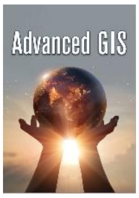
Advanced GIS
http://publish.mersin.edu.tr/index.php/agis/index
e-ISSN:2822-7026



Automatic construction of a knowledge base for transport networks

Ozgun Akcay *1

[1]Canakkale Onsekiz Mart University, Faculty of Engineering, Department of Geomatics Engineering, Canakkale, Turkey

Keywords
GIS,
Spatial Analysis,
Ontology,
Transportation,
Knowledge Base

Research Article
Received: 24/01/2022
Revised: 31/03/2022
Accepted: 27/05/2022
Published: 25/06/2022

ABSTRACT

Abundant spatial transport data is accessible from governmental institutions or internet sources contributed from all over the world. This kind of data obtained from various sources exists in different structures and prevents interoperability. The aim of the research, in this paper, is to develop an ontology compatible with specifications of transport networks of Infrastructure for Spatial Information in the European Community (INSPIRE) and to obtain a knowledge base of transportation data from non-semantic data sources automatically. In the first stage, ontological classes and relations in the transport network are explained. Then, defining algorithms in three main phases, non-semantic spatial data is transformed into the ontological structure. Some ontological queries are implemented to retrieve information from a knowledge base that is produced because of the transformation algorithms. As a result of the study, the transportation datasets in the relational database have been successfully converted into a semantic data format. The functionality of the obtained knowledge base was validated with queries that allow semantic reasoning.

1. Introduction

Today transportation networks that are equipped with sensors and built with agents have been widely used in the various services of transportation systems. To obtain a proper intelligent transportation system that can derive logical conclusions and make decisions successfully, a transport network system needs to correspond with knowledge representation and ontology. To decide how to behave upon instant environmental inputs gathered from sensors, agents need an internal declarative body of knowledge (Arara & Laurini, 2005). Declarative languages are described with description logics (DLs) as a family of knowledge representation formalisms that are mainly characterized by constructors to build complex concepts and roles from atomic ones (Horrocks et al., 1999).

In the last two decades, ontology languages that are supported with DLs are the standards of knowledge representation. In order to state a semantic structure that computers are able to understand and communicate with each other, ontological domains should be composed of ontology languages (Gómez-Pérez & Corcho, 2002). Ontology languages define classes (concepts), properties (roles), data types, and individuals for a specific domain. Property characteristics such as transitive, symmetric, functional, and inverse specify a property to provide enhanced reasoning about the property (Smith et al., 2004). As explained in Horridge (2011), existential, universal, and cardinality restrictions determine individual ranges of a property in a domain. Today, ontology-based applications are particularly implemented with OWL-DL (Ontology Web Languages – Description Logic) based on the RDF (Resources Description Framework) as a language of semantic data (Horrocks et al., 2003).

Recently, many semantic approaches have also been proposed in Geographical Information Systems (GIS) so as to solve semantic problems which can be categorized into three main topics: The first topic is the geospatial data heterogeneity causing interoperability problems (Lutz, 2007; Yang et al., 2008; Zhang et al., 2010a). The second one is the information retrieval of the geospatial data over the internet providing geospatial search engines (Jones et al., 2004; Fu et al., 2005; Alazzawi et al., 2012). The third topic refers to the aspect of geospatial intelligence by retrieving implicit geospatial data from explicit one with knowledge reasoners which proposes context-awareness in GIS (Wang et al., 2004; Weissenberg et al., 2006; Akcay & Altan, 2011). All this scientific research apparently reveals the importance of the semantic knowledge base of geospatial data.

Transport networks as a kind of geometric network is also a considerable subject within the realm of

---

***Corresponding Author**

*(akcay@comu.edu.tr) ORCID ID 0000-0003-0474-7518

Cite this article

Akcay, O. (2022). Automatic construction of a knowledge base for transport networks. *Advanced GIS*, 2(1), 08-17.

semantic geospatial information sciences. In order to take advantage of the semantic data, in the transportation field, many approaches have been exploited by scientists (Fernandez & Ossowski, 2007; Dong et al., 2008; Gregor et al., 2012; Oliveira et al., 2013). A long-term project has been initiated, called Infrastructure for Spatial Information in the European Community (INSPIRE), to provide more collaboration and interoperability for geospatial data among European countries. The aim of the project is to define directives for spatial data infrastructure in various stages until 2019. In 2010, data specifications of Transport Networks were published by the INSPIRE Thematic Working Group (INSPIRE, 2010). The transport networks specifications explain features of all transport themes (road, rail, water, air, and cable) in Unified Modelling Language (UML) diagrams to develop intelligent systems such as location-based services (LBS), telematics and navigation services.

Geometric linestring objects as one of the fundamental GIS data, are the basic need of transport networks, can be obtained from various sources. Some of them are free GIS data web sites such as OpenStreetMap (OSM) (OpenStreetMap, 2012) and Tiger (Topologically Integrated Geographic Encoding and Referencing System) Product (U.S. Census Bureau, 2011). The format of the distributed data is shape file developed by ESRI as a common GIS format. Other data sources can also be accepted as aerial and satellite images. Automatic extraction methods might provide linear feature data so as to prepare a road network of an urban area (Park et al., 2002; Mena, 2003). The raw data of transport networks, however, need converting to knowledge representation formalism because of obstacles in using them semantically.

Despite the definition of a domain ontology being considered as a really difficult task, handling huge amounts of a non-semantic dataset as a part of the knowledge base (e.g., instances of the ontology) might also bring about complicated obstacles. Most of the researchers who are mentioned above are concerned about the meticulous construction of domain ontology and focus on inference abilities of the ontology. The appropriate presence of instances, however, is as important as the building of ontological concepts and properties to provide a complete knowledge base for semantic applications. Regarding domain ontology constraints, dataset mapping for instance, from a non-semantic database onto a knowledge base is an obligatory process. As datasets are traditionally stored and maintained in relational database management systems (RDBMS), geospatial datasets are handled in RDBMS, extended with spatial functions such as PostGIS, PostgreSQL, MySQL, and Oracle Spatial.

The purpose of this paper is to describe a transformation model that can achieve a knowledge base in compliance with INSPIRE data specification on transport networks from the geospatial transport dataset. As a result of the transformation, the large amount of spatial data available from many sources is obtained as INSPIRE-compliant ontological data in OWL-DL. The next sections of the paper are structured as follows. Section 2 presents an expanded review of the related research. The definition of the semantic road network model is proposed in section 3. To generate a semantic road network model the system architecture is explained in section 4 and then, section five describes transformation algorithms. The results of the implementation are given in section 6. Concluding remarks are discussed in section 7.

2. Related Works

Eight basic logical relations, also known as RCC8 (Region Connection Calculus) were defined by Randell et al. (1992) in order to express possible states of the two spatial objects in the space for spatial representation and reasoning. Eight relations of RCC were described as follows: equal, disconnected, externally connected, partial overlap, tangential proper part, non-tangential proper part, the inverse of tangential proper part, and inverse of the non-tangential proper part. Depending on the development of DLs which provides well-defined semantics, formal properties (complexity, decidability), reasoning algorithms, and implemented systems (reasoner engines), RCC8 was reconsidered by Wessel (2001) within the scope of DLs. These studies apparently enlightened some semantic geospatial research of today.

Transportation as a field of geospatial sciences is also influenced by advances in semantic applications in order to enable more intelligent and more interoperable systems. Data sharing is particularly discussed by the transportation community so as to establish interoperability between different data providers. About fifteen years ago, some semantic approaches in transportation were developed in order to overcome obstacles of data exchange and problems of knowledge representation (Bishr et al., 1999; Bishr & Kuhn, 2000).

Obitko & Maríк (2005) explained transportation ontologies enabling communication between agents. The defined ontologies implement a multi-agent system with agent-based platforms for intelligence. The ontologies, however, do not concern comprehensive data interoperability. A semantic transit planning service proposed by Chen et al. (2008) is able to retrieve a transit trip plan using data aggregation and composition schemes. Houda et al. (2010) described ontologies for public transportation in Protege (2013). Public transportation ontologies presented trip plans as a result of some SWRL (Semantic Web Rule Language) rule-based queries. Oliveira et al. (2013) defined transportation ontology for user interface personalization. The study, however, considers only limited kinds of transport modes to solve the travel planning problem.

A new problem that appears after the building stage of transportation ontology is to obtain the knowledge base appropriately from non-semantic data sources. Zhang et al. (2008) developed an algorithm that is able to transform UML transportation data to OWL data form. The algorithm transformed UML packages, classes, attributes, and instances to OWL classes, relations, and individuals. Some issues, however, remained unsolved between UML and OWL. Furthermore, the more general transformation models lead to less comprehensive results for different data sources. Zhang et al. (2010a) and Zhang et al. (2010b) expressed semantic discovery and composition algorithms to eliminate semantic heterogeneity of transportation data.

Katsumi & Fox (2018) analyzed and compared the performance of various transportation ontologies. A transportation geoportal has been developed in accordance with the INSPIRE standard in the study, which deals with heterogeneous data (Gunay et al., 2014). Yu et al. (2017) developed a prediction method based on deep learning algorithms that require a long training time after transforming the traffic flow into static images. However, the development of the deep learning and knowledge base relationship put forward by Das et al. (2016) is another important problem that needs to be resolved. In the study of Ali et al. (2017), fuzzy logic ontologies and transportation properties were discussed. However, transferring non-semantic data to a knowledge base conforming to standards such as INSPIRE does not take place adequately in the literature (Table 1).

Table 1. Comparison among some related works

Study	The ability of semantic data construction	Compliance with a transport standardization
Gunay et al. (2014)	-	+
Das et al. (2016)	+	-
Yu et al. (2017)	+	-
The proposed study	+	+

As the importance of the transportation ontologies is understood from all related works mentioned above, the standardization problem of semantic transportation data and constructing a proper knowledge base, including instances and relations among them are remained unsolved. In this paper, a semantic model and an algorithm which implements the generation of a transportation knowledge base that is compatible with INSPIRE transportation network specifications is introduced.

3. Semantic Road Network Model

Generic Conceptual Model as a part of the data specification development framework of INSPIRE describes Generic Network Model (GNM) based on ISO 19148 (2012) (Geographical Information - Linear referencing) for network term definitions (INSPIRE, 2009). As an extension of the standardization, common transport elements application schema including road, rail, cable, water, and air transports are developed upon GNM (Figure 1). According to the application schema of road transport networks in INSPIRE 2010, features of RoadLinkSequence, RoadLink and RoadNode are defined as follows:

RoadLinkSequence: A linear spatial object composed of an ordered collection of road links, which represents a continuous path in a road network without any branches.

RoadLink: A linear spatial object that describes the geometry and connectivity of a road network between two points in the network.

RodeNode: A point spatial object that is used to either represent connectivity between two road links or to represent a significant spatial object such as a services station or roundabout.

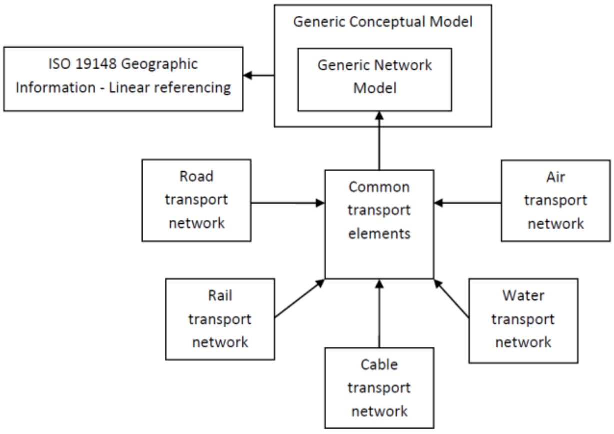


Figure 1. Inspire data specification on transport networks

As an example, Figure 2 depicts road link sequences, road links, and rode nodes in a road network. The figure includes 3 road link sequences, 7 road links, and 8 road nodes.

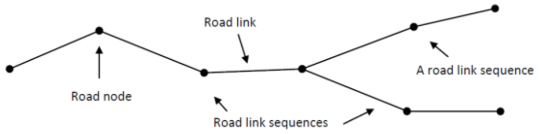


Figure 2. Road link sequence, road link, and road node

A semantic road network model (SRNM) consisting of classes, properties, and individuals, is built in conformity with INSPIRE road transport network application schema. The model has four classes (RoadNetwork, RoadLinkSequence, RoadLink, and RoadNode) and three object properties (contains, isPartOf, and isConnectedWith). The ontological object properties represent topological relations among the road classes (concepts). In Figure 3, in OWL syntax, three object properties and their types are defined. isConnectedWith property is a symmetric property while contains and isPartOf properties are transitive properties. Property types like symmetry or transitivity make contribution to a knowledge base so as to provide more inferences from implicit data. Figure 4 shows details of the ontological RoadLink class in road network in OWL syntax. According to the ontology, a road link is connected with other road links. Other written statements are that "RoadLink contains RoadNode" and "RoadLink isPartOf RoadLinkSequence".

```
<ObjectProperty rdf:about="&srnm;contains">
  <rdf:type rdf:resource="http://www.w3.org/owl#TransitiveProperty"/>
</ObjectProperty>
<ObjectProperty rdf:about="&srnm;isConnectedWith">
  <rdf:type rdf:resource="http://www.w3.org/owl#SymmetricProperty"/>
</ObjectProperty>
<ObjectProperty rdf:about="&srnm;isPartOf">
  <rdf:type rdf:resource="http://www.w3.org/owl#TransitiveProperty"/>
</ObjectProperty>
```

Figure 3. Object properties in semantic road networks

```
<Class rdf:about="&srnm;RoadLink">
  <equivalentClass>
     <Restriction>
        <onProperty rdf:resource="&srnm;isConnectedWith"/>
        <allValuesFrom rdf:resource="&srnm;RoadLink"/>
     </Restriction>
  </equivalentClass>
  <rdfs:subClassOf rdf:resource="&srnm;RoadNetwork"/>
  <rdfs:subClassOf>
     <Restriction>
        <onProperty rdf:resource="&srnm;contains"/>
        <allValuesFrom rdf:resource="&srnm;RoadNode"/>
     </Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
     <Restriction>
        <onProperty rdf:resource="&srnm;isPartOf"/>
        <someValuesFrom rdf:resource="&srnm;RoadLinkSequence"/>
     </Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
     <Restriction>
        <onProperty rdf:resource="&srnm;contains"/>
        <someValuesFrom rdf:resource="&srnm;RoadNode"/>
     </Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
     <Restriction>
        <onProperty rdf:resource="&srnm;isPartOf"/>
        <allValuesFrom rdf:resource="&srnm;RoadLinkSequence"/>
     </Restriction>
  </rdfs:subClassOf>
</Class>
```

Figure 4. RoadLink class and its relations between other road network classes

While defining classes with relations, there are two complementary matters in order to establish a robust semantic model. The first matter is that the class has an equivalent definition (defined class) or subclass definition (also known as a primitive class) by relation and its filler class (Horridge, 2011). In other words, a relation and its filler might be an upper (⊑) or an equivalent class (≡) of another class. The second matter is that if closure axiom is applied by a relation: Closure axiom is the combination of existential restriction (someValuesFrom ∃) and universal restrictions (allValuesFrom ∀). OWL inherits Open World Assumption (OWA) from Description Logics so as not to accept one reality as wrong without an explicit definition. OWA is a feature of reasoning that distinguishes OWL from relational databases. Table 2 expresses the ontological relations among classes using existential and universal restrictions under a given condition. In the table, (⊑) indicates necessary conditions, while (≡) represents necessary and sufficient condition.

Table 2. Conditions and definitions of relations with their filler classes. Used abbreviations: F.: Filler, Seq: Sequence, connWith: connected With

Class	Class	Relation	F. Class
RoadLink	⊑	∀contains	RoadNode
RoadLink	⊑	∃contains	RoadNode
RoadLinkSeq	⊑	∃contains	RoadNode
RoadLinkSeq	≡	∃contains	RoadLink
RoadNode	≡	∃isPartOf	RoadLink
RoadNode	⊑	∃isPartOf	RoadLinkSeq
RoadLink	⊑	∀isPartOf	RoadLinkSeq
RoadLink	⊑	∃isPartOf	RoadLinkSeq
RoadLink	≡	∀connWith	RoadLink
RoadLinkSeq	≡	∀connWith	RoadLinkSeq

4. Generation of Semantic Road Network

Transformation architecture has been designed in order to convert a non-semantic spatial data of a road network into a road network knowledge base. The algorithms in the architecture aim at obtaining road links, road link sequences and road nodes as individuals from a geospatial database as seen in Figure 5. The proposed system produces a knowledge base as an ontology file in OWL syntax upon SRNM. Before the transformation process, definitions of class taxonomy and relations (object properties) are written to the OWL file, as explained in section 3. The system input is geospatial data of the road network that is composed of a set of linestring representing the geometry of the roads.

As shown in Figure 5, a spatial database in the architecture includes three tables: a link table, a sequence table, and a node table. The link table stores geospatial attributes of the road network which is derived from the file of the network (input file), the table is then used to populate instances of RoadLink. The sequence table and node table are non-spatial tables that support determining and defining instances of RoadNode and RoadLinkSequence.

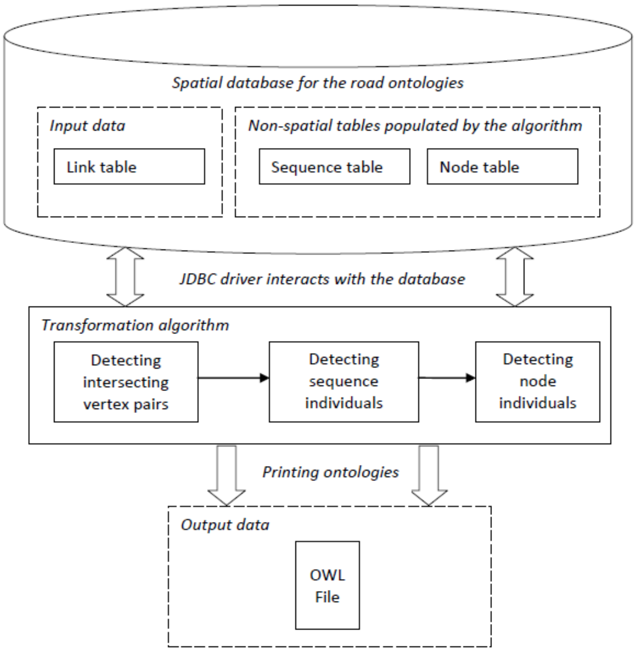


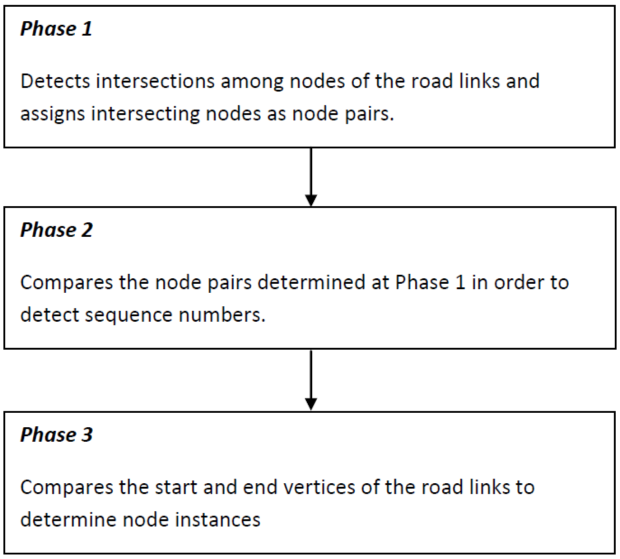
Figure 5. System architecture to produce road ontology

The center of the system which manages the transformation is the algorithm. The algorithm, including three main phases, is implemented in the Java programming language code. The phases 1 and 2 retrieve instances of the RoadLinkSequence in the database while the third phase defines instances of the RoadNode for the knowledge base. Interactions between the algorithm and PostgreSQL/PostGIS spatial database are provided by a Java Data Base Connectivity (JDBC) driver. In the next section, a transformation algorithm that retrieves instances will be elaborated on further.

A produced OWL file of the knowledge base includes all road network data semantically except object coordinates. The coordinate information of the spatial objects in the network could be possibly added to the ontology file as a data property. It is however intentionally excluded because a semantic reasoner is not as powerful as a spatial database concerning calculations for a huge amount of numerical data. Instead, a proposed semantic model infers implicit conclusions using topological relations.

5. Transformation Algorithms

A transformation algorithm produces a knowledge base of transportation network considering INSPIRE standards using topological relations of a geometrically featured spatial database. The aim of the algorithm is to retrieve instances of the classes (RoadNode, RoadLink and RoadLinkSequence) and relations among these instances using query functions of spatial database, such as intersection, equality, and touch. Each row of the link table represents a road link as defined in the INSPIRE road transport network. In the algorithm, the most obstructive tasks are to retrieve the node instances and sequence instances. The road link sequences, and road nodes are determined, using spatial features of the link table. The algorithm processes are as stated in the flow diagram in Figure 6.



Phase 1

Detects intersections among nodes of the road links and assigns intersecting nodes as node pairs.

Phase 2

Compares the node pairs determined at Phase 1 in order to detect sequence numbers.

Phase 3

Compares the start and end vertices of the road links to determine node instances

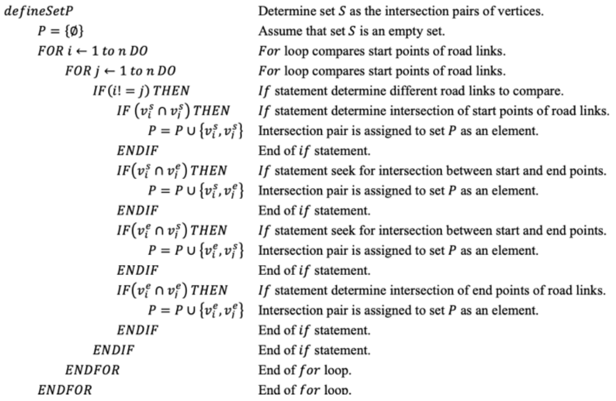
Figure 6. Process flow of the algorithm

5.1. Phase I

Phase I detects intersections between the start and end nodes of the road links and assigns intersecting nodes as a pair. It is assumed that n indicates the number of road links in the link table, v_i^s and v_i^e represents the start and end node of the i^{th} road link, v_j^s and v_j^e represent the start and end node of the j^{th} road link and $P = \{p_1, p_2, .., p_m\}$ is the set of the node intersection pairs. Between two road links, one of the four intersection possibilities is (v_i^s, v_j^s) , (v_i^s, v_j^e), (v_i^e, v_j^s), (v_i^e, v_j^e), therefore, might occur as a unique member of set P.

The algorithm is mainly based on the relations between the start and end points of the links. As shown in pseudo algorithm I, set P which has individuals of pairs among the start and end points of the links is obtained Figure 7. Intersection pairs (individuals of set P) are assigned into the n1 and n2 columns of the sequence table. After the algorithm of phase, I is implemented, multiple intersections (pairs), more than two at one point among road links indicate a junction point for different road link sequences, are deleted from columns n1, n2.

To clarify the transformation algorithm, the road network in the Figure 8 will be elaborated as an example. Figure 8 shows a simple road network composed of 31 road links. Figure 9a shows intersection pairs while Figure 9b depicts a sequence table for road links of the road network at Figure 8. To prevent complexity, labels of vertices and labels of some road links and are omitted in Figure 8. For instance, the start point of the 21th road link and the end point of the 28th road link form an intersection pair (v_{21}^s, v_{28}^e) as seen at the last row of the table in the Figure 9a. After phase I is applied and multiple intersections are eliminated, the table is obtained as shown in Figure 9a. Furthermore, the start point of the road link 28 intersects with road link 9 and road link 12 (v_9^e, v_{28}^s), (v_{12}^s, v_{28}^s) are determined as junction point and deleted from Table 9a. Multiple intersections are deleted from the table as they are not part of the same sequence.



```
defineSetP                                  Determine set S as the intersection pairs of vertices.
    $P = \{\emptyset\}$                      Assume that set S is an empty set.
    FOR $i \leftarrow 1$ to n DO             For loop compares start points of road links.
        FOR $j \leftarrow 1$ to n DO         For loop compares start points of road links.
            IF (i! = j) THEN                 If statement determine different road links to compare.
                IF $(v_i^s \cap v_j^s)$ THEN If statement determine intersection of start points of road links.
                    $P = P \cup \{v_i^s, v_j^s\}$   Intersection pair is assigned to set P as an element.
                ENDIF                        End of if statement.
                IF$(v_i^s \cap v_j^e)$ THEN  If statement seek for intersection between start and end points.
                    $P = P \cup \{v_i^s, v_j^e\}$   Intersection pair is assigned to set P as an element.
                ENDIF                        End of if statement.
                IF$(v_i^e \cap v_j^s)$ THEN  If statement seek for intersection between start and end points.
                    $P = P \cup \{v_i^e, v_j^s\}$   Intersection pair is assigned to set P as an element.
                ENDIF                        End of if statement.
                IF$(v_i^e \cap v_j^e)$ THEN  If statement determine intersection of end points of road links.
                    $P = P \cup \{v_i^e, v_j^e\}$   Intersection pair is assigned to set P as an element.
                ENDIF                        End of if statement.
            ENDIF                            End of if statement.
        ENDFOR                               End of for loop.
    ENDFOR                                   End of for loop.
```

Figure 7. Pseudo code of Phase I

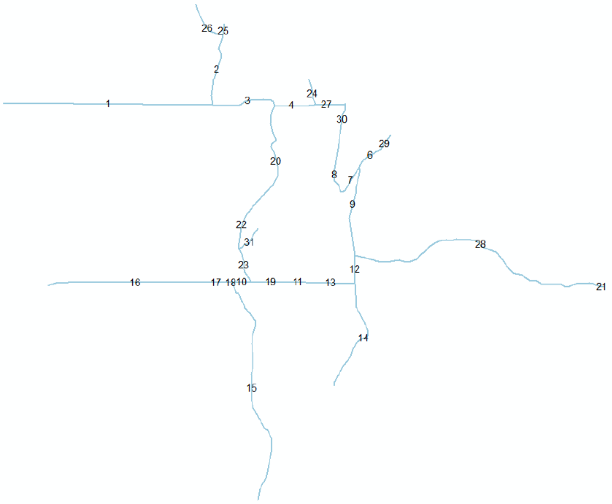
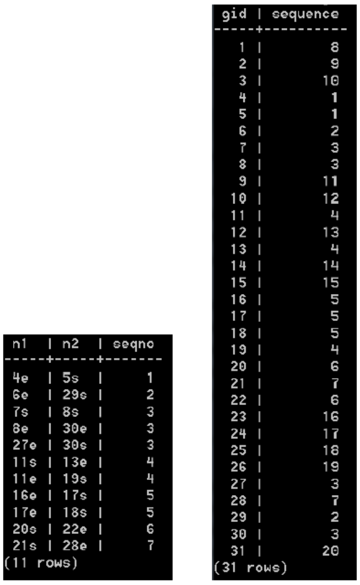


Figure 8. A road network with their road link numbers



```
 n1  | n2  | seqno
-----+-----+-------
 4e  | 5s  |     1
 6e  | 29s |     2
 7s  | 8s  |     3
 8e  | 30e |     3
 27e | 30s |     3
 11s | 13e |     4
 11e | 19s |     4
 16e | 17s |     5
 17e | 18s |     5
 20s | 22e |     6
 21s | 28e |     7
(11 rows)
```

```
 gid | sequence
-----+----------
   1 |        8
   2 |        9
   3 |       10
   4 |        1
   5 |        1
   6 |        2
   7 |        3
   8 |        3
   9 |       11
  10 |       12
  11 |        4
  12 |       13
  13 |        4
  14 |       14
  15 |       15
  16 |        5
  17 |        5
  18 |        5
  19 |        4
  20 |        6
  21 |        7
  22 |        6
  23 |       16
  24 |       17
  25 |       18
  26 |       19
  27 |        3
  28 |        7
  29 |        2
  30 |        3
  31 |       20
(31 rows)
```

Figure 9 Intersection pairs between road links start and end vertices in link sequence table (9a is the left figure); final sequence numbers in road link table (9b is the right figure)

5.2. Phase II

Phase II determines road link sequences from vertex pairs (Figure 10). The algorithm of phase II assigns a unique number to each sequence that includes more than one road link (Figure 9b). Sequence numbers which are therefore made of only one road link are assigned later. Sequence numbers of each road link are inserted into the "seqno" column of the sequence table (Figure 9a).

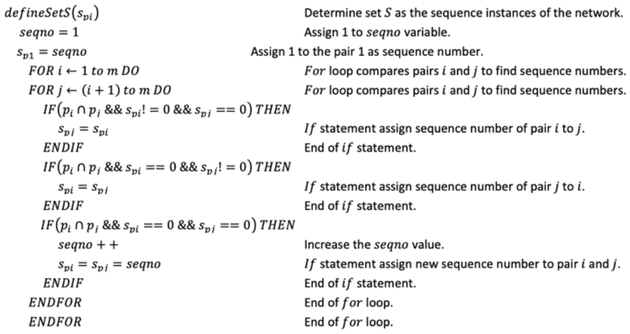
Let's assume that m indicates the number of individuals in the set P. S represents the sequence numbers of pairs while s_{pi} indicates the sequence number of the pair p_i. Seqno is a variable that assigns sequence numbers of the pairs. In the given example, $P = \{p_1 = (v_4^e, v_5^s), p_2 = (v_6^e, v_{29}^s) \dots (v_{21}^s, v_{28}^e)\}$ and $m = 11$ (the total number of the row at figure 9a).

When phase II is implemented, $s_{p1} = 1, s_{p2} = 2, s_{p3} = 3, s_{p4} = 3, s_{p5} = 3, s_{p6} = 4, s_{p7} = 4, s_{p8} = 5, s_{p9} = 5, s_{p10} = 6$ and $s_{p11} = 7$ are obtained, as seen in Figure 9a. Then, in Figure 9b, link table is extended with the "sequence" column in order to assign a final sequence number to each road link object. Additional to the sequences are explained in Figure 9a, other sequences, including only one road link, are determined uniquely. Sequence assignment of one road link sequence is not indicated in phase II due to its excessive code length.

5.3. Phase III

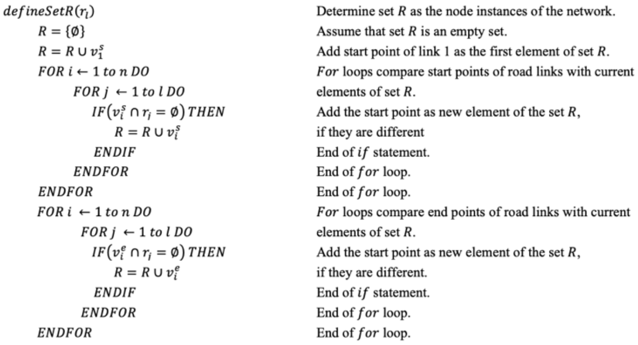
Phase III compares the start and end vertices of the road links to determine node instances (Figure 11). In the algorithm, set R is defined as an empty set. Then, the start and end points of the road links are compared with the element of set R and, a point is assigned to set R once, if they have not been assigned before to set R as an element. Consequently, each element of set R represents a node instance in the knowledge base.

Let's assume that l indicates the number of nodes in the node table, while R is the set of nodes and r is an individual of set R: $R = \{r_1, r_2, \dots, r_l\}$.



```
$defineSetS(s_{pi})$                              Determine set S as the sequence instances of the network.
  $seqno = 1$                                     Assign 1 to seqno variable.
  $s_{p1} = seqno$                          Assign 1 to the pair 1 as sequence number.
    $FOR\ i \leftarrow 1\ to\ m\ DO$              For loop compares pairs i and j to find sequence numbers.
    $FOR\ j \leftarrow (i+1)\ to\ m\ DO$          For loop compares pairs i and j to find sequence numbers.
      $IF(p_i \cap p_j \,\&\&\, s_{pi}! = 0 \,\&\&\, s_{pj} == 0)\ THEN$
        $s_{pj} = s_{pi}$                         If statement assign sequence number of pair i to j.
      $ENDIF$                                     End of if statement.
      $IF(p_i \cap p_j \,\&\&\, s_{pi} == 0 \,\&\&\, s_{pj}! = 0)\ THEN$
        $s_{pi} = s_{pj}$                         If statement assign sequence number of pair j to i.
      $ENDIF$                                     End of if statement.
      $IF(p_i \cap p_j \,\&\&\, s_{pi} == 0 \,\&\&\, s_{pj} == 0)\ THEN$
        $seqno++$                                 Increase the seqno value.
        $s_{pi} = s_{pj} = seqno$                 If statement assign new sequence number to pair i and j.
      $ENDIF$                                     End of if statement.
    $ENDFOR$                                      End of for loop.
    $ENDFOR$                                      End of for loop.
```

Figure 10. Pseudo code of Phase II



```
$defineSetR(r_l)$                                 Determine set R as the node instances of the network.
    $R = \{\emptyset\}$                           Assume that set R is an empty set.
    $R = R \cup v_1^s$                            Add start point of link 1 as the first element of set R.
    $FOR\ i \leftarrow 1\ to\ n\ DO$              For loops compare start points of road links with current
        $FOR\ j \leftarrow 1\ to\ l\ DO$          elements of set R.
            $IF(v_i^s \cap r_j = \emptyset)\ THEN$    Add the start point as new element of the set R,
                $R = R \cup v_i^s$                if they are different
            $ENDIF$                               End of if statement.
        $ENDFOR$                                  End of for loop.
    $ENDFOR$                                      End of for loop.
    $FOR\ i \leftarrow 1\ to\ n\ DO$              For loops compare end points of road links with current
        $FOR\ j \leftarrow 1\ to\ l\ DO$          elements of set R.
            $IF(v_i^e \cap r_j = \emptyset)\ THEN$    Add the start point as new element of the set R,
                $R = R \cup v_i^e$                if they are different.
            $ENDIF$                               End of if statement.
        $ENDFOR$                                  End of for loop.
    $ENDFOR$                                      End of for loop.
```

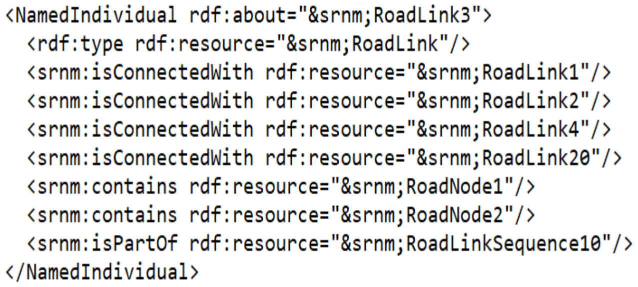
Figure 11. Pseudo code of Phase III

6. Results

Ontologies enable complex queries which retrieve implicit information from explicit data. Infinite query combinations can be derived from an ontology domain. For the given example above, two different complex queries and their results are explained in this section in order to clarify what ontologies are able to do. Queries, in the example, are written in the nRQL (New RacerPro Query Language), (Haarslev et al., 2007). nRQL is a query standard compatible with RacerPro. RacerPro stands for Renamed ABox and Concept Expression Reasoner Professional and is a reasoner engine for semantic web languages.

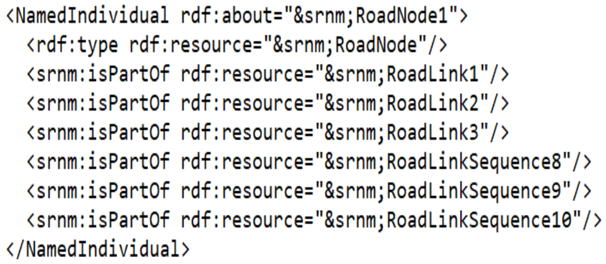
Individuals are determined with the algorithms that is explained in section 5. An individual definition also includes ontological relations with other individuals. The similar algorithms based on topological relations are used to define mentioned relations among individuals. However further algorithms about the relations are omitted here due to repeated versions of explained algorithms and possibility of an overlong paper. As a result of the implementation of the algorithm, 31 RoadLink individuals, 31 RoadNode individuals and 20 RoadLinkSequence individuals with numerous relations are obtained for the sample data set presented above. In excerpts from the knowledge base, individuals of

RoadLink, RodeNode and RoadLinkSequence are illustrated in figure 12, 13, and 14, respectively.



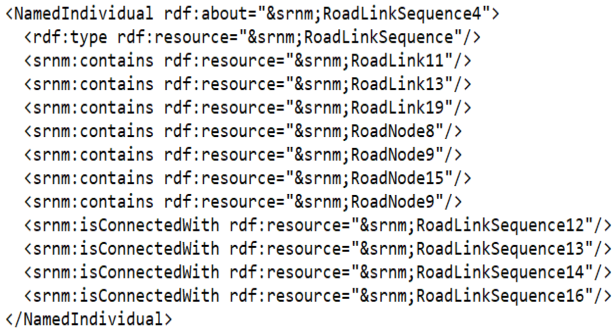
```
<NamedIndividual rdf:about="&srnm;RoadLink3">
  <rdf:type rdf:resource="&srnm;RoadLink"/>
  <srnm:isConnectedWith rdf:resource="&srnm;RoadLink1"/>
  <srnm:isConnectedWith rdf:resource="&srnm;RoadLink2"/>
  <srnm:isConnectedWith rdf:resource="&srnm;RoadLink4"/>
  <srnm:isConnectedWith rdf:resource="&srnm;RoadLink20"/>
  <srnm:contains rdf:resource="&srnm;RoadNode1"/>
  <srnm:contains rdf:resource="&srnm;RoadNode2"/>
  <srnm:isPartOf rdf:resource="&srnm;RoadLinkSequence10"/>
</NamedIndividual>
```

Figure 12. Individual of RoadLink3 and its relations



```
<NamedIndividual rdf:about="&srnm;RoadNode1">
  <rdf:type rdf:resource="&srnm;RoadNode"/>
  <srnm:isPartOf rdf:resource="&srnm;RoadLink1"/>
  <srnm:isPartOf rdf:resource="&srnm;RoadLink2"/>
  <srnm:isPartOf rdf:resource="&srnm;RoadLink3"/>
  <srnm:isPartOf rdf:resource="&srnm;RoadLinkSequence8"/>
  <srnm:isPartOf rdf:resource="&srnm;RoadLinkSequence9"/>
  <srnm:isPartOf rdf:resource="&srnm;RoadLinkSequence10"/>
</NamedIndividual>
```

Figure 13. Individual of RoadNode1 and its relations



```
<NamedIndividual rdf:about="&srnm;RoadLinkSequence4">
  <rdf:type rdf:resource="&srnm;RoadLinkSequence"/>
  <srnm:contains rdf:resource="&srnm;RoadLink11"/>
  <srnm:contains rdf:resource="&srnm;RoadLink13"/>
  <srnm:contains rdf:resource="&srnm;RoadLink19"/>
  <srnm:contains rdf:resource="&srnm;RoadNode8"/>
  <srnm:contains rdf:resource="&srnm;RoadNode9"/>
  <srnm:contains rdf:resource="&srnm;RoadNode15"/>
  <srnm:contains rdf:resource="&srnm;RoadNode9"/>
  <srnm:isConnectedWith rdf:resource="&srnm;RoadLinkSequence12"/>
  <srnm:isConnectedWith rdf:resource="&srnm;RoadLinkSequence13"/>
  <srnm:isConnectedWith rdf:resource="&srnm;RoadLinkSequence14"/>
  <srnm:isConnectedWith rdf:resource="&srnm;RoadLinkSequence16"/>
</NamedIndividual>
```

Figure 14. Individual of RoadLinkSequence4 and its relations

As seen in figure 12, individual RoadLink3 is topologically connected with road links 1, 2,4 and 20; contains road nodes 1 and 2; and is part of road link sequence 10. Figure 13 explains individual RoadNode1. According to the definition of RoadNode1 in the knowledge base, RodeNode1 is part of road links 1, 2, 3 and; road link sequences 8, 9 and 10. Figure 14 is about individual RoadLinkSequence4 that contains road links 11, 13, 19; road nodes 8, 9, 15, 19; and is connected with road link sequences 12, 13, 14, 16.

Query 1 retrieves road link individuals which have at least one neighbour road link each belonging to the same road link sequence at each of its sides. Table 3 defines query 1. Variables in NQRL are prefixed with "?". Injective variables are prefixed with "$?". In the queries of the example, all variables are assigned as injective variables. Haarslev et al., (2007), an injective variable is explained as a variable which can only be bound to an ABox individual that is not already bound to another injective - the mapping from variables to ABox individuals is thus injective. Concepts and relations in the query must be written with its URL (Uniform Resource Locater) as defined in the owl file. For the owl file and the queries presented in this work, URL is defined as "http://www.semanticweb.org/ontologies/2013/02/srnm.owl#". In order to abbreviate and simplify the query sentence, prefix URLs are omitted. Result of query 1 yields road links individuals 8, 11, 17, 30 as stated in Table 4 and Figure 15.

Query 2 asks for any individuals at intersection of three neighboring road link sequences as explained in Table 5. Reasoning engine answer to the query 2 is stated in Table 6 (see Figure 16 for a graphic version of the answer of query 2).

Table 3. Definition of query 1

Line	Parts of query 1 statement	Definition
1	(retrieve ($?y)	Variable will be retrieved after execution
2	(and	Logical operator connects sentences from line 3 to 11
3	($?x RoadLink)	Variable for road link individuals
4	($?y RoadLink)	Variable for road link individuals
5	($?z RoadLink)	Variable for road link individuals
6	($?x $?y isConnectedWith)	Variable pairs for connecting road link individuals
7	($?y $?z isConnectedWith)	Variable pairs for connecting road link individuals
8	($?t RoadLinkSequence)	Variable for road link sequences
9	($?x $?t isPartOf)	Variable pairs for road link individual covered by sequence
10	($?y $?t isPartOf)	Variable pairs for road link individual covered by sequence
11	($?z $?t isPartOf))	Variable pairs for road link individual covered by sequence

Table 4. Answer of query 1

Line	Search variable	Retrieved instances
1	((|$?y|	RoadLink8)
2	(|$?y|	RoadLink11)
3	(|$?y|	RoadLink17)
4	(|$?y|	RoadLink30))

Table 5. Definition of query 2

Line	Parts of query 2 statement	Definition
1	(retrieve ($?n)	Variable will be retrieved after execution
2	(and	Logical operator connects sentences from line 3 to 11
3	($?x RoadLinkSequence)	Variable for road link sequence individuals
4	($?y RoadLinkSequence)	Variable for road link sequence individuals
5	($?z RoadLinkSequence)	Variable for road link sequence individuals
6	($?x $?y isConnectedWith)	Variable pairs for connecting road link sequence individuals
7	($?y $?z isConnectedWith)	Variable pairs for connecting road link sequence individuals
8	($?x $?z isConnectedWith)	Variable pairs for connecting road link sequence individuals
9	($?n $?x isPartOf)	Variable pairs for individuals covered by sequence
10	($?n $?y isPartOf)	Variable pairs for individuals covered by sequence
11	($?n $?z isPartOf]))	Variable pairs for individuals covered by sequence

Table 6. Answer of query 2

Line	Search variable	Retrieved instances
1	((|$?n|	RoadNode1)
2	(|$?n|	RoadLink2)
3	(|$?n|	RoadLink4)
4	(|$?n|	RoadLink6)
5	(|$?n|	RoadLink7)
6	(|$?n|	RoadLink9)
7	(|$?n|	RoadLink18)
8	(|$?n|	RoadLink19)
9	(|$?n|	RoadLink20)
10	(|$?n|	RoadLink22))

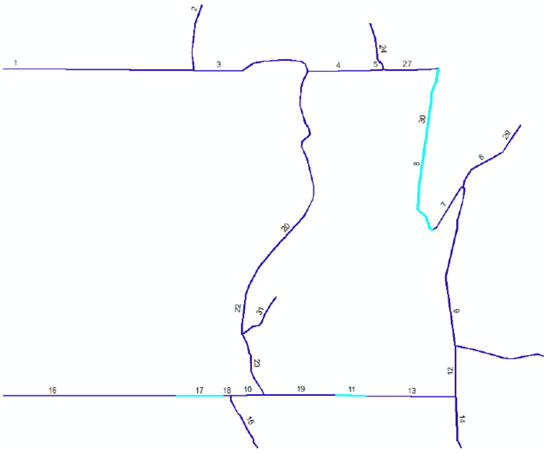


Figure 15. Graphic result of query 1

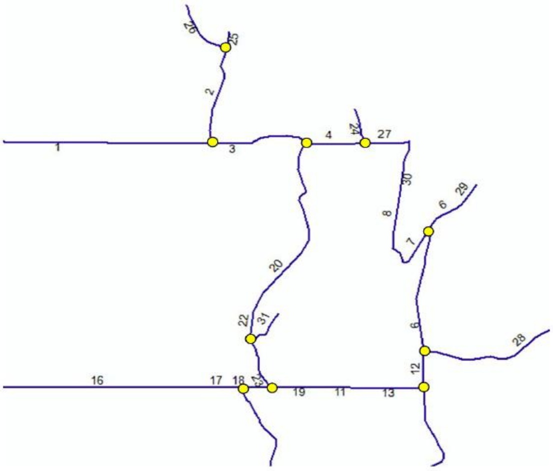


Figure 16. Graphic result of query 2

7. Conclusion

Semantic approaches applied on either transportation systems, or any other systems deal with some geospatial obstacles. The main obstacle is to set up an appropriate ontological structure enabling reasoning as much as possible. For example, two different semantic transportation systems having the same explicit information but built up with different ontological taxonomy and object properties (two knowledge bases with different expressive power) might have different performances to produce implicit information as a result of ontological queries and reasoning. Capability of obtaining implicit information from a knowledge base indicates its computational complexity. Setting up an ideal ontology having enough expressive power for transportation is still an open question.

Another bottleneck is to obey international data standards while building an expressive knowledge base. Standards are supportive documents to establish semantic data structure from non-semantic one. To be obliged to codes of standards, however, make it difficult to obtain a fully expressive knowledge base.

Semantic data is superior to a traditional geospatial data built in a database regarding query speed and capacity. In knowledge bases, answer time is shorter than queries of relational database management systems (RDBMS), as queries are based on sentence syntax in semantic engines. Data in RDBMS spatial databases is stored as numerical codes, therefore the performance of geometric functions is slower when considering huge amounts of data. This drawback of spatial databases has tried to be overcome with indexing. Actually, converting semantic data might be called as a kind of indexing. Furthermore, acquired semantic data as a result of transformation will be ready for agent-based smart applications and provide advantages of interoperability.

Due to the wide scope of INSPIRE data specifications on a transport network, a more comprehensive transformation should be discussed than the subjects handled in this paper. Transportation algorithms therefore should be a research subject, including various data types, such as, road speed limit, traffic flow direction and traffic signs. Especially, this kind of information is an inevitable requirement for applications of smart navigation systems and location-based services.

Author Contributions

The study was carried out by a single author.

Statement of Conflicts of Interest

The author declares no conflicts of interest.

Statement of Research and Publication Ethics

Research and publication ethics were complied with in the study.

References

Akcay, O., & Altan, O. (2011). An ontology-based visualization for mobile geoinformation services. *Scientific Research and Essays*, 6(4), 993-1000. https://doi.org/10.5897/SRE10.416

Ali, F., Kwak, D., Khan, P., Islam, S. R., Kim, K. H., & Kwak, K. S. (2017). Fuzzy ontology-based sentiment analysis of transportation and city feature reviews for safe traveling. *Transportation Research Part C: Emerging Technologies*, 77, 33-48. https://doi.org/10.48550/arXiv.1701.05334

Alazzawi, A. N., Abdelmoty, A. I., & Jones, C. B. (2012). What can I do there? Towards the automatic discovery of place-related services and activities. *International Journal of Geographical Information Science*, 26(2), 345-364.

Arara, A. A., & Laurini, R. (2005). Formal contextual ontologies for intelligent information systems. *Proceedings of World Academy of Science, Engineering and Technology*, 303-306.

Bishr, Y., & Kuhn, W. (2000). Ontology-based modelling of geospatial information. *3rd AGILE Conference on Geographic Information Scienc*e, Helsinki, Finland.

Bishr, Y., Pundt, H., Kuhn, W., & Radwan, M. (1999). Probing the concept of information communities—a first step towards a road of semantic interoperability interoperating geographic information systems. *Interoperating geographic information systems*, 55–69.

Chen, K. H., Dow, C. R., & Guan, S. J. (2008). Nimble transit: public transportation transit planning using semantic service composition schemes. *11th International IEEE Conference on Intelligent Transportation Systems*, Beijing, China, 723-728.

Das, R., Neelakantan, A., Belanger, D., & McCallum, A. (2016). Chains of reasoning over entities, relations, and text using recurrent neural networks. *arXiv preprint arXiv,1607.01426*. https://doi.org/10.48550/arXiv.1607.01426

Dong, H., Hussain, F. K., & Chang, E. (2008). Transport service ontology and its application in the field of semantic search. *IEEE International Conference on Service Operations and Logistics, and Informatics*, Beijing, China 820-824. https://doi.org/10.1109/SOLI.2008.4686512

Fernandez, A., & Ossowski, S. (2007). Role-based discovery and coordination of semantic transportation services. *IEEE Intelligent Transportation Systems Conference*, Seattle, WA, USA.

Fu, G., Jones, C. B., & Abdelmoty A I (2005). Ontology-based spatial query expansion in information retrieval. *OTM Confederated International Conferences*, Berlin, Heidelberg, 1466-1482. https://doi.org/10.1007/11575801_33

Gómez-Pérez, A., & Corcho, O. (2002). Ontology Languages for the Semantic Web. *IEEE Intelligent Systems* 17(1), 54-60. https://doi.org/10.1109/5254.988453

Gregor, D., Toral, S. L., Ariza, T., & Barreo, F. (2012). An ontology-based semantic service for cooperative urban equipments. *Journal of Network and Computer Applications*. 35(6), 2037-2050. https://doi.org/10.1016/j.jnca.2012.08.002

Gunay, A., Akcay, O., & Altan, M. O. (2014). Building a semantic based public transportation geoportal compliant with the INSPIRE transport network data theme. *Earth Science Informatics*, 7(1), 25-37. https://doi.org/10.1007/s12145-013-0129-z

Haarslev, V., Möller, R., & Wessel, M. (2007). *RacerPro user's guide and reference manual version 1.9.1.* Racer Systems GmbH & Co. KG.

Horridge, M. (2011). *A Practical guide to building owl ontologies using protégé 4 and co-ode tools edition 1.3.* The University of Manchester. Retrieved October 03, 2021, from http://mowl-power.cs.man.ac.uk/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_3.pdf

Horrocks, I., Sattler, U., & Tobies, S. (1999). Practical reasoning for expressive description logics. *6th International Conference on Logic for Programming and Automated Reasoning*, 161–180.

Horrocks, I., Patel-Schneider, P. F., & Van Harmelen, F. (2003). From SHIQ and RDF to OWL: The making of a web ontology language. *Journal of Web Semantics,* 1(1), 7-26. https://doi.org/10.1016/j.websem.2003.07.001

Houda, M., Khemaja, M., Oliveira, K., & Abed, M. (2010). A public transportation ontology to support user travel planning. *4th International Conference on Research Challenges in Information Science (RCIS)*, https://doi.org/10.1109/RCIS.2010.5507372

INSPIRE. (2009). *D2.5 INSPIRE generic conceptual model. drafting team data specifications*. INSPIRE. Retrieved Retrieved October 03, 2021, from http://inspire.jrc.ec.europa.eu/documents/Data_Specifications/D2.5_v3.2.pdf

INSPIRE (2010). D2.8.I.7 *INSPIRE data specification on transport networks – guidelines. INSPIRE Thematic Working Group.* INSPIRE. Retrieved October 03, 2021, from https://inspire.ec.europa.eu/id/document/tg/tn

ISO 19148. (2012). *Geographic information - linear referencing system*. ISO. Retrieved October 03, 2021, from https://www.iso.org/standard/75150.html

Jones, C. B., Abdelmoty, A. I., Finch, D., Fu, G., & Vaid, S. (2004). The SPIRIT spatial search engine: architecture, ontologies and spatial indexing. *International Conference on Geographic Information Science* 125-139. https://doi.org/10.1007/978-3-540-30231-5_9

Katsumi, M., & Fox, M. (2018). Ontologies for transportation research: A survey. *Transportation Research Part C: Emerging Technologies*, 89, 53-82. https://doi.org/10.1016/j.trc.2018.01.023

Lutz, M. (2007). Ontology-based descriptions for semantic discovery and composition of geoprocessing services. *Geoinformatica*, 11, 1-36. https://doi.org/10.1007/s10707-006-7635-9

Mena, J. (2003). State of the Art on automatic road extraction for GIS update: A novel classification. *Pattern Recognition Letters*, 24, 3037–3058.

Obitko, M., & Marík, V. (2005). Integrating transportation ontologies using semantic web languages. *International Conference on Industrial Applications of Holonic and Multi-Agent Systems,* 99–110. https://doi.org/10.1007/11537847_9

Oliveira, K. M., Bacha, F., Mnasser, H., & Abed, M. (2013). Transportation ontology definition and application for the content personalization of user interfaces. *Expert Systems with Applications*, 40(8), 3145-3159. https://doi.org/10.1016/j.eswa.2012.12.028

OpenStreetMap (2012). *OpenStreetMap*. Retrieved October 03, 2021, from http://www.openstreetmap.org/

Park, R. J., Saleh, R., & Yeu, Y. (2002). Comprehensive survey of extraction techniques of linear features from remote sensing imagery for updating road spatial databases*. ASPRS-ACSM Annual Conference and FIG XXII Congress*.

Protégé. (2013). *Protege Project*. Retrieved October 03, 2021, from http://protege.stanford.edu/

Randell, D. A., Cui, Z., & Cohn, A. G. (1992). A spatial logic based on regions and connections. *International Conference on Principles of Knowledge Representation and Reasoning (KR'92)*, 165-176.

Smith, M. K., Welty, C., & McGuinness, D. L. (2004*). OWL web ontology language guide. world wide web consortium, w3c recommendation*. Retrieved October 03, 2021, from http://www.w3.org/TR/owl-guide/

U.S. Census Bureau. (2011). TIGER/Line Retrieved October 03, 2021, from http://www.census.gov/geo/www/tiger/

Wang, X. H., Zhang, D. Q., Gu, T., & Pung, H. K. (2004). Ontology based context modelling and reasoning using OWL. *2th IEEE International Conf. on Pervasive Computing and Communications PerCom'04*, Orlando, USA, 18-22. https://doi.org/10.1109/PERCOMW.2004.1276898

Weissenberg, N., Gartman, R., & Voisard, A. (2006). An ontology-based approach to personalized situation-aware mobile services. *Geoinformatica*, 10, 55-90. https://doi.org/10.1007/s10707-005-4886-9

Wessel, M. (2001). Obstacles on the Way to qualitative spatial reasoning with description logics: some undecidability results. *2001 Description Logic Workshop (DL'2001)*, Stanford, California, USA, 1–3.

Yang, C., Li, W., Xie, J., & Zhou, B. (2008). Distributed geospatial resources to support digital earth. *International Journal of Digital Earth*, 1, 259-278. https://doi.org/10.1080/17538940802037954

Yu, H., Wu, Z., Wang, S., Wang, Y., & Ma, X. (2017) Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks. *Sensors*, 2017(17), 1501. https://doi.org/10.3390/s17071501

Zhang, C., Peng, Z. R., Zhao, T., & Li, W. (2008). Transformation of transportation data models from unified modelling language to web ontology language. *Transportation Research Record*, 2064, 81-89. https://doi.org/10.3141/2064-11

Zhang, C., Zhao, T., Li, W., & Osleeb, J. P. (2010a). Towards logic-based geospatial feature discovery and integration using web feature service and geospatial semantic web. *International Journal of Geographic Information Science* 24(6): 903–923. https://doi.org/10.1080/13658810903240687

Zhang, C., Zhao, T., & Li, W. (2010b). Automatic search of geospatial features for disaster and emergency management. *International Journal Applied Earth Observations and Geoinformation,* 12, 409–418. https://doi.org/10.1016/j.jag.2010.05.004