoop platform must be done on multiple nodes. We need an architecture platform "distributed", and therefore several machines that will use DataNode/TaskTracker (data storage and treatment nodes). The cost of a platform with multiple servers is high, but thanks to the Cloud with the particularity of the IaaS cloud computing, it is now possible to get the infrastructure (virtual machines) and pay according to the duration of use.

In addition, cloud is suitable for the scalable storage (scale-out) of very large data volumes. It is an ideal complement to the Hadoop framework because both technologies benefit from the maturity of virtualization, by providing opportunities for storage and expandable treatment. We adopt this architecture to enable the use of a file system located in the cloud for data storage and execution of data processing. Analytical processing of data is distributed on the compute nodes that form the cluster, these can be virtual machines and take advantage of the scalability of Hadoop nodes (ability to vary the number of nodes as required). However, MapReduce provides fault tolerance functionality with its ability to revive a node or assign a task to another node.

The flexibility of cloud brings agility in infrastructure management and resource allocation. The virtual abstraction layer is also centralized, and reveals more manageable.

Kang *et al.* (2013) compared the use of physical clusters and virtual machines in the cloud. The result of the study shows that running Hadoop in virtual machines of a private cloud provides more than 110.76% performance of the physical server.

Therefore, we adopted this architecture (Figure 5) for analysing the trajectories that consists in treating the data with Hadoop using Map Reduce of the Framework Hadoop in virtual machines in the cloud. Communication between MongoDB and Hadoop is made using the connector MongoDB-Hadoop-Connector (MongoDB, 2013).

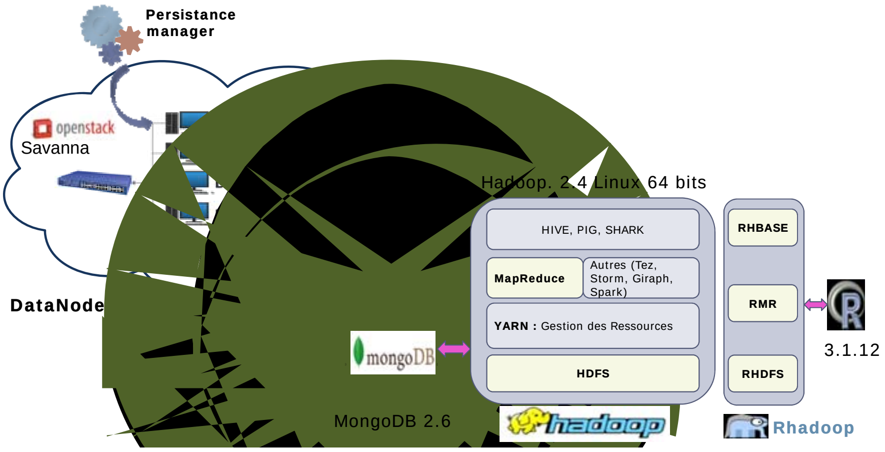

Figure 5. Analysis trajectory architecture using MongoDB and Hadoop clusters in a private cloud environment

4.3. Hazmat space time path data warehouse analysis

The proposed HazMat Space Time Path Data warehouse in a NoSQL database MongoDB is designed for query and analysis the big volume of spatio-temporal data. MongoDB supports a rich, *ad-hoc* query language of its own. Therefore, in a scalable, fast and agile way, complicated HazMat Space Time Path Data warehouse analytical queries can be reduced to nearly line Mongo queries as there is no joins (documents are embedded). Using our system we present, in figure 6, an example of dangerous regions in a urban area that hundreds of trucks carrying Hazardous Materials goes through on afternoons.

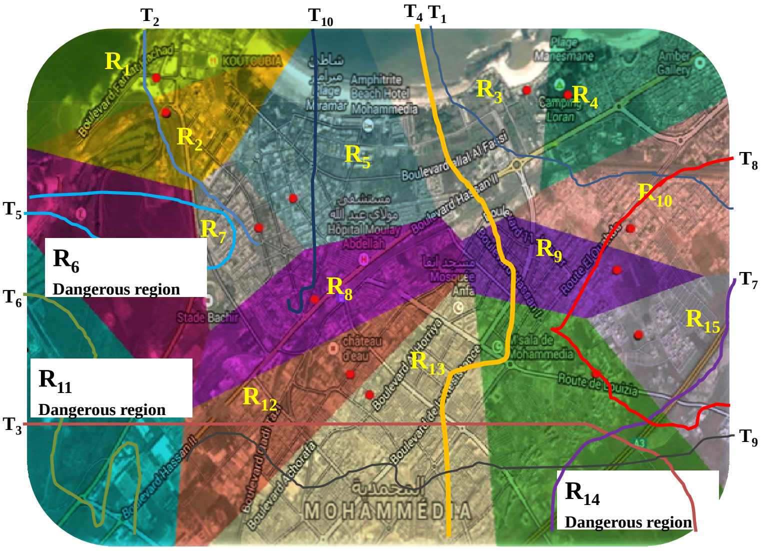

Figure 6. Example of dangerous regions in urban area that hundreds of trucks carrying Hazardous Materials goes through on afternoons

5. Conclusion

In this paper we have proposed hazmat space time path data warehouse conceptual model for establishment of a decisional database that capture HazMat trajectories, shipments and occurring incidents or accidents.

We opted for the creation of our storage system trajectories above the Big Data mainly because it supports different external data sources, not based on a fixed scheme with a capacity to handle very large data, allows an analysis in near real time, has predictive capabilities of the results, has the power to make queries on the fly without a first building query that must be pre-calculated in OLAP cubes.

Furthermore, we have established a conceptual schema for trajectories objects. This schema is based on the hierarchical documents that contain several related levels compatible with NoSQL database type document. The dimensions and measures are inherited from the unified trajectories meta-model.

The proposed system can be exploited in different applications domains and is able to handle in near real time GeoStream amount of spatio-temporal data of hazardous trajectories system from different moving objects and analyzing them in a scalable, fast and agile way.

Decision Makers can mine hazmat space time path data warehouse in MongoDB database using Hadoop framework and its MapReduce paradigm to benefit from the maximum of performance and scalability.

The perspective of this work is, in the short term, to implement more complex aggregate functions to perform the space time path data warehouse analytical operations, and to continue experimentations of the proposed hazmat space time path data warehouse by increasing the load and using the MapReduce paradigm in a cloud computing environment.

References

Alex Holmes. (2012). *Hadoop in Practice*. Manning Publications Co.

Benitez E., Collet C., Adiba M. (2001). Entrepôts de données : caractéristiques et problématique. *Revue TSI,* vol. 20, n° 2.

Boulmakoul A., Karim L. (2013). A framework for scalable NoSQL storing moving objects' trajectories. *Conférence Maghrébine sur les Avancées des Systèmes Décisionnels, ASD'13*.

Boulmakoul A., Karim L., Lbath A. (2012). Moving Object Trajectories Meta-Model and Spatio-temporal Queries. *International Journal of Database Management Systems*, vol. 4, n° 2, p. 35-54.

Boulmakoul A, Karim L (2014). Construction et entreposage des trajectoires. *Work. Int. sur l'Innovation Nouv. Tend. dans les Systèmes d'Information,* 4e Ed.

Damiani M. L., Vangenot C., Frentzos E., Marketos G., Theodoridis Y., Veryklos V., Raffaeta A. (2007). *Geographic privacy aware Knowledge Discovery and Delivery*.

Doucet A., Gangarski S. (2001). Entrepôts de données et Bases de Données Multidimension-nelles, Chapter 12 Book: *Bases de Données et Internet, Modèles, langages et systèmes*. Hermès editions.

Elzbieta M., Esteban Z. (2008). Advanced Data Warehouse Design: From Conventional to Spatial and Temporal Applications. Data-Centric Systems and Applications. Springer; 1st ed. 2008. Corr. 2nd printing edition (April 6, 2011).

Fitzke J., Greve K. (2010). Frei oder umsonst? - Nutzergenerierte Geoinformation zwischen Freiheit und Kostenlosigkeit. In: *Angewandte Geoinformatik - 22. GIT-Symposium*. 1. Ed., Wichmann, Berlin, p. 732-741.

Freitas G., Alberto M., Laender H., Luiza M. (2002). Getting Users Involved in the Development of Data Warehouse Application, In Proc. of the *4th International Workshop (DMDW)*, Toronto, Canada, p. 3-12.

Fubédard Y., Merrett T., Han J. (2001). Fundamentals of spatial data warehousing for geographic knowledge discovery. *Geographic Data Mining and Knowledge Discovery,* London: Taylor and Francis, p. 53-73.

Giannotti F., Nanni M., Pedreschi D., Pinellin F. (2007). Trajectory Pattern Mining. *International Conference on Knowledge Discovery and Data Mining*, p. 330-339.

Güting R.H., Behr T., Almeida V., Ding Z., Hoffmann F., Spiekermann M. Secondo (2004). An extensible DBMS architecture and prototype. Technical report.

Hongbo Y., Shaw, S. (2007). Revisiting Hägerstrand's time-geographic framework for individual activities in the age of instant access, *Societies and Cities in the Age of Instant Access*. In H. Miller (ed.) Dordrecht, The Netherlands: Springer Science, p. 103-118.

http://www.spatial-eye.com/Engels/Applications/Spatial-DWH/page.aspx/117.

Jason Venner. (2009). Pro Hadoop. Build scalable, distributed applications in the cloud.

Kang Y., Kang K.-W. (2013). An Empirical Study of Hadoop Application running on Private Cloud Environment. *Adv Sci Technol Lett* 35, p. 70-73.

Leonardi L. Marketos G., Frentzos E., Giatrakos N., Orlando S., Pelekis N., Raffaeta A., Roncato A., Silvestri C., Theodoridis Y. (2010). T-Warehouse: Visual OLAP analysis on trajectory data. *Data Engineering (ICDE), IEEE 26th International Conference*.

Levene M., Loizou, G. (2003). Why is the Snowflake Schema a Good Data Warehouse Design? *Information Systems*, vol. 3, n° 28, p. 225-240.

MacEachren A. M., Kraak M. (2001). Research challenges in geovisualization. *Cartography and Geographic Information Science*.

Meng X., Ding Z. (2003). DSTTMOD: A Discrete Spatio-Temporal Trajectory Based Moving Object Databases System. DEXA, LNCS 2736, Springer; p. 444-453.

Mike L. (2012). Planning for Big Data. *O'Reilly Media*. chapter 8 The NoSQL Movement. ISBN: 978-1-449-32967-9.

MongoDB 10gen. (2013). Available from: http://www.mongodb.org.

Newson P., Krumm J. (2009). Hidden Markov Map Matching Through Noise and Sparseness. *Proc. 17th ACM SIGSPATIAL Int. Symp. Adv. Geogr. Inf. Syst.*, p. 336-343.

OGC 07-022r1 Version: 1.0. (2008). Available from: httpnt Systems and Machine Learning.

Quine W. V. O. (1985). Events and reification. *Actions and events: Perspectives on the philosophy of Donald Davidson*, LePore E., McLaughlin B. P. (Eds.). Oxford, p. 162-171.

Salvatore O., Renzo O., Alessandra R., Alessandro R. (2007). Trajectory Data Warehouses: Design and Implementation Issues. *Journal of Computing Science and Engineering*, vol. 1, n° 2, December 2007, p. 211-232.

Shaw S. (2011). A Space-Time GIS for Analyzing Human Activities and Interactions in Physical and Virtual Spaces. Center for Intelligent Systems and Machine Learning.

Simone C., Macedo J., Spinsanti L. (2011). St-Toolkit: A Framework for Trajectory Data Warehousing. *AGILE 2011*, April 18-22.

Song I., Medsker W. (2001). An Analysis of Many-to- Many Relationships Between Fact and Dimension Tables in Dimension Modeling. *In Proc. of the International Workshop on Design and Management of Data Warehouses*, vol. 6, Interlaken, Switzerland, p. 1-13.

Spaccapietra S., Parent C., Damiani M.D., Macedo J.A., Porto F., Vangenot C. (2008). A conceptual view on trajectories. *Data and Knowledge Engineering*, p. 26-146.

Spider Monkey (2013). https://developer.mozilla.org/en/ SpiderMonkey.

Teste O. (2001). Towards Conceptual Multidimensional Design in Decision Support Systems. In *Proc. of the 5th East-European Conference on Advances in Databases and Information Systems (ADBIS)*, Vilnius, Lithuania, p. 77-88.

Vaisman A., Zimányi E. (2009). What is spatio-temporal data warehousing? In *DAWAK*.

Wolfson O., Xu B., Chamberlain S., Jiang L. (1998). Moving objects databases: Issues and solutions. *Proceeding of the 10th International Conference on Scientific and Statistical Database Man-agement (SSDBM)*, USA, IEEE Computer Society, p. 111-122.

Yan Z., Parent C., Spaccapietra S., Chakraborty D. (2010). Hybrid Model and Computing Platform for Spatio-Semantic Trajectories. *7th Extended Semantic Web Conference*, Heraklion, Greece**.**