Doctoral Dissertation
Doctoral Program in Urban and Regional Development (30th Cycle)

Geomatics for Mobility Management
A comprehensive database model for Mobility Management

By

Emere Arco

******

Supervisor:
Prof. Piero Boccardo

Doctoral Examination Committee:
Prof. Gabriele Bitelli, Referee, Università di Bologna
Prof. Maria Brovelli, Politecnico di Torino
Prof. Luigi Buzzacchi, Politecnico di Milano
Prof. Mattia Crespi, Referee, Università di Roma “La Sapienza”
Prof. Francesco Laio, Politecnico di Torino

Politecnico di Torino
2018

Declaration

I hereby declare that, the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

Emere Arco

2018

* This dissertation is presented in partial fulfillment of the requirements for **Ph.D. degree** in the Graduate School of Politecnico di Torino (ScuDo).

Acknowledgment

I would like to acknowledge my advisor Prof. Piero Boccardo, for the support of my Ph.D study and research.

I would also to thank the staff of 5T and in particular the Chief Technology Officer Fabrizio Arneodo, the IT Team and Traffic Control Room Team for their valuable guidance and their patience in answering my questions.

Last but not the least, I would like to thank my colleagues at ITHACA, for involving me in inspiring and variuos projects and above all for the fun moments.

Abstract

In urban and metropolitan context, Traffic Operations Centres (TOCs) use technologies as Geographic Information Systems (GIS) and Intelligent Transport Systems (ITS) to tackling urban mobility issue. Usually in TOCs, various isolated systems are maintained in parallel (stored in different databases), and data comes from different sources: a challenge in transport management is to transfer disparate data into a unified data management system that preserves access to legacy data, allowing multi-thematic analysis. This need of integration between systems is important for a wise policy decisions.

This study aims to design a comprehensive and general spatial data model that could allow the integration and visualization of traffic components and measures. The activity is focused on the case study of 5T Agency in Turin, a TOC that manages traffic regulation, public transit fleets and information to users, in the metropolitan area of Turin and Piedmont Region.

In particular, the agency has set up during years a wide system of ITS technologies that acquires continuously measures and traffic information, which are used to deploy information services to citizens and public administrations. However, the spatial nature of these data is not fully considered in the daily operational activity, with the result of difficulties in information integration. Indeed the agency lacks of a complete GIS that includes all the management information in an organized spatial and “horizontal” vision.

The main research question concerns the integration of different kind of data in a unique GIS spatial data model. Spatial data interoperability is critical and particularly challenging because geographic data definition in legacy database can vary widely: different data format and standards, data inconsistencies, different spatial and temporal granularities, different methods and enforcing rules that relates measures, events and physical infrastructures.

The idea is not to replace the existing implemented and efficient system, but to built-up on these systems a GIS that overpass the different software and DBMS platforms and that can demonstrate how a spatial and horizontal vision in tackling urban mobility issues may be useful for policy and strategies decisions.

The modelling activity take reference from a transport standards review and results in database general schema, which can be reused by other TOCs in their activities, helping the integration and coordination between different TOCs.

The final output of the research is an ArcGIS geodatabase, tailored on 5T data requirements, which enable the customised representation of private traffic elements and measures. Specific custom scripts have been developed to allow the extraction and the temporal aggregation of traffic measures and events.

The solution proposed allows the reuse of data and measures for custom purposes, without the need to deeply know the entire ITS environment system. In addition, The proposed ArcGIS geodatabase solution is optimised for limited power-computing environment.

A case study has been deepened in order to evaluate the suitability of the database: a confrontation between damages, detected by Emergency Mapping Services (EMS), and Traffic Message Channel traffic events, has been conducted, evaluating the utility of 5T historical information of traffic events of the Piedmont floods of November 2016 for EMS services.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 General context

Urban transport systems are strictly connected to the general socio-economic growth of urban areas. While urban transport allows more and more people to move, increasing the quality of life, it generates serious problems that affect human activities and environment.

The main problems that affect urban transport are both related to private traffic movements and public transport supply. In particular, the high congestion level of private traffic impacts on travel times and on vehicles parking availability, may result in accidents that reduce the security perception for low mobility users, and has a negative impact on natural and urban environment, with high levels of noise, reduction of natural and public spaces and greenhouse gas emissions. On the other hand, public transport systems are often lacking, in both peak-hour, due to congestion, and in off-peak hours, due to insufficient supply. The inadequate supply of road networks and public transport networks may cause accessibility difficulties and social exclusion, with significant effects on the citizens quality of life.

To cope with these problems the concept of “Sustainable Transport” has emerged in last years: since 2006, the European Commission has included the concept in its main objectives for the European development strategy (Council of the European Union, 2006). The *Sustainable Transport* concept aims to “ensure that our transport systems meet society’s economic, social and environmental needs whilst minimising their undesirable impacts on the economy, society and the environment". The European White Paper published in 2011 has stressed on this strategy, in order to offer high quality mobility services using resources more efficiently at once (European Commission, 2011). Between initiatives reported in the White Paper, the need of a core network of strategic European infrastructures emerges, to be achieved through the deployment of large-scale intelligent and interoperable technologies, in order to optimise the capacity and the use of infrastructures. In this context, **transport data integration** is central. Data integration concerns both information about physical infrastructures and information on the status of the infrastructure (level of service, traffic flow, timetables …), in particular between administrative levels.

Analysis and management of transport and mobility is a multidisciplinary activity: geographic and spatial perspective always supports specific transport related methods. Nowadays, the increasing capacity of producing and acquiring a large volume of spatial data, including flows and movements of passengers and freight, lead to specific spatial questions. Technologies as Geographic Information Systems (GIS) and Intelligent Transport Systems (ITS) are becoming a key factor to manage this data flow.

In urban and metropolitan context, Traffic Operations Centres (TOCs) uses GIS and ITS technologies to tackling urban mobility issue. ITS tools in particular, turn data into information on which a large number of services rely on: monitoring and forecasting the traffic status, routing engines for user-oriented information services, analysis and reports to inform public administration in order to deploy new mobility strategies.

Usually in Traffic Operation Centres, various isolated systems are maintained in parallel (stored in different databases), and data comes from different sources: a challenge in transport management is to transfer disparate data into a unified data management system that preserves access to legacy data, allowing multi-thematic analysis. This need of **integration** between systems is important for a wise policy decisions. The creation of an integrated and efficient transport system represents indeed the solution for organizing and monitoring mobility across different modes, in order to increase the sustainable transport through the creation of new opportunities for collective mobility, enhancing TOCs supervision capacity and management efficiency, and users' accessibility.

1.2 Aims and content of the study

This study concerns the activities of 5T Agency in Turin, a Traffic Operation Centre that manages traffic regulation, public transit fleets and information to users, in the metropolitan area of Turin and Piedmont Region.

In particular, the agency has set up during years a wide system of ITS technologies that acquires continuously measures and traffic information, which are used to deploy information services to citizens and public administrations. 5T has developed lot of innovative ITS implementations and a specific communication protocol for real-time information, which work in parallel to guarantee an adequate level of services (e.g. S.I.MO.NE, http://simone.5t.torino.it/).

Sources of information in the context of 5T vary widely: real-time and dynamic data about traffic events (accidents, road works, delays…), flow and speed measures over the road network, prediction and analysis on traffic flow from specific software platform, real-time tracking of public transport fleets, are some of these. The spatial nature of these data is not fully considered in the daily operational activity, mainly managed through ITS components, with the result of difficulties in information integration. Indeed the agency lacks of a complete GIS that includes all the management information in an organized spatial and "horizontal" vision.

GIS solutions can integrates TOCs data and methods, in a way that other systems do not offer. GIS can be used as a logical and physical data integrator of all types of data necessary to the TOCs operational activities. However, TOCs can be easily frustrated by the high cost of GIS implementation and data maintenance: as highlighted by D. J. Buckley (1997), the availability of software tools, disks spaces, server infrastructures, updated data and general performance of the systems are some of the reasons of the difficulty to exploit such solutions. It has also to be underlined that GIS has generally high initial implementation costs but long-term profit expectations, as shown in Figure 1.

The main idea is to develop a spatial database model built on top of the existing efficient systems, without replacing them. This solution allows to build a GIS that overpass the different software and DBMS platforms, and may be therefore generally applied in other TOCs contexts, helping the data integration and coordination between different mobility agencies. The goal is to demonstrate how a spatial and horizontal vision in tackling urban mobility issues may be useful for policy and strategies decisions.

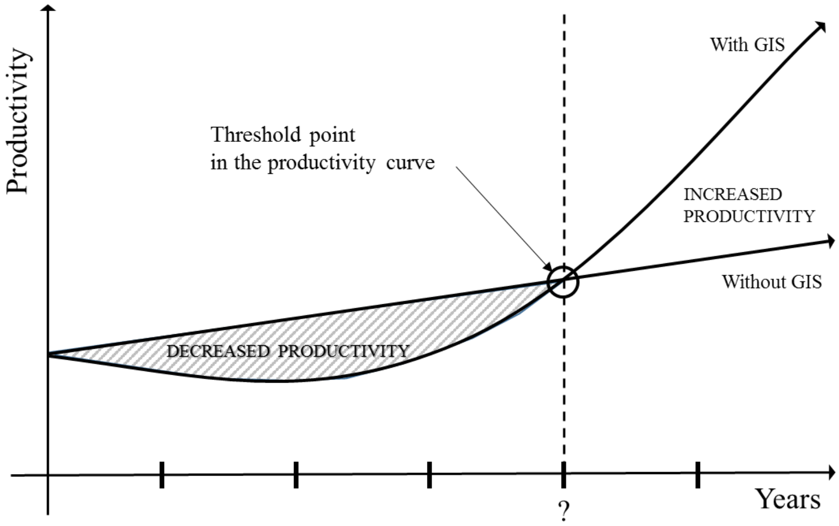


Figure 1: Typical situation that occurs with respect to comparing long-term productivity with, and without, GIS technology. Readapted from D. J. Buckley, 1997.

In order to design a spatial database, the different nature of a wide range of transport data must be taken into account. Spatial data interoperability is critical and particularly challenging because geographic data definition in legacy database can vary widely: different data format and standards, data inconsistencies, different spatial and temporal granularities, different methods and enforcing rules that relates measures, events and physical infrastructures.

Some of the issues to which this research work want to respond are:

- Which is the spatial and temporal quality of transport data in terms of precision, accuracy, spatial and attribute completeness?
- Is it possible to map all physical infrastructural elements (roads, sensors, cameras, parking…) and their related information and measures (speed, flows, videos, events…) in an integrated manner from a spatial point of view?
- What are the needs for data to accomplish such integrations?
- How geographic data standards can helps in data integration?
- What results can reward the effort to produce a GIS enterprise system?

The overall structure of the thesis takes the form of eight chapters, including this introductory chapter. The “General Concept” Chapter presents an overview of the GIS and GIS applications for transport in order to define a common vocabulary for this research: general GIS concepts are introduced, with particular attention to data formats and database implementation, and then typical GIS methods for transport applications are described. Chapter 3 is devoted to international standards in the field of transport and in particular in its spatial representation. The chapter ends with a comparison of the presented models considering geometry, semantics and network as well as fitness for temporal representation. In Chapter 4, a description of road network datasets available in this research context has been conducted: at the end, a simple comparison (total km, number of features, accuracy, attributes, topological correctness) between data is achieved, in order to evaluate possible usability for the tasks of a TOC. The Chapter 5 presents the 5T company environment: the sensors in use, the services provided by the company, a general presentation of the ITS software environment. The Chapter ends with a catalogue of 5T data used in this research. Chapter 6 is devoted to the modelling activity: conceptual, logical and physical data model are described as results of the research activity. Last, the large amount of complex transformations needed to populate the extensive model have are presented. Chapter 7 presents a set of possible applications that can be performed using the data model proposed. In addition, a proposal for integration of the traffic events within the operating procedures of the Copernicus EMS-Mapping is described as a case study. Chapter 8 briefly summarises the work, highlighting the contribution of the research, discussing similarities and differences with the standard models and the possibility of reuse of the proposed model by other TOCs. Finally, lacks and further developments of the research are described.

Chapter 2

General Concepts

This chapter provides the basic context of the research, through the identification of a common language and terminology, and reviews GIS techniques for transport applications.

2.1 GIS Fundamentals concepts

The concept of GIS traces its roots starting from research initiatives in the US, Canada and Europe during the late 1950s, but is widely acknowledged that the first real GIS was the Canada Geographic Information System set up for the Canada Land Inventory (CLI) in 1960 (Tomlinson and Toomey, 1999).

Several definitions of Geographic Information System have been proposed, each one emphasising on various aspects of GIS. Most definitions argues that GIS comprise three main components (Dickinson and Calkins, 1988): the GIS technology (hardware and software), the GIS database (geographical and related data) and the GIS infrastructure (staff, facilities and supporting elements). Below some definitions that underline different perspectives of GIS.

Following Tomlin (1990) a GIS is “a facility for preparing, presenting, and interpreting facts that pertain to the surface of the earth […], a configuration of computer hardware and software specifically designed for the acquisition, maintenance, and use of cartographic data”. This definition stresses the cartographic perspective and map view purpose of a GIS.

Chorley (1987) defines GIS as a computer-based data management system for storing, editing, manipulating, analysing, and displaying geographically referenced information. A user, viewing different features together on the same map, can answer questions about the spatial relationships between features, such as proximity, adjacency, containment and connectivity. Chorley underlines the spatial analysis view perspective that a GIS can allow. Huxhold (1991) also endorse the importance of the spatial analysis view, specifying how the topology structure of GIS data represents the key for spatial analysis.

Finally, ESRI (1990), one of most prominent commercial vendor of GIS software, define a GIS as "an organized collection of computer hardware, software, geographic data, and personnel designed to efficiently capture, store, update, manipulate, analyse, and display all forms of geographically referenced information".

2.1.1 Spatial data introduction

All GIS software have been designed to handle spatial data. Spatial data (also called geospatial data) is how geographic information is captured in a GIS.

Vector and **raster** data are the two primary data types used in GIS. These data are characterized by a spatial reference system, which defines the coordinates, tolerance and resolution, allowing the correct positioning of a geographical object on the Earth. All locations and shapes can be defined in terms of x and y coordinates from a given grid system: these numerical values are used to translate map information into digital form.

Vector data are based on a coordinate-based data model that allows the representation of vertices and paths. Three types can be defined: polygon, line (or arc) and point data (ESRI, 2017 a). The decision to choose vector points, lines or polygons for the representation of certain geographic element is governed by the cartographer and scale of the map.

Point is most commonly model used to represent non-adjacent features and to represent discrete data points. Their geometry is defined as pairs of xy coordinates. Points are also used to represent abstract points, likes city locations or place names at global scale. *Line* is used to represent linear features. Common examples would be rivers, trails, and streets. Line features have a starting and ending point. *Polygon* is used to represent areas such as the boundaries of a city, lakes, or forests. When a set of vertices are joined in a particular order and closed (start and end node must be the same), they become a vector polygon feature.

Raster data (also known as grid data) represents surfaces. The raster spatial data model defines space as an array of equally sized cells arranged in rows and columns, and composed of single or multiple bands. Each cell contains an attribute value and location coordinates (ESRI, 2017 a). Raster models are useful for storing data that varies continuously in space, as in an aerial photograph, an elevation surface or a satellite image. The cell size defines their spatial accuracy.

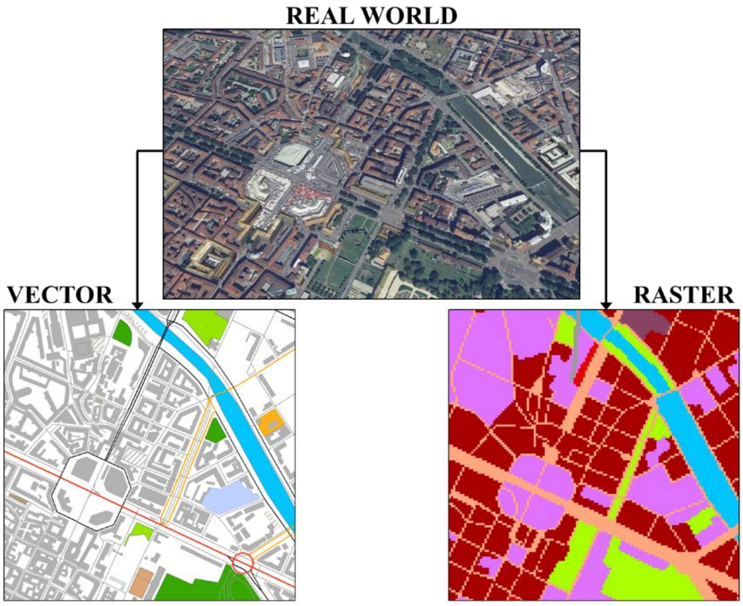


Figure 2: The vector and raster model of the reality

Raster data models can be *discrete* or *continuous*. Discrete raster have definable boundaries, likes land cover classification raster. Discrete raster are also referred to as thematic or categorical raster data. Continuous raster are grid cells with gradual changing data such as elevation, temperature or a satellite imagery. Continuous data is also known as non-discrete or surface data.

Vector data has some advantages respect to raster ones, like the possibility to add logic to the conceptual model: rules for topology and connectivity can be defined in order to perform analysis or define a behaviour. In addition, vector data allows for visually smooth and easy implementation of overlay operations, especially in terms of graphics and shape-driven information like maps, routes and custom fonts, which are more difficult with raster data. Vector data can be easier to register, scale, and re-project, simplifying the combination of vector layers from different sources. Vector data are also more compatible with relational database environments, where they can be part of a relational table and processed using a multitude of operators, allowing an easy maintenance. However, data analysis with raster data is usually quick and easy to perform. With map algebra, quantitative analysis is intuitive equally with discrete or continuous raster, but graphic output and quality is based on cell size so it can have a pixelated look and feel (see Figure 2).

Spatial data can be stored in various **formats**, different for raster and vector data. The easy way to store this kind of information are text files, where by including a numeric field for *xy* coordinates, it is possible to associate references to point entities in a text file. A specific spatial referencing system must be specified in the metadata in order to relate data to specific place on the Earth.

Regarding *vector data* one of the first format developed are DWG and DXF, proprietary formats of the AutoDesk Corporation. The ability to attach semantic information to geometry in these formats is very limited and fixed. These formats are also very flexible in terms of the ability to mix all sorts of geometry types (point, line, polygon) in a single dataset, which can make it difficult to create suitable models from a GIS perspective, where usually each geometry types is stored in separated file.

Table 1: File formats which compose a shapefile, readapted from ESRI, 1998.

File format	**Description**
.shp	Shape format: the feature geometry itself
.shx	Shape index format: a positional index of the feature geometry to allow seeking forwards and backwards quickly
.dbf	Attribute format: column attributes for each shape, in dBase IV format
.prj	Projection format: the coordinate system and projection information, a plain text file describing the projection using well-known text format
.sbn and .sbx	Spatial index of the features
.shp.xml	Geospatial metadata in XML format, such as ISO 19115 or other XML schema

The **shapefile** (ESRI, 1998) is a digital vector storage format for storing geometric location and associated attribute information. It is a format developed by ESRI as a (mostly) open specification and nowadays is the most common format for sharing vector data. A shapefile is actually a set of several files (located in the same folder), which are described in Table 1. Each shapefile stores only one type of vector geometry (polygon, line or point). Advantages in using shapefiles are due to light structure, which not support topological geometries enhancing drawing speed and edit ability respect to other data sources.

Table 2: List of common formats for the storage of vector and raster data.

Raster Formats

Binary file	An unformatted file consisting of raster data written in one of several data types, where multiple band are stored in BSQ (band sequential), BIP (band interleaved by pixel) or BIL (band interleaved by line). Georeferencing and other metadata are stored one or more associated files.
ECW	Enhanced Compressed Wavelet (from ERDAS). A compressed wavelet format.
ESRI grid	Proprietary binary and metadata ASCII raster format, used by ESRI.
GeoTIFF	TIFF variant enriched with GIS relevant metadata.
IMG	ERDAS IMAGINE image file format.
JPEG2000	Open-source raster format. A compressed format, allows both lossy and lossless compression.
netCDF-CF	netCDF file format with CF medata conventions for earth science data. Binary storage in open format with optional compression. Allows for direct web-access of subsets/aggregations of maps through OPeNDAP protocol.

Vector Formats

AutoCAD DXF	Contour elevation plots in AutoCAD DXF format (by Autodesk).
ESRI TIN	Proprietary binary format for Triangulated Irregular Network data used by ESRI.
Geography Markup Language (GML)	XML based open standard (OpenGIS) for GIS data exchange.
GeoJSON	A lightweight format based on JSON, used by many open source GIS packages.
GeoMedia	Intergraph's Microsoft Access based format for spatial vector storage.
ISFC	Intergraph's MicroStation based CAD solution attaching vector elements to a relational Microsoft Access database.
Keyhole Markup Language (KML)	XML based open standard (OpenGIS) for GIS data exchange.

MapInfo TAB format	MapInfo's vector data format using TAB, DAT, ID and MAP files.
Spatialite	Is a spatial extension to SQLite, provide vector geodatabase functionality. It is similar to PostGIS, Oracle Spatial, and SQL Server with spatial extensions.
Shapefile	A popular vector data GIS format, developed by ESRI.
Spatial Data File	Autodesk's high-performance geodatabase format, native to MapGuide.
GeoPackage (GPKG)	An standards-based open format based on the SQLite database format for both vector and raster data.

Vector files can also be stored in a database with a specific **spatial extension**, which defines how geometries are stored and specific function to perform spatial operation on the data. Databases spatial extensions allow the application of specific behavioural rules.

Raster data is stored in various formats: from a standard file-based structure of TIFF, JPEG, etc. to binary large object data (BLOB) stored directly in a relational database management system. The Table 2 show a list of the most common formats for the storage of vector and raster data.

2.1.2 Database and Spatial Databases

As highlighted in the introduction of this Chapter, lot of attention is given to the data structure of the geographic data. In particular, the management of geographic information is demanded to database structure that have several advantages in comparison with flat file data format listed in previous section.

From a conceptual point of view, a database is essentially an abstract representation of things in the real world. From a technical point of view, a database can be defined as one or more structured sets of persistent data, managed and stored as a unit and generally associated with software (Database Management Systems - DBMS) to update and query the data (Butler, 2008).

There are several type of database model. The most prominent is the **relational database model**, invented by Dr. Edgar Codd in early 1970s at IBM. Such databases management system is based on relational algebra, a kind of math that controls what can happen to the data in such storage structure. Relational algebra supports several functions: read data, write data, define virtual relations (table views), create snapshot relation, define and implement security rules, establish and meet stability requirements, operate under integrity rules (Date, 2000).

The *Structured Query Language* (SQL) is the declarative language developed for *manipulate* and *retrieve* data in relational database management systems, which allow to support the relational algebra functions previously listed. From the 1986 is become a standard, developed by ISO/IEC 9075, though usually every database management systems have its own implementation.

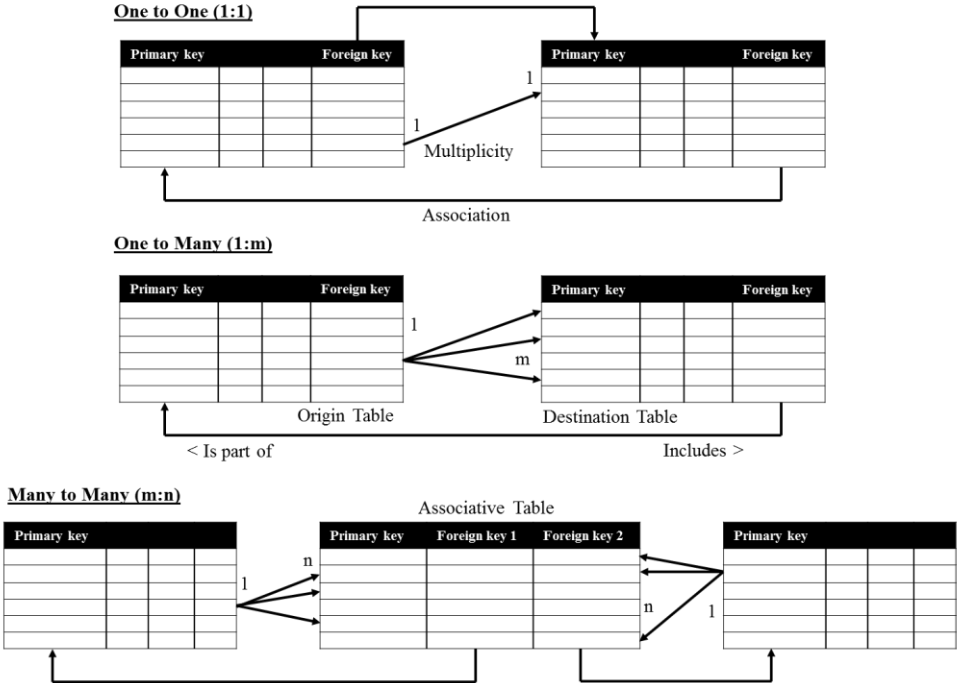


Figure 3: Association and cardinality between tables.

The big advantage offered by relational database is the ability to represent and manage relationships between tables: those relationships ensure the right modelling of the behaviour of the objects represented by tables (Butler, 2008). An **association** between tables is established by placing the same column or a set of columns in both tables (see Figure 3), or creating an association table that links the common columns. Associations are characterized by a *role*, which clarify the mean of the relationship ("is part of" in Figure 3), and by *cardinality* (or multiplicity), which defines the correspondence or equivalency between sets of data: for one or many rows in a table, there can be zero, one or many rows in the related table.

A more modern paradigm in database modelling is the **object oriented/object relation database**. The Object Oriented Paradigm (OOP) is a software programming approach, and allows writing programs that can be easily reused and maintained. Object-oriented software encapsulates data and the software that uses the data into an object class. In particular, the OOP approach thanks to its attention to requirements, and passing through concepts like objects and classes, forces to deepen the vocabulary and problem domain (Booch, 1994). The *encapsulation* is one of the main principles of OOP: users can view and manipulate data as in traditional relational DMBS without being aware of the structure of the object class that is exposed through *interfaces,* which defines views, behaviours and operations on the object class. This approach allows programmers to evolve the internal data structure without affecting the view for the users, minimising interdependency among separately written modules (Snyder, 1986).

The OO paradigm is built on the concepts of **Objects** and **Classes**: a *class* describes a set of *objects*, with specific common properties (or attributes) and behaviours (or methods). A single object is also called an *instance* of a class. To make a comparison with the relational model, objects can be seen as the rows and the class as the table, the methods instead are the rules applied to the table. The OOP allows, in addition of the capabilities offered by traditional relational databases, another common and important relationship called *inheritance*, a parent-child relationship (“is a” relationship among classes). An example of OO database is the ArcGIS geodatabase.

A **spatial database** is a database that is optimized to store and query data that represents objects defined in a geometric space. Most spatial databases allow representing simple geometric objects such as points, lines and polygons. Some spatial databases handle also more complex structures such as 3D objects, topological coverages, linear networks, and TINs.

Spatial databases require additional functionality to process spatial data types efficiently: in particular, the Open Geospatial Consortium[1] (OGC) developed the *Simple Features Access* (first released in 1997), now part also of ISO standards (ISO 19125), that specifies a common storage and access model of geometries used in GIS. The first part of the standard describes how to model and store the geometry and its representation, defining the *Well-Known Text* (WKT) and *Well-Known Binary* (WKB) data types, used by DBMS to store the geometry column.

---

[1] The OGC (Open Geospatial Consortium) is an international not for profit organization committed to making quality open standards for the global geospatial community. Further explanation can be found in Chapter3.

The second part defines the SQL implementation for retrieve and manipulate geographic data. The main spatial operations added to SQL by the standard are:

- *Spatial Measurements*: allow to compute line length, polygon area, distance between geometries, etc.
- *Spatial Functions*: allow to modify existing features to create new ones, for example by providing a buffer around them, intersecting features, etc.
- *Spatial Predicates*: allows true/false queries about spatial relationships between geometries.
- *Geometry Constructors*: allow to create new geometries, usually by specifying the vertices (points or nodes) which define the shape.
- *Observer Functions*: allow queries that return specific information about a feature such as the location of the centre of a circle.

Database management systems usually use *indexes* on attributes to quickly look-up values. However, these indexes are not optimal for spatial queries. Instead, spatial databases use a *spatial index* to speed up database operations. A common spatial index method is *R-tree*, typically the preferred method for indexing spatial data: spatial objects are grouped using the Minimum Bounding Rectangle (MBR). Objects are added to an MBR within the index that will lead to the smallest increase in its size (see Figure 4).

Using a spatial database for storing and managing geographic data substantially allow more advanced analysis and techniques. A DBMS indeed allows adding behaviours and rules to manage spatial relationships and integrity (e.g. topology, geometry networks), and provides, through a client/server architecture, the support for concurrency access to the data, in order to view, query and edit the data without conflict. Other advantages that a spatial database storage can assure (in opposition to local file storage) regards the increased scalability (big data volumes), managing users and permissions, increased performances, replication and versioning.

Some of the most common DBMS that implements a spatial extension are:

- Oracle, with Spatial for Oracle DB;
- Microsoft SQL Server with support for spatial types (since version 2008);
- SQLite with SpatiaLite;
- PostgreSQL with PostGIS;
- ESRI Geodatabase.

In the next section, an introduction to data modelling and UML language is given, in order to deepen in the following section the ESRI geodatabase model, as an example of spatial object-relational database.

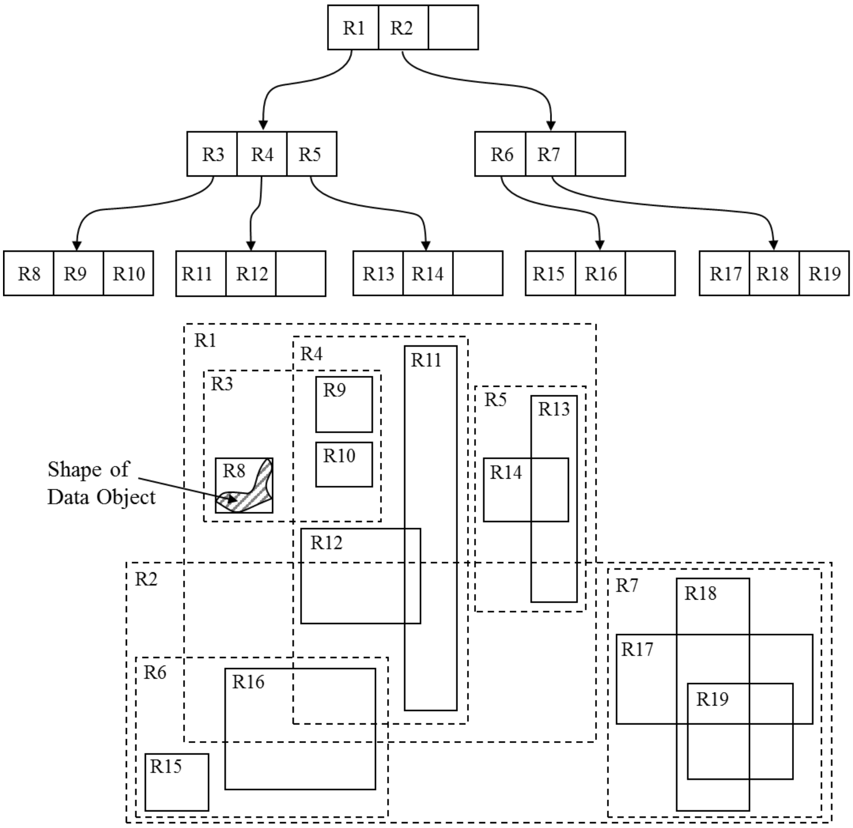


Figure 4: R-Tree index principles, readapted from Guttman, 1984.

2.1.3 Data Modelling

As already stated, data modelling is a structured process that allow to abstract the reality in the most appropriate way to respond to requirements for an application. In this section, the general methodology used in this research to build a database model is described.

The Agile methodology is an approach to develop and maintain information systems. Main characteristic (Qumer and Henderson-Sellers, 2008) regard the use of iterative and people-focused approach, which ensures a design and development process characterised by better communication and flexibility, focused on single-step objectives and deliveries and easy to be readapted.

Following the Agile methodology, six main steps can be defined in order to produce a good data modelling process (Butler, 2008; Ambler 2003):

1. Define the user requirements – define the purpose of the data and the applications requirements to be supported;
2. Develop a conceptual data model – identify the basic elements and relationships between elements of the database that meets the requirements;
3. Develop a logical data model – specify attributes and eventually redefine elements and relationships;
4. Develop a physical data model – define the structures of single tables, domains values and implicit and explicit relationships;
5. Test the data model – realize the implementation of the database and test functionality and requirements;
6. Product implementation – Create a default database and load data.

Conceptual data modelling concerns the activity to identify entities and relationships between entities, without including attributes. Conceptual data modelling translates the requirements into data structures. The elements must be defined in an unambiguous form.

Logical data modelling expands and specifies the elements defined in the conceptual data model: the UML[2] is wide used in this phase. In the logical data modelling phase, also abstract class are used, in order to underlines inheritance relationships between elements. Attributes are added to elements, and relevant attributes that enables relationships between elements are explicated. Enumerations, that are list of possible values for specific attributes are also added.

In the *physical data modelling* phase relationships may be implied by foreign keys or explicitly included as relationship classes. Enumeration are converted into domain values and many to many associations are defined through associative tables.

As support to database design, the *Unified Modelling Language* (UML) is widely used in this research. Using UML to document and design a database schema has several advantages: improves communication, improves interoperability between systems, reduces implementation costs in case of changes during the design process and allows the reuse of the general implementation.

---

[2] UML - "Unified Modelling Language" is a standard modelling language.

The UML is an ISO/IEC official standard (ISO, 2005). The UML may be used in a variety of ways to support a software development methodology, but in itself does not specify a methodology or a process. In fact, the standard proposes various ways to visualize design diagrams, in different domains (Ambler 2003): describing classes and objects that will make up the system, but also interactions, states, conditions, workflows, software and/or hardware components or physical architecture and deployment of components.

In this research context, the focus is on *logical and class model*, which shows the building blocks of the system describing attributes and behaviours. The model allows to illustrate clearly relationships between classes, as generalizations, aggregations, and associations.

In Figure 5, a class model in UML is exemplified, showing the notation of main concepts. As already introduced, *inheritance* is one of the core concept of OOP. Indeed, inheritance simplify logical data modelling allowing to omit the repetition of the *parent* class attributes in the *child* class. In data models, the child class includes only attributes that have been redefined or added to the parents. Parents classes are often called *stereotypes*. Many stereotypes are *abstract*, which means that their purpose is solely to serve as class template. In Figure 5, the class Car represents an abstract class (italic font) that serves as stereotype for the child classes CityCar and SUV (stereotype quoted). Child classes have their own added attributes respect to their parent class Car. Other important relationships that can be used in a class model are:

- *Dependency*: a class can depends on other classes for its existence, or rather one class *instantiates* the other (a function of the class creates objects of the dependent class);
- *Composition*: a class is composed of one or more instances of other classes, and if the sum of the components is deleted, also the component class must be deleted;
- *Aggregation:* a class is a collection of other classes, but deleting the collection not affect the single class component.

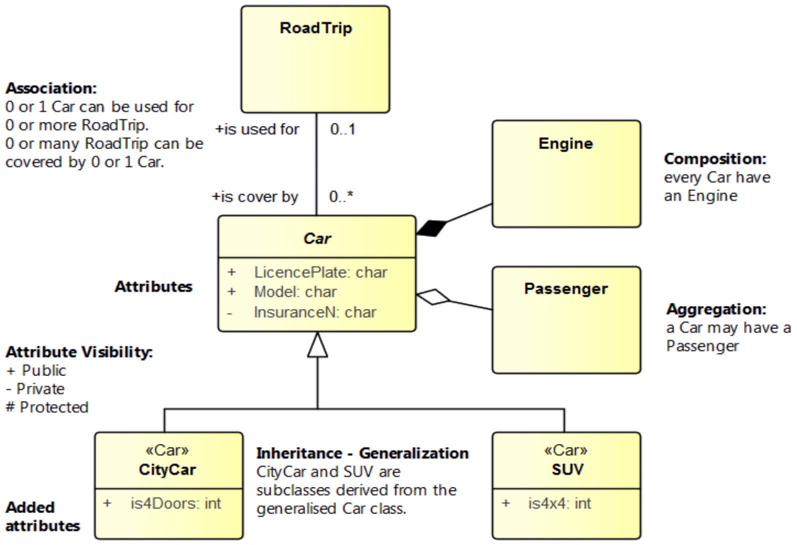


Figure 5: UML notation of OOP class model.

2.1.4 The ESRI Geodatabase

The ESRI ArcGIS software model is an example OOP implementation: the resultant data structure appears as a relational one, but there is more behind it.

The geodatabase is a collection of *ArcObjects* that interacts with each other through interfaces. In general, ESRI restricts the use of objet class to mean a type of table that stores non-spatial objects. An object that includes a shape attribute (geometry) is called feature. A **feature class** adds the software logic and the data structure needed to store and retrieve geometry to the object class: the user will see only the shape attribute, but behind there is lot more.

The geodatabase as OO data model can be represented through a conceptual model. In Figure 6, are shown the main classes that comprise also the geodatabase (abstract class are indicated with italic font). At the top of the diagram, the *Workspace-Factory* instantiates a *Workspace* (dependency relationship between abstract and concrete class). There are three main kind of workspace: file systems (shapefiles and coverages), local databases (personal and file geodatabases), and remote databases (SDE, SQLServer, PostgreSQL and compatible other DBMS). A *Dataset* is an abstract class that represents a named collection of data within a workspace and provides access and edit functions over the data; a workspace must include at least one dataset. A dataset may have a *PropertySet* and *Domain classes*. *Domains*, in particular, are created at workspace level and can be applied to user-supplied attribute values in any dataset that exists within the workspace.

The domains are rules that limits user's data entry to a specific set of valid values, improving data quality and integrity.

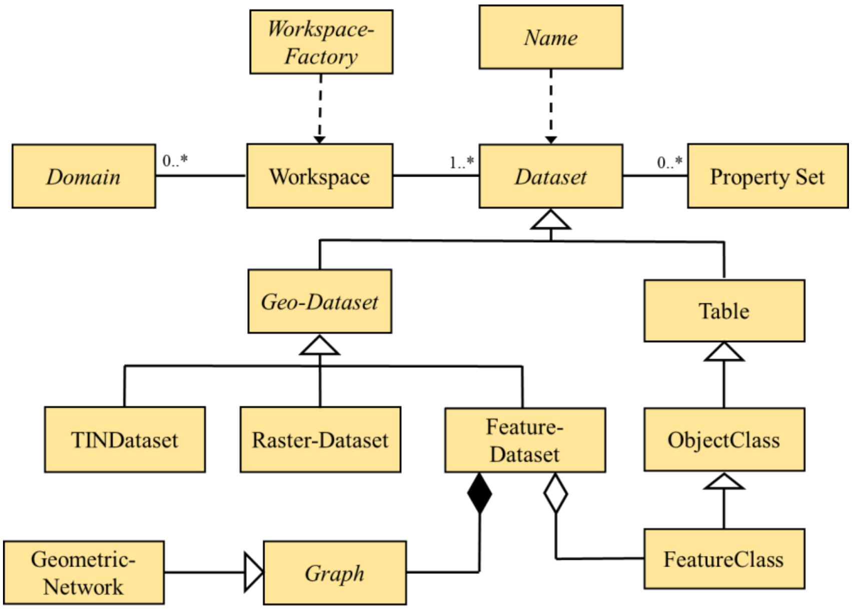


Figure 6: The geodatabase. This conceptual data model shows the core components of the geodatabase support structure, readapted from J.A. Butler, 2008

The *Table* class provides the basic data storage mechanism of a relational table consisting of columns (fields), and an unordered collection of rows containing table members. Since it is a child of *Dataset*, it inherits all the properties and methods of the parent class. The *ObjectClass* adds the necessary properties and methods to turn a relational table into an entity in your geodatabase. *ObjectClass* adds the OBJECTID primary key field to the relational table it represents. Through this series of inheritance relationships, *FeatureClass* has all the interfaces of *Dataset*, *Table*, and *ObjectClass*. A feature class is just an object class that includes also a geometry property and methods to work with it; features classes can have *subtypes*, which can be used to differentiate the control object behaviour inside the same class, usually improving performances.

The abstract class *Geo-Dataset* is specialized in *TINDataset*, *Raster-Dataset* and *Feature-Dataset*, the container for each type of geometry format. The *Feature Dataset* is an aggregation of *FeatureClass* classes and may include *Geometric-Network*. In particular, geometric networks are special feature classes that contain feature attributes, geometries and network topologies (connectivity rules). A particular specification of geometric network are Network datasets, used to represent transport systems for pathfinding applications.

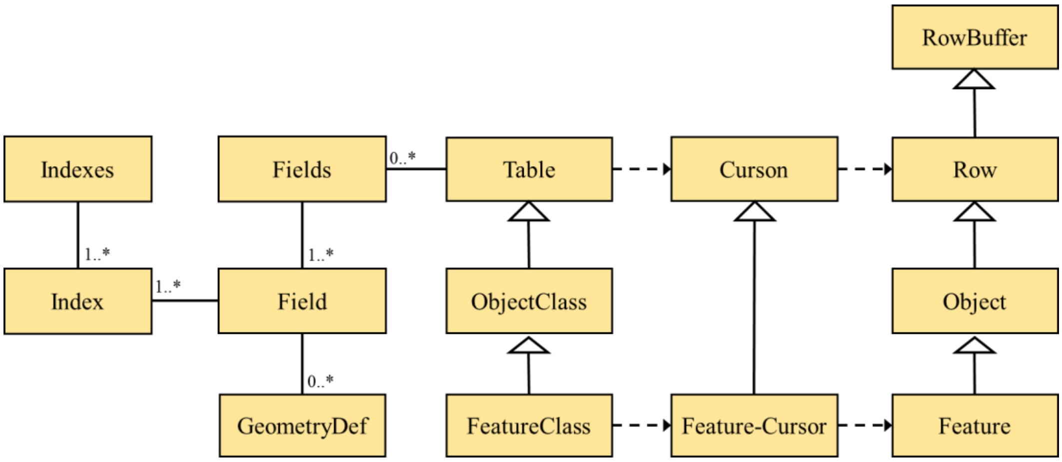


Figure 7: The ArcGIS engine. This conceptual data model shows how the *Table* class manages fields and rows, readapted from J.A. Butler, 2008

In addition, ArcGIS provides classes that help the *Table* class to manage data: fields, rows, cursors, indexes and geometry definitions represent central classes that govern the creation and access to spatial data, or the engine of ArcGIS.

Looking at Figure 7, a *Table* is a set of columns (fields) and unordered rows of data values. The *Fields* class stores the total number of fields in the table. A *Field* class specifies each field. As all rows in a table have the same fields, the *Table* class uses the *Field* class to understand how the row can be managed. In case of *FeatureClass* (which are Table child), also the class *GeometryDef* is added among fields, defining the geometry type. The schema shows also *Indexes* class: it manages all the indexes that exists for a *Table*. In general, ArcGIS automatically add an index on the OBJECTID field, but the user can eventually add other indexes.

Rows are created and retrieved through the class *Cursor* (or *Feature-Cursor*, a specialization of a *Cursor* used in *FeatureClass* class management): it allows to manipulate records, and in particular manages the retrieving, creating, modifying or deleting of rows. The *Table* class instantiates the *Cursor* class to work with instances of the *Row* class. In the same way, the *FeatureClass* object works through a *FeatureCursor* class to access *Feature* instances.

This structure allows to define a data structure (the table) independently from its real data container, which can be a shapefile, file geodatabase or an ArcSDE instance, managing a relational database (Butler, 2008).

The object-oriented foundation of the geodatabase offers advantages over a purely relational database, as the application logic supplies predefined behaviours for data objects, domain checking and rule enforcement, which must be manually coded in other spatial DBMS. In addition, the separation of the application logic from the storage is what allows support for many different DBMS and data formats.

2.2 GIS for Transport

Transport is inherently a geospatial activity, involving the movement of people and/or freights from one geographic location to another. Not surprisingly, therefore, much of the data needed to support transport-planning, operations, and policy decisions include location as a key attribute.

The need for reliable data and information has motivated and favoured the application of GIS to transport systems (Thill, 2000). GIS for transport (GIS-T) denotes a specific expression that encompasses all the activities that use GIS for some aspects of transport planning and management (Curtin et al., 2003). The applications of GIS-T research has led to significant improvements in the general GIS techniques.

In general, topics related to GIS-T studies can be grouped into three categories, even if interdependent one to each other (Rodrigue, 2017):

- *Data representation*: how can various components of transport systems be represented in a GIS data model?
- *Analysis and modelling*: how can transport methodologies be used in a GIS?
- *Applications*: what types of applications are particularly suitable for GIS?

Data representation is a core research topic of GIS-T. Some transport problems tend to fit better with one type of GIS data model than the other. For example, network analysis based on the graph theory typically represents a network as a set of nodes interconnected with a set of links. The object-based GIS data model therefore is a better candidate for such transport applications.

GIS offers a wide framework for *analysis* and *modelling* transport problems: mobility applications have benefited from many of the standard GIS functions (query, geocoding, buffer, overlay, etc.) to support data management, analysis, and visualization needs. Like many other fields, transport has developed its own unique analysis methods and models. Examples include shortest path and routing algorithms, spatial interaction models, network flow problems, conflation and map-matching problems, facility location problems, travel demand models, and land use-transport interaction models.

GIS-T *applications* have been implemented at various transport agencies and private firms. They cover much of the broad scope of transport and logistics, such as infrastructure planning and management, travel demand analysis, traffic monitoring and control, public transit planning and operations, routing and scheduling, vehicle tracking and dispatching, location-based services (LBS).

These applications have different needs, which are directly relevant to the GIS-T data representation and the GIS-T analysis and modelling issues. When a need arises to represent transport networks of a study area at different scales, what would be an appropriate GIS-T design that could support the analysis and modelling needs of various applications?

This research is focused on GIS-T data models to allow a better data representations in support of various transport applications.

2.2.1 The network data model

Following Goodchild (1998), three classes of GIS models can be used for transport:

- *field models*, thus a representation of a continuous variation of a phenomenon over the space, usually achieved by raster data format, used for example in digital terrain model;
- *discrete models* in which discrete entities (points, lines, polygons) populate the space, using vector format, useful for representation of real geographic objects, like in topographic maps, and
- *network models*, that aims to model the connectivity properties of a real objects: its represents topologically connected linear entities, that are fixed in the continuous reference surface.

Between models in transport, **network models** are the most prominent, thanks to their simplicity, elegant form and functionality, as they allow to represent the flows of passengers, cars, freight moving through the networks lanes (Thill, 2000). Since Euler's time it has been known how to efficiently model a transport network by using graphs, as he demonstrated with the famous example of the Königsberg bridges and, following the rise of Operations Research in the 1950s and 1960s, a number of optimization problems have been successfully resolved with efficiency and elegance through graph modelling (Mathis, 2007).

A **graph** can be defined as a finite set of points (nodes or vertices) and a set of relations between these points (edges or arcs). Formally, a graph G is a triple consisting of a set of vertices $V(G)$, a set of *edges* $E(G)$ and a relationship of *incidence*, which associates an edge to a couple of vertices (or *extremities*) (Iovanella, 2009).

When a graph has no multiple arcs or loops[3] is called *simple*. Considering an arc $e = ab$, if $ab \in E(G)$ then a and b are called *adjacent*, and *incident* on the arc ab. Two arcs are *adjacent* if they share a vertex.

The *order* of a graph is the number of vertices in the set $V(G)$. The *dimension* of a graph is the number of edges in the set $E(G)$. The dimension m of a simple graph is dependent from its order n and can range between:

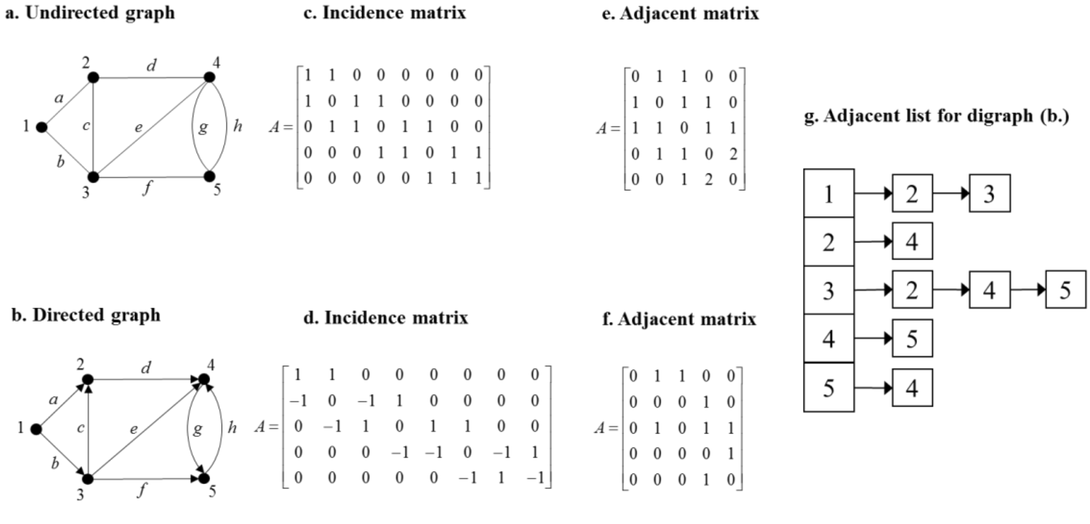
$$0 \leq m \leq \frac{n(n-1)}{2}$$

The measure of *density* of a graph represents the rate of the number of arcs with the maximum possible number of arcs:

$$D(G) = \frac{m}{n(n-1)/2}$$

The *degree* $d(x)$ of a node $x \in G$ is the cardinality of set of adiacent vertices of the node x.

A graph is *directed* (*digraph*) when the relationship of incidence is true for $e = ab$ and not for $e = ba$ (see Figure 8.a and 8.b).



a. Undirected graph

c. Incidence matrix

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

e. Adjacent matrix

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 2 \\ 0 & 0 & 1 & 2 & 0 \end{bmatrix}$$

g. Adjacent list for digraph (b.)

b. Directed graph

d. Incidence matrix

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 \end{bmatrix}$$

f. Adjacent matrix

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Figure 8: Graph types and matrix representations, readapted from Iovanella, 2009.

---

[3] A loop is an arc where extremities are coincident. An arc is multiple when the extremities are the same.

In the graph theory, the localisation of nodes is unimportant respect to the existence of a relationship between two nodes. This characteristic allows multiple matrix representations of the graph, on which efficient algorithms can be defined.

In particular, the *incident matrix* A of an undirected graph G, is the matrix of dimension $n \times m$ where each row represents a vertex (n) and each column an edge (m). The column of the edge (i, j) will have values 1 in correspondence of node i and j (see Figure 8.c). The sum of the value for each row corresponds with the degree of the equivalent node. In case of directed graph, the incident matrix for the h^{th} edge (i, j), in the correspondent column h, the element a_{ih} is equal to 1, whereas the element a_{jh} is equal to -1 (see Figure 8.d). The representation of a graph through incident matrix is not so efficient, due to the storage volume needed for store it on a computer, which increases with the growth of n and density.

The *adjacent matrix* A of an undirected graph G is the matrix of dimension $n \times n$ where each row and each column represent a vertex (n). The elements of the matrix a_{ij} are equal to 0 if the edge (i, j) not exists, else are equal to the sum of the edges that join nodes (i, j) (see Figure 8.e). This matrix is symmetric and has n^2 elements. The sum of values in a row and the sum of the values in a column indicates the degree of the correspondent node. In case of directed graph (see Figure 8.f), the matrix is no more symmetric and for each row the sum of values indicates *indegree* (the number of head ends adjacent to a vertex), whereas for each column the sum indicates the *outdegree* (the number of tail ends adjacent to a vertex). Finally, the *adjacent list* of nodes $A(i)$ of a graph G with n nodes is a vector of dimension n, in which each cell points a list that contains the set of nodes j such that $(i, j) \in E\,(G)$ (see Figure 8.g). The adjacent list of a graph optimizes calculation procedures as it removes zero values and retains only the existing arcs.

A **network** is a type of mathematical digraph that captures relationships between objects using **connectivity**, which can be modelled through incident and adjacent matrixes. The connectivity may or may not be based on spatial proximity, and is usually guaranteed by the application of topology rules. In many applications, objects are modelled as *nodes* and *links* in a network. The network model contains logical information such as connectivity relationships among nodes and links, *directions* (or *navigability*) of links, and *costs* (or *impedances)* of nodes and links. The navigability of link is particularly important in GIS-T applications, as it reflects the real-world characteristics of transport networks such as one-way streets.

The following are some key terms related to a general network data model (not necessarily referred to mobility networks) and used in this research:

- A *node* represents an object of interest;
- A *link* represents a relationship between two nodes;
- A *path* is an alternating sequence of nodes and links, beginning and ending with nodes, and usually with no nodes and links appearing more than once;
- A *cost* is a non-negative numeric attribute that can be associated with links or nodes to computing the minimum cost path, which is the path that has the minimum total cost from a start node to an end node.

Logical networks contain only connectivity information, while a *spatial network* contain geometric and connectivity information. In most of transport applications, a spatial network model is used: nodes can represent intersections and links represents streets.

In order to construct a basic network data model, two table are required (Rodrigue, 2017): a node table, which contains the unique identifier and the position (as geometry or as coordinates) of each point, and a link table, which contains the unique identifier of each link, two additional field to store the identifiers of the origin and destination nodes and a field to describe directionality.

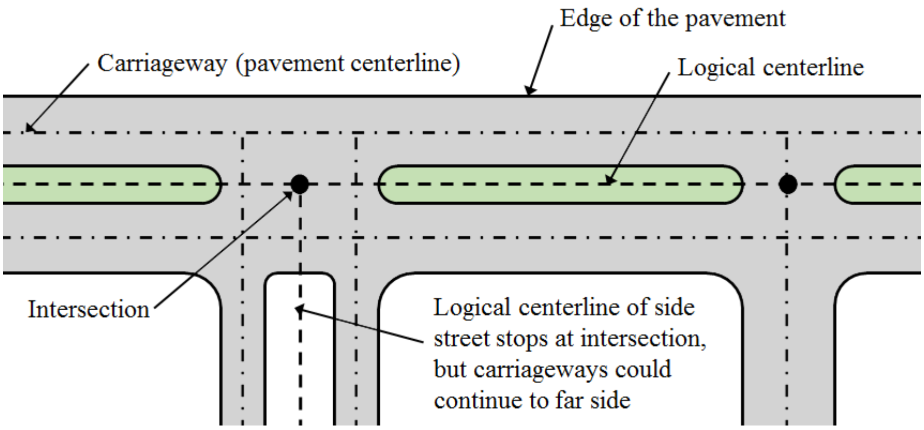


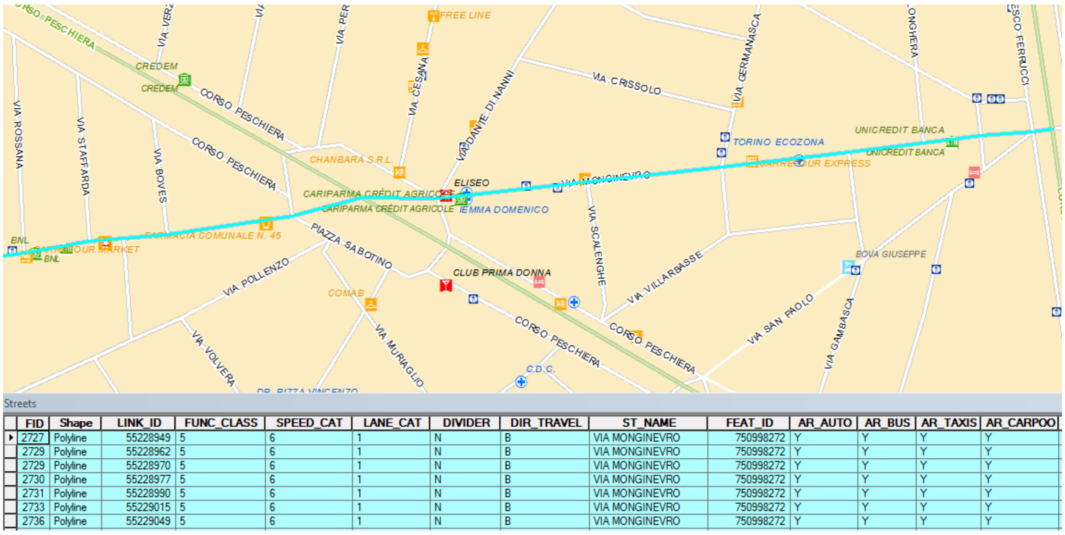
Figure 9: Road geometry representation: logical centreline versus physical centreline. Readapted from Butler, 2008.

However, as streets in reality are areas, there are multiple ways to model a street. A common abstraction uses *logical centreline* or *physical centreline* (see Figure 9). Logical centreline represents the approximate midpoint of the street across its width, in order to show the general location and shape of the facility at a relatively small scale. Physical centreline usually represents the approximate midpoint of carriageway across its width and are most used at bigger scale.

When dealing with a logical centreline, a one-to-one relationship between a facility and its representative geometry can be constructed by breaking the geometry into segments. One segment has one centreline and one set of attribute values. In case geometry represents carriageways, for each facility segment there will be one or two linear features, still manageable but with increased complexity. The complexity will increase over if the geometry representation involves edge line features (as edge of pavement), as its number can be very large. Choosing an appropriate level of abstraction that cope with the application purpose and manageable complexity is never a trivial task.

At the same time, intersections also can have different levels of abstraction and their representation is dependent from centreline abstraction used. It is usual to think to an intersection as an atomic feature in real world, but if the intersection of two divided roads can be represented by one point, in a carriageway representation, there will be four points, where the physical centrelines cross. In case of pathfinding applications for example, solutions imply the use of internal turns at intersection. Other approaches treat an intersection as a multipoint feature.

The simplest approach to making a transport dataset is to create independent segments representing linear portions of the total system (Butler, 2008; Rodrigue, 2017). A typical way to do this is to segment longer facilities at intersections. Each segment gets the same set of attributes so that values extending across a road are duplicated for all segments to which the value applies (see Figure 10). The segment attributes can be used to determine line symbology, or create separate map layers, to represent each kind of facility. Some database designers may argue that this simple design is inefficient, but it is indeed the most used in transport databases.



Streets

FID	Shape	LINK_ID	FUNC_CLASS	SPEED_CAT	LANE_CAT	DIVIDER	DIR_TRAVEL	ST_NAME	FEAT_ID	AR_AUTO	AR_BUS	AR_TAXIS	AR_CARPOO
2727	Polyline	55228949	5	6	1	N	B	VIA MONGINEVRO	750998272	Y	Y	Y	Y
2729	Polyline	55228962	5	6	1	N	B	VIA MONGINEVRO	750998272	Y	Y	Y	Y
2729	Polyline	55228970	5	6	1	N	B	VIA MONGINEVRO	750998272	Y	Y	Y	Y
2730	Polyline	55228977	5	6	1	N	B	VIA MONGINEVRO	750998272	Y	Y	Y	Y
2731	Polyline	55228990	5	6	1	N	B	VIA MONGINEVRO	750998272	Y	Y	Y	Y
2733	Polyline	55229015	5	6	1	N	B	VIA MONGINEVRO	750998272	Y	Y	Y	Y
2736	Polyline	55229049	5	6	1	N	B	VIA MONGINEVRO	750998272	Y	Y	Y	Y

Figure 10: A road split in segments with repeated attributes (e.g. “ST_NAME”).

In addition, nodes can be added at start and end of segment links, in order to construct a network for pathfinding. Furthermore, following this approach, node features can be used for storing intersection attributes, organizing traffic signal inventory, or adding impedances to cross the intersection.

2.2.2 Characteristics of GIS for transport

In this section, some general concepts on the use of georeferenced data in mobility management are introduced. GIS for Transport indeed utilizes some different concepts, nomenclature, and functionality compared to “mainstream” GIS, due to the particular issues that it has to deal with.

One of the first issues is related to the dynamic nature of the attributes that a network model can have. Transport databases usually include a very large number of segment, with a rich number of attribute table associated. Overlooking the case of the construction of a new road, many of these segments will change periodically, due to projects, maintenance works, and crash accidents that may affect capacity, navigability or other attributes. Sometimes changes can be permanent or temporary and not always they affect the entire segment, but only portions. Editing and re-segment each time the features can overload the maintenance of the network and the operational activity of a mobility management centre.

Linear referencing systems (LRS) represent the solution to these issues. LRS existed long before GIS and computers, developed by the railroad industry to identify positions on a rail line, and can be defined as “a system of determining the position of an entity relative to other entities to some external frame of reference” (Goodwin et al., 1998).

A LRS is essentially a one-dimensional coordinate system, which uses cumulative distance from a point of origin to identify the location of a point on the facility. Each facility thus becomes its own datum for stating a location along its length. Such method reduces the maintenance workload and improves the database performance, allowing to associate multiples set of attributes to a portion of a linear feature without splitting it each time the values change (Butler, 2008).

A *linear referencing method* (LRM) is the set of technical processes used to determine, specify, and recover a location within an LRS. It includes rules for managing the measurement process, as how to define the origin, the unit of measure, the precision and scale for measured distances (see Figure 11). The distance from the origin to the point of interest is called *measure*. The *route* is the facility on which measures are determined and applied. Linear referencing imply to view an attribute as a spatial (linear) *event* occurring on the network (Scarponcini, 1999). The event can be a punctual when only a measure is specified or linear when a start and end measure are specified.

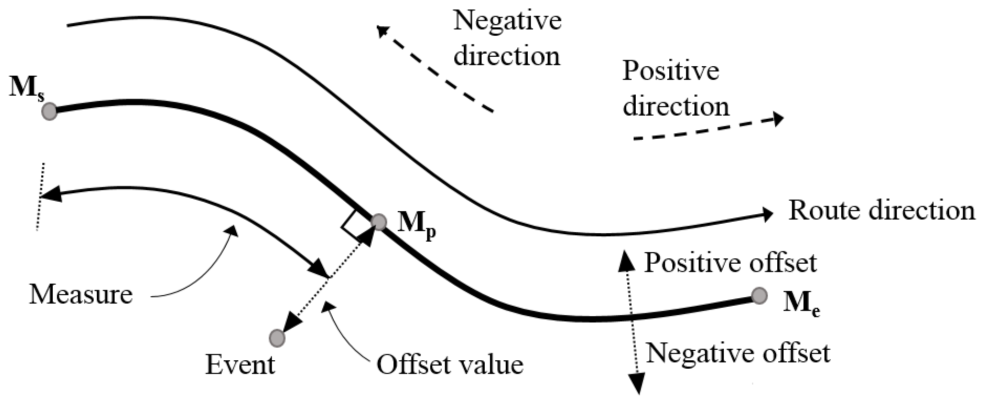


Figure 11: Linear Referencing and Dynamic segmentation.

There are literally an infinite number of methods used to describe locations on a transport network: some of methods includes mile or kilometre posts, street addresses and locational proxies such as state, county, zip code or Census Tract. The address geocoding activity may be interpreted as a linear referencing activity, where the street name represent the route and number address the point event.

Since multiple location referencing methods exist in most agencies, the ability to compare and convert among multiple LRMs is needed to enable meaningful data comparisons. In addition, many agencies have increasingly been collecting data in latitude-longitude formats using GPS. Managing and converting among these types of LRMs has become feasible using computerized conversion tools as part of GIS-T software.

Dynamic segmentation is the software-based process of transforming linearly referenced data stored in a table into a virtual (on-the-fly) feature that can be displayed on a map (Cadkin, 2002 a, b). The process allows also to offset a point from the centreline, in order to represent those objects that not lie on the centreline, and to generate route segments by pairing two "addresses" to define starting and ending points. These new data relationships (events) can be saved and maintained permanently (if desired) in the database, while the topological integrity and structure of the original roadway links can still be preserved without physical changes. Eventually also an angle can be set in order to orient properly the object. Right and left offsets can be accommodated by using negative and positive values, where the sign indicates the direction of offset relative to the direction of increasing measures (see Figure 11).

In general, every object or property of a road can be view as a route event, like signs, intersections, and crashes as punctual event or speed limits, functional classification, number of lanes, and pavement condition as linear event. However, it has to be underlined that modelling everything as event is acceptable when there is only one value to store for each event type (Butler, 2008).

Another issue in a transport organization, is related to the management of multiple versions of the same transport network. This may occur because the data source was different (with different scales, for instance), because of changes applied for specific application purposes, because of techniques for collecting and modelling data can change over time. Individual links common to multiple networks may not have identical shapes or the same lengths. Often, is needed to take attribute data associated with an older, less precise or less accurate network and transfer it to a newer, higher-quality, more precise and/or more accurate network. GIS-T tools can help perform this task through the process of **conflation.** *Conflation* is defined as "the process of combining geographic information from overlapping sources so as to retain accurate data, minimize redundancy, and reconcile data conflicts" (Saalfeld, 1988). The problem of network matching has been studied in different disciplines of GIS and transport but also in image processing: due to diversified references different terms has been used in literature, including map-matching, conflation and linear alignment.

The matching process can be achieved through spatial adjustment and attributes transfers, involving setting distance tolerances and mostly combining automated and manual process. Even if several algorithms have been developed, all conflation approaches usually need a final manual evaluation: conflation results may include multiple matching between features, due to scale effects, definitional discrepancies and data model differences (Nystuen et al., 1997).

One of the first algorithm was developed by the U.S. Census to integrate data from U.S. Geological Survey and Census Bureau (Saalfeld, 1988). This algorithm is applied to link-node structure data, with a bottom-up computational approach. First step is *node matching*, and then a *rubber sheeting* operation is performed on the links, adjusting the geometry of one data to better fit the correspondent data. Finally, matches of corresponding links on the two dataset are identified.

A more sophisticated algorithm introduced by Gabay and Doytsher (1994) is also able to find not only common elements but also the unique elements appearing only in one dataset, allowing to better recognize and handle geometric inconsistencies and topological differences. The workflow evaluates firstly lines with matched end nodes, and then evaluates the unmatched lines. *Edge matching* is performed by matching end nodes of edges and by searching the corresponding edges in a given buffer: in case where multiple edges are close to each other, matching can be difficult. An additional solution is to use also a *segment matching*, which compare edge at segment level, allowing a precise measure of similarity between edges.

Finally, Walter and Fritsch propose an approach based on *relational matching* in 1999. The approach takes into account not only the single link matching but also whether their neighbours are matched or not. The major strength of the relational matching is that uses, in addition to the characteristics of individual participating components, also topological and geometric relationships between these components and other corresponding components, when matches are evaluated. In addition, it uses the information theory to searching for the link that best match in term of common information.

2.2.3 The ArcGIS Network Analysis Extension

The ArcGIS Network Analysis Extension allow building a connected transport network model and performing analysis over it.

Network datasets, in particular, are designed for transport network where travel on edges is allowed in both directions by default.

In a network dataset three kinds of network elements exists (ESRI, 2017 b):

- The **Edge** is the linear element that represents the link over which agents travel.
- The **Junction** is the point element that connects edges and facilitate navigation from one edge to another.
- The **Turn** is the stored information that can affect movement between two or more edges.

Edges and junctions form the basic structure of any network (links and nodes as stated in Section 2.2.1). In order to build a network dataset in ArcGIS, at least a source of line feature must be used, whereas junctions and turns can be omitted, as the software can create junctions automatically starting from line feature, creating a point at each end of an edge. Turns are optional elements that store information about a particular turning movement; for instance, a left turn is restricted from one particular edge to another. The software do not need neither the application of embedded topology within the features, as it keeps track of which source features are coincident. Ensuring that edges and junctions are formed correctly is important for accurate network analysis results. Consequently, setting topology rules constitutes a good practice of pre-processing data before creating the network, and using specific source features for junctions allow applying more constraint on connectivity rules, using predefined attributes (Butler, 2008).

Therefore a correct topology between network elements is the base on which the software can built the connectivity model. The ArcGIS Network Analyst extension (ESRI, 2017 b) allows to define a *connectivity policy*: this is made through the creation of *connectivity groups* which define if coincident features are truly connected. This makes it possible to model overpasses and underpasses without having the roads connect. Each edge source (or its subtypes) can be assigned to exactly one connectivity group, and each junction source can be assigned to one or more connectivity groups. A connectivity group can contain any number of sources. How network elements are connected depends on which connectivity group the element belongs. Connectivity groups can be used to model multimodal transport systems (Figure 12).

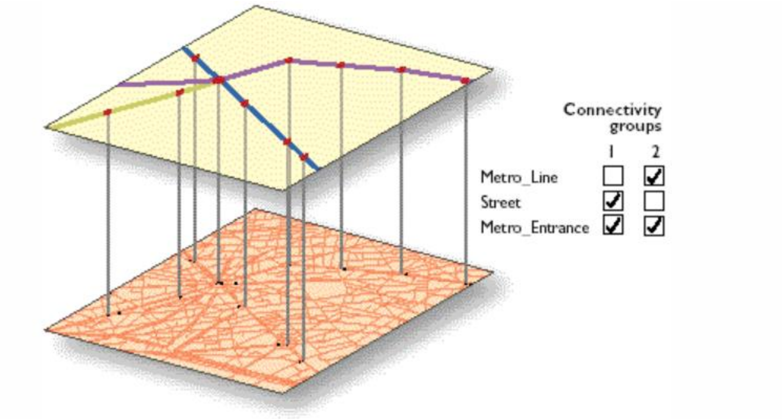


Figure 12: Connectivity groups example: connection between streets and metro transport system. Reprinted from ESRI, 2017 b.

The network dataset also possesses a rich network **attribute model** that helps model costs, restrictions, and hierarchy that control navigation over the network (ESRI, 2017 b). In particular, a *cost* attribute is used to measure and model impedances, such as travel time (transit time on a street) or demand (the volume of garbage picked up on a street). These attributes are divided proportionately along the length of an edge. *Restrictions* define where traversing an edge or junction is prohibited entirely, avoided, or even preferred. Restrictions can be used to model one-way streets, or traversable edges for vehicle types. A *hierarchy* attribute define the rank of a network element: streets are categorized depending on their capacity or classification level. The use of this attribute can reduce computing time and better simulate drivers' behaviour. Finally, a *Descriptor* is an attribute that describe characteristics of the network or its elements. This value is not dependent from the length of an edge, but can be used in conjunction with others attributes to create a new cost attribute.

Each attribute have five basic properties (ESRI, 2017 b): name, usage type, units (of measure), data type, and use by default. The usage type specifies if the attribute is a cost, a descriptor, a restriction, or a hierarchy. Data types can be either Boolean (like in case of restriction), integer (in case of hierarchy or cost), float, or double (for cost attributes). The "Use by default" property automatically sets the attribute on a newly created network analysis layer.

Each attribute defined in the network must have values for each source participating in the network. This is achieved through an *evaluator* (ESRI, 2017 b), which assigns values for the attribute of each source. ArcGIS offers different types of evaluators: constant, field expression, function and script evaluators. When an attribute is defined, an evaluator must be set in order to define the value of the attribute. Edges have always two evaluators, one for each digitized direction of the edge, whereas junctions and turns can have one evaluator for each attribute defined.

When a network dataset is built and added in ArcMap as a layer, it is called a network dataset layer or, more simply, a network layer (ESRI, 2017 b). A *Network analysis layers*, or analysis layers for short, can be thought of as a framework for setting up and solving a network problem. Network analysis classes are feature classes and tables, predefined for the network problem type to be solved. The features and records they contain serve as input and output data for network analysis layers: for instance, points and lines can be added as input (e.g. origin and destination of a trip) and the result of the analysis (e.g. the line representing the trip) is then added as output and stored as a network analysis classes[4] (Figure 13). In addition, the network analysis layer has *properties* that allow to further define the problem, for instance deciding which attributes of the network must be evaluated during the analysis.

Modelling real transport network in network data model allow solving a wide range of problems, which can be classed as “optimization problems” (Rodrigue, 2017). The ArcGIS Network Analyst extension allows solving common network problems, such as finding the best route across a city, finding the closest emergency vehicle or facility, identifying a service area around a location, servicing a set of orders with a fleet of vehicles, or choosing the best facilities to open or close. Below the set of analysis the extension allows to perform.

Network Analyst (ESRI, 2017 b) can find the best route to get from one location to another or to visit several locations (*Route Analysis*). This is the most common use of network analysis algorithms in GIS-T (Rodrigue, 2017). Many types of impedances can be included and modelled with a set of rules that reflects movement speeds. The best route can be the quickest, shortest, or most scenic

---

[4] Network analysis feature classes are displayed in the ArcMap table of contents as sublayers, but they are not maintained on disk; rather, they are stored in memory and saved in the map document, but results can be exported and properly saved into a database.

route, depending on the impedance chosen in the properties of the analysis layer. If there are more than two stops to visit, the best route can be determined for the order of locations as specified by the user. Alternatively, Network Analyst can determine the best sequence to visit the locations, which is known as solving the *traveling salesman problem,* the shortest path route finding problem applied to multiple origins and/or destinations (Rodrigue, 2017).

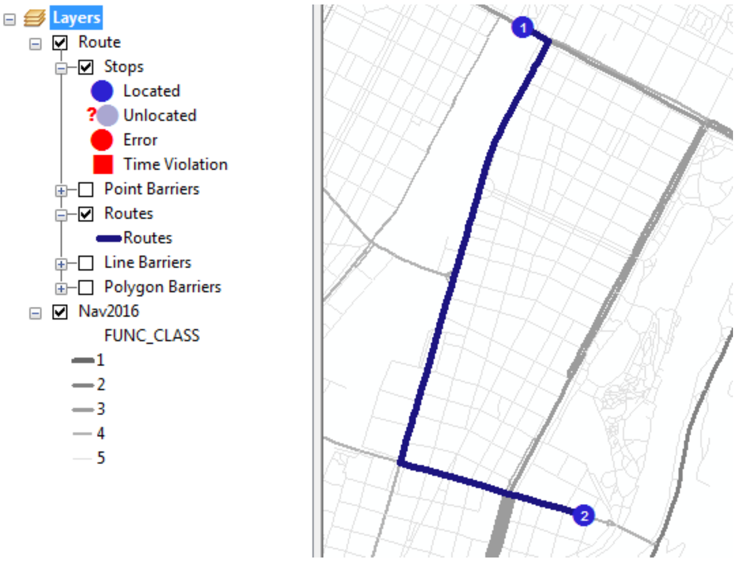
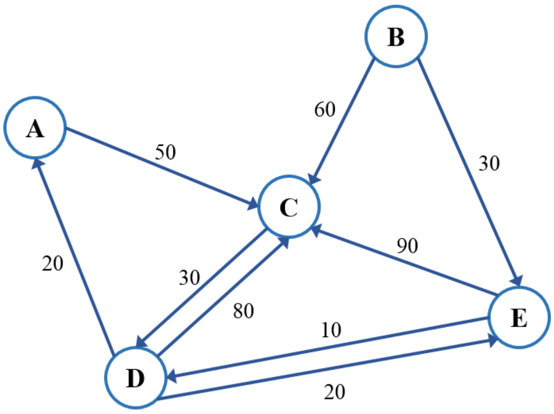


Figure 13: Network analysis Route layer example.

Service Area analysis creates service areas around specified location on a network. A network service area is a region that encompasses all accessible streets, that is, streets that lie within a specified impedance. Multiple concentric service areas can be created setting different cut-off values in the properties of the layers. This algorithm is a variation of classical route finding and can be applied to evaluate the accessibility from or to a certain facility (ESRI, 2017 b).

Another variation of the same algorithm is the *Closest Facility* analysis (ESRI, 2017 b), which allows to solve problems like the closest hospital to an accident. In the properties of the layer, it can be specified how many facilities have to be found, cut-off values and whether the direction of travel is toward or away from them. The output will be the routes to or from the facility, with an associated travel cost.

Network Analyst, support also the creation of *Origin-Destination* (OD) cost matrix from multiple origins to multiple destinations. The matrix is at the basis of the *travel demand model*, which allow to understand the transport patterns and trends in a given urban system, assessing the impact of changes in land use, and evaluate the system-user behaviours under various scenarios. In its most basic form, a travel demand model is a set of tables that define the network, trip generation rates, and the probability for a given trip to exist between *origins* and *destinations,* in a matrix form (Rodrigue, 2017). Origin-destination flow data are usually acquired through surveys. A travel demand model is purely conceptual, however, a reasonably good geographic positioning of origin and destination nodes will help the user visualize the network in terms of the real-world facilities it represents (see Figure 14). In ArcGIS Network Analyst an OD cost matrix is a table that contains the minimum network impedance from each origin to each destination (ESRI, 2017 b). Even though the lines are straight for performance reasons, they always store the network cost, not straight-line distance.



	A	**B**	**C**	**D**	**E**	**T_i**
A	0	0	50	0	0	50
B	0	0	60	0	30	90
C	0	0	0	30	0	30
D	20	0	80	0	20	120
E	0	0	90	10	0	100
T_j	20	0	280	40	50	390

Figure 14: O/D matrix between five locations and representation of spatial interactions. Readapted from Rodrigue, 2017.

The closest facility and OD cost matrix solvers perform very similar analyses; the main difference, however, is in the output and the computation speed. OD cost matrix generates results more quickly but cannot return the true shapes of routes or their driving directions (Butler, 2008).

The *Vehicle Routing Problem* analysis layer is used for complex fleet management tasks (ESRI, 2017 b). The algorithm decides how to best assign a group of customers to a fleet of vehicles and how to sequence and schedule their visits. In the layer's properties can be set the constraints of the analysis, such time window, driver work shifts, driving speeds, and customer commitments. The objectives in solving such vehicle routing problems are to provide a high level of customer service by honouring any time windows while keeping the overall operating and investment costs for each route as low as possible.

Finally, the *Location-Allocation* analysis helps in choosing which facilities from a set of facilities to operate based on their potential interaction with demand points, answering “what if” scenarios related to multiple service locations (ESRI, 2017 b). The objective may be to minimize the overall distance between demand points and facilities, maximize the number of demand points covered within a certain distance of facilities, maximize an apportioned amount of demand that decays with increasing distance from a facility, or maximize the amount of demand captured in an environment of friendly and competing facilities.

Chapter 3

Standards

Standardization is a process that aims to help in technological development and to share good ideas and solutions, through documenting practical information, best practices and agreed solutions on global problem (ISO, 2017). Thanks to standardization initiatives promoted from 1995 in the contest of spatial data handling, significant technological development has made. In the context of this research, standards represent the main source of reference, documenting the state of art of the spatial data modelling activity. In this chapter, relevant standards in the field of GIS and transport are deepened and compared.

3.1 General overview of geospatial standards

The standardisation process in the field of geospatial information is focused on the concepts of interoperability, harmonisation and integration (Longhorn, 2005). Standards indeed enable data sharing and integration between data producers and users, and also between different computer systems (interoperability).

Harmonisation indicates the process applied to a dataset in order to prepare it for a more efficient exploitation in conjunction with other datasets. The process can be achieved using consistent metadata tags, or through the development of a data model and schema which describe the dataset. It essentially a matching activity, finding correspondences between two data structures.

The *integration* process is performed by software or users knowledge: two datasets are integrated when their content is used in combination to examine a specific problem. The harmonisation can ease the integration between datasets.

The *interoperability* between datasets is achieved by computer software applications: these can correctly interpret the information contained in metadata in order to extract and process data, allowing data flows between applications and in general in reducing cost of data sharing.

The standardisation process helps organisations in acquiring data, for instance, when they cannot afford the cost for them, or when they need data outside their operational areas (acquired by other organisations); but also helps in integration when a common semantic is established and allow a major reuse of existing data without a duplication (Berendsen et al., 2010) .

Between organisations dealing with standards production, two main groups can be identified: organisations de-jure and organisations based on voluntary consensus. Organisations based on voluntary consensus often lead the first implementation of a standard that will be then integrated in the de-jure standard sets: standards produced by the former organisations remain publicly available, typically for free, via download from organisation's Web site, while de-jure standards are not.

De-jure organisations include the *International Organization for Standardization* (ISO), which operates at the international level, the *European Committee for Standardization* (CEN), which operates at the European level, creating "European profiles" of ISO standards, the *American National Standards Institute* (ANSI), the same of CEN but for the USA, and, for the Italian level, the *"Ente Nazionale Italiano di Unificazione"* (UNI). These organisations operate in collaboration in order to discuss and publish technical solutions. Inside these organisations, specific Technical Committees (TC) follow the standardisation process for a specific topic. In the contest of spatial data management and transport, the committees of interest are the **ISO/TC 211 – Geographic Information/Geomatics**, the **ISO/TC 204 – Intelligent transport systems**, the **CEN/TC 287 – Geographical Information**, the **CEN/TC 278 – Intelligent transport systems**.

As ISO are the established global norm, two main needs arise: firstly, formal standard published by ISO are normative specification, which are not addressed to non-expert, rising the need of user-oriented implementation guidelines and material to make geospatial standards more accessible. Secondly, as regional/national profiling of the norms exist, there is the need to provide ‘crosswalks’ between existing (national, thematic) geospatial standards and the international standards. This activity is not trivial as there is never a convenient one-to-one correspondence between them (Longhorn, 2005).

The Technical Committee (TC) **ISO/TC 211 - Geographic Information/ Geomatics** was instituted in 1995 and started publishing several standards and technical specifications (the 191xx series), subsuming relevant works from CEN TC 287, OGC and FGDC and collaborating with them for new standards (ISO/TC 211, 2017a). Standards produced ISO/TC 211 specify, for geographic information, methods, tools and services for data management (including definition and description), but also methods for acquiring, processing, analysing, accessing, presenting and transferring such data in digital/electronic form between different users, systems and locations. The set of standards produced provide a framework for the development of sector-specific applications using geographic data.

The policy of the European standards organisation CEN calls for the production of new regional European standards only if no international standard (ISO) already exists. In particular, the **CEN/TC 287 – Geographical Information** work focuses mainly on creating European “profiles” for existing ISO standards from ISO TC 211 (Longhorn, 2005).

One of the main and well-known activity of the CEN/TC 287 has regarded the Working Group 1, which have the aim to develop standards required for building a European Spatial Data Infrastructure (ESDI). The development of an ESDI represents the main objective of **INSPIRE Directive** (acronym for INfrastructure for Spatial Information in Europe), which came into force on 15 May 2007. The goal of the directive is to create a technology infrastructure accessible via the Internet, maintained by the EU Member States and their agencies, to act as a link between and within national and/or regional producers and users of geospatial information, to facilitate the sharing of data, overcoming the problems regarding the availability, quality, organization and accessibility data and, therefore, improve decision making in Europe. The INSPIRE Directive is a legislative document stating that “international standards” will be used to record and publish metadata about geospatial data holdings at all levels of government, from local to national. The European Commission has established five areas in respect of which are defined rules of implementation (Implementing Rules) useful to qualify as an SDI compatible with INSPIRE. These areas are metadata, data, networking, sharing, access and re-use of data, monitoring and control (Litwin and Rossa, 2011).

Other relevant standards for the field of GIS and transport have been produced by the **CEN/TC 278 Intelligent transport systems**, which operates in close cooperation with ISO/TC 204 Intelligent transport systems, responsible for developing international standards in ITS field. As already introduced, Intelligent Transport Systems are often considered a separate area of transport, although they utilize existing transport infrastructure as their deployment base. Between standards produced by CEN/TC 278, particular relevant for this work is the DATEX and now DATEXII [CEN 16157 series], which will be further deepened in next section.

The **Open Geospatial Consortium** (OGC) and in the USA, the **Federal Geographic Data Committee** (FGDC) represent instead the main organisations based on voluntary consensus, which operates in the field of GIS and transport.

The **Open Geospatial Consortium** (OGC) formerly known as the Open GIS Consortium, is a global, not-for-profit, international voluntary consensus standards organization involved in development of open geospatial standards (OCG, 2017). It was founded in 1994, has almost 500 public and private sector member organizations, and thus far has developed almost 40 standards. OGC's standards cover geospatial services and formats, which overcome interoperability problems, including Web services, XML encodings, sensor Web enablement, and open location services.

OGC produces mainly specification documents, that detail engineering aspects and rules for implementing interfaces or encoding that solves a specific geospatial interoperability problem. The specifications are freely and publicly available. OGC makes every effort to ensure that no specification has Intellectual Property Rights restrictions.

The OGC operates in a defined framework, the *Abstract Specification*, which supports the vision of geospatial technology and data interoperability and provides the conceptual foundation and reference model for the development of OGC *Implementation Standards*. The Implementation Standards have a more technical content and detail the interface structure between software components. In addition, other useful documents published and freely available in OGC Web Site are Best Practices, Engineering Reports, Discussion Papers and White Papers. Note also that some of the OGC implementation specifications have already become, or are proposed to become, ISO 19xxx series international standards. OGC enjoys indeed a Class “A” liaison with ISO TC 211 under which certain OGC specifications eventually become ISO standards[5] (ISO/TC211, 2017b).

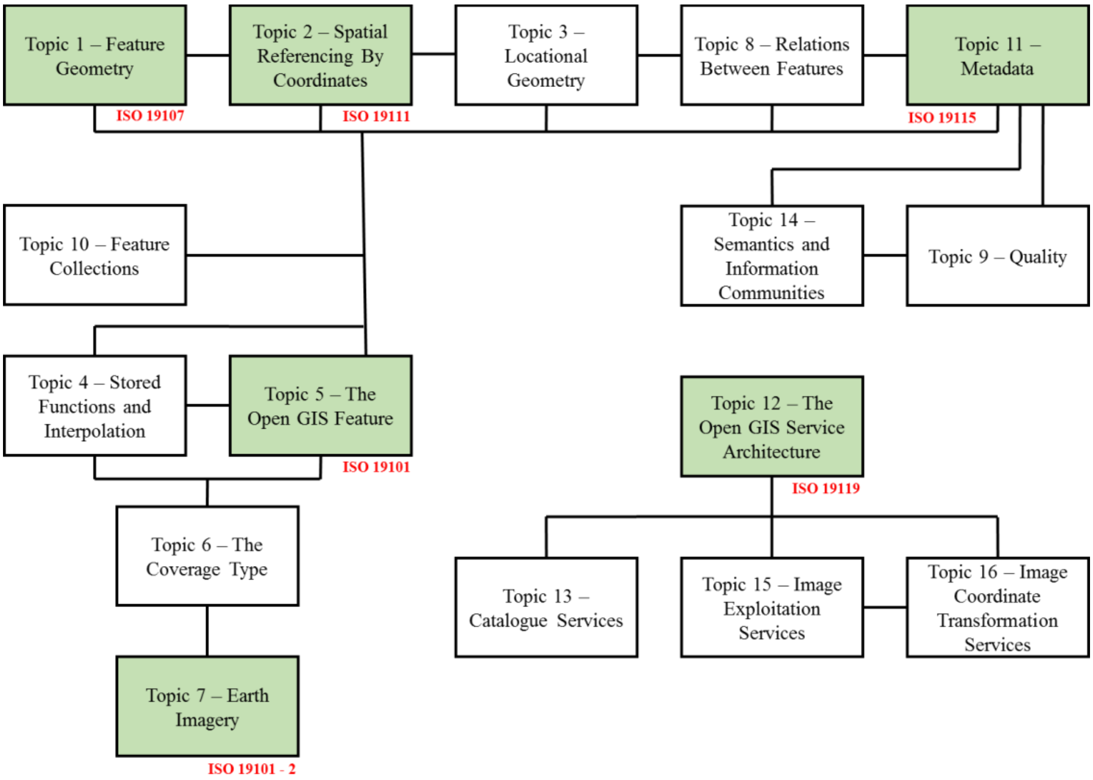


Figure 15: Abstract Specification Topic Dependencies, readapted from OGC, 2005.

In Figure 15, the Abstract Specification Topics and dependencies among them are shown, the topics already incorporated by ISO are highlighted in green.

Most of the OGC specifications are embedded into software tools provided by the vendor members, like Web Feature Services (WFS), Web Map Services (WMS), and Web Coverage Services (WCS), and are published with an “OGC compliant” tag. Therefore, it is important to be updated with the OGC works, as

---

[5] For instance, in relation to metadata, one of the major specifications to come from OGC deals with the XML (eXtensible Markup Language) implementation of GML (Geography Markup Language), which is now the ISO 19136 standard.

these specifications are crucial tools for implementing interoperable geospatial data infrastructures. Most OGC interoperability specifications are of little relevance to end-users, other than to know that the added functionality exists in ‘OGC-compliant’ toolsets available to their information systems developers.

In the USA, the **Federal Geographic Data Committee** (FGDC) was created in 1990 with the aim to develop geospatial data standards that would enable sharing of spatial data among producers and users and support the growing National Spatial Data Infrastructure (NSDI). These processes are developed in cooperation with state, local, tribal, private, academic and international communities. FGDC standards are included in this research because they produced relevant standard for transport.

In general, standards for geospatial data can be subdivided in three groups: standards for data, metadata, and services. Standards on data deal with methods used to store spatial data in common formats and the methods transfer data from one computer system to another. Standards on metadata are used to store, organize, manage and share descriptions of geographic data. Standards on services are used to transfer data or to allow remote access to data stored in the web. These services allow users to interact with the data typically through simple Web clients that operate in real-time. These applications typically allow to view maps, access, retrieve and query data, and make analysis processes.

Between standards on **data**, firstly two geographic data definition and encoding are described:

- **OGC/ISO Simple Features**: common storage and access model of mostly two-dimensional geometries (point, line, polygon, multi-point, multi-line, etc.) used by geographic information systems.
- **GML - Geography Markup Language**: is an eXtensible Markup Language (XML) grammar written in XML Schema to enable modelling, transport, and storage of geographic information in a standardised way.[6]

Following the OGC *Abstract Specification – Topic 5 – Features*, the standard name adopted for the digital representation of abstractions of real world

---

[6] However, GML should not be interpreted as a grammar to describe geographical data, but as a basic grammar and a set of rules for generating XML grammars that describe geographical data. XML grammars are defined through the XML Schema language. An XML Schema file usually has the extension .xsd.

phenomena is that of **feature** (OGC, 2017). The feature is the basic representation construct defined in the reference spatial data model developed by the OGC and endorsed by *ISO 19101 Geographical Information – Reference model*. Between OGC Implementation Standards, the *Simple Feature Access – Part 1: Common Architecture* and *Simple Feature Access – Part 2: SQL option* are part of ISO and CEN standard within the *19125* series.

Features are spatial when they are associated with locations on the Earth otherwise they are non-spatial. Features have a distinguishing name and have a set of attributes (OGC, 2017). Moreover, features may be defined at instance and type level: feature instances represent single phenomena; feature types describe the intentional meaning of features having a common set of attributes. Spatial features are further specialized to represent different kinds of spatial data, as coverages.

The XML plays an important role in how data are handled by various applications, how are logically transported between computers and how applications are able to interpret data contents that have been tagged using XML conventions (Longhorn, 2005). The **GML** is a XML grammar, which specifies different kinds of ‘objects’ for describing geography generally, including features, coordinate reference systems, geometry, topology, time, units of measure and generalized values. OGC has for first developed encoding rules for GML, which is now arrived to the third version (GML 3.0). OGC GML 3.0 specification is compliant with the *ISO 19118:2011 Geographical Information – Encoding, ISO 19107:2003 Geographical Information – Spatial schema, ISO 19108:2002 Geographical Information – Temporal schema* and *ISO 19123:2015 Geographical Information – Schema for coverage geometry and functions* standards. The key concepts used GML to model the world are taken from the OGC Abstract Specification. OGC has also developed specific extension of GML for managing data of specific topics (e.g. CityGML, GMLCOV, InfraGML, IndoorGML). GML is defined by ISO standard *ISO 19136:2007 Geographic information – Geography Markup Language (GML)* and *ISO 19136-2:2015 Geographic information – Geography Markup Language (GML) – Part 2: Extended schemas and encoding rules*. The ISO 19136 is also part of CEN standard from 2009 (*EN ISO 19136:2009*).

OGC has also defined other widespread data formats as KML and GeoJSON.

In addition, as for the GML extensions, standards on geographic data regard also the implementation of *common data models*, which helps in sharing geospatial data. Data models can be expressed in various way, from data dictionaries (detailed description of database content), DDT (data definition table) and UML as graphical representations. These implementations allow to sharing a common semantics, as people use the same terms to identify the same objects or processes, making data sharing easier. In the field of transport for instance, is particularly important to define a common vocabulary in order to easy understand and reuse available datasets. In the next section, relevant data model in the field of transport will be described.

Metadata is "data that provides information about other data" (Litwin and Rossa, 2011), and is crucial for the correct sharing and reuse of data, as through it is possible to decide if a dataset is appropriate for a certain purpose.

Between standards relevant for metadata, the *ISO 19115* series define the basis for geographic metadata definition. As the widespread use of XML eased the development of mechanisms for exchanging information on the web, the *ISO 19139* series define the XML schema implementation for metadata. These standards are also acknowledged by CEN, which has defined them as a core standards for INSPIRE. The ISO 19115, in particular, establishes which properties needs to be described in geographical information and associated services metadata (identification, extent, quality, spatial and temporal aspects, content, spatial reference, portrayal, distribution, and other properties). The XML schema defined by ISO 19139, allows the linkage of disparate geospatial datasets from multiple sources in innovative way and with less effort and cost, and is built on OGC compliant interoperability tools. Indeed metadata definition is strictly connected with web services, and in this context, OGC has for instance published specifications for the deployment of Catalogues Services, which are services for searching and retrieving data built on metadata encoding.

Another relevant standard for metadata is the Dublin Core, from the Dublin Core Metadata Initiative (DCMI), a consensus derived, discovery level metadata standard, became in 2009 the *ISO 15836* series and revised in 2017 (*15836-1:2017*). The ISO norm establishes 15 core metadata elements for cross-domain resource description, without implementation rules. These terms are part of a larger set of metadata vocabularies maintained by the Dublin Core Metadata Initiative. Cross-walks have been prepared between Dublin Core and ISO 19115, with FGDC, and with other metadata standards. In particular within the INSPIRE initiative, the Metadata Implementing Rules take into account also the Dublin Core.

Looking at local level, in the Italian context the RNDT (Repertorio Nazionale dei Dati Territoriali) represents the specific profile implementation of the INSPIRE and ISO standard on metadata.

Today the main trend is to serving up spatial data to interested users through **web services**. The OGC is the organisation that has started to define specifications for web services within the Topic 12 of Abstract Specification, the *"OpenGIS Service Architecture"*, an infrastructure type supporting "always-on" data serving. The XML/GML data serving format supports service requests, core datasets exchange and download, and also dynamic data serving (e.g. on line road traffic data). These mechanisms have a role in supporting a decentralised approach to data management and exchange, in opposition to centralised data warehousing (Longhorn, 2005). This Abstract Specification is now part of ISO and CEN. In particular, the *ISO 19119:2016 – "Geographic information – Services"* defines how geographic services shall be categorised according to a service taxonomy based on architectural areas, providing support for easier publication and discovery of services.

Implementation standards for geospatial services defined by OGC include the following Web services:

- *WMS - Web Map Service*: provides map images;
- *WFS - Web Feature Service*: for retrieving or altering feature descriptions. This is also an ISO and CEN standard (*19142:2010 Geographic Information – Web Feature Service*);
- *WCS - Web Coverage Service*: provides access, sub setting, and processing on coverage objects;
- *WMTS - Web Map Tile Service*: provides map image tile;
- *WPS - Web Processing Server*: remote processing service, allowing data transformation;
- *WCPS - Web Coverage Processing Server*: remote processing service, allowing processing of multi-dimensional geospatial coverages representing sensor, image, or statistics data;
- *GeoRSS - Geographically Encoded Objects for RSS feeds*: is a proposal for geo-enabling, or tagging, "really simple syndication" (RSS) feeds with location information;
- *CSW - Catalogue Service for the Web*: defines common interfaces to discover, browse, and query metadata about data, services, and other potential resources for several bindings (Z39.50, CORBA, and HTTP).

- *SOS - Sensor Observation Service*: provides access and representation for observations and sensor data and metadata;
- *SPS - Sensor Planning Service*: provides information about the capabilities of a sensor and how to task the sensor;
- *GeoSPARQL - Geographic SPARQL Protocol and RDF Query Language*: representation and querying of geospatial data for the Semantic Web.

In conclusion, is important to be updated with geospatial standards in particular when working on data exchange. Several data models have been created to harmonize data set from different sources, but there are still particular situations not managed. In addition, the implementation of standards requires a huge initial effort, mainly related to the deeply understanding of the documentation, which is usually highly technical. The production of implementation guidelines is useful to reduce this effort, and allows an easier spreading of the standard itself.

3.2 Standard data model for transport

In this section a set of standards useful for the definition of transport network data model are described.

3.2.1 INSPIRE - Transport Network

The INSPIRE Directive has been already introduced in last section. The Directive addresses 34 spatial data themes needed for environmental applications, grouped in three annex, with key components specified through technical guidelines. Between this data themes the Transport Network Theme, part of the first Annex, is the one described in this research. The content of this section is mainly derived from the Generic Network Model and Transport Networks technical guidelines (“D2.8.I.7 Data Specification on Transport Networks – Technical Guidelines” version 3.2 and “D2.10.1: INSPIRE Data Specifications – Base Models – Generic Network Model”, version 1.0rc3), freely available from the INSPIRE Web site.

The INSPIRE Directive defines the spatial data theme Transport Networks as road, rail, air and water transport networks and related infrastructure. The documentation defines a **Generic Conceptual Model of Network**, from which different types of Transport Networks **Application Schemas** derives (INSPIRE, 2014): Road Transport Network, Railway Transport Network, Cable Transport Network, Water Transport Network, Air Transport Network and Common Transport Elements.

The *Generic Conceptual Model* include base models and base types packages. Base Types describes some common elements like the data type *Identifier*, the feature type *SpatialDataset*, and other code lists, enumerations and data types, which are reused inside specific data models. Between Base Model, a **Generic Network Application Schema** is defined as Figure 16 and Figure 17. The central element of the model is the class *NetworkElement* that is the abstract type representing an element in a network. This class is specialized in *GeneralisedLink*, *LinkSet*, *GradeSeparatedCrossing* and *Node*. A *LinkSet* represent a collection of *Link* or *LinkSequence*, which are the child classes of *GeneralisedLink*.

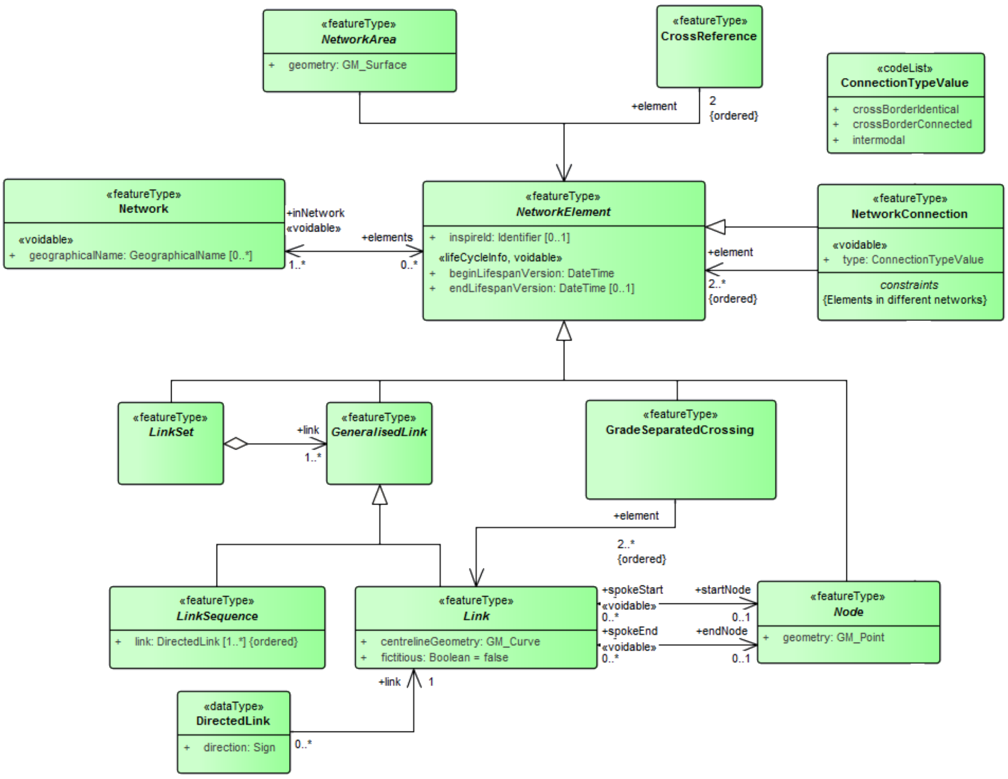


Figure 16: INSPIRE Network Application Schema – Overview. Readapted from INSPIRE, 2013 a.

The *GeneralisedLink* is an abstract type representing a linear network element that may be used as target in linear referencing. The concrete class *Link* contains the attribute "centerlineGeometry" which allow to define the geometry representing the curvilinear network element that connect two position (usually pavement centreline). Each link have a relationship with the data type *DirectedLink*, which indicates if the links' direction of flow agrees (positive) or disagrees (negative) with the positive direction of the link. The concrete class *LinkSequence* indicates a network element, which represents a continuous path in the network without any branches, and where the attribute “Link” allow to identify the single link and its order in the path. The *Node* concrete class represents a position in the network, which occurs at the beginning or the end of a link; its attribute “geometry” allows to define the position of the point, whereas the attributes “spokeEnd” and “spokeStart” define respectively the link that enters and the one that leaves the node. The class *GradeSeparatedCrossing* indicates which of two or more intersecting elements is below or above the other (in case elevation coordinates are not present). Network may be also represented by polygonal geometry within the class *NetworkArea*.

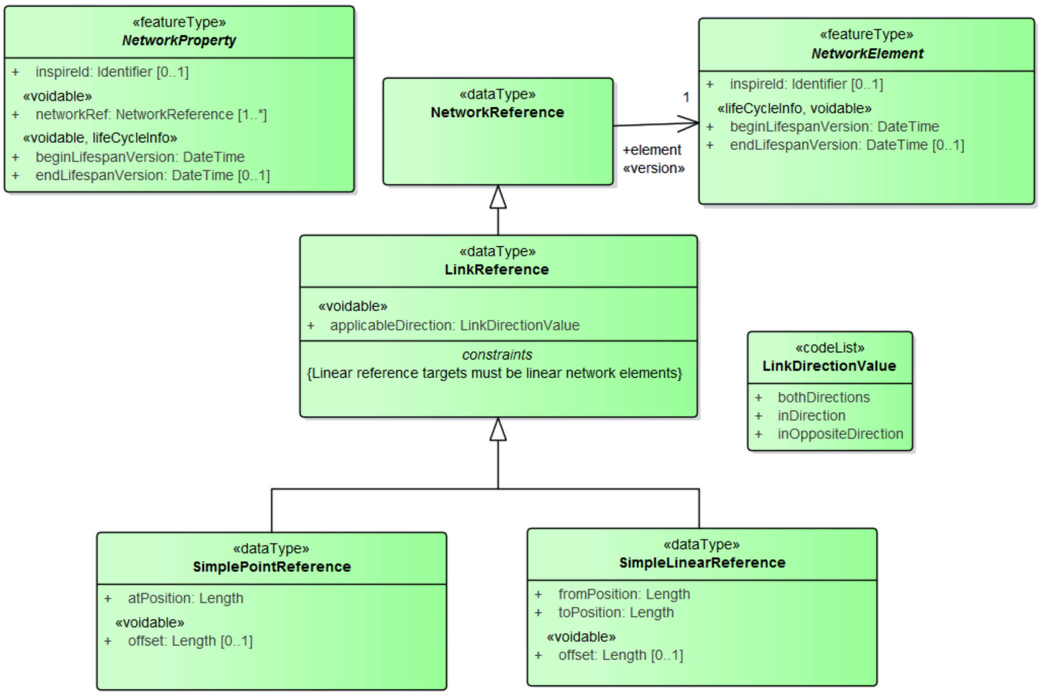


Figure 17: INSPIRE Network Application Schema – Network Properties. Readapted from INSPIRE, 2013 a.

In addition, the class *Network* represents a collection of network elements, the class *CrossReference* represents a cross reference between two elements which have different representation but indicates the same spatial object, and the class *NetworkConnection* represents a logical connection between two or more network elements in different networks, where the attribute “type” is coded through the predefined codeList *ConnectionTypeValue.*

Usually in the generic conceptual model, phenomena are often spatial referenced indirectly by a reference to a network element instead of a geometry. For phenomena located along a link, the spatial reference usually is applied to a part of the link only. Linear referencing may be used to describe such reference in accordance with the standard ISO 19148, requiring for all linear reference one or two expressions representing the distance from the start of the link along its curve geometry, eventually with an offset. The class *NetworkProperty* is an abstract type representing phenomena located at or along a network element, and it provides the general properties to associates phenomena with network elements (see Figure 17). Each *NetworkElement* has a relationship with the data type *NetworkReference* and its child class *LinkReference*, which allow to identify a linear network element and the direction where the property is applicable through the attribute "applicableDirection" coded by the codeList *LinkDirectionValue*. The *LinkReference* data type can be expressed by its child classes *SimplePointReference* and *SimpleLinearReference,* where the attributes "offset", "atPosition", "fromPosition" and "toPosition" allow to precisely locate the phenomena over the network element.

Looking at the specific **Road Transport Network Application Schema,** some classes are specialised in elements characteristic of a road network infrastructure.

In Figure 18, is shown the general overview of the road transport network model. Most of these classes are derived from the Generic Conceptual Model of Network (as can be view from the top-right of the main boxes). Main classes define spatial object like areas (blue in Figure 18), lines (red in Figure 18) and node (yellow in Figure 18) and an additional class defines a set of properties (green in Figure 18). The Figure 19 and give a graphical example of the use of these concepts.

The *TransportLink* is a generic linear spatial object that describes the geometry and connectivity of a transport network between two points in the network (see Figure 19). It is represented by the concrete class *RoadLink*, which can represent paths, bicycle roads, single carriageways, multiple carriageway roads and even fictitious trajectories across traffic squares (see Figure 20).

The *TransportLinkSequence* is a generic linear spatial object, composed of an ordered collection of transport links, which represents a continuous path in the transport network without any branches (see Figure 19). It describes an element of the transport network, characterized by one or more thematic identifiers and/or properties. It is represented by *RoadLinkSequence*, which has a defined beginning and end. Every position on the *RoadLinkSequence* is identifiable with one single parameter such as length. It describes an element of the road network, characterized by one or more thematic identifiers and/or properties.

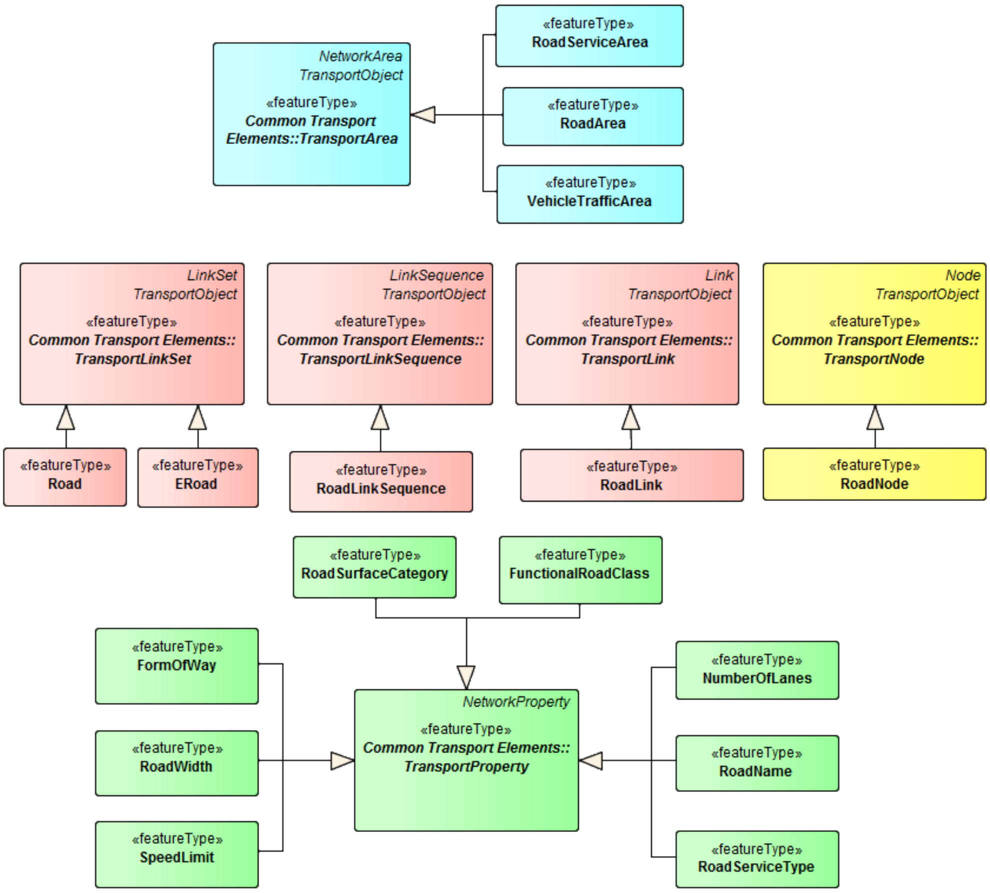


Figure 18: The overview of the INSPIRE Road Transport Network Data Model. Readapted from INSPIRE, 2014.

The *TransportLinkSet* is a collection of transport link sequences and or individual transport links that has a specific function or significance in a transport network. This spatial object type supports the aggregation of links to form objects with branches, loops, parallel sequences of links, gaps, etc. (see Figure 19). It is represented by *Road* and *Eroad* (see Figure 20). A *Road* is a collection of road link sequences and/or individual road links that are characterised by one or more thematic identifiers and/or properties. Examples are roads characterized by a specific identification code, used by road management authorities or tourist routes, identified by a specific name. An *Eroad* is a collection of *RoadLinkSequences* and or individual *RoadLinks* that represent a route that is part of the international E-road network, characterized by its European route number like (“E40”).

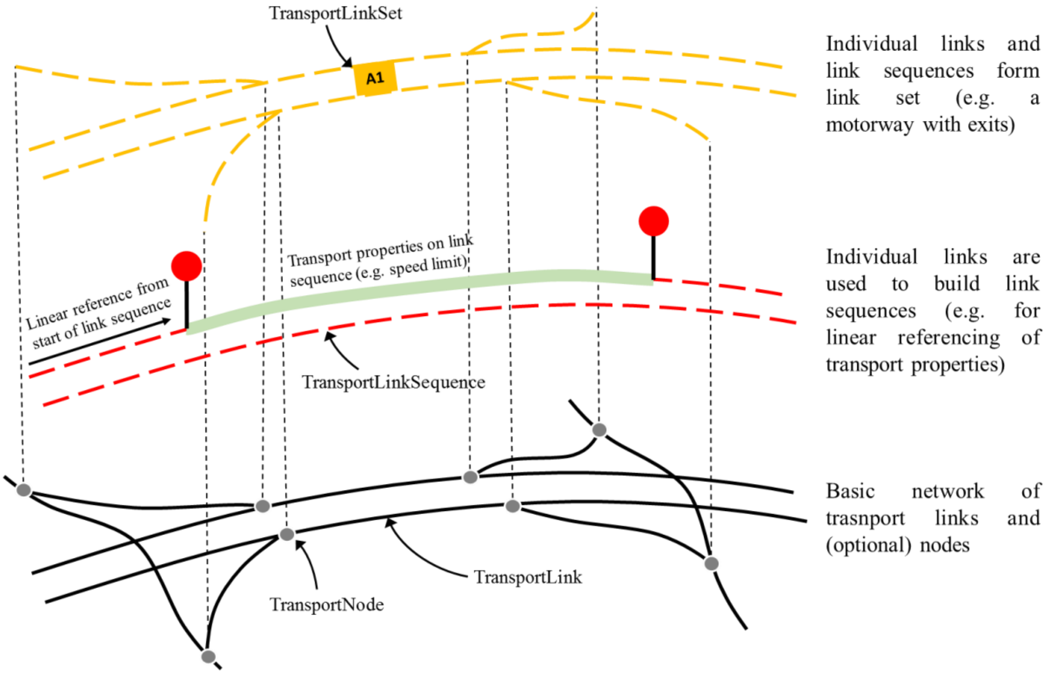


Figure 19: Example of the use of Link, Node, LinkSequence and LinkSet. Readapted from INSPIRE, 2014.

The *TransportNode* is a point spatial object, which is used for connectivity. Nodes are found at either end of the *TransportLink* (see Figure 19). Its child class *RoadNode*, defines a point spatial object that is used either to represent connectivity between two road links or to represent a significant spatial object such as a service station or roundabout (see Figure 20). It has a coded attribute “FormOfRoadNode” that describe the physical meaning of the node (enclosedTrafficArea, junction, levelCrossing, pseudonode, roadEnd, roadServiceArea, roundabout, trafficSquare).

The *Transport Area* is a surface that represents the spatial extent of an element of a transport network. It has three child classes: *RoadArea*, *VehicleTrafficArea* and *RoadServiceArea* (see Figure 20). A *RoadArea* is the surface that extends to the limits of a road, including vehicular areas and other parts of it, like pedestrian areas. A *VehicleTrafficArea* is the surface that represents the part of a road, which is used for the normal traffic of vehicles. A *RoadServiceArea* is the surface annexed to a road and devoted to offer particular services for it, as Gas station, rest area, toll area.

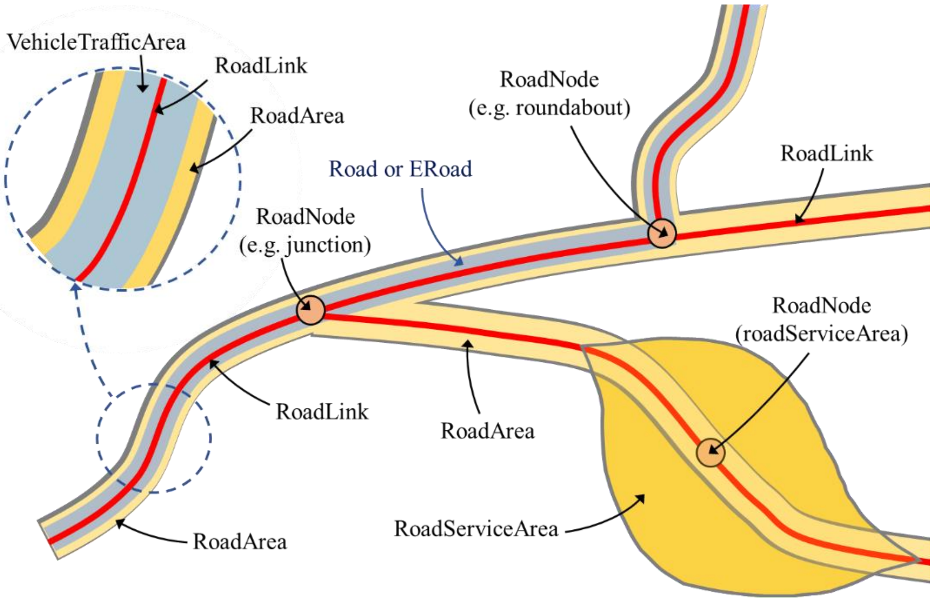


Figure 20: Example of use of elements forming the Road Transport Network. Readapted from INSPIRE, 2014.

Finally, the class *TrasportProperties* defines a reference to a property that falls upon the network. These properties can be applied to the whole network element associated with it or, for linear and punctual spatial objects, be described using linear referencing (see Figure 19). As for the generic network model, these properties may be applied on specific direction (through the attribute “direction”). Child classes of *TrasportProperties* define the specific set of properties of a road network element, and each child class can have further attributes and codeList associated (see Figure 21). Child classes for *TrasportProperties* are:

- *SpeedLimit* is the limit for the speed of a vehicle on a road. It is applied only on road elements. Specific associated codeLists and enumerations are: *SpeedLimitSourceValue*, *VehicleTypeValue*, *SpeedLimitMinMaxValue*, *WeatherConditionValue, AreaConditionValue*.
- *RoadServiceType* describes the type of road service area and the available facilities, through the associated codeList *RoadServiceTypeValue.*
- *Roadname* is the name of a road, as assigned by the responsible authority.

- *FunctionalRoadClass* is the classification based on the importance of the role that the road performs in the road network, through the associated enumeration *FunctionalRoadClassValue.*
- *RoadWidth* indicates the width of the road, measured as an average value through the associated codeList *RoadPartValue.*
- *FormOfWay* is the classification based on the physical properties of the road element, through the associated codeList *FormOfWayValue.*
- *RoadSurfaceCategory* is the specification of the state of the surface of the associated road element, specifying whether a road is paved or unpaved, through the codeList *RoadSurfaceCategoryValue.*
- *NumberOfLanes* indicates the number of lanes of a road element as integer. It also have an associate enumeration *MinMaxLaneValue* indicating whether number of lanes are counted as the maximum, minimum or average number.

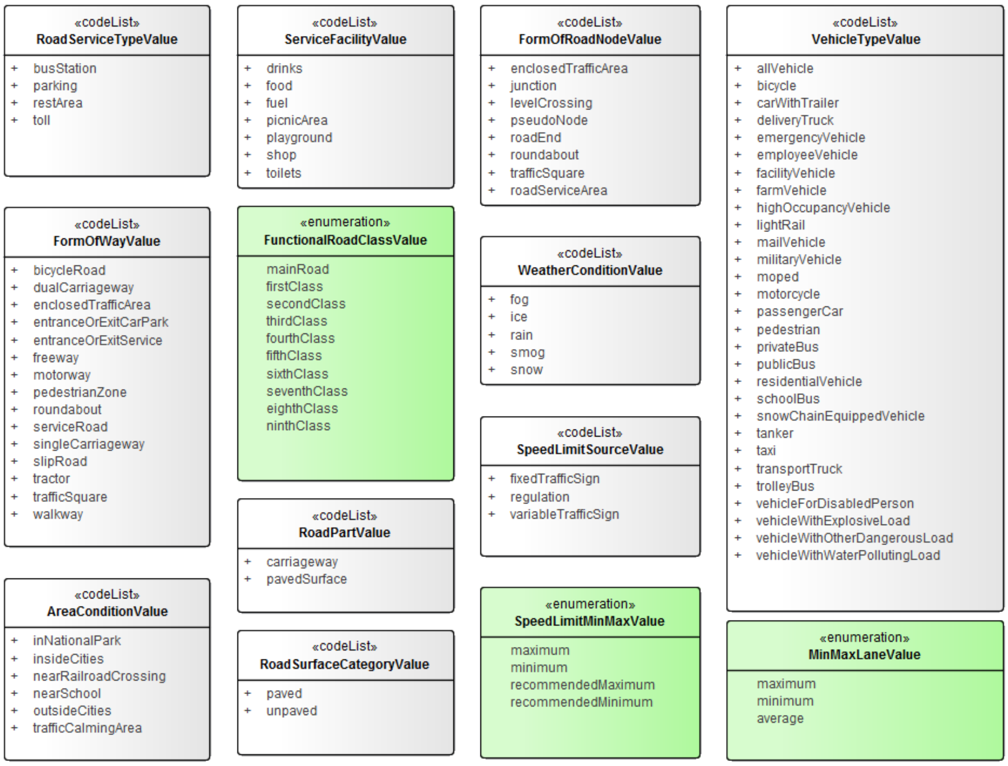


Figure 21: INSPIRE Road Transport Network Data Model: Enumerations and Code Lists. Readapted from INSPIRE, 2014.

The INSPIRE Directive identifies 34 data themes for which it describe a spatial data schema. Nevertheless the Directive also take into account measured, modelled or simulated data which can be integrated into INSPIRE spatial data themes.

In particular, the Base Models package contains also the sub-package Observations, where is defined how the *ISO 19156:2011 – Geographic information – Observations and Measurements* (*O&M*) standard is to be used within INSPIRE. The *ISO 19156:2011* standard indeed defines a conceptual schema for observations and for features involved in sampling when making observations, providing models also for the exchange of information (XML schema). The standard was initially developed by OGC under the work on Sensor Web Enablement, becoming then an ISO standard, later endorsed by CEN and FGDC. Even if this specification is not explicitly applied to Transport Network, it can be used to model road sensors and related data. The standard is structured upon explicit relationships between results and the feature of interest, sampling features or procedures.

An *observation* is an action whose *result* is an estimate of the value of some *property* of the *Feature-of-Interest* (FoI), at a specific point in *time*, obtained using a specified *procedure* (INSPIRE, 2013 b).

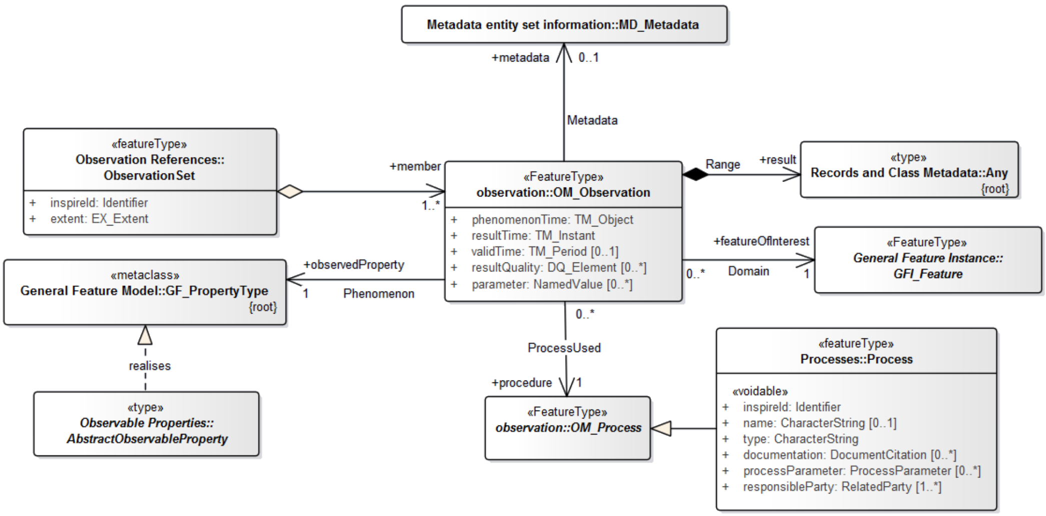


Figure 22: The overview of Observation Pattern, readapted from INSPIRE, 2013 b.

As can be seen in Figure 22, the main class in the diagram is the *OM_Observation*, with a series of attributes that specifies it. In particular, the “phenomenonTime” is the valid time of the observation's result, whereas the “resultTime” represents its creation time. The “validTime” represents the time span for which an observation is valid. Other attributes are “resultQuality”, which describes the quality of the result, and “parameter”, which describes an event-specific parameter (environmental parameter, instrument setting…). Those last two attributes are optional and their presence is strictly tied to the process definition.

One or more *OM_Observation* objects are related with one *GFI_Feature*, an abstract class that represents the set of all classes that are feature types, which constitutes the domain on which observations are applied. In this relationship, the *GFI_Feature* has the role of *FoI*, a representation of a real-world object that carry the observed property. It can be either a domain feature (a s*ampledFeature*), as a specific road, or a *samplingFeature*, e.g. a stretch of road. A *FoI* can exists only because exists an observation, and a specific sampling feature must be defined in this case. In other cases, a feature also used in other contexts within the domain will also serve as a *FoI* for an observation. Giving an example in the transport field, the sampling Feature (that is the feature of interest of an observation) is the area on which a loop sensor insists, and the sampled feature can be the entire lane on which the measure can be projected.

The *OM_Observation* objects are also related to one *OM_Process,* an abstract class that can represent an instrument, a sensor (or a sensors system), but also a computation or algorithm applied to primitive results used as input. *OM_Process* is an instance of a Process class, characterized by a series of attributes.

The association Phenomenon shall link the *OM_Observation* to the *GFI_PropertyType*. The property type has the role *observedProperty* with respect to the observation: an observed property supports semantic or thematic classification of observations and describes the phenomenon associated with the type of the *FoI*, usually using specific ontology concepts. INSPIRE has developed a specific model for observed properties which provides a framework for extending a pre-defined term in a vocabulary, such as constraints or statistical measures. In particular, statistical measures describe data grouping: an attribute describes the statistical function being applied (i.e. mean), and additional attributes (as “aggregationTimePeriod”) describe the level of aggregation (which can be over time, spatial dimensions or other defined aggregation types).

Finally, the association *Range* shall link the *OM_Observation* to the value generated by the procedure (observation as a composition of results). The value has the role *result* with respect to the observation. The type of the result is shown as *Any*, since it may represent the value of any feature property, and can be a scalar value as well as a complex multi-dimensional array. The type of the observation result must be consistent with the observed property, and the scale or scope for the value shall be consistent with the quantity or category type.

OM_Observation can be a member of an *ObservationSet*, which links multiple related observations together (in a certain spatial and temporal extent).

The O&M standard arranges a series of Design Patterns, which helps in implementation design. In particular, different designs belongs to different dimensionalities of the *FoI* of the observation and of the results. *FoI* can be of type Point, Curve (or trajectory) and Surface. Results can be *Single* or *Multiple* in time (a set of time, value pairs).

As in this research we are interested in modelling also traffic detectors (fixed and not) and their measures, two Design Patterns seem appropriate. Gathered data from fixed sensors can be interpreted as *FoI* of the type *Point*, with *Multiple Results in Time*: the *SF_SamplingPoint* is the class that represents the geographic location of the measurement and could be specialised in an extension schema to be a station feature type (or similar) to provide further information about the fixed station (e.g. a name). Eventually also a *FoI* of type *Curve*, representing a trajectory along with individual measurements are provided with the results, can be used in case data are gathered through Floating Car Data.

3.2.2. FGDC – Geographic Information Framework Data Content Standard – Transportation Data Model

As outlined in the general section of this Chapter, one of the main activities of the Federal Geographic Data Committee (FGDC) is the coordination in the development of the National Spatial Data Infrastructure (NSDI) in the USA. This project can be seen as the parallel one to INSPIRE initiative in Europe. Between objectives, a relevant one is to develop the “Framework Data”, a set of seven common themes of geospatial data that provide the basic data "skeleton" needed by GIS users.

In this section, Framework Transportation model is described, with a focus on road and transit data model. The content in this section is mainly derived from the Guidance Document “Developing NSDI Framework Data: ANSI Framework Data Content Standards”, and the Geographic Information Framework Data Content Standard “Part 7: Transportation Base”, “Part 7c: Transportation – Roads” and “Part 7d: Transportation – Transit” freely downloadable from the FGDC Web site.

The Framework Standard defines a minimal level of data content and requirements. The goal is indeed to structure data and information about data in order to allow computer systems interacting each other to determine whether data are compatible. The transport system defined in this Framework includes physical and non – physical components: physical elements take into account all modes of travel, representing them through subthemes (road, railroad, transit, airport facilities, and waterway networks), whereas non – physical elements are represented through a generic event model. The Framework is compliant with the *ISO 19107* standard: the “Geometry” attribute (optional) is a *GM_Object* class, which can include point, line, polygon geometry and other variations. In addition, the optional “Topology” attribute also is complaint with the *ISO 19107* and is used to describe relationships between features (such as connectivity). The Framework includes optional elements in order to support linear referencing following the *ISO 19133* standard.

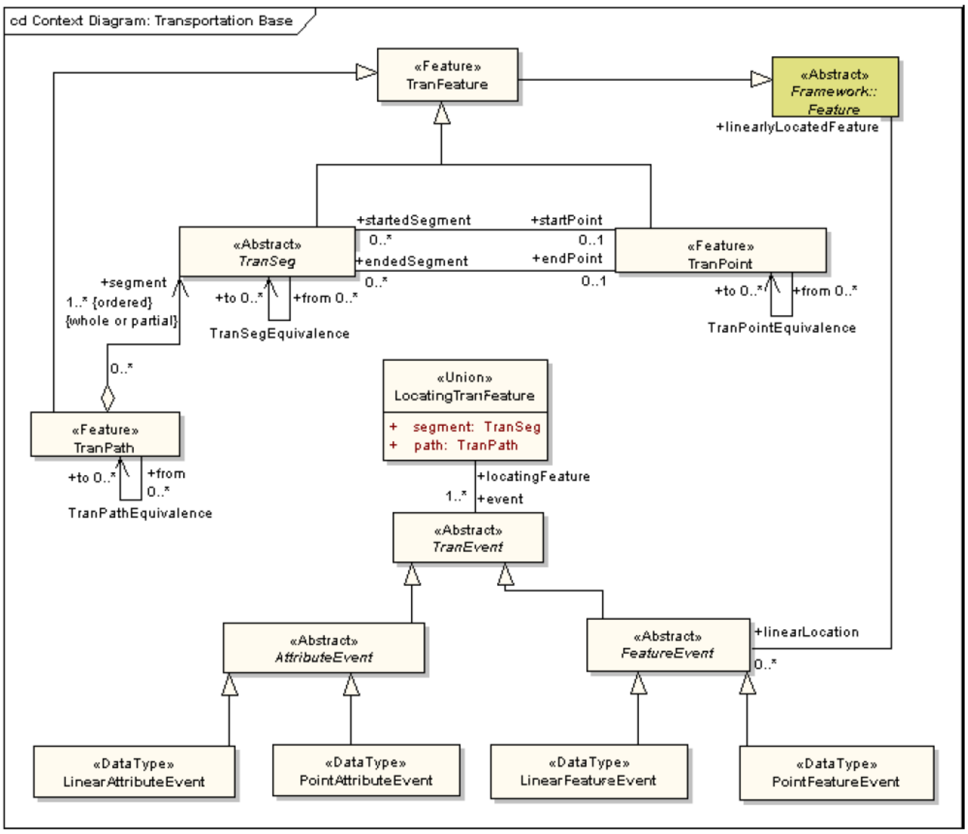


Figure 23: Transportation Base Model, reproduced from Federal Geographic Data Committee, 2008 a.

The general schema of Transportation Base model is shown in Figure 23. The model includes a **segmentation model** to represent geometry, topology and relationships between features and an **event model**, which through defined attributes allows to linearly locate features. At the most general level, the *TranFeature* class is used to model transport features; it is derived from the general abstract class *Feature*, from which it inherits the required field “identifier”, a specific data type introduced for the permanent identification of a feature, and other optional attributes as “geometry” and “topology”. From the *TranFeature* class three child classes are derived, as can be more explicitly seen from Figure 24. These classes are used for managing linear features with connectivity: *TranPath*, *TranSeg*, and *TranPoint*. These child classes are abstract types, and can be instantiated by their analogues classes in specific sub model: for instance, in the Transportation – Road model there will be *RoadPath*, *RoadSeg* and *RoadPoint* classes. The model can be extended creating user-defined *TranFeature* child classes, in order to represent all other non-linear transport elements.

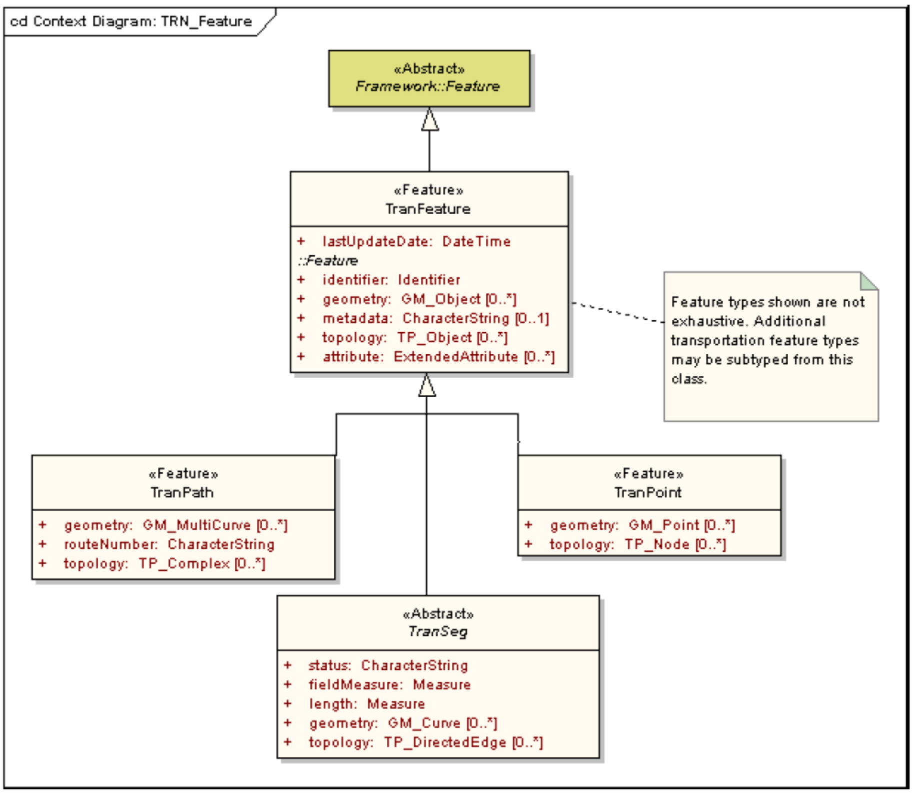


Figure 24: Transport feature type hierarchy, reproduced from Federal Geographic Data Committee, 2008 a.

The *TranSeg* is an individual linear section of the physical transport network without branches and allows to represent a single segment between two points, or separated segments for each direction of travel. The geometry attribute is optional as the feature can be instead represented by topology. In the road segmentation model this class can be instantiated by the class *RoadSeg*: this class has an additional attribute in comparison to *TranSeg*, “isAnchorSection”, of Boolean type. An anchor section is a road section between two known locations, called anchor points. These points represent unambiguous physical locations in the real world, linking the computer representation of the road system to the real world.

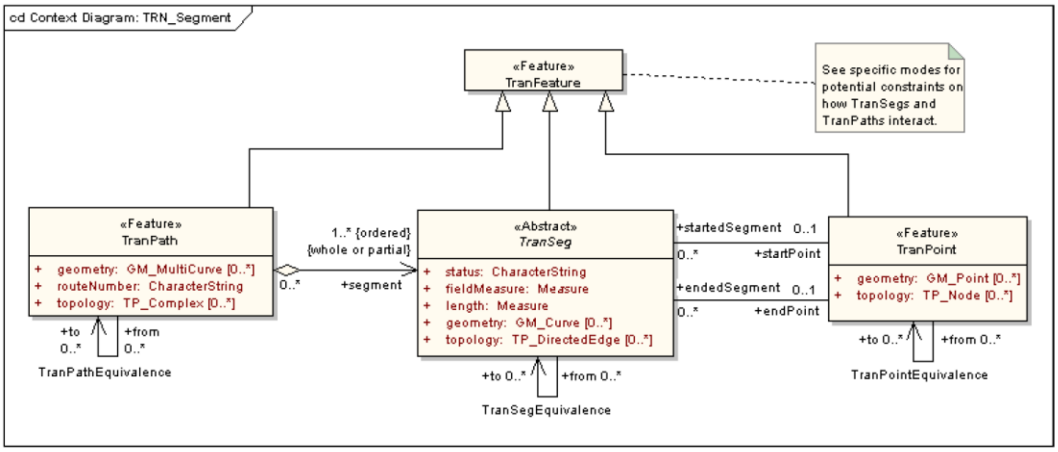


Figure 25: Relationships among *TranSeg*, *TranPoint*, and *TranPath*, reproduced from Federal Geographic Data Committee, 2008 a.

TranPoint class (and the relative *RoadPoint* class in the road model) represents the junctions of the networks and is used to provide topological connections between *TranSeg* elements, of which they represent the start and end. In addition, through the relationship with the *TranSeg*, shown in Figure 25, direction of flow can be provided. This relationship can be implemented several ways, like using a table that relates each segment with the relative start and end point. As *TranPoint* class has not additional attributes, point feature that not represent junctions have to be modelled using user-defined child classes of *TranFeature* or through the event model.

The *TranPath* class represent an aggregation of one or more, whole or partial, *TranSeg*, and in case of *RoadPath* it can represent a usage of part of the road network, where the attribute “routeNumber” identify the path (see Figure 25). The geometry can be inherited from *TranSeg* or a new one can be defined (e.g. a more generalised representation). The *TransSeg* elements that compose the *TranPath* may be connected, but is not mandatory.

The **event model** allows to linearly locate features or attributes along a *TranSeg* or a *TranPath*. A measure, calculated from the beginning point of the linear elements, specify the location. Events can also have a length and an offset associated with them.

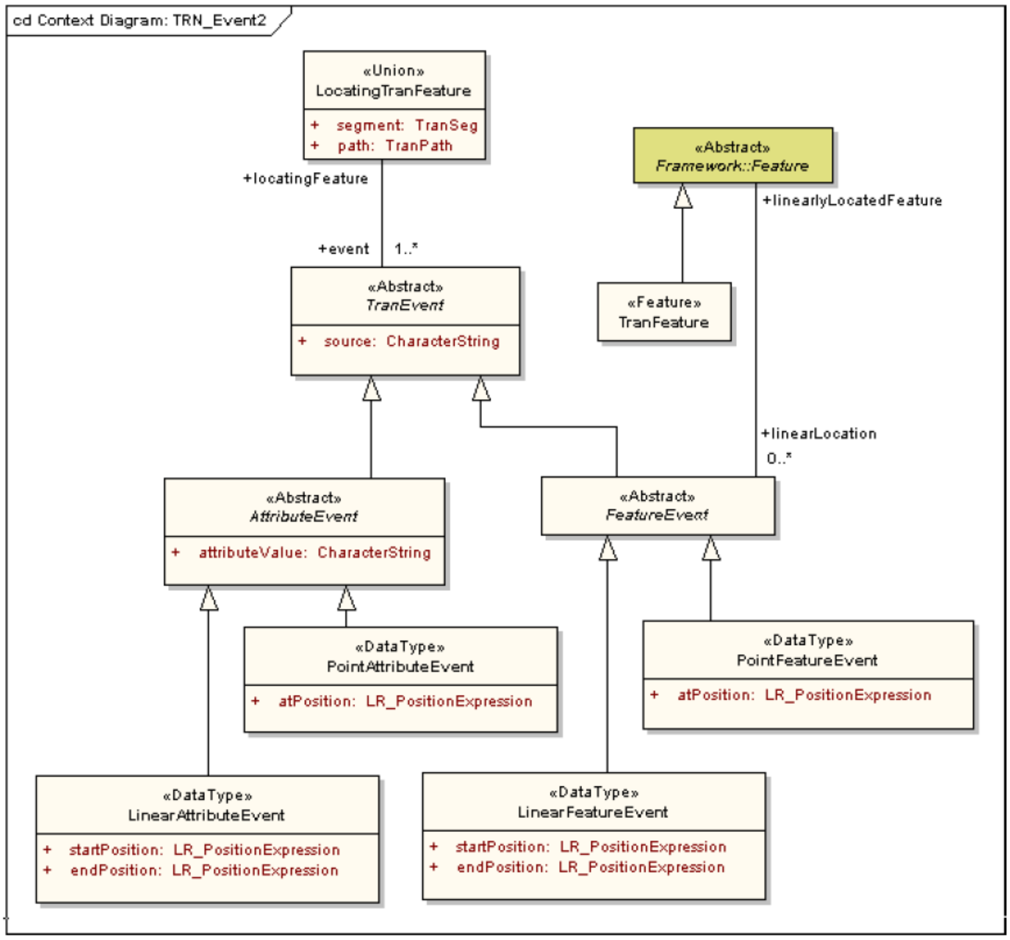


Figure 26: Transportation event model, reproduced from Federal Geographic Data Committee, 2008 a.

Looking at the schema in Figure 26, the class *LocatingTranFeature* is a union abstract class, which consists in a set of alternatives, where only one can be used for each instance. This implies that the child class *TranEvent* must be located on either a *TranPath*, or a *TranSeg*, but not both. A *TranEvent* is specialized in two abstract classes: *AttributeEvent* and *FeatureEvent*, which are further specialized in the data types *PointAttributeEvent*, *LinearAttributeEvent*, *PointFeatureEvent* and *LinearFeatureEvent*.

The *AttributeEvent* class is used when an attribute can change in single location on a *TranSeg* or *TranPath*. *AttributeEvent* class has required attributes: “source”, “attributeValue”, the reference to the *TranSeg* or *TranPath* that are used to locate the event, which derives from the *LocatingTranFeature* parent class, and the measured location which are “startPosition” and “endPosition” for a *LinearAttributeEvent* and “atPosition” for a *PointAttributeEvent. AttributeEvent* instances have not their own geometry, but inherit the reference geometry on which they are defined.

In parallel, in the road transport model, two code lists support the *RoadPointAttributeEvent* and the *RoadLinearAttributeEvent* classes: *RoadPointEventType*, which includes “tollbooth”, “tollCharge”, “maxElevation”, “sign”, and “pass” and *RoadLinearEventType,* which include “speedRestriction”. Other values can be added to these code lists.

FeatureEvent class is used in the case where attributes have an unchanging or rarely changing value, or when the feature to be represented has also some additional attributes. Attributes are the same of *AttributeEvent*, plus many other optional attributes and geometry; in addition, user-defined attributes can be added. *FeatureEvent* must be preferred to *AttributeEvent* when features have been collected with GPS, as they can have better accuracy than the road geometry to which are referenced. *LinearFeatureEvent* class supports also features with area geometries, defining what part of the segment or path is within the area geometry.

As for *AttributeEvent*, a *RoadPointFeatureEvent* and a *RoadLinearFeatureEvent* classes exist for the road transport model, and they support the same code lists defining the type of the event as for the *AttributeEvent*.

Finally, the **transit data model** is described apart from road and general transportation model, because the transit system depends and should be rely on road (bus) and rail model (subway, light rail). In addition, the data model defines most broadly used elements of the public transport system like stops, interconnections, facilities and routes, temporal elements aspects such as trips and arrival and departure times, and information that does not have its own geographic location, as fares and public transport vehicles, which are not covered in the other transport data models. In Figure 27, a general overview of the data model is given, describing relationships between features and data types of transit system. In Figure 28, an overview of the specific features of the transit model is given.

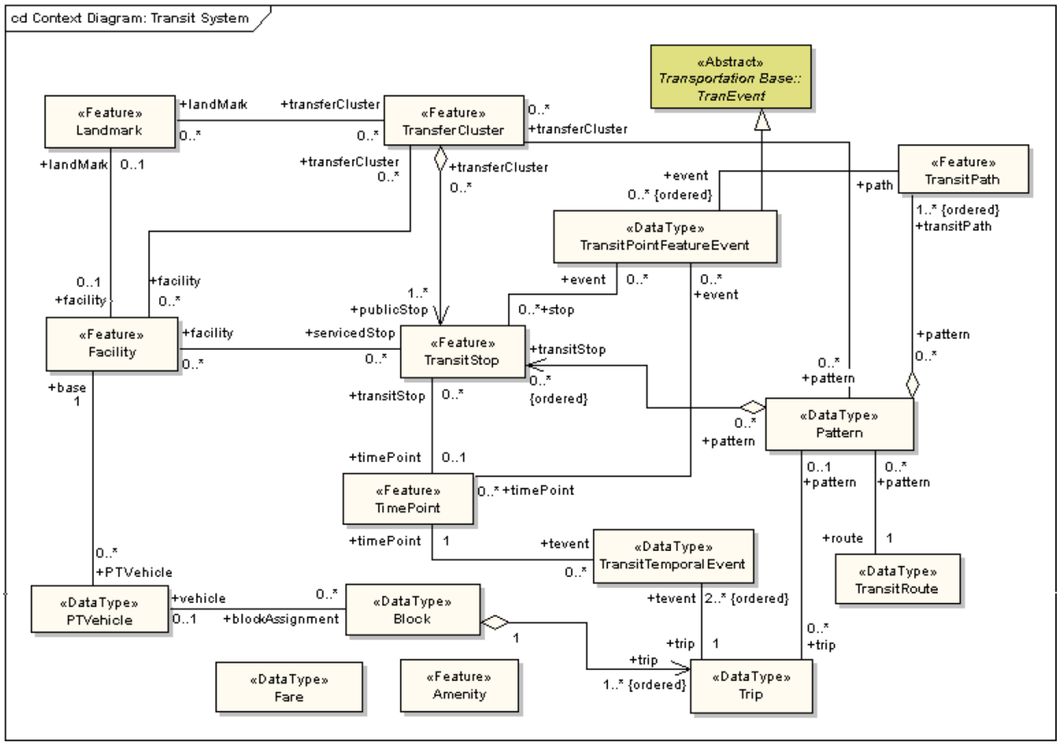


Figure 27: Overview of transit system data model, reproduced from Federal Geographic Data Committee, 2008 c.

As a child of *TranPath*, *TransitPath* inherits topology and geometry attributes, which are both optional: a *TransitPath* may be defined also as an ordered collection of *TransitStop* (or *TimePoint*), or as an ordered collection of transport segments (inheriting geometry or topology from *TranPath*). *TransitPath* may optionally have one or more *Patterns* associated with it (see Figure 29).

A *ConnectionSeg* class is defined as child of *TranSeg* and is the linear path allowing the riders to move from one *TransitStop* to another. Attributes include “distance”, “fromStop”, “toStop”, and connection “Instruction”(s), as can be seen in Figure 30.

TransitStop is the central feature of the transit model: the class represents locations where customers can access the transit system. The class has three mandatory attributes: “identifier”, “lastUpdateDate” and “statusInfo” (with a code list associated). Optional attributes include "stopId", "stopOwner", "Heading" (direction of travel or orientation of a transit vehicle), and different ways to describe the location: "geometry", "relativeLocation", which is a verbose description of the location and has an associated code list with additional attributes (as can be seen in Figure 31), "address" and "alongLocation", describing the linear reference. When the location of a *TransitStop* is described through linear referencing, the *TransitPointFeatureEvent* class must be used. *TransitStop* may be associated with several other transit features, such as *TimePoint*, *TransferCluster, Pattern*, *Facility*, and *TransitPath*.

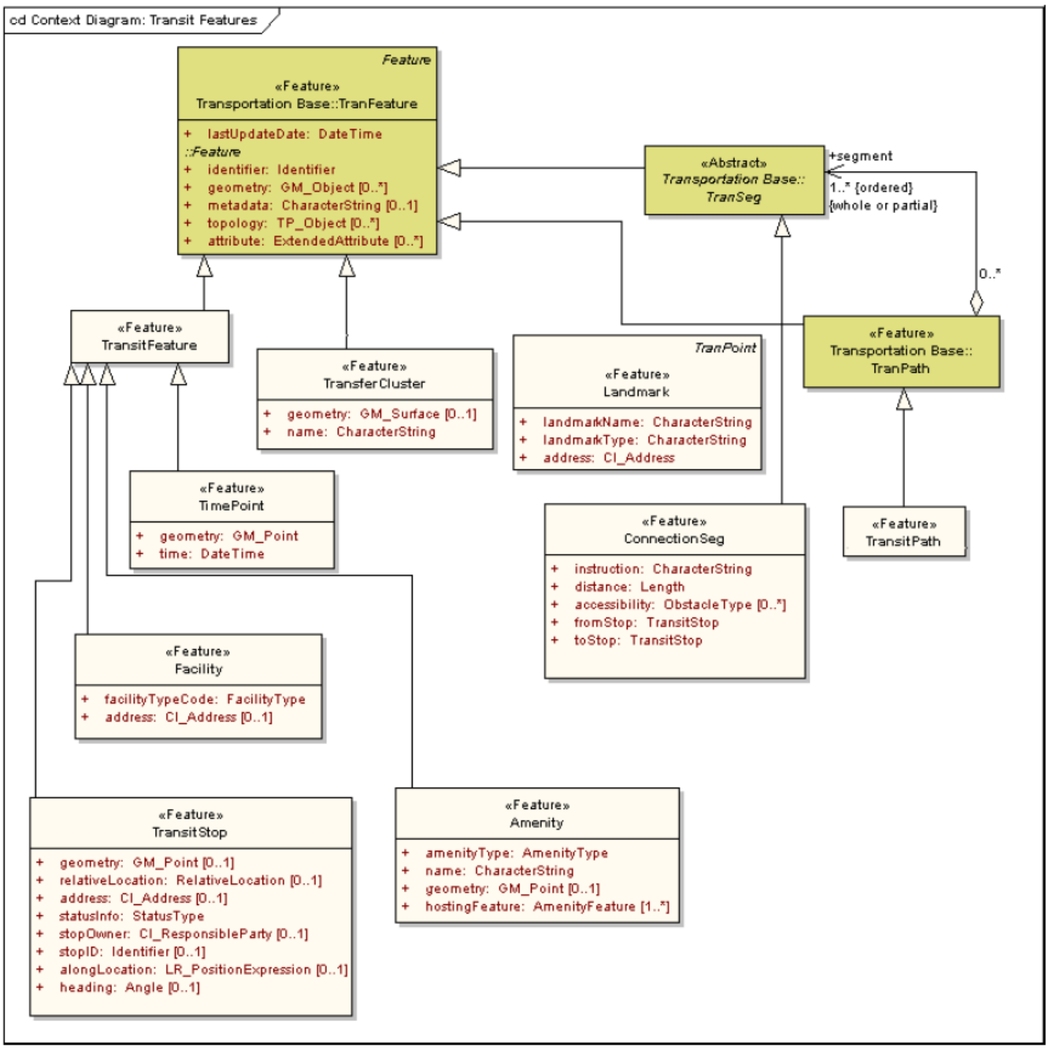


Figure 28: Overview of transit features, reproduced from Federal Geographic Data Committee, 2008 c.

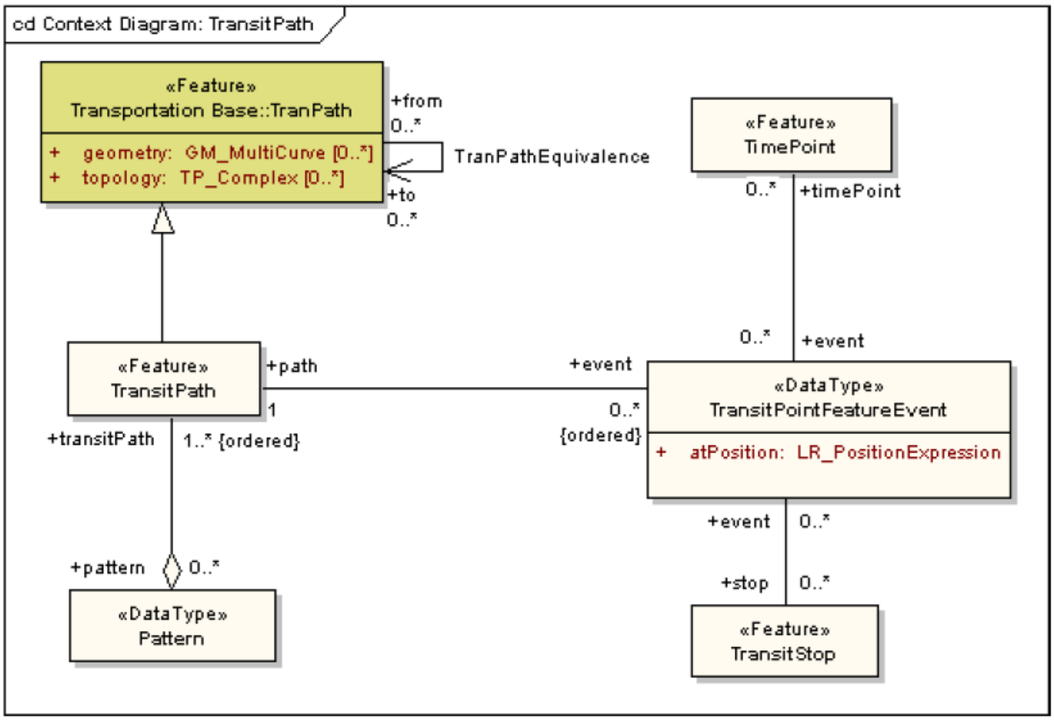


Figure 29: The *TransitPath* class, reproduced from Federal Geographic Data Committee, 2008 c.

The *TimePoint* class represents a location where *Trips* are assigned arrival, dwell, or departure time periods. A *Trip* is a one-way scheduled movement of a transit vehicle between starting and ending *TimePoints*. A *TimePoint* is basically the same thing of a *TransitStop,* but may include also other scheduled stop locations as garage for storage, maintenance or re-fueling. A *TimePoint* has four mandatory attributes, “identifier”, “lastUpdateTime”, “geometry” and “time”. The geometry must be identified by a point coordinate. A *TimePoint* can be associated with zero to many (non-ordered) *TransitTemporalEvents* (see Figure 31).

Other important features characterising the transit model are *TransferCluster*, *Landmark*, *Amenity* and *Facility*, which are all child classes of *TranFeature*.

A *TransferCluster* class describes interchange stations, where transit passengers can change routes and, being a collection of *TransitStop*, it is represented by a polygon geometry. It can be associated with zero or many *Facility*, *Landmark* and *Pattern*.

A *Landmark* is a point of interest, with a point geometry and attributes as "type", "name" and "address". A *Landmark* class can be associated with zero or many *TransferCluster* and with zero or one *Facility*.

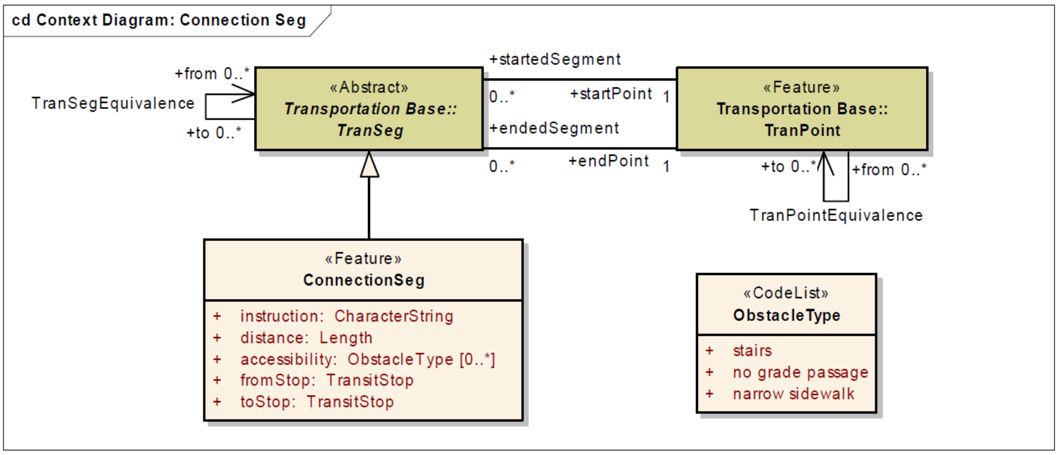


Figure 30: The *ConnectionSeg* class, reproduced from Federal Geographic Data Committee, 2008 c.

A *Facility* is a physical place used by the transit agency, like parking locations or administrative offices. The class has a code list associated indicating the possible type of the *Facility*. It can be associated with zero or one *Landmark* and with one or more *TransferCluster*, *TransitStop*, and *PTVehicle*.

An *Amenity* refers to the elements of a physical feature, a fixed location, or a transit facility, likes shelter, schedule displays, and bike racks. Attributes consist of an optional point geometry, "name", "amenityType" (with an associated code list) and "hostingFeature", which reference a *TransitStop* or a *Facility*.

In the transit data model also several data types have been defined: *Trip*, *Pattern*, *PTvehicle*, *Block*, *TransitRoute* and *Fare*.

The *Trip* data type is a one-way scheduled movement of a transit vehicle between two consecutive *TimePoint* elements. A *Trip* is composed of two or more ordered "times" (*TransitTemporalEvent* instances). Each *TimePoint* in the sequence becomes a *TransitTemporalEvent* of the *Trip*. A *TimePoint* may occur more than once in a single trip, however, each occurrence is a unique temporal event. Each trip is an instance of a *Pattern* and one *Pattern* may optionally be associated with a *Trip*. One or more ordered *Trips* are aggregated to create one *Block*. It can have *Fare* as optional attribute.

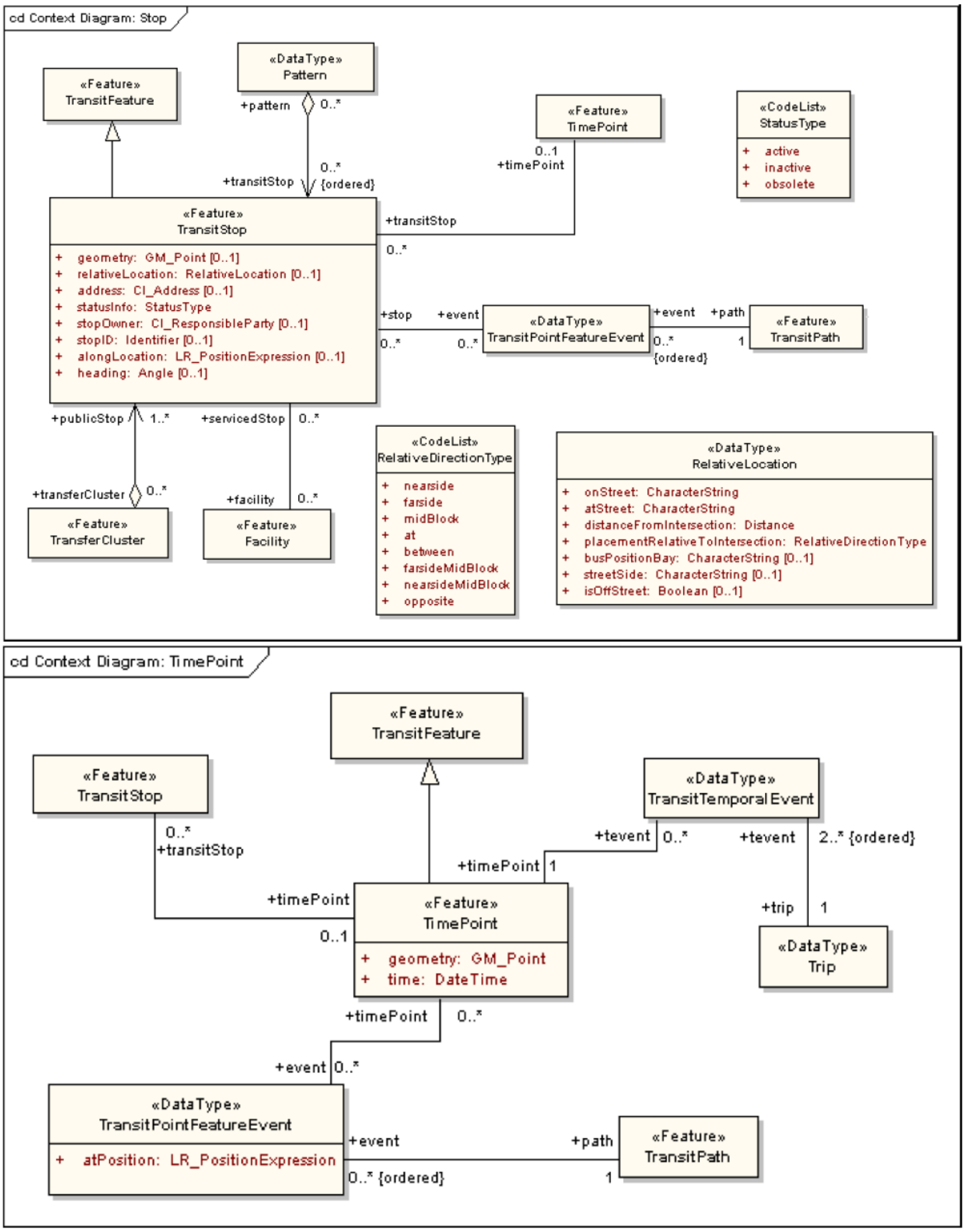


Figure 31: The *TransitStop* and *TimePoint* features model, reproduced from Federal Geographic Data Committee, 2008 c.

A *Pattern* is defined as an ordered sequence of *TransitStops* or *TransitPaths* that is followed by a transit vehicle in scheduled service. While a *Trip* is a construct that is useful to costumers, a *Pattern* is a construct that is useful for transit managers. A *Pattern* may consists of zero to many ordered *TransitStop* or one to many ordered *TransitPath* and it has one *TransitRoute* associated with it. It has several optional attributes: “patternType”, “routeDirection” and “transitServiceType”, which have associated code lists, and “timetableVersion”, to be used when there are multiple time tables associated with a pattern. A *Pattern* can be associated with one or more *TransferCluster* and one or more *Trip.*

A *TransitRoute* is a collection of patterns in revenue service with a common identifier. The *TransitRoute* is the representation of the time-table for a customer that defines a one-way *Trip*. The class has three mandatory attributes: “routeNumber”, “name”, and “timeTableHeader”, which is a summary of publically recognized *TimePoint* contained in a group of *Pattern* oriented in the same route direction, and is used to generate timetables.

PTVehicle is the data type that represents any public transport vehicle. The class has one mandatory attribute, “vehicleId”, optional attributes that describe the vehicle itself as “vehicleCapacity” and “vehicleType” (with a code list associated) and other attributes used for real-time routing and scheduling status. A *PTVehicle* is assigned to one vehicle base or *Facility* (at a time), and zero to many *Blocks* may be associated with a *PTVehicle*.

A *Block* is a sequence of *Trip* over which a *PTVehicle* is assigned from pull out time to pull in time. *Block* includes scheduling information such as “pullInTime”, “pullOutTime”, “pullInBase”, “pullOutBase”, “status”, and “timetableVersion”. There may be one-to-many ordered *Trip* associated with a *Block*, and one *PTVehicle* may optionally be associated with a *Block*.

Finally, a *Fare* is a data type that describes the cost for riding a transit vehicle. It has five mandatory attributes: “fareValue”, “fareType” and “farePolicy” (with associated code lists), “fromStop” and “toStop”, which contains the identifier of a TransitStop.

3.2.3 Public Transport Reference Data Model – TRANSMODEL v5.1 (EN 12896:2006)

Transmodel is the short name for the European Standard “Public Transport Reference Data Model” (EN 12896:2006). The standard aims to facilitate interoperability between information processing systems of transport operators and agencies, developing a common language in order to make EU-wide multimodal travel information services accurate and available across borders to

ITS users, following the Priority Action A of the European ITS Directive 2010/40/EU. Most of the content in this section is readapted from documentation available in the Transmodel Web Site (http://www.transmodel-cen.eu/).

The Transmodel is configured as a reference standard: one of the main objective is to build a **common vocabulary** on which specifiy database structures. It is comprehensive of all aspects of public transport: it can support operational activities of public transport agencies, as drivers and vehicles management, fare management, collection of service performance data, as well as passenger information services. For those reason the standard is useful both for organisations who manage the public transport service and for organisations who develop products for the public transport industry. In particular, the modular structure of the standard allows great flexibility: it can be implemented as whole or partially and it can be extended or reduced in order to be adapted to specific organisation requisites.

Transmodel has been developed within a range of European projects of several European Programmes. The first development of Transmodel has started with the Cassiope project (1989-1991), within the Drive I programme. Cassiope project results were then considerably enriched by the EuroBus and Harpist projects, within the Drive II programme. The Telematics Applications Programme (TAP) project TITAN (1996-1998) continued to validate and enhance Transmodel, implementing it in three European pilot sites and securing the standardisation of Transmodel, which was voted in 1997 as the European pre-standard ENV 12896. Further projects, notably the French SITP and SITP2 projects (1999-2002), further extended and validated the pre-standard.

The CEN TC 278, with the WG 3 - Public Transport, WG 17 - Urban-ITS (created in May 2016) and the Project Team PT0302, are currently in charge of the update of the Transmodel v.6.0.

As Transmodel is a reference standard, there is no need for an agency to implement it as a whole; however it is structured in eight parts which can be separately developed with specific physical data models. Transmodel constitutes an important reference inside the European standardisation process for Intelligent Transport Systems in urban areas. In particular, schemas can be applied to vehicle scheduling or fare management, or to develop interfaces, such as between a ticket machine and a management system, or to integrate data of two neighbouring transport operators.

Each part of the normative is built on other Standards and Technical Specifications, as can be view in Figure 32, which are developed separately but in connection with the Transmodel updates. As most of them have Technical Specifications, the implementation of Transmodel must be follow them.

Operating Raw Data (**OpRa**) is CEN initiative focused on the identification of a set of Public Transport raw data, which can be exchanged, gathered and stored to support studies and monitoring of the Public Transport Service. Between raw data, delays, cancelled vehicles journeys and others operational measures has been considered, defining sampling intervals and possible aggregations.

Standard Interface for Real-time Information (**SIRI**) is a CEN Technical Standard (CEN/TS 15531 series from Part 1 to 5) that specifies a European interface standard for exchanging information about real-time public transport operations between different computer systems. The Standard can be used:

- to provide real-time departure from stop information, real-time progress information about individual vehicles and performance information;
- to manage the movement of buses roaming between areas covered by different servers and monitor the status of operational activities;
- to exchange planned and real-time timetable updates.

Transmodel has been used as first input and currently last updates and extensions of SIRI in 2016 are intended to be taken into account in the relevant parts of the update to Transmodel v.6.0.

The **Identification of Fixed Objects in Public Transport** (**IFOPT**) is a CEN Technical Specification (EN 28701:2009), which provides a reference model for the main fixed object relevant for public transport, in particular for stops and points of interest and related physical points of access. The IFOPT is organised in four related sub models: the *Stop Place Model*, the *Point of Interest Model*, the *Gazetteer Topographical Model*, and the *Administrative Model*. In particular, the Stop Place Model defines a conceptual model and identification principles for places of access for all modes of transport, distinguishing all physical points of access to transport in order to describe the navigation paths between such points, allowing a multimodal journey routing.

Network Timetable Exchange (**NeTEx**) is a CEN Technical Standard (CEN/TS 16614 series from Part 1 to 3) for exchanging data for passenger information, such as stops, routes timetables, fares and related data, among different computer systems (using an XML schema). It is built upon Transmodel and IFOTP, and it is divided into three parts, as can be view in Figure 32, each covering a functional subset of the Transmodel.

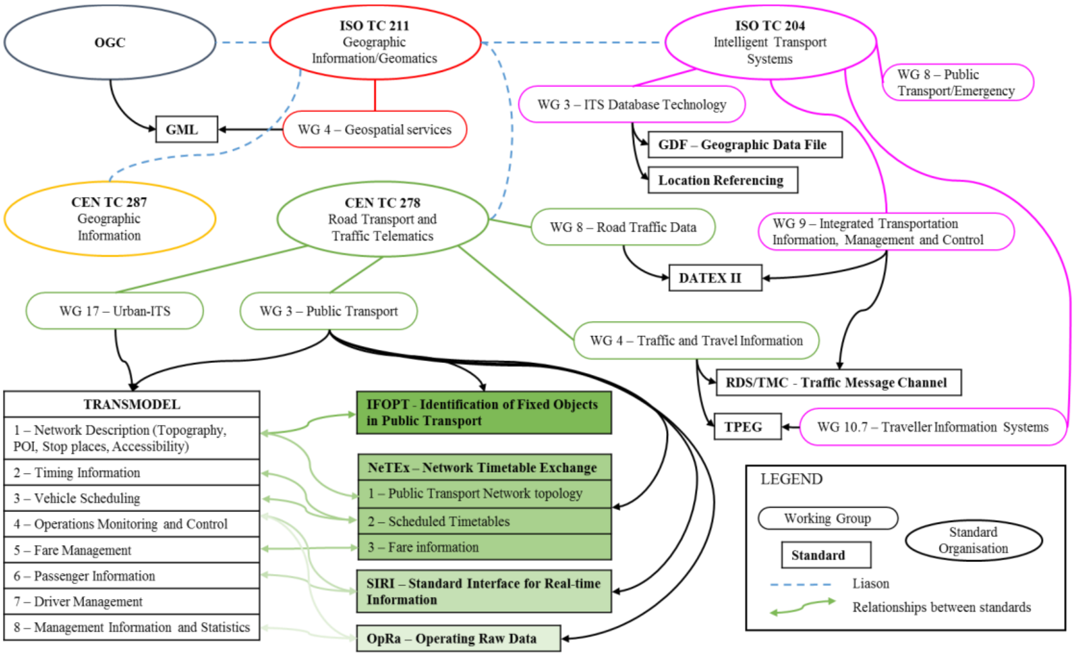


Figure 32: General context of standards for Public Transport in Europe. Readapted from Tibaut et al., 2017.

NeTEx can be used to collect and integrate data from many different stakeholders, and to reintegrate it, as it includes container elements called “version frames”, which allows to group data into coherent sets for efficient exchange across distributed systems. NeTEx schema, thanks to the network topology definition, allows sharing routes with complex topologies, connections between routes, stops and map coordinates. It also allows to model and exchange scheduled information (departures times, frequencies…), exceptions, composite journeys, accessibility to passengers with restricted mobility and fares (structures, products, and prices). From the operational point of view, it also includes the service operator, positioning runs, garages, layovers and duty crews.

Other minor CEN initiatives connected to Transmodel are the *Open API for distributed journey planning* (*OJP*) initiative, which aims to define a single Open Journey Planning API to support all distributed journey planning systems, and the *Data Communication on Vehicles* (*DCV*) project.

The version 6.0 of the Transmodel Standard, with UML schemas and data dictionary, is available for download in the Standard official web site. It is organised in eight part: Common Concepts (CC), Public Transportation Network Topology (NT), Timing Information and Vehicle Scheduling (TI), Operations Monitoring and Control (OM), Fare Management (FM), Passenger Information (PI), Driver Management (DM), Management Information and Statistics (MI). Each package has several sub-packages, each one describing a particular set of concepts. From the available documentation, the last version of the standard is published only for the first three packages, of which main concepts are addressed in this section (only these three parts concern 433 specific terms in the data dictionary).

Transmodel standard tries to clearly distinguish between spatial and temporal concepts. In particular, spatial concepts are covered by the Common Concepts part, in the sub-packages Generic Framework Model and Reusable Components Model, and in the Public Transport Network Topology part, in the sub-packages Network Description Model and Fixed Object Model. Time-related concepts are presented in the Common Concepts part, in the sub-package Reusable Components Model (Service Calendar Model), in the Public Transport Network Topology part, in the sub-package Tactical Planning Components Model (Time Demand Type Model), and in the Timing Information and Vehicle Scheduling part, which contains numerous timing information models.

In the **Common Concept** (CC) part are introduced several concepts shared by the different functional domains covered by Transmodel. In particular, it is composed by five sub-packages:

- the *Version and Validity Model*, which describes the versioning systems of data elements (version elements, frames, delta tables) and the validity conditions for the use of those versioned data,
- the *Responsibility Model*, which describes roles (with predefined list) and responsibilities of organisations over data,
- the *Generic Framework Model*, which describes generic objects and representations not specific of transport domain,
- the *Reusable Components Model*, which describes some common low-level components, widely used in several domain areas of the Transmodel (modes of transport, calendars, vehicles types…), and

- the *Explicit Frames Model*, which describes the mechanism to build coherent sets of versioned data.[7]

In particular, in the *Generic Framework Model* are introduced the generic concepts of *POINT* and *LINK*, where *LINK* may be defined between two *POINT*s to store spatial information (and may be oriented from start point to end point). Main concepts and relevant relationships can be seen in Figure 33.

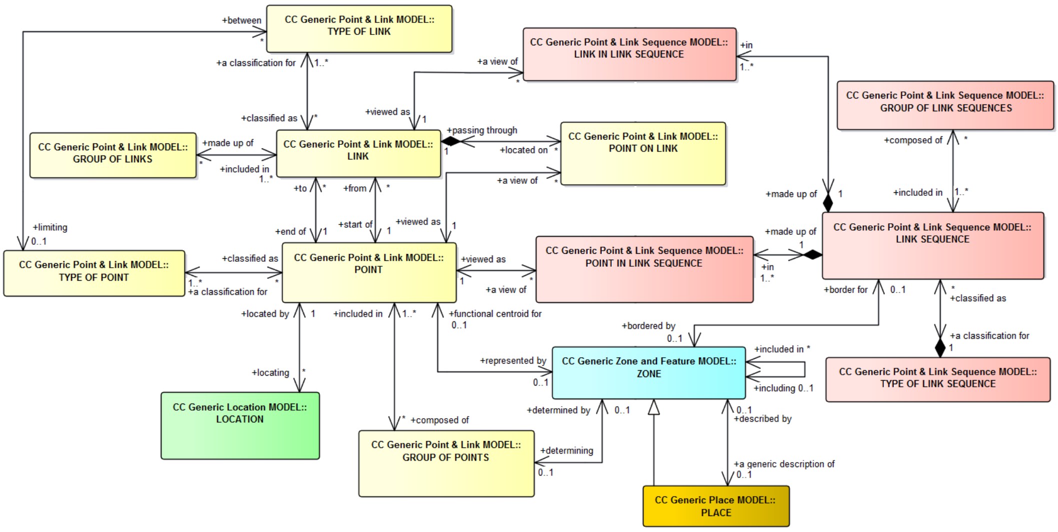


Figure 33: Overview of the *Generic Framework Model* into the *Common Concepts* package. Readapted from TRANSMODEL, 2015.

A *LOCATION* class allows providing representation for spatial objects, using GML schema to define coordinates. A *POINT ON LINK* is defined as a point located along a *LINK*.

LINK SEQUENCE is an ordered set of *LINK*s or *POINT*s that may represent a public transport path. *LINK*, *LINK SEQUENCE* and *POINT* are characterised by a *TYPE* and may be included in *GROUP*s for certain purposes.

A polygonal class *ZONE* is introduced, where groups of entities may belong to a certain *ZONE*, defined by a closed *LINK SEQUENCE* and/or by *GROUP OF POINTS* (as centroid).

Other concepts described are:

---

[7] This sub-package is included in each part of the standard.

- the concept of *PLACE*, as a geographic place of any type which may be specified as origin or destination of a trip;
- the concept of *COMPLEX OBJECT*, as a simple representation for complex objects (e.g. one point to represent a complex station), and
- the concept of *ACCESSIBILITY ASSESSMENT*, which allow to express the accessibility of a site in term of suitability for specific *USER NEEDs* or in term of *ACCESSIBILITY LIMITATIONs*.

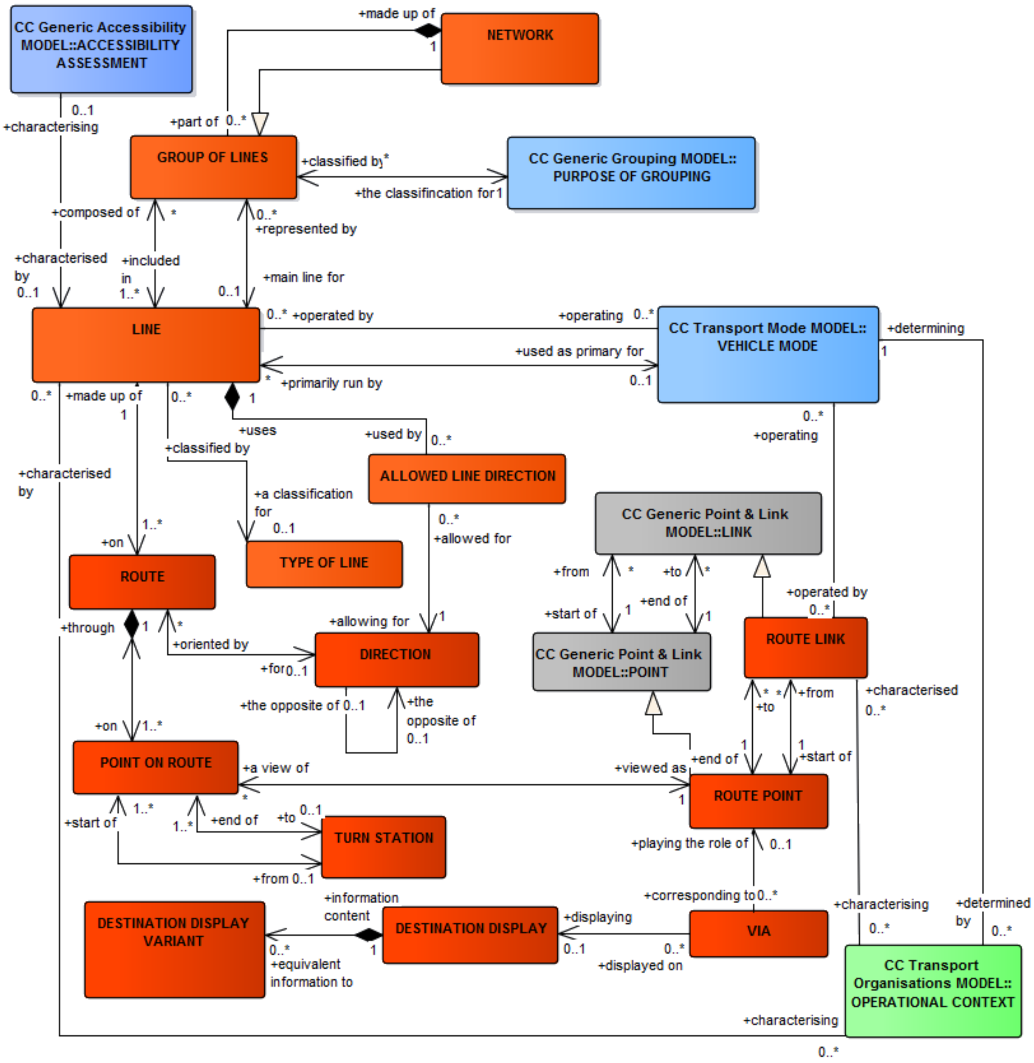


Figure 34: Overview of the *Route Model* into the *Public Transport Network Topology* package. Readapted from TRANSMODEL, 2015.

The package **Public Transportation Network Topology** (NT) has the aim to describe the structure of the network topology and includes most of the fixed object defined by IFOPT. It is organised in three sub-packages (plus an Explicit Frame sub-package).

The *Network Description Model* defines the basic components of an infrastructure (*INFRASTRUCTURE POINT*s and *LINK*s) and their specializations (*ROAD*, *RAILWAY* and *WIRE*). In addition, a range of concepts as *OVERTAKING POSSIBILITY*, *IMPOSSIBLE MANOEVRE* etc., define the possible types of restrictions over the network.

A *ROUTE* is conventional way of describing paths of public transport through the network, is represented as an ordered list of located *POINT*s defining one single path and can be view as a *LINK SEQUENCE*. A *LINE* is a group of *ROUTE*s (very similar from a topological point of view) that is generally known to the public by a similar name or number. A general overview of the Route Model is given in Figure 34. The schema also defines additional properties that can be used to describe flexible systems.

The sub-package *Fixed Object Model* is an extract of IFOPT adapted for NeTEx. It defines spatial concepts as *SITE*, *STOP PLACE*, *POINT OF INTEREST*, *NAVIGATION PATH,* and *PARKING*. In particular, a *STOP PLACE* is place comprising one or more locations where vehicles may stop and where passengers may board or leave vehicles or prepare their trip; it is characterised by one or more well-known names.

The sub-package *Tactical Planning Components Model* provides reusable components useful for service definition and planning. In Figure 35, an overview of *Journey Pattern*, *Service Pattern* and *Timing Pattern* models is given. One of the central concepts in this sub-package is *PATTERN*, a specialisation of *LINK SEQUENCE* on a single *ROUTE* used to define the work of vehicles, which is further specialised in:

- *JOURNEY PATTERN*, an ordered list of *SCHEDULED STOP POINT*s and *TIMING POINT*s on a single *ROUTE;*
- *SERVICE PATTERN*, a *JOURNEY PATTERN* from the passenger point of view, made up of *SCHEDULED STOP POINT IN JOURNEY PATTERN*, and
- *TIMING PATTERN*, a subset of a *JOURNEY PATTERN* made up only of *TIMING POINT*s *IN JOURNEY PATTERN*.

A *SCHEDULED STOP POINT* is a *POINT* where passengers can board or alight from vehicles. Two ordered *SCHEDULED STOP POINT* might be connected through a *SERVICE LINK*, which can be used by different transport modes. A *SCHEDULED STOP POINT* can be viewed as a *CONNECTION END*, enabling the physical (spatial) transfer between *SCHEDULED STOP POINT*s included in the same *STOP AREA*.

A *TIMING POINT* is a *POINT* against which the timing information necessary to build schedules may be recorded: the *TIMING LINK*, comprised by two ordered *TIMING POINT,* could be used to record the run time between two stops.

The **Timing Information and Vehicle Scheduling** package is organised in several sub-packages and defines main temporal concepts and how they are referenced to topological ones.

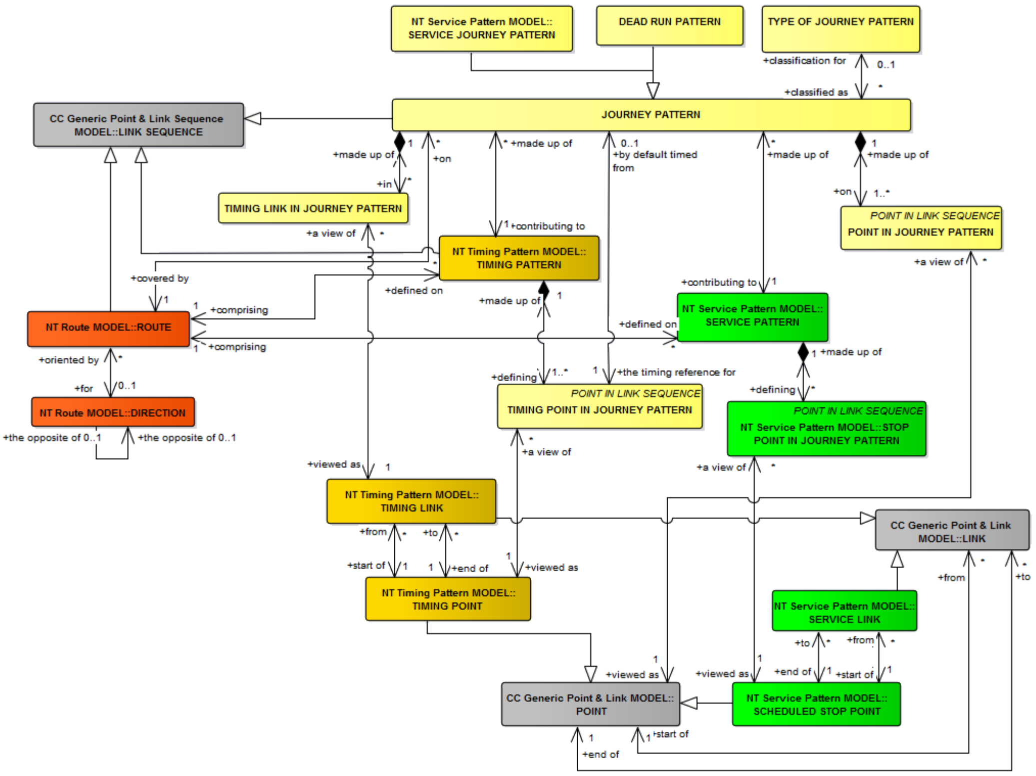


Figure 35: Overview of the *Journey Pattern Model* into the *Public Transport Network Topology* package. Readapted from TRANSMODEL, 2015.

One central concept is *VEHICLE JOURNEY*: the planned movement of a public transport vehicle on a *DAY TYPE* from the start point to the end point of a *JOURNEY PATTERN* on a specified *ROUTE*. In general, Transmodel considers a journey as a purely time-related concept. A *VEHICLE JOURNEY* is only linked to the topological object *JOURNEY PATTERN*.

A *SERVICE JOURNEY* is the specialisation of *VEHICLE JOURNEY* from the point of view of the service offer provided to the passengers. *DEAD RUN* is a specialisation of *VEHICLE JOURNEY* and represents movements necessary to transfer vehicles where they are needed (mainly from the depot into service and vice versa).

PASSING TIME represents time data of public transport vehicle passing a particular *POINT*. A set of *PASSING TIME* linked to a *VEHICLE JOURNEY* (spatial representation) is a *TIMETABLED PASSING TIME* (temporal representation). To calculate the timing of a journey must be take into account the *TIMING POINT* belonging to the *JOURNEY PATTERN* where a "departure time" is specified for the journey at this point, then run and wait times for the different *TIMING LINK*s have to be considered to determine the *PASSING TIME* at *TIMING POINT*.

Finally, an *INTERCHANGE* is an operational time constraint for a transfer: it is defined as the scheduled possibility for transfer of passengers between two *SERVICE JOURNEY*s at the same or different *SCHEDULED STOP POINT*s. Is the temporal representation of a transfer.

From this general description of main concepts, it can be easily assumed the high complexity of Transmodel. Some European countries, nevertheless, have already developed practical implementations of specific parts of Transmodel for public transport management, which can be used as starting point for new possible implementations.

In Belgium, the implementation of Transmodel has been realized through the project *MobilitX*, which aims to gather reference and real-time information related to transport in order to produce added value services and information and deliver information both to transport actors and passengers.

In Sweden and Denmark, the *Nordic Public Transport Interface Standard (NOPTIS)*, represents a Transmodel implementation focused on interfaces, which supports the interconnection of subsystems within a public transport information system, including planning systems, schedule databases, GIS-systems, real-time vehicle reporting systems, traveller information systems and travel-planning systems.

In France, which is one of the leader country in the process of definition of the standard, the already mentioned *TITAN* project in Lyon pilot site has implemented the network topology, the vehicle and driver schedules, timetables and passenger information of Transmodel, validating the v.4.1 of the Standard. Actually, the Greater Lyon Area is one of the examples of a full-scale implementation of Transmodel. More recently the project *Chouette* (open source software for implementing multimodal traveller information systems) has validated the Transmodel v.5.1 through an implementation in the Greater Paris Region.

In United Kingdom, the Transmodel has played an important role in the development and evolution of national standards. Different parts of Transmodel are implemented through three different national standards. The *National Public Transport Gazetteer (NPTG)* provides a topographic context, identifying towns and settlements, and relative bodies responsible for managing public transport data. The *National Public Transport Access Node (NaPTAN)* is a nationwide system for uniquely identify all the points of access to public transport. The *Transport Exchange (TransXChange)* is the standard for exchanging bus schedules and related data. Originally, these were three separated formats with differences in how they described stops and other features. Transmodel gave a systematic basis for converging these into a single national model with shared subschema and a uniform set of entities.

In Italy, the Exbus® system implements its databases following several parts of the Transmodel standard (from Transport Network Topology Model to Driver Management). Exbus is an automatic vehicle monitoring system designed for the monitoring of the public transport service. In particular, it enables real-time monitoring, traveller information, voice and data communication with drivers, automatic data collection, on-board integration with existing or new systems, like fare collection system, passenger-counting system and video-surveillance system. The system is actually used in daily operation in six Italian cities (nearly 2.000 vehicles, and more than 8.500 stopping points).

Another relevant Italian project is BIP (Biglietto Integrato Piemonte), coordinated by 5T agency. The project implements several part of the Transmodel standard, and in particular the Fare Management part. The BIP project has introduced an innovative integrated ticketing system for public transport operators, railways, and for virtually all other transport systems, which allows users to access transport with a single Smartcard. It also introduced an Automatic

Vehicle Monitoring system (AVM), enabling real-time and off-line monitoring of services, and a video-surveillance system for passenger safety. Data exchange among Service Providers, Public Administrations and CSR-BIP (the Regional Service Centre instituted with the project) is made using a tailored version of the NeTEx developed by 5T and called BIPEx. The BIP Project involves over 100 public transport operators, nearly 3.400 vehicles, more than 8.600 stopping points, and nearly 400 train stations.

3.2.4 GTFS - General Transit Feed Specification

Sometimes standards come out for their widely use, becoming a standard "de-facto". One of major example is **General Transit Feed Specification** (**GTFS**), developed by Google and the Portland (Oregon) TriMet transit agency, which defines a common format for collecting and publishing data and for sharing public transport schedules and associated geographic information.

In 2005, the Google employee Chris Harrelson started a side project exploring ways to incorporate public transport data into Google Maps: the first implementation was realised in Portland (December 2005) with the first version of Google's "Transit Trip Planner". In September 2006, five more US cities were added to the Google Transit Trip Planner, and the data format was released as the "Google Transit Feed Specification". In 2009, it was proposed to change the name of the Specification switching from Google to General, which is now part of the official name "General Transit Feed Specification". Indeed, the widespread of GTFS format has drove transit agencies to open their data, but lot of agencies were still not persuaded to share them. Changing the name was a way to increase the potential users of the specification (Roush, 2012).

Looking at the Web page of the "GoogleTransitDataFeed" project[8], there are actually 319 public transport agencies that publish data in GTFS format, of which 12 from Italy (including the Metropolitan Area of Turin, where 5T agency manages the GTFS publication).

A GTFS data feed is a collection of CSV files (with extension ".txt") contained within a ".zip" file, of which six mandatory and seven optional. It is an essential set of tables, with some implicit relationships sufficient to provide a trip planning functionality, which describes the scheduled operations of public transport service from the point of view of passengers. It can be also useful for

---

[8] https://code.google.com/archive/p/googletransitdatafeed/wikis/PublicFeeds.wiki

other applications such analysis of service levels and some general performance measures.

GTFS only includes scheduled operations that are meant to be distributed to passengers. It is also limited to scheduled information and does not include real-time information. However, real-time information can be related to GTFS schedules thanks to the GTFS-Realtime specification, an extension that allows public transport agencies to provide real-time updates about their fleet.

Table 3: Description of main CVS tables composing a GTFS data feed.

Table	Description	Fields
agency	Provides information about the transit agency as such, including name, website and contact information.	agency_name agency_url agency_timezone
routes	Identifies distinct routes. This is to be distinguished from distinct routings, several of which may belong to a single route.	route_id (primary key) route_short_name route_long_name route_type
trips	Trips for each route. A trip is a sequence of two or more stops that occurs at specific time.	trip_id (primary key) route_id (foreign key) service_id (foreign key)
stop_times	Times that a vehicle arrives at and departs from individual stops for each trip.	stop_id (primary key) trip_id (foreign key) arrival_time departure_time stop_sequence
stops	Defines the geographic locations of each and every actual stop or station in the transit system as well as, and optionally, some of the amenities associated with those stops.	stop_id (primary key) stop_name stop_lon stop_lat
calendar	Defines service patterns that operate recurrently such as, for example, every weekday. Service patterns that don't repeat such as for a one-time special event will be defined in the calendar_dates table.	service_id (primary key) monday tuesday wednesday thursday friday saturday sunday start_date end_date

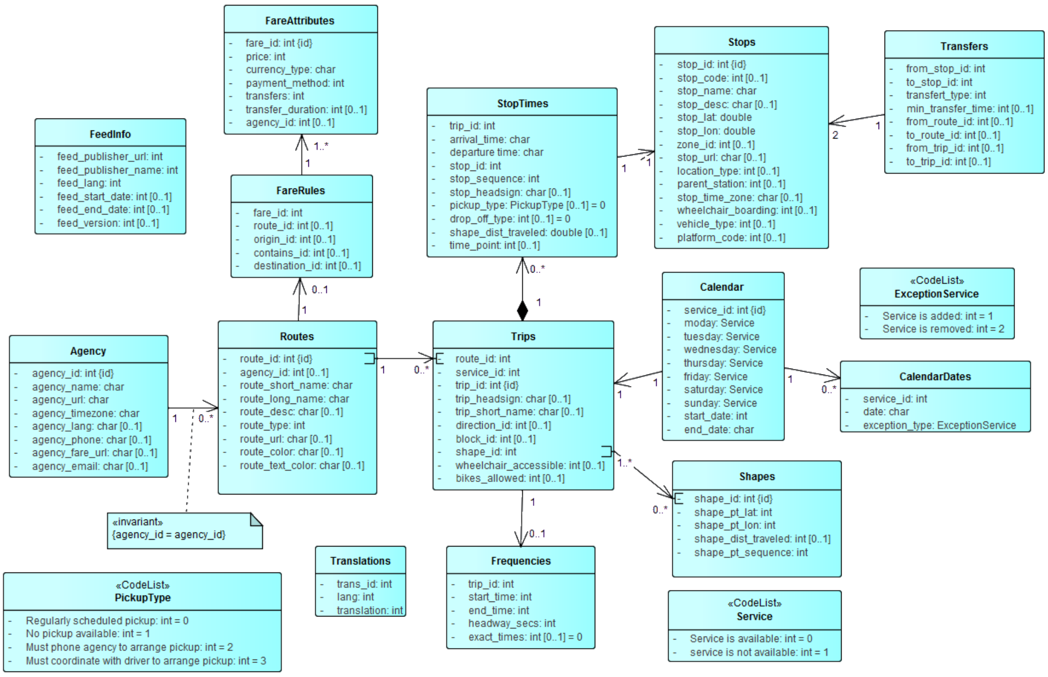


Figure 36: GTFS specification data model.

ciated geographic information.

In 2005, the Google employee Chris Harrelson started a side project exploring ways to incorporate public transport data into Google Maps: the first implementation was realised in Portland (December 2005) with the first version of Google's “Transit Trip Planner”. In September 2006, five more US cities were added to the Google Transit Trip Planner, and the data format was released as the “Google Transit Feed Specification”. In 2009, it was proposed to change the name of the Specification switching from Google to General, which is now part of the official name “General Transit Feed Specification”. Indeed, the widespread of GTFS format has drove transit agencies to open their data, but lot of agencies were still not persuaded to share them. Changing the name was a way to increase the potential users of the specification (Roush, 2012).

Looking at the Web page of the “GoogleTransitDataFeed” project, there are actually 319 public transport agencies that publish data in GTFS format, of which 12 from Italy (including the Metropolitan Area of Turin, where 5T agency manages the GTFS publication).

A GTFS data feed is a collection of CSV files (with extension “.txt”) contained within a “.zip” file, of which six mandatory and seven optional. It is an essential set of tables, with some implicit relationships sufficient to provide a trip planning functionality, which describes the scheduled operations of public transport service from the point of view of passengers. It can be also useful for other applications such analysis of service levels and some general performance measures.

GTFS only includes scheduled operations that are meant to be distributed to passengers. It is also limited to scheduled information and does not include real-time information. However, real-time information can be related to GTFS schedules thanks to the GTFS-Realtime specification, an extension that allows public transport agencies to provide real-time updates about their fleet.

Table 3 lists the mandatory table for GTFS data feed, with the required fields. Other optional tables are "calendar_dates.txt", "fare_attributes.txt", "fare_rules.txt", "shapes.txt", "frequencies.txt", "transfers.txt", "feed_info.txt". The GTFS does not provide a UML model of entities and relationships, but an implicit model can be inferred through reverse engineered process, as can be seen in Figure 36.

3.2.5 TMC - Traffic Message Channel

TMC is the acronym that stands for **Traffic Message Channel** and indicates, globally, a standard service for traffic information distribution. It is a technology for spreading digital traffic information either to final users using the FM-RDS (Radio Data System) on FM radio transmissions or among TIC using DATEX (DATa EXchange) or DATEX II protocols, via dedicated operational nodes. Moreover, using the TMC, all traffic information services can also be transmitted on other transmission channels including digital radio (Digital Audio Broadcasting - DAB), satellite radio, Internet or GSM (Global System for Mobile communications)/GPRS (General Packet Radio Service) mobile network (Arco et al., 2017).

TMC is an ISO/TC 204 standard, acknowledged also by CEN, defined and maintained by the TMC Forum, a non-profit organization. The full name of the norm is *EN ISO 14819 – Traffic and travel information messages via traffic message coding* and is subdivided in four parts (Coding protocol, Event and information codes, Location referencing, Encryption and conditional access). The standard is strictly connected with RDS system and the DATEX protocols. RDS systems is defined by the standard developed by the International Electrotechnical Commission (IEC) *62106:2015*. The DATEX standard is here involved mainly for the *traffic and travel data dictionary*, defined by the *ENV 13106:2000 – Road* *transport and traffic telematics – DATEX traffic and travel data dictionary* (version 3.1.a). The DATEX is still in use by most of TICs and TCCs, but is now replaced by DATEX II, defined by *CEN/TS 16157* series (six parts). Other TMC related standards are:

- the *ENV 12313 - Traffic and traveller information (TTI)* and in particular the part 4 (Coding protocol for radio data system traffic message channel using ALERT-Plus with ALERT-C);
- the *EN 28601 – Data elements and interchange formats – Information exchange – Representation of dates and times*.

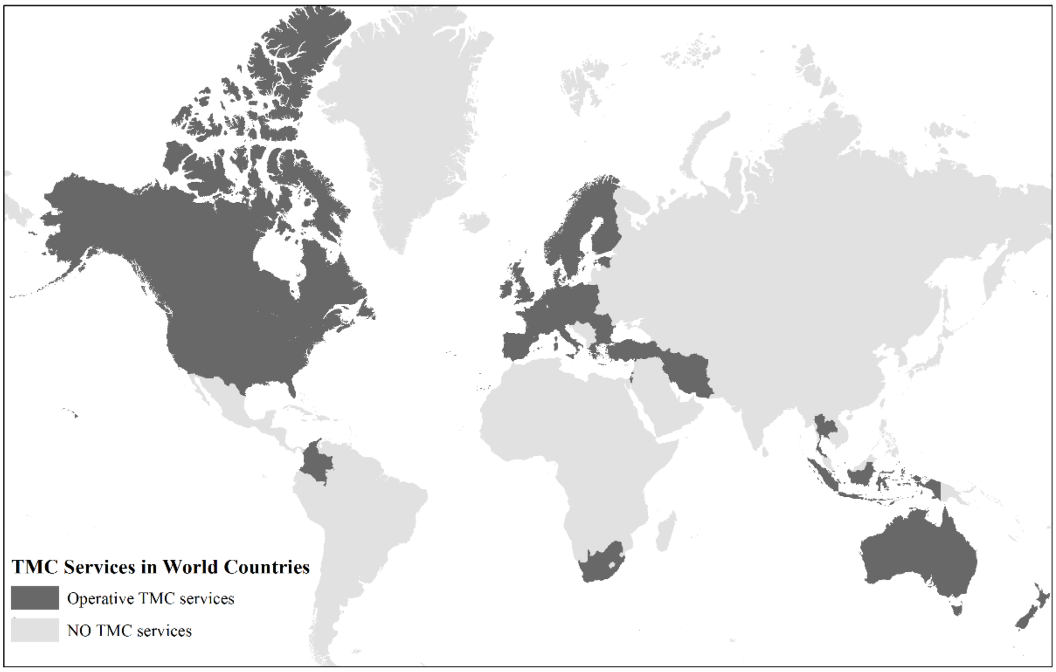


Figure 37: Operative TMC services per country at 2017. Readapted from Wikipedia contributors, 2017 a.

RDS-TMC messages are based on the ALERT-C protocol, which conveys information that is related to real world objects (TMC Forum, 2003). These objects are called locations, and in ALERT-C protocol are identified by a location reference or *Location Code* (LCD). Locations can be points on the road network, specific roads or part of roads, but also areas like municipality or other administrative units. *Location codes* are stored in location tables, together with additional information about the locations. Basically, the TMC standard defines the implementation schema of a TMC locations database, which represents the main road network implementing a sort of raw graph, with a set of points associated to a specific geographical point on the real road (e.g. main intersection, ramps, etc.) and arcs that connect couple of points, coded for machine-to-machine communication.

In this perspective, a TMC Location Code (LCD), since is associated to a specific geographical location on the road network, it is used as reference for TMC events codification and localization. Thus transmitting only the numerical code all TICs and TCCs, as well as all mobile information services of a country, can understand exactly in which locations and on which way the event is located.

The TMC databases are one for each state of the European Union and were made following the standards CEN. This allows, for example, to an RDS-TMC enabled (using the chip card of the Nation); to receive the information and direction pertaining to the country in which the motorist is in that moment. Currently the Italian TMC database consists mainly of points placed on TERN (Trans European Road Network), which includes all roads in major traffic flow (Arco et al., 2017). In Figure 37, the world coverage of TMC services is shown.

The TMC database for Italy (now at version 4.3) is hosted by the "Centro di Coordinamento Informazioni sulla Sicurezza Stradale" (CCISS), National TIC for Italy, which regularly update it and make it available on the Ministry of Infrastructures and Transport web site[9]. It has to be noted that TMC databases are not freely downloadable in each country: publishing it or making it available only to interested users is a national choice.

---

[9] http://www.cciss.it/portale/cciss.portal?_nfpb=true&_windowLabel=quicklinks_1&quicklinks_1_actionOverride=%2Fportlets%2Fquicklinks%2FgoRdsTmc

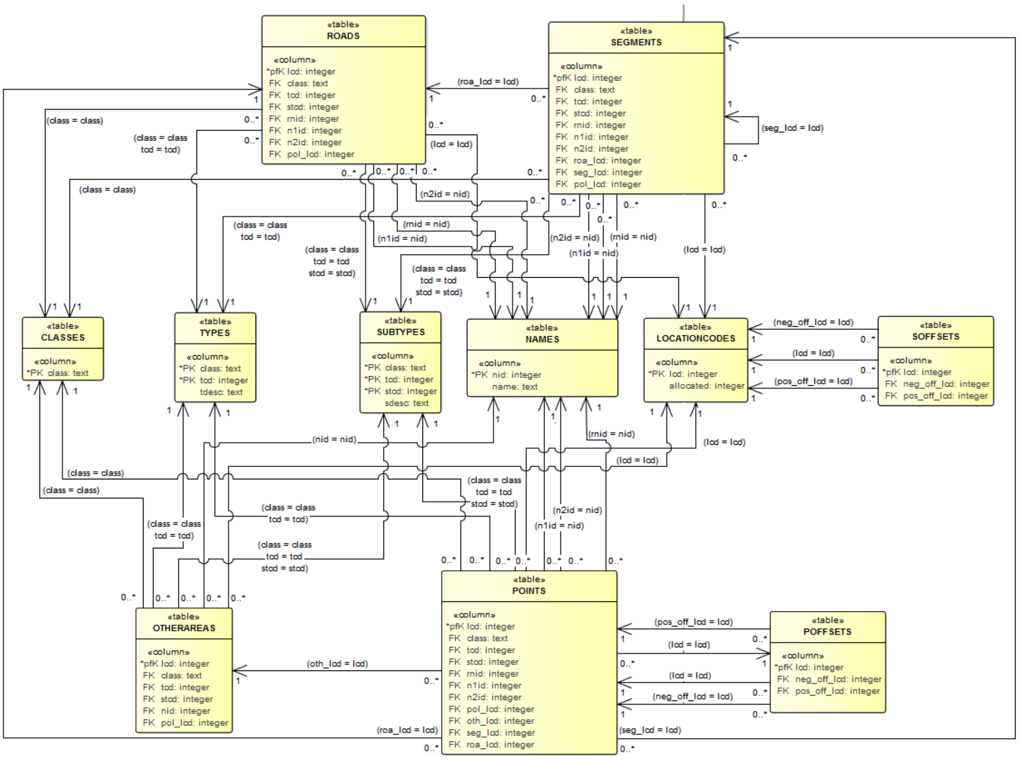


Figure 38: The TMC database schema.

The TMC (Traffic Message Channel) database, with tables, attributes and relationships is shown in Figure 38. A *POINT* of the database TMC corresponds to a well-defined point on the road network, with respect to which they are related, "events" (congestion, delays, etc.) that occur on the roads in the database. Major is the number of points included in the database TMC and roads, and more is the detail with which can be provided information on mobility.

A TMC message consists of an event code, a localization code and some additional information such as the expiration time. A TMC message is encoded and may be encrypted, according to the specifications defined by the TMC protocol. The event code is associated to a list of events that can be translated by a TMC receiver in the user's language. The receiving system decodes messages and can display alerts on the map, via the TMC protocol and the localization tables already integrated in the maps of the major vendors. It can also present the message in text form, translating it into the user's language; all TMC receivers use the same list of event codes as the location database contains a set of country-specific location codes (Arco et al., 2017).

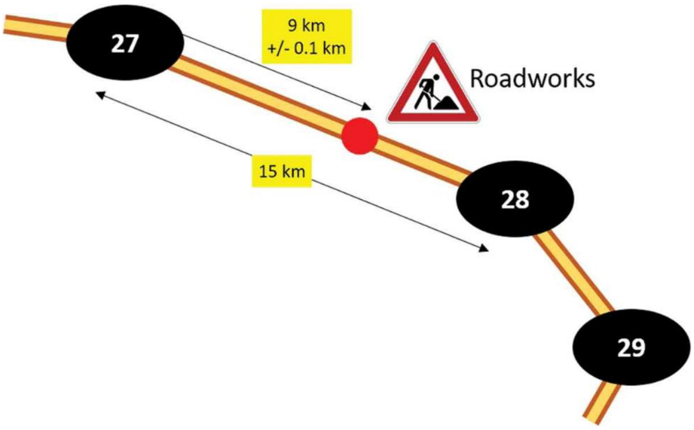


Figure 39: Georeferencing in TMC, readapted from Kamalski, 2005.

The localisation of the event is achieved through three main attributes: the primary location, where the event is located, the extent, which indicates other points involved, and the direction bit, which indicates the direction of the event-affected traffic flow. Linear referencing is applied in order to locate the event along a road, as can be seen in Figure 39. In practise, several fields are used to locate a TMC event. These fields, reported in Table 4, are:

- *Location code*: identifier between 0 and 65536 describing segment of road, intersection or region.
- *Type*: defines the type of location in three main categories and a defined number of sub-categories: Area (A), Linear Location (L) and Points (P). In the above example, L3 identifies a link road, A6.2 describes a metropolitan area, P3.2 a bridge, P1.3 an intersection and P3.3 a service area.
- *Road number*: the street reference number.
- *Name 1*: the name of the primary location that must be displayed by the receiver.
- *Name 2*: the name of a secondary location required to describe a road segment.
- *Ref A*: a pointer to an area where the segment belongs.
- *Ref L*: a pointer to the road section to which the location belongs.
- *Negative offset*: a pointer to the previous area or location (e.g. on the same road segment).
- *Positive offset*: a pointer to the next area or location (e.g. on the same road segment).

Table 4: Localization example, adapted from Arneodo and Gagliardi, 2010.

Location Code	**Type**	**Road number**	**Name 1**	**Name 2**	**Ref A**	**Ref B**	**Negative offset**	**Positive offset**
949	L3	E1	X-town	Y-town	2009	-	948	950
2009	A6.2	-	Greater neighbourhood	-	1	-	-	-
4420	P3.2	E1	Bridge	-	2009	949	4456	4423
4423	P1.3	E1	Place A	N207	2009	949	4420	4459
4459	P3.3	E1	Parking	-	2009	949	4423	4460
4460	P1.3	E1	Place B	-	2009	949	4459	4461

This technology, consolidated, reliable and economical, nowadays allows for the “silent” spread of information not only to FM radio receivers but also to connected satellite navigators enabled to receive TMC signals. Satellite navigators, thanks to the geo-referencing in the RDS-TMC information, visualize TMC events on map in real-time as it occurs on the journey path, allowing the navigator’s computer to calculate and propose new travel path to drivers, according to the real-time traffic conditions (Arco et al., 2017).

An analysis focused on change detection between Version 3.3 and 4.1 of Italian TMC has been performed in the pertinence area of 5T, for internal purposes of the company. The analysis has been achieved through specific queries in Postgresql/PostGIS, which have led to the following results:

- 34 Location Points have changed their geographic position, improving their spatial accuracy, without changing their code;
- 20 Location Points have been eliminated: 4 of them were located on a road that has been eliminated;
- 10 Location Points have been added: 7 of them on a new road.

Since changes between versions were not numerous, the company has decided to implement them manually instead of replacing the whole dataset, reducing the impact on the systems which relying on the TMC data.

3.2.6 DATEX II

The DATEX technology has been developed in the 1990s under projects funded by the European Union and consists of a standard methodology, functions and message structures for the exchange of data between traffic and travel information centres managed by different road operators.

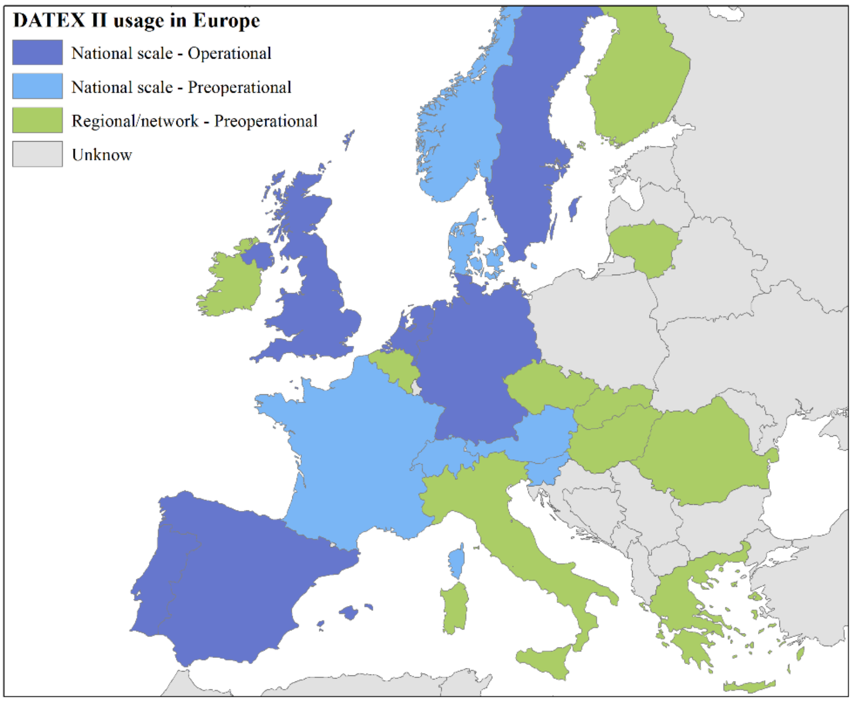


Figure 40: DATEX II usage in Europe as of September 2011, readapted from Dölger and Geißler, 2011.

Currently data exchange between the various national TICs of the EU member states is done in DATEX, enabling users to obtain traffic information across Europe via RDS-TMC. As stated in the previous section, DATEX is now replaced by the DATEX II, published as *CEN/TS 16157*, organised in six parts (Context and framework, Location referencing, Situation publication, Variable Message Sign (VMS) publications, Measured and elaborated data publications, Parking publications). At September 2011, the usage of DATEX II in European countries is shown in Figure 40.

DATEX II improves and enhances the localization mechanisms of previous DATEX, supporting different localization modalities like OpenLR and WGS84 coordinates. Moreover, DATEX II is of relevance for all applications where dynamic information on the transport systems and notably the road system is concerned. The main usage areas are (Dölger and Geißler, 2011):

- Rerouting, network management and traffic management planning. Motorway networks and urban networks are regarded as closely connected here.
- Lane or line control systems and related applications like ramp metering, dynamic speed limits and overtaking control.
- Linking traffic management and traffic information systems.
- Applications where information exchange between individual vehicles and traffic management is crucial, like for car-to-infrastructure systems.

As today the XML is widely acknowledged for information exchange, thus the DATEX II implementation follows this approach, designing an XML Schema to fit a certain area of applications, founded on a data model and a data dictionary. Indeed, the standard allows exchanging several kinds of data through several operating modes, and not all data content and operating modes need be implemented. Each organisation can choose a subset to implement, which is called a DATEX II Profile, allowing great flexibility.

Is not easy to give simple overview of such complex standard. In this try of summary, most of the information are taken by the User Guide documentation available for download in the DATEX II web site (DATEX II, 2006).

The information exchanged through DATEX II is composed of a series of basic elements:

- *Road and traffic related event*: events that are not initiated by the traffic operator and force him to undertake (re)actions. They are classified in six main categories: Abnormal traffic, Accidents, Obstructions, Activities, Incidents on infrastructure equipment, Specific conditions.
- *Operator actions*: Network management, Roadworks, Roadside assistance, Sign settings.
- *Advice*: speed, lane usage, winter driving….
- *Impacts*: delays and traffic status information (free flow, heavy, congested, impossible, unknown).
- *Non road event information*: events which are not directly on the road as transit information, service disruption, car parks.
- *Elaborated data*: data derived on a periodic basis by the Traffic Centre from input data for specified locations, as travel times, traffic status (the same as for impact), traffic values (flow, speed, headway, concentration and individual vehicle measurements), weather values (precipitation, wind, temperature, pollution, road surface condition and visibility).

- *Measured data*: data derived from direct inputs from outstations or equipment at specific measurement sites (e.g. loop detection sites or weather stations), which values are the same of elaborated data.

Those elements can be exchanged individually or grouped. Four type of publications are defined: *Situation publication*, *Elaborated data publication*, *Traffic View publication*, *Measured data publication*.

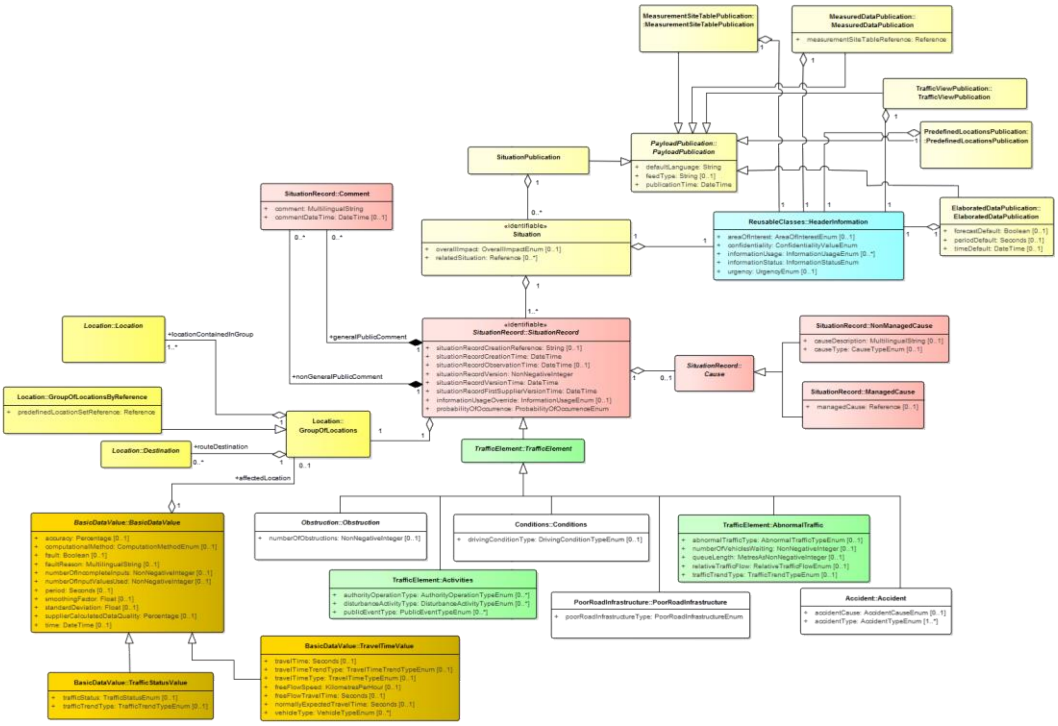


Figure 41: Overview of *Situation Record* and related classes, readapted from DATEX II, 2006.

A *Situation Publication* can contain several different situations. A situation represents "a traffic/travel situation comprising one or more traffic/travel circumstances which are linked by one or more causal relationships and which apply to related locations" (DATEX II, 2006). A *Situation Record* represents each traffic/travel circumstance. It is characterised by a time, which defines the version of this element. A situation record can be: a road or traffic related event, an operator action, a non-road event information. It can also contains advice and impact elements.

An *Elaborated Data Publication* is used to send periodically elaborated data related to specified locations, which may be explicitly defined in the publication or, may be referred to by references to predefined locations already exchanged ("predefined locations" publication). Similarly, a *Measured Data Publication* is used to send periodically measured data at specific measurement sites. Each site is identified by reference through a measurement site table (exchanged via "measurement site table" publication).

The *Traffic View Publication*, introduced only in DATEX II, is a view of all the elements related to a one specific itinerary, in one direction at a given time. A traffic view is organised in "oriented road sections". Each section can contain Elaborated data, Road or traffic related events and Operators actions.

The UML model comprises four diagrams: Analysis diagrams, Dynamic diagrams, Functional diagrams, Logical diagrams. The most interesting part for this research is included in Logical Diagrams. This package contains four sub-packages:

- Exchange: contains exchange diagram and exchange enumeration definitions;
- Payload: contains descriptions of publication data to be exchanged (publication time, publication creator);
- General: contains four general classes: Data types, Location classes, Payload enumerations, reusable classes;
- Management: contains situation management.

In Figure 41 is presented a UML schema illustrating the possible type of publication and their relationships, in particular the *Situation Publication* and the *Situation Record,* and all classes related to the *Situation Record*, as the relevant classes used for location. As can be seen, there are several enumeration, which define the attribute values.

3.2.7 S.I.MO.NE.

S.I.MO.NE., acronym for "Sistema Innovativo di gestione della MObilità per le aree metropolitaNE" (Innovative System for Metropolitan Area Mobility management), is an Italian project, coordinated by 5T company, devoted to implement a Decision Support Systems (DSS) and standard communication protocol to address private mobility management (Arneodo et al., 2009).

One of the main motivation on which the project is built is the growing use of Floating Car Data (FCD) and the need to define a standard way to use and share those data. Floating Car Data is a method to gather traffic information as travel times over specific road sections and traffic congestion events, from vehicles fleets, which act as a dynamic sensors moving through the monitored areas, integrating measures coming from fixed sensors installed on the roads. Floating Car Data systems allow to increase coverage and extensiveness of the monitored area, without adding new physical fixed sensors or infrastructures, and at the same time to improve the timeliness and the significance of the information provided to users (Arneodo et al., 2009).

The project has developed a communication protocol to exchange information from vehicles fleets to Fleet Manager Centres and different TCCs in a bidirectional way, and a software component which aggregates measures. This last aspect, which regards the definition of a level of accuracy of the traffic estimates, respect to the quantity of vehicles in the traffic, and methods to aggregate and integrate measures, will be not deepened in this research. What is of interest here is the structure of the XML Schema realised to exchange those information, as S.I.MO.NE. is considered an Italian Standard, already adopted by several cities as Bologna, Genoa, Florence, Cagliari and Turin (which was the test sites of the project) and have the potential to be spread across the country. In addition, the protocol is built on DATEX II standard, using a similar structure implementation and especially the same code lists and dictionary. This allows an easy translation between DATEX II and S.I.MO.NE. messages.

```xml
<!-- Traffic_data_1.8.xsd -->
<!-- root level traffic_data -->
<?xml version="1.0" encoding="UTF-8"?>
<td:traffic_data xmlns:td="http://www.5t.torino.it/simone/ns/traffic_data"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.5t.torino.it/simone/ns/traffic_data traffic_data_1.8.xsd"
datatype="misura"
generation_time="2014-04-10T10:50:10"
start_time="2014-04-10T10:45:00"
end_time="2014-04-10T10:50:00"
source="a1"
schema_version="1.8">

<!-- location_reference element -->
    <td:location_reference>
        <td:tmc_info tabcd="1" cid="25" />
    </td:location_reference>

<!-- RD_data element example -->
    <td:RD_data veh="526" timestamp="2014-04-10T10:45:10" event="sampling" lat="45.78843" lng="7.54321" vehicle_type="M1" />

<!-- MRD_data element example -->
    <td:MRD_data veh="754" lcd1="1312" offset="54" timestamp="2014-04-10T10:45:10" event="sampling" speed="60" lat="45.78843" lng="7.54321"
    bearing="57" vehicle_type="M1" tracking_distance="55"/>

<!-- TT_data and TT_profiled_data elements example -->
    <td:TT_data lcd1="1310" lcd2="1312" speed="53" q_idx="1" vehicle_type="M1-AU"/>
    <td:TT_profiled_data lcd1="1310" lcd2="1312" speed="53" n_vehicles="15" std_dev="3" accuracy="90" vehicle_type="M1-AU" day_type="FBS"
    start_time="08:00:00" end_time="09:00:00" />

<!-- OD_data and OD_profiled_data element example -->
    <td:OD_data lcd1="1310" lcd2="1312" trips="123" vehicle_type="M"/>
    <td:OD_profiled_data day_type="FBS" start_time="08:00:00" end_time="09:00:00" lcd1="1312" lcd2="1310" trips="321" vehicle_type="M"/>

</td:traffic_data>
```

Figure 42: An example of traffic data XML, readapted from Arneodo et al., 2014.

The protocol considers an XML encoding of data. As the volume of the data is significant, it has been chosen to develop two operating modes for publishing data: a *push mode*, where the data provider periodically arranges and sends data, and a *pull mode*, where data provider periodically arranges data and the consumer explicitly request it according to their own timing and needs.

Three XSD (XML Schema) have been developed (freely downloadable from S.I.MO.NE. project site):

- Traffic_data_1.8.xsd: manages traffic data gathered from FCD and other sensors, and following aggregation as historical profiling (traffic monitoring and control);
- Traffic_info_1.5.xsd: manages traffic events (info mobility);
- Access_control_1.1.xsd: manages Restricted Access Area (RAA).

Looking at the "Traffic_data_1.8.xsd" structure (see Figure 42), a root element *traffic_data* is defined, as a container for all traffic data types: it have attributes defining the type of data contained, the time generation of the data, the start and end validity time for data and the identifier of the data provider. The following elements are all child of the root *traffic_data*.

The *location_reference* element define the spatial reference. It has an attribute WGS84 indicating whether or not coordinates are in WGS84, and two child element: *tmc_info*, which contains information about the version of the TMC network used, and *detailed_graph_info*, which contains the reference to a generic detailed graph shared between parties.

Following elements define the possible data types (see Figure 42), which can be included in this XML (Arneodo et al., 2014):

- *RD_data – Raw data*. FCD gathered from private vehicles or from public transport fleets and commercial fleets (distinct through the attribute vehicle type).
- *MRD_data – Map-matched Raw data*. FCD with a partial elaboration: the GPS position is already referred to a road network (TMC or other).
- *TT_data & TT_profiled_data – Travel Time data*. Travel time along an edge of the network (derived from FCD), eventually profiled by day type.
- *OD_data & OD_profiled_data – Origin Destination data*. Number of trips for each couple of origin – destination. Attributes are lcd1 and lcd2, defining respectively origin and destination zone, vehicle type and number of trips, eventually profiled by day type.

- *FDT_data – Traffic Flow data*. Aggregated data gathered from fixed sensor or deduced from further information. Relevant attributes describe information about fixed sensor, location information (as TMC references or WGS84 coordinates), measures information and other related information.
- *TLight_data – Traffic Light data*. Duration of the traffic light cycle and green rate with respect to this duration, in correspondence with a determined traffic light manoeuvre.
- *PK_data – Parking data*. Real-time information about parking occupancy. Attributes identify the parking area, locate it along a road or by coordinates and give information about capacity, actual occupancy and tendency in real-time.
- *Tstate_data – Traffic state data*. Aggregated data related to real-time or forecasted flow conditions. Localisation is given by two location code, a wide range of quantitative and qualitative measures is given.

```
<!-- Traffic_info_1.5.xsd -->
<!-- root level EVTS_GEN_DELIVERY -->
<?xml version="1.0" encoding="UTF-8" ?>
<EVTS_GEN_DELIVERY xmlns="http://www.autostrade.it/schema/road-events"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.autostrade.it/schema/road-events
http://simone.5t.torino.it/ns/traffic_info.xsd" date="2014/06/10 16:42:45" numevts="43" supplier="ITTOC" schema_version="1.4" >
<!-- EVTS_RECORD_ROAD element example -->
    <EVTS_RECORD_ROAD for="N">
        <SOURCE>ITTOC</SOURCE>
        <SITUATION_ID>A11382353357_40</SITUATION_ID>
        <EVT_ID>1</EVT_ID>
        <ROAD desc="SS33 - Del Sempione">351</ROAD>
        <LOCATION>
            <LOC_SEC desc="Iselle" km="3.950" x="8.199620" y="46.207290">5152</LOC_SEC>
            <LOC_PRI desc="Sempione/Svizzera" km="3.950" x="8.152470" y="46.196310">5153</LOC_PRI>
        </LOCATION>
        <CAUSE desc="Lavori in corso">RMT</CAUSE>
        <EVENT desc="senso unico alternato">SAT</EVENT>
        <DIVERSION>n</DIVERSION>
        <DIRECTION desc="Entrambe">3</DIRECTION>
        <DATE>
            <EVT_START_DATE>2013/10/07 00:00:00</EVT_START_DATE>
            <EVT_UPDATE_DATE>2013/10/21 16:06:00</EVT_UPDATE_DATE>
            <EVT_END_DATE>2014/09/04 17:30:00</EVT_END_DATE>
        </DATE>
        <ADDING_INFORMATION />
        <ATTRIBUTES>
            <ATTRIBUTE code="PHC" desc="Causa lavori in corso" um="">RWK</ATTRIBUTE>
            <ATTRIBUTE code="SUR" desc="Note" um="">REGOLATO DA IMPIANTO SEMAFORICO</ATTRIBUTE>
        </ATTRIBUTES>
    </EVTS_RECORD_ROAD>
</EVTS_GEN_DELIVERY>
```

Figure 43: An example of traffic info XML, readapted from Arneodo et al., 2014.

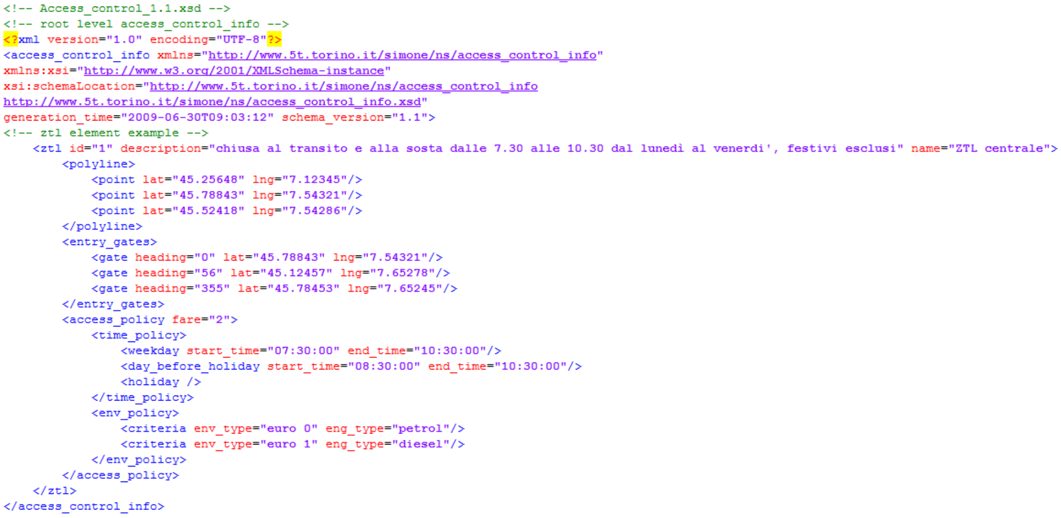
The “Traffic_info_1.5.xsd” is designed to manage traffic events. Following DATEX standard, a traffic *situation* is identified, which includes a series of traffic *events* related by the same cause. The protocol also adopts the same code lists that describe the type of the event, the “Data Object” (DOB) and the “Phrase” (PHR). The structure is lighter than the previous one (see Figure 43): it contains a root element *EVTS_GEN_DELIVERY*, with some associated attributes that identify it (supplier, number of events, date). The child element *EVTS_RECORD_ROAD* includes all child elements useful to describe a single traffic events: these elements are characterised by a “desc” attribute and eventually a “code” attribute, and are described below (Arneodo et al., 2014):

- *SOURCE*, the identifier of the event provider, which has validated and inserted the event;
- *SITUATION_ID*, the identifier of the situation to which the event belongs;
- *EVT_ID*, the identifier of the event within the situation;
- *ROAD*, TMC location code of the road along which is localized the event, where the attribute “desc” include the common name of the road;
- *LOCATION*, block composed of sub-elements LOC_SEC and LOC_PRI that localises the event: the values are the TMC location code and other attributes allow to add coordinates and a linear measure;
- *CAUSE*, the DATEX DOB code of the event, with its verbose description;
- *EVENT*, the DATEX PHR code of the event, with its verbose description;
- *DIVERSION*, indicates whether or not a deviation is occurred (Boolean);
- *DIRECTION*, code that identifies the direction (0 – no information, 1 – positive TMC direction, 2 – negative TMC direction; 3 – both direction, 4 – direction is specified only in the “desc” attribute);
- *DATE*, block composed of sub-elements which indicates the start, the update and the end date and time of the event;
- *ADDING_INFORMATION*, block composed of a series of *ATTRIBUTE* elements, which are additional attributes describing the event using a DATEX encoding;
- *SAD*, Additional advice and information on the event.

Finally, the “Access_control_1.1.xsd” manages Restricted Access Area (RAA). The schema allows to describe the perimeter of the RAA, the position of the gates and the associated access rules (see Figure 44). The root element *access_control_info* has some attributes describing the XML (generation time, schema version) and a child element called *ztl* that identifies the area subject to access control, through “id”, “name” and “description” attributes. The *ztl* element includes (Arneodo et al., 2014):

- *polyline*, block composed of a series of ordered point element, characterised by a latitude and a longitude attribute, which defines the polyline that delimitate the area;

- *entry_gates*, block composed of a series of gate elements, characterised by a heading, a latitude and a longitude attribute, which identify the access point to the area;
- *access_policy*, block composed element that describe the access policy of the area. The policy is defined by the two child elements time_policy and env_policy, which describe time windows and vehicles allowed.



```
<!-- Access_control_1.1.xsd -->
<!-- root level access_control_info -->
<?xml version="1.0" encoding="UTF-8"?>
<access_control_info xmlns="http://www.5t.torino.it/simone/ns/access_control_info"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.5t.torino.it/simone/ns/access_control_info
http://www.5t.torino.it/simone/ns/access_control_info.xsd"
generation_time="2009-06-30T09:03:12" schema_version="1.1">
<!-- ztl element example -->
    <ztl id="1" description="chiusa al transito e alla sosta dalle 7.30 alle 10.30 dal lunedì al venerdi', festivi esclusi" name="ZTL centrale">
        <polyline>
            <point lat="45.25648" lng="7.12345"/>
            <point lat="45.78843" lng="7.54321"/>
            <point lat="45.52418" lng="7.54286"/>
        </polyline>
        <entry_gates>
            <gate heading="0" lat="45.78843" lng="7.54321"/>
            <gate heading="56" lat="45.12457" lng="7.65278"/>
            <gate heading="355" lat="45.78453" lng="7.65245"/>
        </entry_gates>
        <access_policy fare="2">
            <time_policy>
                <weekday start_time="07:30:00" end_time="10:30:00"/>
                <day_before_holiday start_time="08:30:00" end_time="10:30:00"/>
                <holiday />
            </time_policy>
            <env_policy>
                <criteria env_type="euro 0" eng_type="petrol"/>
                <criteria env_type="euro 1" eng_type="diesel"/>
            </env_policy>
        </access_policy>
    </ztl>
</access_control_info>
```

Figure 44: An example of access control info XML, readapted from Arneodo et al., 2014.

3.2.8 CityGML

In order to give a complete background of spatial data models also the CityGML has been taken into account. Information in this section are mainly derived from the “OGC City Geography Markup Language (CityGML) Encoding Standard” document (OCG, 2012).

CityGML is a semantic model that wants to respond to the emerging exigence of building virtual 3D city models for different application purposes. It is an XML-based model that uses the application schema of GML 3.0. CityGML defines classes and relations for the most relevant topographic objects in cities and regional models with respect to their geometrical, topological, semantical, and appearance properties. It also includes generalisation hierarchies, aggregations, relation between objects and spatial properties. In particular, CityGML gives lot of attention to surface topology and relationships and surface representation, decomposing each object in surfaces with well-defined relationships between them.

Between relevant features, the CityGML standard provides a quite complete ontology for urban landscape and specific elements for 3D representation: a 3D geometry definition based on standard (*ISO 19107*) and elements as textures and materials, which allow to control the surface styling process through computer graphics. In addition, a strength of the model is the multiscale organisation: five well-defined Level of Detail (LOD), from 0 (regional level) to 4 (interior architectural models), facilitate an efficient visualisation and data analysis, and enable for a same object different degrees of resolution. Finally, CityGML provides Application Domain Extensions (ADE), which allow to define application specific extensions.

The CityGML is thematically decomposed into a core module and thematic extension modules (CityGML profiles). Profiles of interest in this research context are: Transportation, Tunnel, Bridge as regards the representation of road network, some elements of CityFurniture, useful to describe elements as stops, signs, or traffic lights, and the Generic module, which allows to defines specific objects and attributes not covered by other modules.

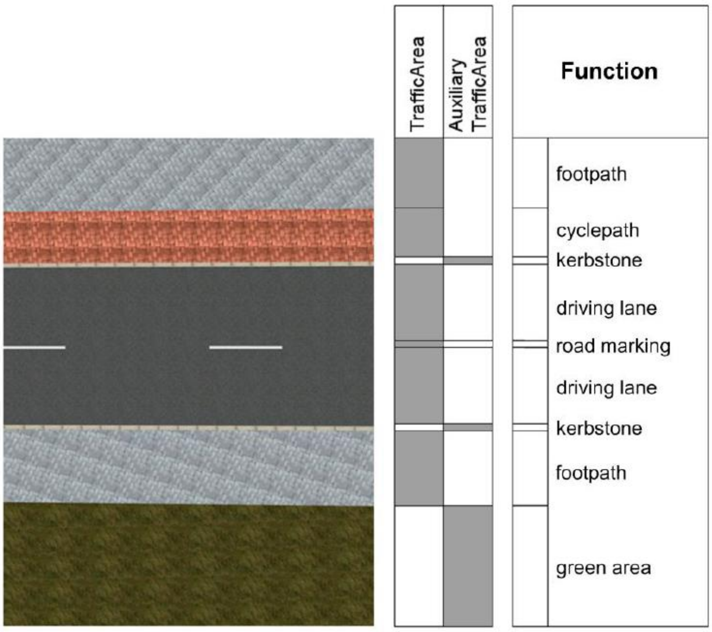


Figure 45: Example of a road with a functional classification. Reprinted from OGC, 2012.

In particular, the Transportation module is focused on the functional and thematic representation and on geometrical and topological aspects: linear networks are used in the LOD0 (comprising also tunnel and bridge paths) and from LOD1 transport features are described as 3D surfaces (where tunnel and bridge are described in a separated module). In Figure 45, an example of a road of LOD2 is given, differentiated by functional classification (which follow a predefined code list).

Specific aspects of transport characteristics are not taken into account by the standard, as the purpose is intended to complement and not replace existing standards: however, if specific objects or functional aspects have to be associated with CityGML transport objects, generic objects and attributes provided by CityGML’s Generics module can be used. In addition, the flexible implementation of the standard, which allows the use of ExternalReferences, may guarantee the connection of specific transport models and the CityGML in a unique environment, enabling the use of specific features of transport models (e.g. speed limits) inside the CityGML model.

The 3D representation aspect is not considered in this research: indeed, most transport applications related to mobility management do not rely on 3D models, which may also need high computing performances to manage this huge flow of information. Possible applications of the 3D model in the transport context can be the evaluation of the diffusion of air and noise pollution generated by private traffic, training simulator for guidance, or eventually 3D models as support for autonomous driving, working in connection with on board unit on cars.

3.3 Considerations on standard data models

The standards described in previous sections are related to different aspects of mobility management, depending on their purpose. It is possible to divide them in three main groups, despite some overlaps may exist between them: standards related to transport network representation and management (INSPIRE and FGDC – Roads, CityGML), standards related to public transport management (TRANSMODEL, GTFS, and FGDC – Transit) and standards related to info mobility (DATEX, S.I.MO.NE and TMC).

From the point of view of the **purpose** of the modelling activity, a similarity exists among INSPIRE and FDGC - Road: indeed both are chiefly designed to allowing the data sharing across countries (European countries for INSPIRE, States for the USA). In addition, they are well structured for road visualisation (geometric representation) and categorisation (thematic representation), and thought also for routing purposes, with a well-defined topology schema and connectivity rules. CityGML instead is mainly tailored for 3D visualisation and analysis.

The INSPIRE Road Transport Network data model lacks of several elements useful for traffic and mobility management: in particular elements as induction loops, cameras, traffic lights, informative panels does not have a definition, even if the model contains classes (as *SimplePointReference* in the Network Application Schema) that can be further specialised in order to represent those kind of objects. In the same way, the class *RoadLinkSequence* can be eventually specialised to represent a public transport vehicle journey. The problem is not related to the possibility of class specialisation and customisation where possible, but in the lack of a precise dictionary, which can allows an unambiguous definition and interpretation of the new defined object. Indeed, road traffic management as public transport management are out of scope of the INSPIRE Directive.

On the other hand, the Road Framework defined by FGDC is generally more flexible in class definition than INSPIRE, allowing to be extended in several ways. As for INSPIRE, there is not a precise dictionary of traffic management elements, but the documentation contains several practical implementations which can be used as reference to easily implement specific extensions. A strength of this standard is related to its strong link among road and transit part: the road and the rail models represent the infrastructure on which the transit service is build, assuring a solid integration between them, in particular from a spatial and topological point of view.

The CityGML standard generally lacks in the definition of elements and dictionary related to ancillary traffic elements. It also has a poor topological definition respect to classical network model for routing purposes (edges-nodes), even if quite deepened for areal routing.

Looking at the *public transport data models*, pronounced differences arise from different purposes. In Europe in particular, where the public transport assets are important and well developed, the situation varies across countries: service provision can be offered by single or multi-agency configurations, operated by both public and private sector organisations, and there is a strong exigence of transport modes integration, which leads to complex fares management. As European Standard, the Transmodel reflects the complex European situation and aims to model public transport from all points of view, supporting back office and operational systems managements, fleets monitoring and management, fares management, real-time data production and communication and passengers' information as well as geometric representation.

On the other hand GTFS and FDGC – Transit Framework ensue from the USA context, where public transport is less developed and with a minor grade of complexity. In particular, the GTFS is clearly devoted to passengers' information: all the elements are structured in order to be easy interpreted by the riders. The Transit Framework indeed is especially designed for data sharing across states (of both spatial and temporal data) in order to support itinerary planning but also infrastructure inventories and re-routing applications. In addition, it aims to define a common set of definitions for real-world transport features useful for both users and producers. In comparison with the Transmodel, GTFS and FDGC – Transit Framework lack in supporting management and back office operations, and they do not offers a complete multimodal and fares management model.

Finally, as regards standards for info-mobility, different perspectives are considered: the DATEX and S.I.MO.NE standards are mainly a communication protocols devoted to share and spread private traffic information among operators and users, while TMC constitutes a common reference for operators for localisation on the network. In addition, the standards related to public transport have some features devoted to info-mobility, with different level of detail.

Trying to make a comparison between standard approaches two point of view has been considered: the spatial and the temporal point of view.

From a spatial perspective, first is important to looking for a **geometry definition**: the *ISO 19107 – Spatial Schema* is the standard used both in INSPIRE, CityGML and in FGDC Framework. The Transmodel indeed do not specify any standard for geometry definition, even if data types are closer to the *ISO 19107* ones. TMC and GTFS uses only point geometries (routes shapes are optional in GTFS), eventually as ordered sequences, defined using WGS84 coordinates. DATEX and S.I.MO.NE as communication protocols do not implement geometry object, but location-referencing systems supports WGS84 coordinates as coded locations as TMC, with point and linear referencing methods compliant to *ISO 19148 – Linear Referencing*.

From the point of view of **spatial objects semantic**, is possible to compare the different configuration of several concepts as network, infrastructure, junctions and point features, edges and linear features, areas. In addition, a look has been given to topology implementations, connection rules and direction of flow management.

The *network* concept is present in all standards, whatever as a defined class or only as concept. Network is used a grouping element, which comprehend a set of network elements, usually point and line, but also areas (as in the case of INSPIRE). Grouping network elements can be done for various reasons, and only in Transmodel the group is related to a set of validity conditions under which the network is usable.

Point features are defined and managed in various ways. INSPIRE directive mainly defines *nodes*, as significant position in the network always at the beginning or at the end of a link, used to represent junctions and intersections, and to assure the connectivity between links. The *simplePointReference* class indeed does not have an own point geometry, as it is only a data type and not a feature, but can be used to locate point objects other than junctions along the network. The FGDC Road Framework uses the concept of point, as a *TranPoint* with its own geometry when the point represents a junction. The Framework allows also the representation of points which are not a junction through: the definition of a new user-defined *TranFeature* element, or the use of the *PointFeatureEvent* class, with a linear referenced position on a segment, which allows its representation. In addition, looking at the Transit Framework, classes as *Amenity* and *Landmark* allow the definition of point objects that can represent ancillary elements and points of interests.

As already stated, DATEX and S.I.MO.NE allow the representation of point through WGS84 coordinates and through reference to well-known locations as the TMC locations. In particular TMC locations in DATEX are defined by the *AlertCLocations* class. Another DATEX class particular useful for traffic management is the *MeasurementSite*, a point (but also a stretch of a road) from which a stream of measured data come from or may be derived.

Linear features are usually called *links.* In the INSPIRE Directive a link is a homogenous path that connects two positions, substantially representing a stretch of a road or of a railway (centrelines abstraction is preferred). Also the Transmodel uses the term link. The FGDC Framework uses the term *Segment* to identify links, as portion of road or of a railway. In the TMC the *road* is the object that connects two points (here also segment is defined, but with a different meaning respect to FGDC). DATEX and S.I.MO.NE allow a linear representation of an event using, for instance, the road location code of TMC combined with linear referencing methods. In general, a link represents paths: for instance in Transmodel but also in FGDC framework, linear features are used to represent access paths and movements between stops, in order to assure connections among journeys.

In most of standards a link can have or not a spatial representation (logic versus physical representation). For instance, in Transmodel the spatial representation of a link is demanded to the link projection, where the link object is associated to a datum and projection for a spatial representation. In the TMC instead, linear objects are defined only at logical level.

In addition, links are always characterised by a **direction of travel**, which defines their navigability. This can be achieved through attributes as “spokeStart/spokeEnd” or “StartNode/EndNode”, which allow to identify which is previous and next connection element. Alternatively, direction of flow can be defined also as positive or negative (for instance respect to their digitalization direction). In DATEX for instance, direction is defined in two ways: in case one location point is given, one direction of flow of an event is associated (with or without an offset), if two location points are given, the direction is always from the secondary point to the primary point (primary point is downstream of the secondary point).

Most of the standards allows the definition of **areal features**: in INSPIRE areas are used to describe the surface of road infrastructure (as service areas, traffic areas…), in the FGDC Road Framework, although there is not a polygonal class, areas can be defined through *LinearFeatureEvent* as closed path. In Transmodel, polygonal class are defined in order to describe objects as Stop Place. In general, in public transport data models, the concept of administrative zone is usually intended as area feature within a set of conditions are valid, with or without a geometry representation, which can be provided also as simple centroid, as in the case of DATEX. In CityGML the surface concept is predominant, but definition purposes are mainly devoted to representation aspects than to logical behaviour.

Looking at **properties** and **types** definitions, in general the definition of code lists and dictionaries is particularly useful to set up a standard. INSPIRE and FGDC, as their main objective is data sharing between different administrative levels, gives particular attention to the creation of code lists, which concerns road hierarchy, surface, type of a road, lanes, events types.... CityGML also has a dictionary, which can be expanded for application purposes.

Transmodel has a detailed dictionary, and is particularly accurate in technical terms definition in comparison to the others public transport standard.

In DATEX attention is given to the code lists referred to event definition and characterisation: this aspect is not trivial, as several definitions are not so clear in DATEX I, with several overlapping concepts. The DATEX II gives a serious improvement of those code lists.

Finally, standards can be compared on the **temporal** aspect representation.

The INSPIRE directive does not deepen temporal aspects: the main temporal concept defined is related to the valid date time of a data set. In addition, the model has a basic versioning system, managed through the definition of a composite identifier made by: the “local_id” which is the unique identifier within a “namespace”, the identifier for the whole data source, owned by a data provider and the “version_id”, which identifies a particular version of the object. The use of the versioning system is not mandatory in INSPIRE. Temporal aspects are indeed deepened by O&M model, which comprises instants as well as ranges and time series, but practical implementation for traffic monitoring purposes is not deepened.

The FGDC Frameworks use temporal concepts essentially in events definition. An event is in fact something that change in time, although time attribute defined is only the “lastUpdateTime”. The Frameworks moreover do not have any versioning systems.

CityGML generally lacks of temporal representation aspects (to be managed by custom extensions).

Temporal aspects are furthermore deepened by Transmodel and info-mobility standards. In general, time can be modelled as instant or as a range (in Transmodel as in DATEX). In Transmodel in particular, one of the sub-models defined is the “Timing model” (Part 3), which concerns temporal aspects of a public transport service. When temporal aspect has a strong connection with spatial elements, instants are modelled as points (e.g. *scheduled stop point*), ranges as linear elements (e.g. *timing link*). Other types of temporal concepts are related to validity conditions and constraints, as *time table* or *time bands*. Transmodel has also a *version frame model,* which allows to manage different version of entities, using the concept of *frame*. It also defines objects as *deltas tables,* in order to tracks differences between versions. In GTFS there is a "version" attribute in the *FeedInfo* table.

A particular comparison concerns the semantic in the field of **public transport**, where one of the main goals of the standards is to establish a precise vocabulary. The vernacular use of terms as route, journey, trip or stop indeed can cover a wide range of overlapping concepts, leading to confusion when information systems and data needs to be compared. In particular, Transmodel is the most accurate in technical terms definitions and can be used as reference to compare the use of different terms in other standard models (Knowles, N., Miller, P., 2008). Below a comparison between public transport concepts in GTFS, Transmodel and FGCD – Transit Framework.

The term **Trip** is used in Transmodel to describe the journey made by the passenger, whereas it defines a *Vehicle Journey* to delineate the journey made by the vehicle, with scheduled timetable defined in a specific day type. The FGDC Transit Framework give to trip a more generic definition, as the one-way scheduled movement of a transit vehicle between starting and ending time points. It is an instance of a *Pattern* (one pattern for one trip) and is demarcated by an ordered sequence of time points (two or more ordered time). The FGDC Transit Framework never uses the term journey. In GTFS a trip is a sequence of two or more stops that occurs in a specific time, characterised by a header and a direction. A trip is seen from a consumer point of view, where the utility for the user is to know where the vehicle is directed. The GTFS trip is quite similar to Transmodel one as it is tailored on the users' point of view, whereas the FGDC points out on the vehicle point of view. In all cases, attention is given to the temporal aspect: a trip exists within scheduled time of arrival/departure from stop point.

A similar discussion can involve the term **Route**. In GTFS a route is a group of trips displayed to the passengers as a single service. This concept is overlapped to the Transmodel *Line* concept: a group of routes known to the public by a similar name or number. Instead, the concept of route in Transmodel is more general. A route is an ordered list of located points defining one single path through the road (or rail) network. As location is involved in Transmodel route definition, a Transmodel route is more close to the *Shape* table of GTFS.

Specifically, a GTFS shape can be view as a Transmodel *route link projection*. FGDC Transit Framework define the *TransitRoute* as a collection of patterns in revenue service with a common identifier, more similar to the GTFS route concept.

Transmodel distinguishes between **operator**, **authority** and **organisation**. The *operator* is a company providing public transport services: a single *vehicle journey* can be assigned to a single *operator*, whereas a *line*, as group of vehicles journeys, can be assigned to a group of operators operating under contract to an authority. An *authority* is the organisation under which the responsibility of organising the transport service in a certain area is placed. An *organisation* is a more general concept, describing legally incorporated body associated with any aspect of the transport system. In GTFS it is possible to specify one or more transit *agencies*, which are the ones that provide data feeds (*Agencies* class). The agency may be the same one that manages routes, or only the one which provides data. The model does not allow to define single lines managed by more than one operator. In FGDC Transit Framework the only reference to the operator concept is found in the "owner" attribute in *TransitStop* class. Once again, modelling a service shared by multiple operators is difficult.

Differences can be highlighted also in the **stop area** definition and in the stop connection management. A classical case is to have different stop points close to each other that can allow to alight a vehicle and board on another to arrive to destination. Those stops can be close and on the same side of the road, but the spatial proximity of two stop points does not guaranteed that there is a link between them, as there can be any type of obstruction, or stops can be at different heights (metro and bus station), with difficult or impractical route between them.

In order to model such various situations, Transmodel, through the IFOPT *Stop Place Model*, defines a *stopArea* that can be associated to a number of stop points or even with other stop areas, allowing also a hierarchical differentiation (e.g. an airport is composed by several terminals, a metro station, bus stations, trains….). Transmodel uses the concept of *connection* to define the transfer time for a specific user (maybe with reduced mobility) between two (scheduled) stop points: the connection allows indeed to take into account accessibility conditions and constraints. An example can be a metro station, which can be composed of different entrances with different accessibility levels and opening at certain time, each one maybe associated with a different cluster of stop point on the ground level. Transmodel uses also the term *interchange*, mainly as a temporal concept: a *service journey interchange* in particular allows to define the type of connection (*planned*, when passenger can autonomous making the connection, *advertised* in case of specific claims and *guaranteed* when the second vehicle waits for a delayed incoming service). The SIRI part of Transmodel defines also a *connection timetable* and a *connection monitoring* in order to exchange and spread information and monitor connections in real-time.

The GTFS specification allows to distinguish between *station* and *stop* from a hierarchical point of view. A stop indeed can be physically located inside or associated with a station (through the "location_type" and "parent_station" attributes in stop table). The model do not allows to represent more complex situations and associated accessibility conditions. In particular, the "wheelchair_boarding" attribute in stop table, admits different values depending on the configuration of values in "location_type" and "parent_station". As a parallel to the *interchange* Transmodel concept, GTFS uses a *Transfers* table to define connections between two stops, providing a minimum transfer time, and distinguishing among guaranteed and not guaranteed.

The FGDC Transit Framework defines *stops* in a general manner: locations where transport customers may board or alight from a transit vehicle (in revenue service). A *TransitStop* is defined as a point, but is also a facility and can inherit the facility geometry. TransitStop can be clustered in *TranferCluster*, when is possible for the passenger to change route. The class is also related to a *TimePoint*, a concept similar to the scheduled stop point of Transmodel. The stop model lacks of specific time constraints and accessibility conditions and not allow a hierarchical classification of stops. The class *ConnectionSeg* in FGDC Framework can be seen as a parallel of the connection concept in Transmodel: it is characterised by a distance, an accessibility level, and a set of instructions, but it not provides any information about time. It has to be underlined that the FGDC Transit Framework provides, unlike the other standard, a good support for stop inventory also thanks to the *Amenity* class, which describes physical elements that characterise a stop or other transport facilities, with a description and a status attributes, in order to allow periodical maintenance operations.

Differences can be found also in the management of **calendars**. In particular, the Transmodel defines the *Operating Day* class, in order to flexible managing services that operates across midnight. It is a reusable day that not end at midnight and can be associated to a *day type*. Transmodel is flexible in day type definition, which can be both a day of the week (as in the GTFS) but also some other types of day that affect the demand for services (e.g. match day). Transmodel in addition allows to define specific time zones, for modelling services across different time zones.

Table 5: GTFS tables and equivalent classes of Transmodel and FGDC Transit Framework. Readapted from Knowles and Miller, 2008.

GTFS	**Transmodel**	**FGDC Transit Framework**
Agency	Operator/Authority	TransitStop [Owner]
Stop	Scheduled Stop Point, stop place, tariff zone	TransitStop, TransferCluster and Facility
Route	Line	TransitRoute
Trip	Vehicle journey	Trip (temporal aspect), Pattern(spatial aspect)
Stop Times	Passing times, stop point in journey pattern, distance in route link	Time Point
Calendar	Day type, period and day of week	-
Calendar dates	Operating day	-
Fares attributes	Fare element price	Fare, FareType code list
Fare rules	Fare element, distance matrix	FarePolicyType code list, FareType code list
Shape	Route, Link projection	TransitPath
Frequency	(frequency)	-
Transfers	Connection (link), service journey interchange, service journey pattern interchange, default interchange	ConnectionSeg

In GTFS, the table *calendar* defines the temporal boundaries of the operating day, derived from trips associated to a service. The *service* table allows to specify the availability of a trip in a particular day of the week, with exceptions managed through the *calendar_dates* table. Time zones are managed in GTFS through the attribute “timezone” in Agency and Stops tables. The assumption is that an agency operates only in one time zone.

Those aspects are managed in FGDC Transit Framework through the class *TransitRoute*, which contains the attribute “timeTableHeader”. This attribute is a summary of publically recognized *TimePoints* contained in a group of Patterns oriented in the same route direction, and is used to generate timetables. The data model does not support concepts as calendar, day type and time zone.

Differences arises also in the concept of **modes** of transport. In Transmodel, the class *Transport Mode* defines the main mode (bus, rail, ferry…) and the *Vehicle Type* class can be assigned to different service journeys, specifying the types of equipment of the vehicle. The GTFS instead uses the attribute “route_type” in *Route* table, but it is limited, as it does not include the variety of transport mode typical of European situation. In particular, GTFS model lacks concern urban and suburban rail and bus and differences between coaches and buses. Considerations of limited mobility are applied only to the stop and not to the vehicle characteristics. Also in the FGDC Framework specification on transport mode are not taken into account.

Finally, as both Transmodel and FGDC Transit Framework have some elements for supporting operational activities of a public transport agency, the concept of **block** is present in both standard with the same meaning. A block is a sequence of trips over which a vehicle is assigned from pull out time to pull in time for the FGDC Framework, and the work of a vehicle from the time it leaves a parking point until it returns to a parking point for the Transmodel.

A summary of overlapping concepts between public transport standards is given in Table 5.

A final consideration concerns the use of the CSV files versus XML format (Knowles, N., Miller, P., 2008). The use of csv in GTFS has indeed advantages, as it is compact, easy to interpret and use. However, the Transmodel XML encoding (provided by the NeTex) allows a better versioning management, making easier to exchange incremental changes to the data rather than the full dataset. Other technical advantages of XML are a rich object model, a self-describing representation, a packaging management thanks to encapsulation, and a general easier design for applications interfaces.

Chapter 4

Transport network data sources

In this section a review of available data sources for transport networks is performed. In particular, attention in given to open source data, which can be potentially used by Traffic Operator Centres to deploy services and publish open data: strengths, limitations and shortcoming in the use of open data in the transport field are deepened. In addition, the commercial dataset NAVSTREETS Street Data provided by HERE company is described, as it is the one used by the 5T Agency for their daily operational activities.

4.1 OpenStreetMap

OpenStreetMap (OSM) is a collaborative project with the aim to create and provide free geographic data, such as street maps, to anyone (Wikipedia contributors, 2018).

Gathering geographic data is an expensive activity, usually in charge of public administration and governmental agencies, but in recent years there has been a significant growth of private companies, which gather and sell geographic data (TeleAtlas, HERE…). In addition, the advent of inexpensive portable satellite navigation devices (GPSes car devices, smartphones) has raised the capacity for everyone to know where they are and where they go. These devices allow to collect geographic positions, and use frequently private services in order to have a map on which locate itself. As in instance, Androids smartphones are today largely widespread, and they embed the Google Maps app, which allows users to use Google location services. This is a private service, and Google spent a lot of money to buy geographic data from company as HERE and Tele Atlas. As users can access this information through online maps, they cannot download or use it for other purposes.

Restriction of use and high cost are two serious problems in geographic data spreading. This situation exposes the global community to some risks: Wroclawski, in its article of 2014, points out as “place is a shared resource, and when you give all that power to a single entity, you are giving them the power not only to tell you about your location, but to shape it”. When a geographic data provider becomes enough widespread it “becomes the source of “truth””, as it can choose what elements have to be emphasised or what have to displayed or not. This a serious problem of democracy. In addition, private companies as Google and Apple provides locations service, which now collect location information of the users, using it not only to improve their map accuracy, but also to track the correlation between users searches and their movements (McDermott, 2013).

The OpenStreetMap project arises as a solution of these issues, positioning itself as a “cartographical Wikipedia”: the project aims to freely distribute geographic information taking advantage of being built by volunteers. Today OSM is considered a prominent example of volunteered geographic information (Wikipedia contributors, 2018), and, from the point of view spatial coverage and topographic elements, can be considered one of the best available source of open data.

The OpenStreetMap project stresses the difference between maps and geographic data: its main goal is to provide a world geographic database and not a map. Many of third-party GFOSS (Geographic Free Open Source Software) projects use OSM database to provide geographic services as map tiles. The advantage is that anyone can download the data, using it offline and is free to render it as he wants. As negative consequence, its spread to common people is limited in contrast to tools as Google Maps or Google Earth (Wroclawski, 2018).

A distinctive trait of OSM is that, instead of the layered approach used by most of geographic databases, it use a single layer and a system of **tags** (a pair of key and value) for specifying the geographic meaning of a single object. The data model is based on a tree data model (see Figure 46), typical of XML structures. Four types of object can be defined: **nodes**, **ways** (ordered list of nodes), **polygons** (a way that starts and ends on the same node) and **relations**. Each object must have at least one tag and other tags can be added in order to describe the object: they can be view as attributes of the object and there are not strict rules on how many attributes any object should have (no upper limit).

For transport purposes, the most common way to define a road is to edit an open way, an ordered list of nodes which also normally has at least one tag or is included within at relation; in case of road the tag “highway=*” is used. The value of the key “highway” is usually needed to indicate the importance of the road in the network, as “motorway”, “primary”, “secondary”, “service”…. A list of possible values is specified in the Wiki of the project. Other tags can be combined, in order to define specific characteristic of the road (“maxspeed=*”, “oneway=*”…) (OpenStreetMap Wiki contributors, 2017 a, b).

Street as a vector

A one-way residential street, tagged as `highway=residential` + `name=Clipstone Street` + `oneway=yes`

```
<way id="5090250" visible="true" timestamp="2009-01-19T19:07:25Z" version="8" changeset="816806" user="Blumpsy" uid="64226">
  <nd ref="822403"/>
  <nd ref="21533912"/>
  <nd ref="821601"/>
  <nd ref="21533910"/>
  <nd ref="135791608"/>
  <nd ref="333725784"/>
  <nd ref="333725781"/>
  <nd ref="333725774"/>
  <nd ref="333725776"/>
  <nd ref="823771"/>
  <tag k="highway" v="residential"/>
  <tag k="name" v="Clipstone Street"/>
  <tag k="oneway" v="yes"/>
</way>
```

Figure 46: OSM way data structure, reprinted from OpenStreetMap Wiki contributors, 2017 a.

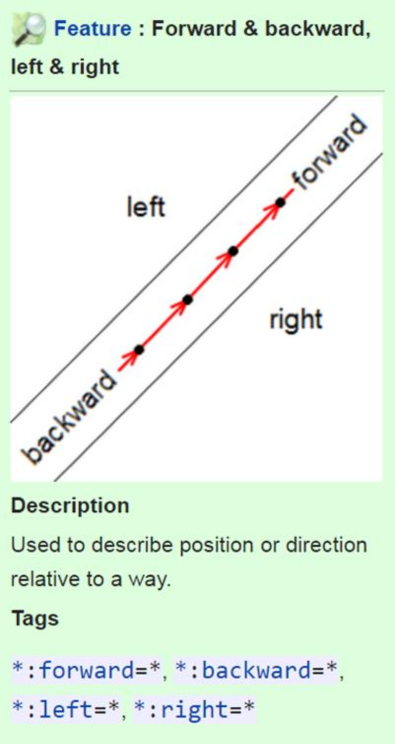
The definition of the direction of travel is of utmost importance for routing applications: in particular four terms have been introduced and can be used to describe the direction of travel and the side of a way. “Forward” and “backward” describe a direction along a way, taking into account the direction in which the way is drawn in OpenStreetMap: “Forward” means the digitized direction of the way, while “backward” means the opposite direction (see Figure 47). For instance the tag key “oneway” with “yes” as value, indicates a way traversable only in the forward direction (as in Figure 46), while a value of “-1” indicates the backward direction (OpenStreetMap Wiki contributors, 2018).

The two other terms introduced are “left” and “right”, used for describe a side of a way. In particular, “left” means the left-hand side of the road when looking in the forward direction (as defined above), while “right” means the right-hand side (see Figure 47).

Below three example of the use of these terms: they can be appended as a namespace to a tag's key, by adding a colon after the tag, as in the example (1) and (2), or can be used as a tag’s value, as in the example (3).

1. maxspeed:forward=* → maximum speed which only applies in forward direction;
2. cycleway:left=* → a cycleway on the left side of the road;
3. sidewalk=left → the side(s) of the road where sidewalks are present.

The OpenStreetMap model allows also the definition of a **relation**, a data element used to define logical or geographic relationships between objects. Nodes, ways and relations can be member of a relation. A relation can be characterised by a role, which describes (through a verbose description) the function that a particular feature plays within a relation (OpenStreetMap Wiki contributors, 2018 b).



Feature : Forward & backward, left & right

Description

Used to describe position or direction relative to a way.

Tags

`*:forward=*`, `*:backward=*`, `*:left=*`, `*:right=*`

Figure 47: Direction of flow logic in OSM, reprinted from OpenStreetMap Wiki contributors, 2018 a.

Uses of relations are multiple: a “street” role has the purpose to group all the elements that make up a street together, with the advantage to avoid the repetition of tags applied to an entire street. For instance, a street relation provide the explicit connection between addresses and the street to which they belong. Roads can also play the role of administrative boundary. Relations are also used to represents public transport trips: those relations usually include the sequences of the stops (nodes) composing the route, a way with tags “type=route”, “route=bus/train/tram…”, “ref=*” (reference to the line number), “operator=*” and others. The relation “type=route_master” includes both direction of travel of the line.

The **tagging system** of OpenStreetMap is very flexible: tags can be arbitrarily added by users in order to define new kind of objects (as ones that usually are not included in topographic elements). The OpenStreetMap Wiki documentation contains general rules for mapping several types of geographic objects (proposed, approved or de facto), and series of tagging schema are recommended (as the public transport scheme, now on version 2), but as a matter of fact anyone is allowed to add a new tag and value in the map. The drawbacks of this approach are related to the potential inconsistences between tags, also due to trivial misspelling errors (which can be reduced using editing interface as “iD” editor), and differences in the choice of representation. As instance, sidewalk and cycleway can be represented as ways itself, with their own geometry or can be only an additional tag of the road on which they belong. This leads to difficulties in writing application tools.

Other problems emerge in the use of OpenStreetMap data: one of the obstacle that limit their use in public administration is the lack of a **permanent ID**. IDs exist for low-level objects, but are not related to high-level concepts. For instance, a building is composed by closed way, with IDs of the nodes and of the way, but the building itself not have an ID. This do not allow to preserve the history of the conceptual object itself: ironically, a node of the building can be moved and used to compose other objects, losing completely the sense of the history of the building as a geographic object. Therefore, administrations which want to reuse OSM data have to remap them in order to obtain a persistent id to use for sharing data and preserve their integrity during time and versions.

Another big issue in OpenStreetMap is related to the fact that **anyone can map**: sometimes users can be beginners, that can maps incorrectly, but there is not a well-oiled review system, making difficult to find and correct errors. In addition, the project is highly exposed to vandalism. This cause a loss of data quality.

Looking at the use of OpenStreetMap data for **transport** purposes, the problem of *data quality* seriously limits it use: in particular, it is not always suitable for high-reliability routing applications. Limitations are related to a non-uniform spatial coverage in all areas: being a collaborative and voluntary project, not all areas of the world reach the same level of detail. The degree of *accuracy* that is reasonable depends on the existing data. Therefore, in areas where there is no coverage, even an inaccurate tracing of a road is an improvement. The data are continuously updated, but it is not possible to obtain information about the quality of the data provided, both at the level of accuracy and precision, as it varies depending on the techniques used for the digitization (OpenStreetMap Wiki contributors, 2018 c).

Figure 48: Some errors of OSM in the area of Turin, reported the web application http://keepright.ipax.at/report_map.php

The high spatial inhomogeneity is also reflected at the *attribute* level. In in the case of a road sometimes only the geometric information is provided, the street names are sometimes included and sometimes not, and not always this names correspond to the official names, making difficult to exploit geocoding applications. Information about restrictions, number of lanes, maximum speed or direction of travel, as they are not mandatory, are often missing. As is highlighted in OpenStreetMap Wiki “mapping completeness has always been a slightly awkward aspect of OpenStreetMap, both in terms of achieving good completeness, and in measuring how good it is”. Establishing metrics for measuring *completeness* is very important in order to evaluate the suitability of data for analysis (OpenStreetMap Wiki contributors, 2018 d). Additionally, topology in network editing is frequently insufficient: in particular, roads are not always split at intersections, generating unconnected network and making OpenStreetMap not routable and not ready to use for analytical tasks. In Figure 48, is shown one of the web applications that shows errors on OSM data, for network and topographic features.

In order to **download** and use OpenStreetMap data, several solutions are available. The first one is through the website of the project (http://www.openstreetmap.org/) through an API (Application Programming Interface), which allows users to fetch and save raw data from/to the read-only copy of the main OpenStreetMap database over a small custom area. This API is optimized for editing purposes and data are available in XML format (.osm or JOSM encoding). All elements are available, also the ones which have not a specific rendering, but the XML is not easy to use as it is.

The OpenStreetMap project has also made available the Overpass API, which can be used to extract large datasets, also applying searching criteria as location, type of objects, tag properties, proximity, or a combination of them. It acts as a database over the web: the client sends a query to the API and gets back the data set that corresponds to the query. As the API has its own specific query language, the interactive frontend Overpass Turbo (http://overpass-turbo.eu/) can be used to set up query in an interactive way: the interface allows to visualize the selected elements and has some pre-built query, which eases the process of data selection. Data can be saved as geoJSON, GPX, and XML formats.

In order to download large areas the third-party website Geofabrik (http://download.geofabrik.de/) represents an optimal solution: OpenStreetMap data are daily updated and organised by countries. Some countries are further subdivided following administrative division. It is possible to download data in .osm format (or .osm with a PBF compression) and in shapefile format. It has to be underlined that the shapefile conversion do not comprehend the whole objects defined in OpenStreetMap: it is a simple default selection of most used elements (road and rail network, buildings, points of interest, forests, water areas …).

Between other third party services, some relevant are:

- “OpenStreetMapData” (http://openstreetmapdata.com/), which provides shapefiles of global coastlines, land polygons or water polygons.
- “Estratti OpenStreetMap” (http://osm-estratti.wmflabs.org/estratti/), which provides daily extracts for the Italian Regions (.shp, .pbf, .osm, .sqlite).

In Table 6, a comparison between download services is shown. The “total score” represents a measure of the flexibility of use of the service itself.

Table 6: Comparison between OSM download services functionalities.

		Evaluation criteria							**Total Score**
		Custom Area		Completeness		Custom Features	Data Export Format		
Services	**Geofabrik**	+	Large Admin.Areas	++		-	++	XML, .pbf, .shp	4/10
	Estratti OpenStreetMap	++	Regional Level	++		-	++	XML,.shp, .pbf, .sqlite	5/10
	OpenStreetMap Data	+	Only global	+	Predefined themes	-	+	.shp	2/10
	OSM API	+++	Limited areas	+++		-	+	Only raw XML	6/10
	Overpass API	+++		+++		+	+++	.geoJSON, .GPX, XML	10/10

4.2 OpenTransportMap

OpenTransportNet (OTN) is a European funded project focused on transport related services across Europe. A City Data Hubs has been constituted in order to aggregate transport-related data and spatial information, with the aim of creating innovative and collaborative ITS applications and services.

One of the first and most important outcome of the project is the **OpenTransportMap** (OTM) (http://opentransportmap.info/). The project aim was to provide a harmonized source of information for transport network, starting from open data, in order to enable routing services and expose data in the form of easy-to-understand visual interpretations, with a focus in the visualization of traffic volumes.

After a first comparison of open data available over the four European cities involved in the project (Birmingham, Antwerpen, Issy-les-Moulineaux, Liberec), OpenStreetMap data has been chosen as main source of information. Indeed open data provided by public administrations use different semantics and data schemas, so OpenStreetMap data, evaluated as enough complete over the areas of interest, has been chosen in order to set up the most replicable approach (Jedlička et al., 2016).

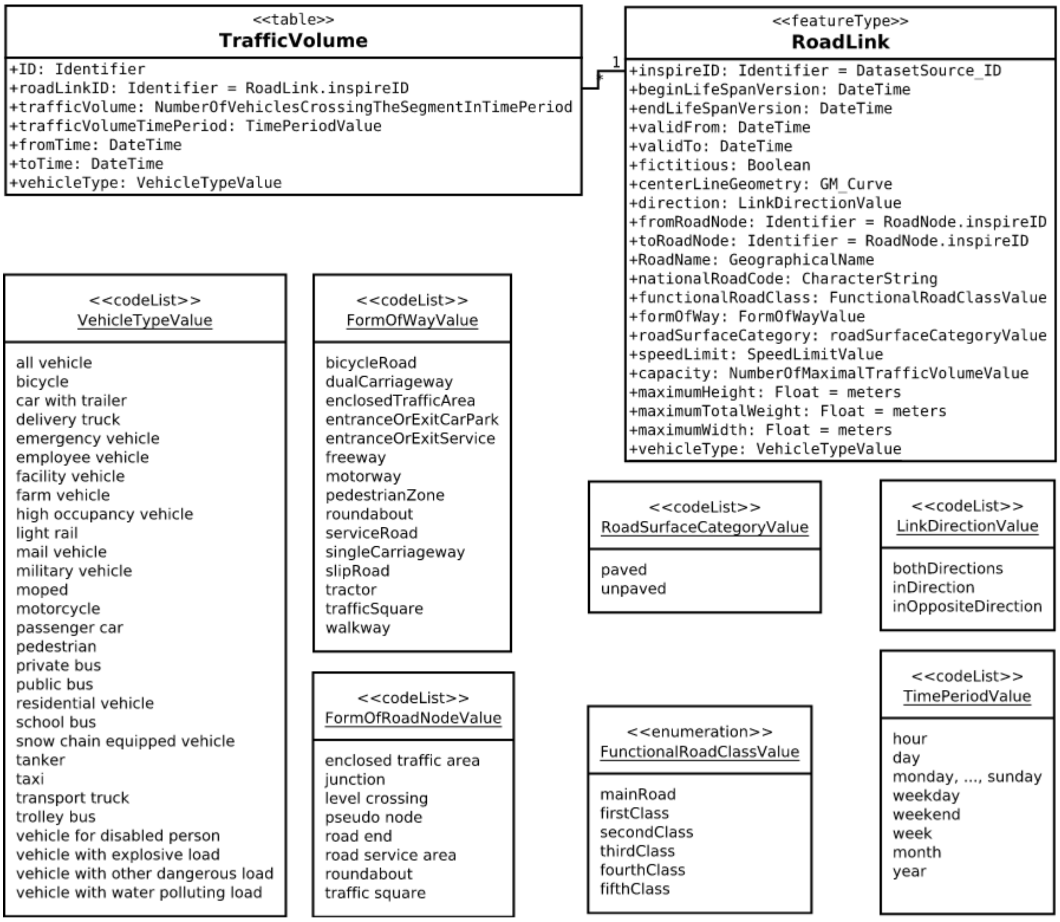
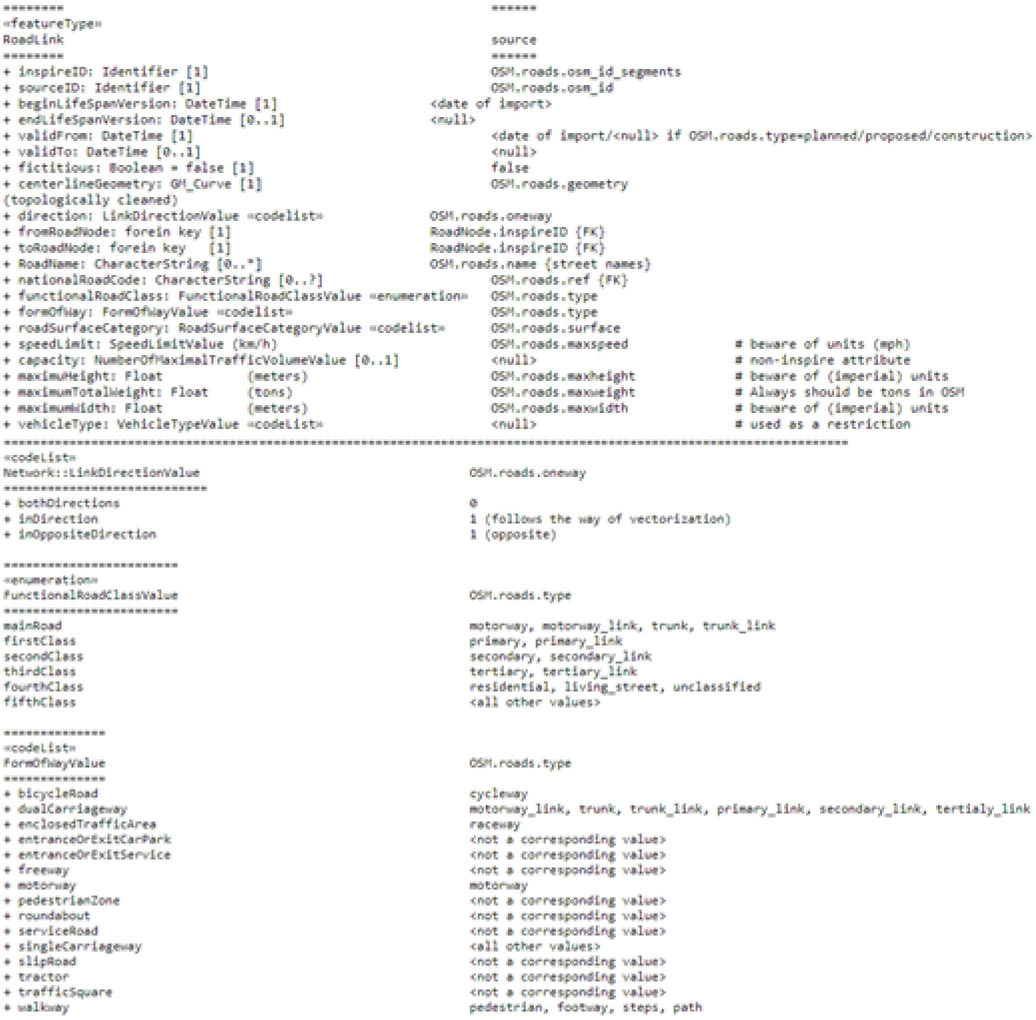


Figure 49: The OpenTransportMap data model, reprinted from http://opentransportmap.info/img/OTM_physicalModelAndCodelists.svg

The INSPIRE Transport Networks data model was chosen as reference in order to build the harmonised data schema, as it addresses the linear topology and is compliant with the EU legislation. Processing steps have involved first the evaluation of geometry and in particular of the original topology of the OpenStreetMap data: some automatic and manual corrections have been applied over data in order to obtain a well-connected topology structure, which is critical to describe allowed movements between places (Jedlička et al., 2016).



```
========                                                        ======
«featureType»
RoadLink                                                        source
========                                                        ======
+ inspireID: Identifier [1]                                     OSM.roads.osm_id_segments
+ sourceID: Identifier [1]                                      OSM.roads.osm_id
+ beginLifeSpanVersion: DateTime [1]                    <date of import>
+ endLifeSpanVersion: DateTime [0..1]                   <null>
+ validFrom: DateTime [1]                                       <date of import/<null> if OSM.roads.type=planned/proposed/construction>
+ validTo: DateTime [0..1]                                      <null>
+ fictitious: Boolean = false [1]                               false
+ centerlineGeometry: GM_Curve [1]                              OSM.roads.geometry
(topologically cleaned)
+ direction: LinkDirectionValue «codelist»              OSM.roads.oneway
+ fromRoadNode: forein key [1]                          RoadNode.inspireID {FK}
+ toRoadNode: forein key   [1]                          RoadNode.inspireID {FK}
+ RoadName: CharacterString [0..*]                      OSM.roads.name {street names}
+ nationalRoadCode: CharacterString [0..?]                      OSM.roads.ref {FK}
+ functionalRoadClass: FunctionalRoadClassValue «enumeration»   OSM.roads.type
+ formOfWay: FormOfWayValue «codelist»                          OSM.roads.type
+ roadSurfaceCategory: RoadSurfaceCategoryValue «codelist»      OSM.roads.surface
+ speedLimit: SpeedLimitValue (km/h)                            OSM.roads.maxspeed              # beware of units (mph)
+ capacity: NumberOfMaximalTrafficVolumeValue [0..1]            <null>                          # non-inspire attribute
+ maximuHeight: Float           (meters)                        OSM.roads.maxheight             # beware of (imperial) units
+ maximumTotalWeight: Float     (tons)                          OSM.roads.maxweight             # Always should be tons in OSM
+ maximumWidth: Float           (meters)                        OSM.roads.maxwidth              # beware of (imperial) units
+ vehicleType: VehicleTypeValue «codeList»                      <null>                          # used as a restriction
==============================================================================================================
«codeList»
Network::LinkDirectionValue                                 OSM.roads.oneway
===========================
+ bothDirections                                            0
+ inDirection                                               1 (follows the way of vectorization)
+ inOppositeDirection                                       1 (opposite)

=======================
«enumeration»
FunctionalRoadClassValue                                    OSM.roads.type
=======================
mainRoad                                                    motorway, motorway_link, trunk, trunk_link
firstClass                                                  primary, primary_link
secondClass                                                 secondary, secondary_link
thirdClass                                                  tertiary, tertiary_link
fourthClass                                                 residential, living_street, unclassified
fifthClass                                                  <all other values>

=============
«codeList»
FormOfWayValue                                              OSM.roads.type
=============
+ bicycleRoad                                               cycleway
+ dualCarriageway                                           motorway_link, trunk, trunk_link, primary_link, secondary_link, tertialy_link
+ enclosedTrafficArea                                       raceway
+ entranceOrExitCarPark                                     <not a corresponding value>
+ entranceOrExitService                                     <not a corresponding value>
+ freeway                                                   <not a corresponding value>
+ motorway                                                  motorway
+ pedestrianZone                                            <not a corresponding value>
+ roundabout                                                <not a corresponding value>
+ serviceRoad                                               <not a corresponding value>
+ singleCarriageway                                         <all other values>
+ slipRoad                                                  <not a corresponding value>
+ tractor                                                   <not a corresponding value>
+ trafficSquare                                             <not a corresponding value>
+ walkway                                                   pedestrian, footway, steps, path
```

Figure 50: An example of mapping between OpenStreetMap and INSPIRE Transport Network, readapted http://opentransportmap.info/OSMtoOTM.html

Then the INSPIRE Transport Networks schema was analysed, selecting *RoadLink* and *RoadNode* as the main classes to be used for routing purposes. The schema of the new harmonised data model is simple: the RoadLink class have a relationship with a TrafficTable class, with attributes and related code lists directly derived from INSPIRE Directive, as shown in Figure 49.

The mapping activity between OpenStreetMap tags and INSPIRE attributes and code lists has been made, as shown in Figure 50. This is an example of *information loss*, when searching for the lowest common denominator: indeed, maybe some tags, which may be useful for roads classification, are not taken into account (Jedlička et al., 2016). Some limitations inherited from the OSM data still persist: in particular at the level of attribute completeness and the general non-homogeneity, as the lack of information about the accuracy. The result is a road network enough suitable for routing and that can support the visualisation of traffic volumes.

In addition, the creation of an "InspireID" attribute can help in solving issues related to the OSM use by Public Administrations and related version tracking.

In Figure 51 is shown the webGIS application that shows the transport network revised by the project and structured in functional class, derived from INSPIRE Directive. Currently, Open Transport Map as an eligible source of information for road transport network, as it is freely available on Open Data Commons Open Database License (ODbL): through the project website is possible to download data at NUTS-3 level [Province] in shapefile format, for the whole European Union countries. The visualisation of mean traffic volumes is instead available only over the pilot cities involved in the project.

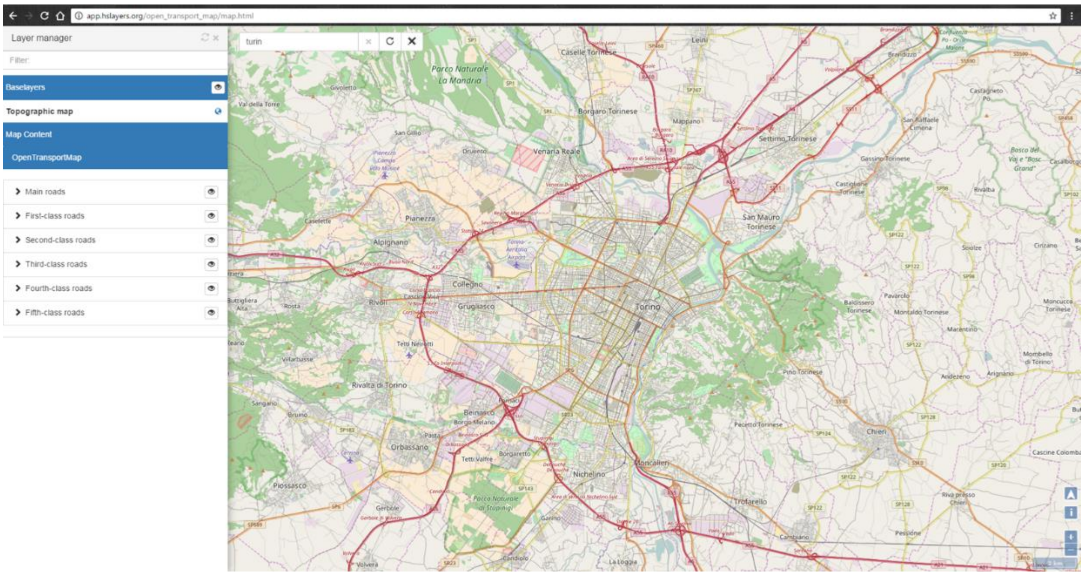


Figure 51: The OpenTransportMap WebGIS application (http://opentransportmap.info/).

4.3 NAVSTREETS Street Data

Navteq is an American provider of GIS data founded in 1985, specialised in navigable transport network data and maps. The company was acquired by Nokia in 2007/2008 and fully merged into Nokia in 2011 to form part of new brand Here (Nokia Location & Commerce). Navteq, in collaboration with third-party agencies and companies, exploits various services: maps for Garmin portable GPS devices, web-based applications such as Yahoo! Maps, Bing Maps, and Nokia Maps (Wikipedia contributors, 2017 b).

5T Agency in Turin uses NAVSTREETS Street Data, one of the main product of HERE, as main source of information for their operational activities.

The NAVSTREETS Street Data is delivered also in shapefile format (the one used in 5T) and provides, over the requested area, in addition to transport network data, other cartographic features that enhance the routing functionality. The delivered product contains several shapefiles and .dbf tables, but the main source of information is represented by the *Streets* shapefile, which are the navigable edges of the transport network, and the *Zlevels* shapefile, which represents nodes. Thanks to the definition of a specific set of rules for editing, the network is consistent over the whole area. The dataset is fully documented in the “NAVSTREETS Street Data Reference Manual v6.0” of the 1st April 2016, produced by HERE, from which information contained in this section is taken.

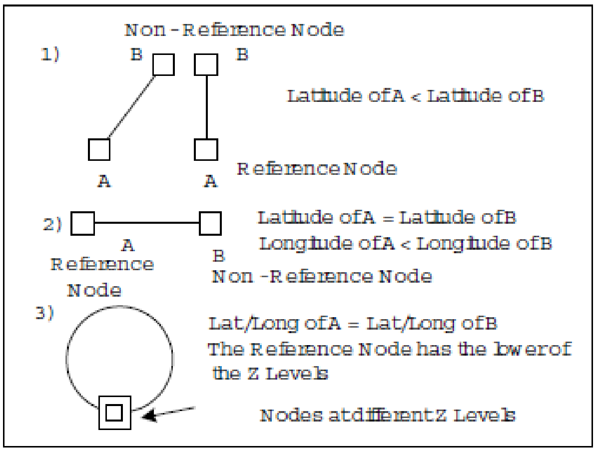


Figure 52: Rules for Reference and Non-Reference Nodes, reprinted from HERE, 2016

Looking into the schema of the *Streets* shapefile, the minimum mapping element is represented by the LINK, which have a minimum length of 2 metres and a maximum length of 10 kilometres (segmentation occurs at intersection and at attributes changes, for instance pavement condition). A LINK is unambiguously identified by the "LINK_ID" attribute as can be seen in Figure 53. The identifier attribute constitutes also the main foreign key to reference additional tables and features to the *Streets* shapefile (see Figure 53). LINKs can be grouped in FEATURE (LINK with the same "FEAT_ID" attribute) in order to defining roads with the same street name. The feature concept is useful from a semantic point of view, as it define as a unique element roads composed by several lanes or frontages, even if they are not continuous (the "FEAT_ID" is equal to zero in LINK representing complex intersection as roundabouts).

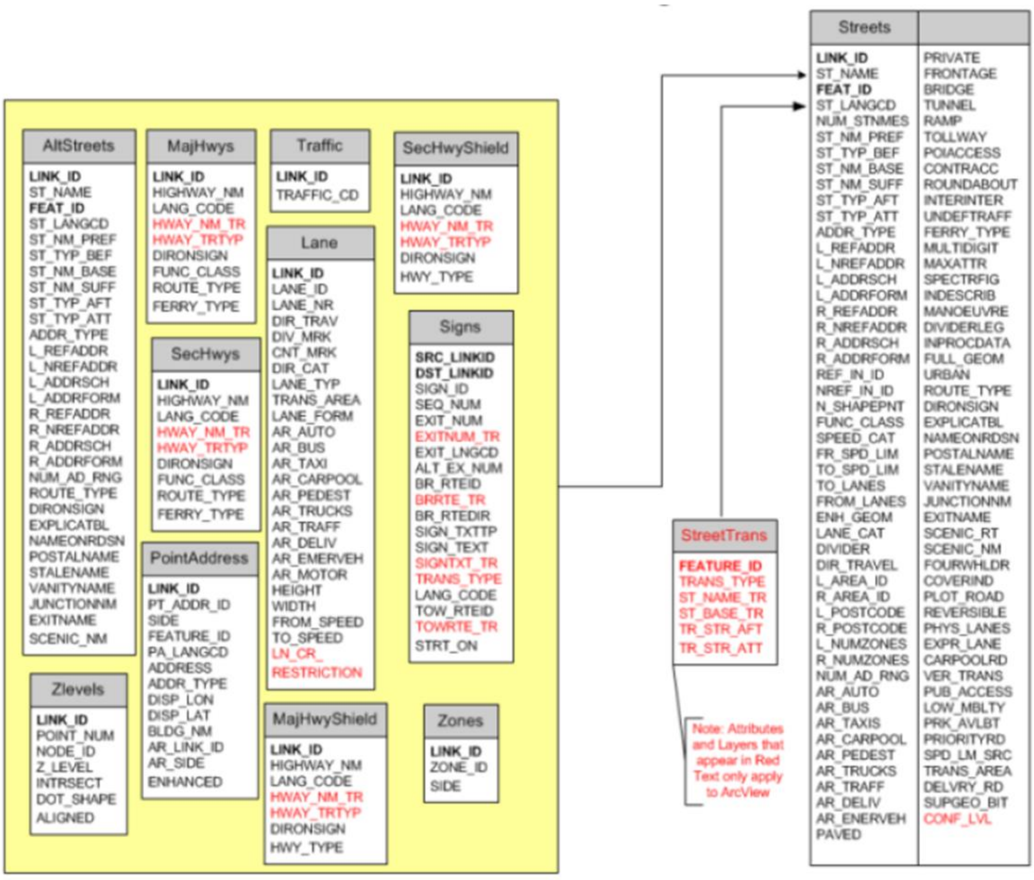


Figure 53: General schema and attribute relationships of the NAVSTREETS Street Data, reprinted from HERE, 2016.

Each LINK is defined by a *Reference Node* ("REF_N_ID" attribute) and a *Non-Reference Node* ("NREF_N_ID" attribute), where first is the one with lower latitude or in case of the same latitude, with lower longitude (see Figure 52).

According to the definition of *Reference Node* and *Non-Reference Node*, the navigability of the edge is defined as “BOTH”, “FROM REF NODE” or “TO REF NODE” values in “DIR_TRAVEL” attribute. The navigability information can be further refined through a join with the table of the *Lanes*, which specifies the direction for each lane (the information is not complete for North-West Italy area).

The *ZLevels* shapefile contains both the necessary nodes for edges digitizing (internal nodes) and the nodes that define the intersections between links (identified by the “Y” value into the “INTRSECT” attribute). In addition, only intersection nodes provide a value in “NODE_ID” attribute (otherwise equal to zero), which allow to reference the link on which they belong (the value reported in “REF_N_ID” and “NREF_N_ID” in the *Streets* shapefile). Each node has the “LINK_ID” attribute that allows knowing which is the referenced edge.

The *ZLevels* shapefile contains the “Z_LEVEL” attribute that is used to represent junctions as the bridges and tunnels, so crossing over or under of links with other links. This attribute is not to be used to indicate actual elevation gain or loss, but to prevent routing between links that do not connect in reality (“0” for ground level, negative values for tunnel and positive for bridges).

As can be seen in Figure 53, the *Streets* shapefile contains over 98 attributes, which describes the characteristics of the road. Between relevant attributes can highlighted:

- Functional class (“FUNC_CLASS”): used to define hierarchical levels between roads for a more efficient routing. Values ranges from 1 (most important) to 5.
- Speed Category (“SPEED_CAT”): allow the classification of streets based on posted or legal speed. It can differs from the Speed Limit value as it takes into account several factors (e.g., physical restrictions or access characteristics).
- Access: a group of fields (“AR_AUTO”, “AR_BUS” …) which identifies the types of traffic allowed. Data types for these fields is Boolean[10] (Y/N).
- Bridge, Tunnel, Ramp, Roundabout (…): group of fields indicating through a Boolean value (Y/N) a characteristic of the street useful for display and route guidance.

---

[10] These attributes are Boolean from a logical point of view. From a physical point of view, they are a character with length equal to one.

Another feature of interest is the Traffic table (.dbf) contained in the NAVSTREETS Street Data set. In particular, this table enables the representation of Traffic Message Channel locations point over the road network. The information is stored at LINK level: for each element that has a correspondence with the TMC network a code is defined (e.g. -501-37825). This code consists of several parts as shown in Table 7.

Thanks to this reference, it is possible to locate traffic events along the network, having for each LINK the location code of the road and the location code of the point in positive and negative directions. In Figure 54, the extent of the TMC referenced to NAVSTREETS Street Data in the area of Turin is shown: red lines are the edges defined into the TMC, in grey the original network.

The NAVSTREETS Street Data, as an expensive commercial dataset, provides some assurances. A new version is provided each year, ensuring a high level of update. In particular, the identifier values (“LINK_ID”, “FEAT_ID”, “NODE_ID”) are persistent between versions, easing the process of update of a graph and related elements.

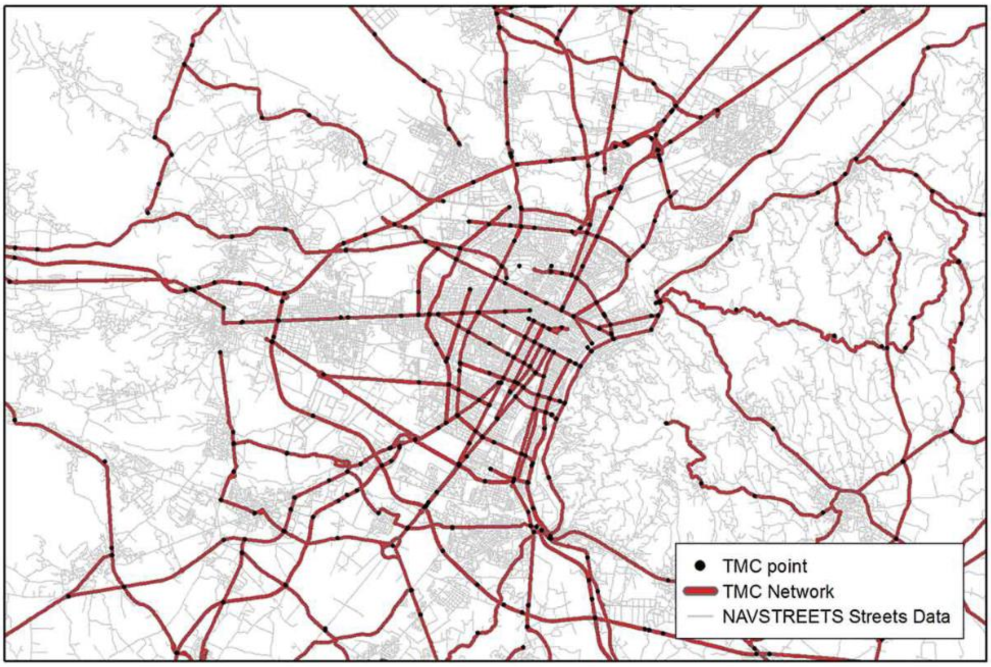


Figure 54: The TMC network referenced to the NAVSTREETS Streets Data.

At the level of attributes, the data is almost complete and reliable, at least for the most important fields for routing (in general the level of attribute completeness varies among countries). It also has multiple names of streets and house numbers, allowing an advanced geocoding and routing applications.

Spatial accuracy is declared and varies between + or - 5 metres for the absolute positioning, and + or - 1 metre for relative positioning. In addition to the spatial coverage it is generally complete, in particular as regards the paved roads traversable by cars. It provides also traces of railroads. At the level of cycling infrastructure instead, no information is provided for the shapefile product (the MapInfo product instead may contain this information at lane level).

Table 7: Definition of the attribute TRAFFIC_CD of the Traffic.dbf table [ABBCCDEEEEE], readapted from HERE, 2016.

Char	Description
A	Is the one character Direction of Road. This is based on the Direction of Travel on the link. + is the direction from the From Node - is the direction towards the From Node
B	B is the one character EBU Country Code [5 for Italy] http://www.interactive-radio-system.com/docs/EN50067_RDS_Standard.pdf
CC	CC is the two digit Location Table number [01 for TMC Italy]
D	D is the one character RDS direction, where: + is in the positive direction and external to the Problem Location - is in the negative direction and external to the Problem Location P is in the positive direction and internal to the Problem Location N is in the negative direction and internal to the Problem Location
EEEEE	the five digit Location Code

4.4 Data sources comparison

In order to evaluate accuracy and completeness of the data sources available in the Piedmont region area, some briskly comparisons have been made.

In general, most of the approaches to evaluate the completeness of spatial data over a certain area are achieved through a comparison with established references ones, for instance data produced by public administrations and institutions.

An global approach proposed by Maron (2015) is to compare OpenStreetMap data with the CIA World Factbook, which contains measurements of paved road lengths in every country in the world. From his work, it is evident that OpenStreetMap data in Italy have reached a good level of completeness (see Table 8). In order to replicate the same approach the following datasets have been chosen:

- OpenStreetMap data downloaded from GeoFabrik web site, as it provides data ready to use (in shapefile format) and with a complete documentation about mapping schema[11], for the whole Italy and for the Piedmont Region.
- OpenTransportMap data downloaded for the whole Italy and for the Piedmont Region[12].
- NAVSTREETS Street Data (2016 version) over the Piedmont Region, which can be used as main reference for completeness (only in the regional area).
- CIA World Factbook data available only for the whole Italy as tabular information and updated to 2007[13], which can be used as reference for completeness for the whole Italy.

In order to properly calculate the total length of roads over the whole Italy, data have been projected in ETRS89 Lambert Azimuthal Equal Area[14] [EPSG:3035], whereas for the analysis in the Piedmont regional area the projection WGS84 - UTM 32N [EPSG:32632] has been applied.

In Table 8, is possible to see a comparison between total kilometres length of CIA World Factbook data and the OpenStreetMap and OpenTransportMap ones. As the CIA World Factbook comprises mainly paved roads, a selection has been made for OpenStreetMap and OpenTransportMap data in order to exclude paths not routable by cars[15], even if, due to data mapping differences between sources (and different update times), there is no assurance to have selected the same dataset. From the total kilometres length, is evident that OpenStreetMap data (both from GeoFabrik and OTM) has an high level of completeness over Italy and

---

[11] The document is available at: http://download.geofabrik.de/osm-data-in-gis-formats-free.pdf

[12] According to attribute information, last update is 20-21/10/2015 over Piedmont Region.

[13] Available at: https://www.cia.gov/library/publications/the-world-factbook/geos/it.html.

[14] Used for European area statistical analysis and display purposes (http://mapref.org/LinkedDocuments/MapProjectionsForEurope-EUR-20120.pdf) [EPSG:3035]

[15] For Geofabrik data, the selection has included all the main roads (motorways, primary, secondary and tertiary), residential roads (prevailing in urban areas) and unknown/unclassified data (particular numerous in Italy – as a general consideration, it is more frequent that a common road is not classified compared to a pedestrian path).

For OpenTransportMap data, only roads classified as “fifthClass” has been excluded.

can be adequate for lot of mapping applications. In addition, the comparison of the number of features highlights the differences between OSM and OTM data: OpenTransportMap data indeed is reprocessed for topology correction, and has a higher number of features compared to the total km length, as features have been supposedly split at intersections.

For the comparison over the Piedmont Area, firstly a selection of all features intersecting the regional area has been applied (for OSM, OTM and NAVSTREETS Street Data). This procedure has been favoured to a “clip” operation in order to maintain the integrity of features that cross the administrative borders.

Table 8: Comparison of total km and number of features between OSM, OTM and CIA World Factbook in Italy.

		GeoFabrik - OpenStreetMap	**OpenTransportMap**	**CIA World Factbook**
Italy - all	Total Km	1.159.892,22	1.031.240,27	-
	Features count	3.842.748	6.037.408	-
Italy - paved	Total Km	605.256,94	592.170,69	487.700,00
	Features count	1.939.200	3.845.153	-

In addition, as NAVSTREETS Street Data contains mainly roads traversable by cars, in order to obtain a more comparable dataset, some selections have been performed. In particular, for NAVSTREETS Street Data, features classified as not paved and features that have access restrictions for cars, buses, taxis, motorcycles and trucks have been excluded from the analysis.

For OpenStreetMap data, following the GeoFabrik mapping documentation, features classified as bridleway, cycleway, footway, path, pedestrian and steps have been excluded, whereas no information about paved roads is reported in mapping schema.

For OpenTransportMap data, features classified as ‘bicycleRoad’, ‘enclosedTrafficArea' and 'walkway' have been excluded. Even if the OpenTransportMap mapping schema reports the information about pavement, this information has been considered not reliable and consequently not used for selection. Indeed, the schema maps as unpaved all roads that not have this tag information, but it is more common (almost in Italy and Piedmont) to not have this tag on common roads than the opposite. Then selecting unpaved roads, in this case, potentially excludes most of paved roads.

Table 9: Comparison of total km and number of features between OSM, OTM and NAVSTREETS Street Data in Piedmont Region.

		NAVSTREETS Street Data	**GeoFabrik - OpenStreetMap**	**OpenTransport Map**
Piedmont - all	Total Km	58.924,85	101.324,40	128.300,77
	Features count	462.875	306.465	306.465
Piedmont - car traversable	Total Km	48.935,19	82.051,99	105.459,93
	Features count	406.291	255.984	609.952

After these steps, it was evident that the OpenStreetMap data sources have more cars routable km compared to the NAVSTREETS Street Data one. This can be due to several reasons: OSM sources have in general more service streets, unpaved roads have not been taken into account, selection criteria applied cannot assure the correctness of data, also for eventual mapping errors inherited from the original OSM source. This discrepancy is particularly evident in mountain areas, as can be observed in Figure 55. In Table 9, the comparison of total km and number of features between OpenStreetMap, OpenTransportMap and NAVSTREETS Street Data is shown.

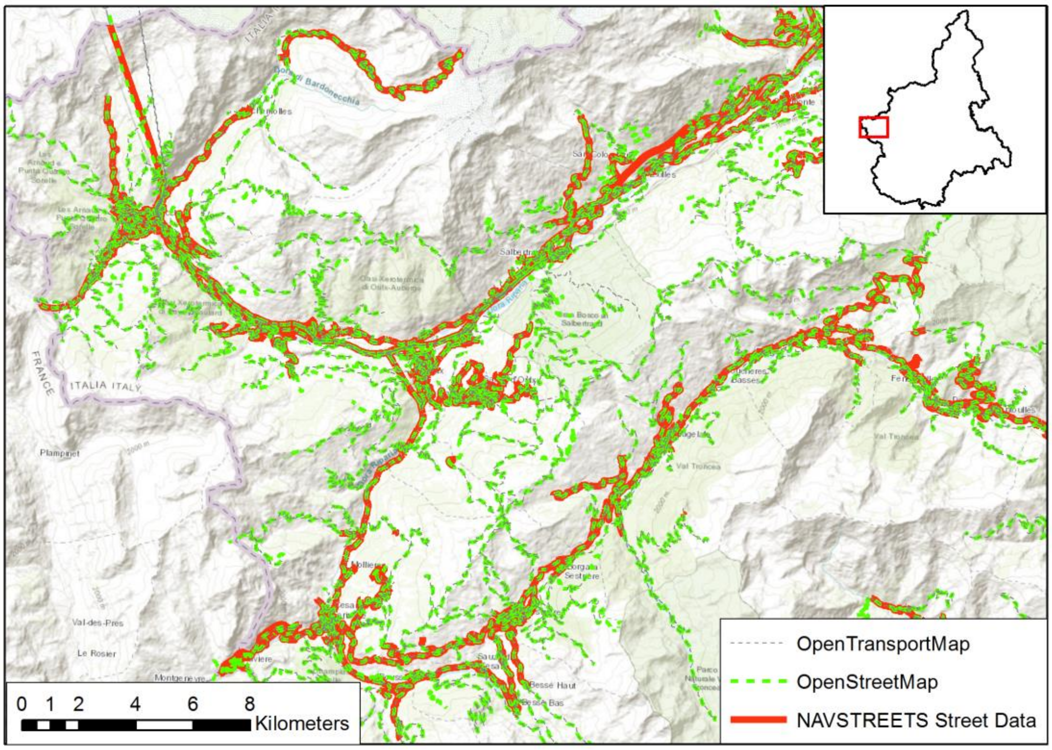


Figure 55: Spatial coverage comparison between road network data sources after the removal of edges not traversable by cars.

Following the approach described by Ludwig et al. in 2011, in order to have a more refined dataset to be compared with NAVSTREETS Street Data, features of OSM and OTM datasets within a distance of 5 m, 10 m and 30 m from NAVSTREETS Street Data have been further selected. This operation has been done in order to refine the matching between datasets, and to find a correspondence at object level. The desirable condition is to find a 1:1 correspondence between objects, which is obviously not possible.

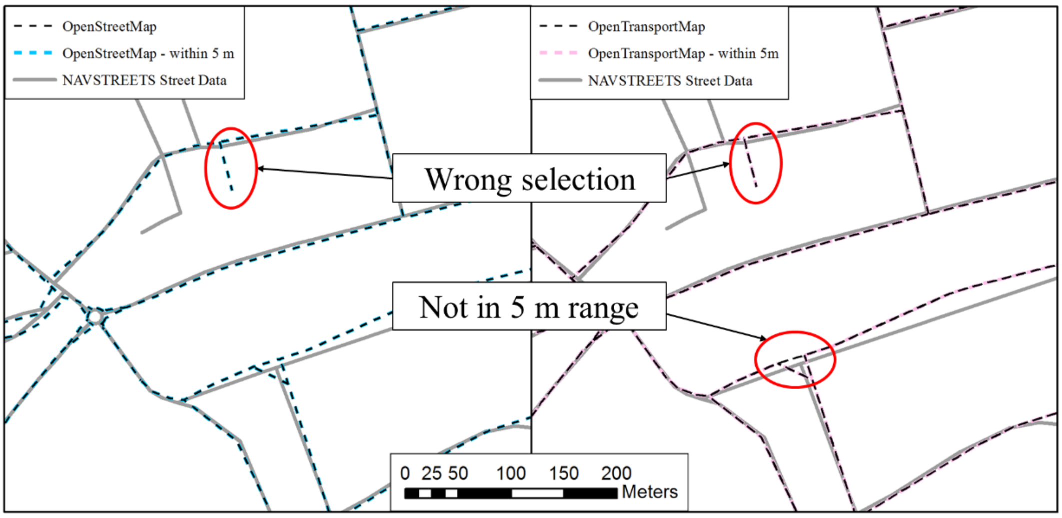


Figure 56: Selection errors applying the “within a distance of 5m” criteria.

The range of 5 m seems the most appropriate, as using larger ranges increase the spreading of matching errors: indeed, even if this operation reduces differences in datasets, results also in the selection of road branches that not corresponds logically to the original road from the NAVSTREETS Street Data, as can be view in Figure 56. Most appropriate techniques to select matching data have to be evaluated but are now out of the scope of this research. Total Km and features count for selected datasets is shown in Table 10, from which can be observed how the 5 m buffer allows to have a better matching between OSM sources and NAVSTREETS Street Data.

Data comparison has been set also checking the distribution of Functional Class. In Table 11, the attribute matching between datasets used for the analysis is shown. This approach is prone to error, as it inferred many simplifications.

Table 10: Comparison of total km and number of features between OSM, OTM and NAVSTREETS Street Data in Piedmont Region, distributed for buffer selection.

	Total km	Features count
NAVSTREETS Street Data	48.935,19	406.291
OpenStreetMap 5m	66.052,86	203.304
OpenStreetMap 10 m	67.454,18	210.398
OpenStreetMap 30 m	69.335,58	220.199
OpenTransportMap 5 m	81.406,44	498.966
OpenTransportMap 10 m	83.259,50	514.430
OpenTransportMap 30 m	85.847,40	534.555

Main differences in the map matching at functional class level concern:

- ramps and junctions classification, where in NAVSTREETS Street Data are usually classified in the lower class whereas in OSM sources in the higher one;
- service roads of major highway, where usually in OSM sources there is not a separation whereas are differently classified in NAVSTREETS Street Data;
- fourth and fifth classes in general.

Table 11: Attribute mapping between NAVSTREETS Street Data Functional Class and OSM and OTM datasets.

NAVSTREETS Street Data Functional Class	**GeoFabrik – OpenStreetMap [fclass]**	**OpenTransportMap [functional]**
1	motorway, motorway_link, trunk, trunk_link	mainRoad
2	primary, primary_link	firstClass
3	secondary, secondary_link	secondClass
4	tertiary, tertiary_link	thirdClass
5	all other values	fourthClass, fifthClass

Results, differentiated for total km and feature count are shown in Figure 57 and Figure 58. In general, more similarities between OSM sources and NAVSTREETS Street Data can be found in the group of 5 m selection. OpenTransportMap also seems to have a more matching correspondence with respect to the GeoFabrik – OpenStreetMap source.

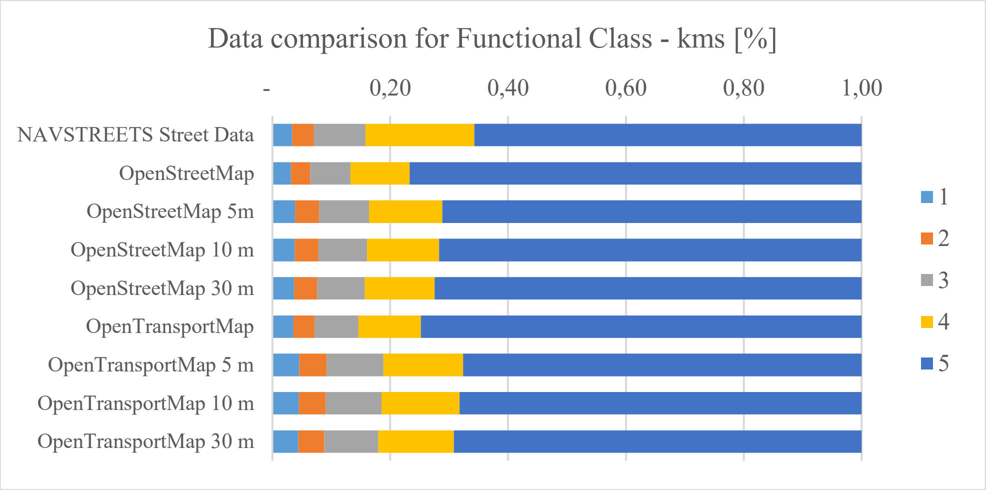


Figure 57: Data comparison for functional class, as percentage of total km.

Figure 58: Data comparison for functional class, as percentage of features count.

Another characteristic that can be assessed is the presence of an attribute name value, which is relevant for routing applications. In NAVSTREETS Street Data, as already stated, edges without name are the ones describing complex intersections. For this data source, features without a name are the 12% of the total. For OSM and OTM data, selection has been performed looking at two attributes: “name” and “ref” for GeoFabrik – OpenStreetMap, and “roadname” and “nationalroad” for OpenTransportMap. If both attributes were not filled, the feature was considered without a name (without considering possible errors in name values). As can be view in Figure 59 , features without a name are over the 50% for OSM, whereas percentages generally lower for OTM (nevertheless considerably high compared to NAVSTREETS Street Data). It is also evident from the figure that the buffer increase leads to select more features without name, probably because it has led to include more roads of the lowest hierarchy level.

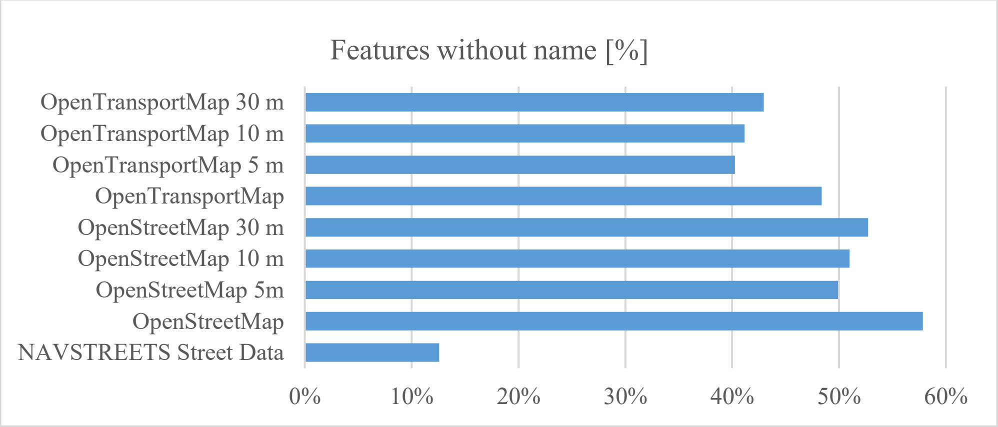


Figure 59: Comparison of number of features without a name, as percentage.

Finally, the topological correctness of data sources has been assessed. In particular, OpenStreetMap data are often not split at intersections: this is evident also in the higher value of mean length for a feature in OpenStreetMap, compared to the other two data sources, as highlighted in Table 12. A ‘Feature To Line’ operation has been performed in ArcGIS in order to evaluate the number of feature not split. New values can be found in the last row of Table 12: it can be stated that after this operation values between OSM and OTM are closer (as probably OTM have already this corrections implemented). It has to be highlighted that the Feature to line operation can also generate some error for routing application if applied to the whole dataset: if it is true that connection must be guaranteed at intersections, this may not be correct in case of tunnels and bridges, where intersection between roads does not mean a real connection.

Table 12: Comparison of mean feature length and topology errors.

	Feature Mean Length	**N° of Features**	**Must Not Self-Overlap**	**Must Not Self-intersect**
NAVSTREETS Street Data	0,127	406.291	0	0
OpenStreetMap	0,331	255.984	44	74
OpenTransportMap	0,183	609.952	5	15
OpenStreetMap - FeatureToLine	0,170	481.524	0	0

Other topological issues can be due to not connected roads and self-intersection. Evaluating through automatic procedures non-connected road is not trivial: for instance the topological rule “Must not have dangles” in ArcGIS can be applied, but there is no way to discriminate errors from ends of road. An evaluation has been performed instead using the rules “Must Not self-overlap” and “Must not self-intersect”. Results are listed in Table 12. Errors are higher in GeoFabrik – OpenStreetMap, even if the number allow a possible manual correction. As already stated, the biggest problem is related to un-connected roads and methods for assessment and correction have to be implemented.

Despite the briskly evaluation approach, it can be stated that OpenStreetMap sources (both from Geofabrick and OTM) in the Piedmont area can be considered adequate in spatial completeness for many applications. For routing purposes on the other hand, topological issues and low level of attribute completeness are the main obstacles, but strategies for correction may be evaluated in further researches, even if the corrections already applied on OTM dataset increase the topological correctness. In general, the needs of a TOC cannot still be fulfilled by the use of an open source road network data source.

Chapter 5

5T Agency

The 5T company in Turin designs, builds and manages Intelligent Transport Systems and info-mobility, for private and public transport management, over the Piedmont regional area. The aim is to improve the private traffic flow and security in urban areas, reducing congestion and pollution, and improve quality and performance of public transport services through fleets monitoring and real-time information spreading.

The 5T Agency in Turin is a related undertaking born in 1992 from a consortium created for the European founded project “Quartet” (Turin, Birmingham, Athens, and Stuttgart). In 2000 the consortium included several public and private partners operating in the field of transport (ATM, AEM, FIAT, CSST and Mizar[16]), with the aim of managing traffic control and traffic information in the city of Turin. In 2006, thanks also to the event of Turin Winter Olympics Games, the Traffic Operation Centre has been instituted, with an enlargement of the operational area outside the municipality borders. In 2008, 5T became a public limited liability company and now operates in Turin and Piedmont, as the in-house company for the City of Turin, GTT (Gruppo Torinese Trasporti), Piedmont Region and Metropolitan City of Turin.

---

[16] ATM – Agenzia Torinese Mobilità (now GTT - Gruppo Torinese Trasporti), AEM – Azienda Energetica Metropolitana (now IREN – Iride Enìa), FIAT Fabbrica Italiana Automobili Torino (now FCA Italy – Fiat Chrysler Automobile), CSST – Centro Studi sui Sistemi di Trasporto.

In the following sections services, systems and data managed by 5T will be described, mainly derived from 5T internal documentation.

5.1 Sensors system

5T company provides several services oriented to mobility management and mobility information to users.

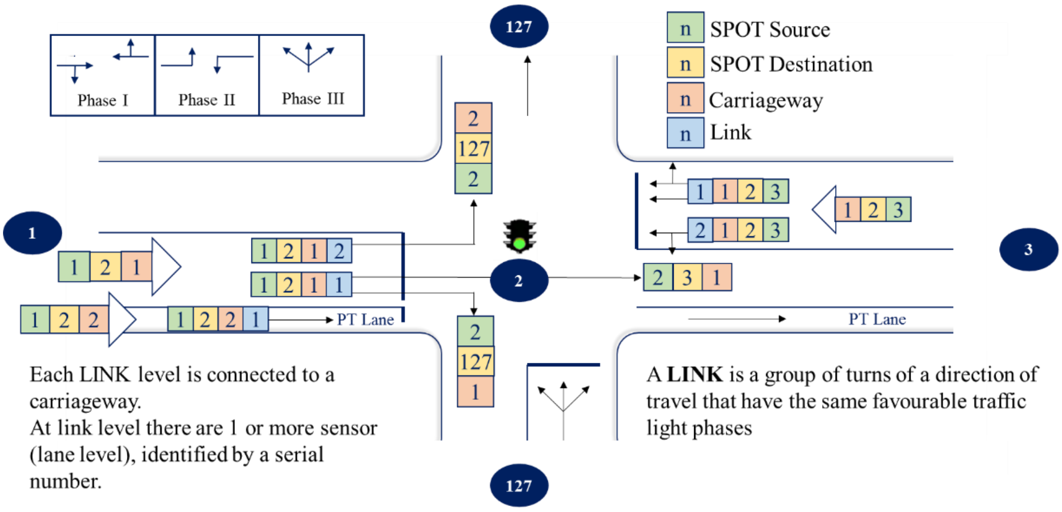
Between their activities, one is related to the management of data gathered by a number of different sensors, mainly located in the Metropolitan Area of Turin. These sensors are owned by different societies and/or public administration offices (ATIVA - Autostrada Torino Ivrea Valle d'Aosta, ANAS - Azienda Nazionale Autonoma delle Strade, SITAF - Società Italiana per il Traforo Autostradale del Fréjus, Vercelli Province, Metropolitan City of Turin, City of Turin, GTT, ARPA…) and are dedicated to different purposes.

Most of sensors are related to private traffic management. In particular, sensors in this group acquire, for one direction of flow, information about vehicle flow (veh/h) and mean speed (km/h), and eventually other parameters as vehicle weight. Different technologies characterise these sensors, in particular:

- Induction loops (with electrical or photovoltaic supply);
- Microwave sensors (with electrical or photovoltaic supply);
- Ultrasound technology sensors (only in the Metropolitan Area), installed on Variable Message Panels (VMS);
- Doppler Radar technology sensors (only in the Metropolitan Area);
- Wireless magnetic field technology.

These sensors enables many of the services in charge of the Traffic Control Centre. In particular, in the City of Turin, a subset of these traffic monitoring sensors constitute the **UTC system** (Urban Traffic Control). This system was installed in the 1990's and is designed to smooth the traffic flow through the adaptive control of traffic light phases over 330 intersections of the most important roads of the City. In addition, it allows to enable the traffic light priority service, which minimize the waiting times at intersections for public transport vehicles (increasing the regularity of the service). Over 3000 induction loop sensors are connected through devices to traffic lights, sending a continuous flow of data: depending on values gathered by sensors, traffic light phases vary in order to minimize the total delay over the network.

The UTC system is based on a complex intersection model, illustrated in Figure 60. Each traffic light intersection is managed by a “SPOT” component (about 1400), a device that receives measures from connected sensors, and, interacting with the adjacent SPOT components and the central system management, sends inputs to the connected traffic lights, modifying the green phase time in order to minimize queues. In Figure 60, the SPOT identified by the number 2 is connected with SPOT 1 and 3, which represent the adjacent managed intersections, and with the generic SPOT 127, a code that indicates an intersection not managed by the system. The intersection needs three phases to allow the vehicles flow in each direction. The coloured boxes in the figure indicate the coding system that identifies single sensors. First, each sensor is characterised by a “source” SPOT and a “destination” SPOT, which allow to define the direction of traffic flow monitored by the sensor. The third number identify the carriageway: in particular, the model considers two types of carriageways: “reserved” if only public transport vehicles are allowed, or “mixed” for all other cases.



Each LINK level is connected to a carriageway.
At link level there are 1 or more sensor (lane level), identified by a serial number.

A **LINK** is a group of turns of a direction of travel that have the same favourable traffic light phases

Figure 60: The intersection model of the Urban Traffic Control System UTC system.

Last component of the coding system is the “LINK”. A LINK is a group of turns of a direction of travel that have the same favourable traffic light phase. For instance in Figure 60, the flow coming from SPOT 1 to SPOT 2 is split in two LINK, one for left turn, and one for right turn and straight direction. Finally, through the calculation of an adjacent matrix, each LINK is associated to all possible access allowed carriageways for that intersection. UTC loop sensors are located at lane level, so for each LINK there are usually more than one sensor, identified through a serial number. In addition, sensors are usually located both on the lane entering in the intersection and on the exit lane.

The UTC system produces a number of measures, which are automatically aggregated on 5 minutes temporal range by SPOT devices, which also send the data to the central system. In particular, vehicles flow and speed are available for single loop sensor, queues and delay are available at LINK aggregation level, and turn rate are available at link level only if the adjacent carriageway matrix is defined.

As stated at top of section, several **other traffic detectors** are distributed over the Metropolitan and Regional area. In particular, as UTC system uses only induction loop sensors, other technologies are used for traffic monitoring inside and outside the City of Turin (about 400 sensors). Most of them are located on top of informative panels, traffic lights or other ancillary road elements: these kind of sensors as well as being more modern than the UTC sensors, have the advantage to be more easy to maintain, as they are not placed under the ground. These group of traffic detectors are internally managed by the PASTA component.

Cameras can also be included between sensors devoted to traffic monitoring: the system includes 80 devices in the City of Turin, which send continuous information to the Traffic Operation Centre. Cameras control 24 important intersections of the City. Other cameras are used in combination with other devices for **traffic enforcement** management: in particular, cameras are located on the access roads of some restricted access pedestrian areas in the City of Turin, where a set of retractable bollards (Pilomat), remotely managed by the TOC, regulate the access. Cameras and other sensors are used to regulate the Restricted Access Area (RAA) in Turin: cameras monitor the incoming and outward traffic on the 36 gates, allowing the semi-automatic recognition of unauthorised vehicles through the scan of the licence plate. Most of these gates are also equipped with informative panels. Between traffic enforcement devices, 5T also manages two speed cameras in Turin (Regina Margherita and Unità d’Italia Avenues), where sensors allow the semi-automatic recognition of the licence plate of vehicles exceeding speed limits and the vehicles flow and mean speed.

All sensors described up to this point are fixed, but 5T agency also manage a group of **mobile sensors**, defined as Floating Car Data (FCD). The technology is based on information gathered by vehicles (and in particular of public transport fleets, which are continuously tracked) and mobile phones that are moving on the road network. Measures gathered are real-time position of the vehicles and, consequently, speed and travel time on specific road stretches.

Finally, a particular kind of sensor can include the overall technological system built to manage the Regional ticketing system (BIP – Biglietto Integrato Piemonte), which allow users to access to the regional public transport (urban and suburban), to the rail transport and also to the bike sharing service, through the use of one single “smart card”. This technology system has several benefits, not only for regular users, but also for the public transport agencies involved, which can monitor (potentially also in real-time) the use of their services.

5.2 Traffic supervisor

Data gathered by the traffic detectors described in previous section constitute the main input for the traffic supervisor component, a software module that enable the traffic behaviour prediction, in order to provide real-time information to users and dynamically manage traffic control.

Actually in 5T, two supervisors co-exists, one operating on the regional area (Regional Supervisor – SVR) and one on the metropolitan area (Metropolitan Supervisor – SVM). In Figure 61, the operating area of the two supervisors is shown.

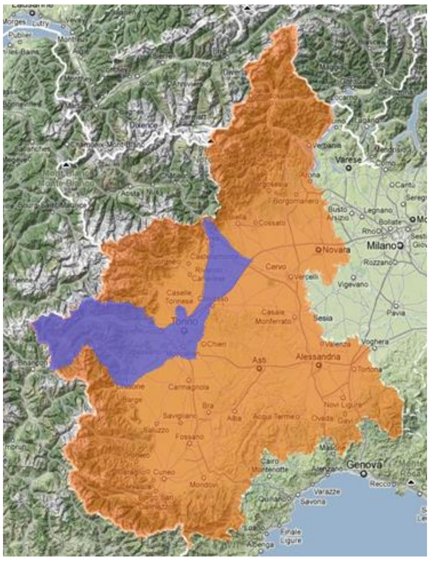


Figure 61: Operating area of the Metropolitan Supervisor (in blue) and of the Regional Supervisor (in orange).

The prediction of the capability of a road system is based on the well-known fundamental equation of traffic flow:

$$f = d * v$$

where:

- f is the number of vehicles traversing a road section in a time unit [veh/h];
- d is the number of vehicles (in a time instance) in a length unit [veh/km], and
- v is the mean speed on the road section [km/h].

The equation describe the fundamental relation between the traffic flux and the traffic density and is expressed through the fundamental diagram of traffic flow, depicted in Figure 62, which allows to identify the conditions of vehicular outflow along a road section. Speed is equal to the slope of the line that intersect the curve. From the diagram, the outflow can be stable when density is lower than critical density, instable (traffic congestion) otherwise.

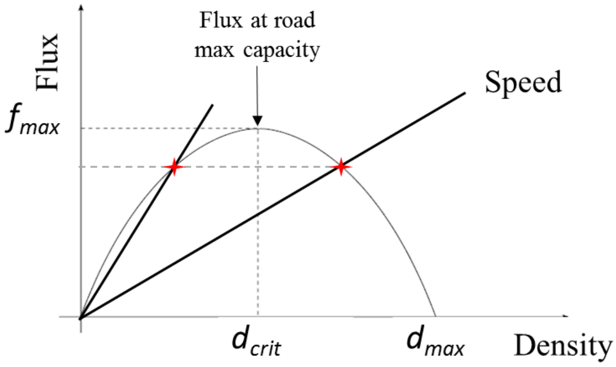


Figure 62: Fundamental diagram of traffic flow.

In order to define the traffic state (actual and forecasted), once the diagram is defined for a road section, as the flux is generally known (derived from traffic detectors), an estimation of the density is needed. This estimation can be achieved through the definition of a traffic model.

The process to build a traffic model start with the definition of an area of study (for the two supervisors, the areas in Figure 58). Then demand and supply in the area must be defined. The transport network represents the supply. The demand definition is achieved through three main steps: generation, distribution and modal choice. These steps lead to the creation of an origin – destination (O/D) matrix: it displays the number of trips going from each origin to each destination, represented by the zones in which the area of study is divided. The O/D matrix is usually derived from periodical statistical summaries. Last step for traffic model definition involves the interaction between demand and supply (assignment), describing routes used for trips from a zone to another. The assignment can be different if the network is congested or not: if usually the shortest path is the better option to move from a zone to another, in a congested network single users choices affect the movements and the behaviour of the others.

Defining actual and forecasted traffic state on the road network cannot be achieved only through the definition of a traffic model: real traffic measures gathered on the network are needed, and the use of software system as traffic supervisor helps in integrating these measures with the traffic model. In Figure 63, an overview of the integration process between traffic model and traffic measure is given.

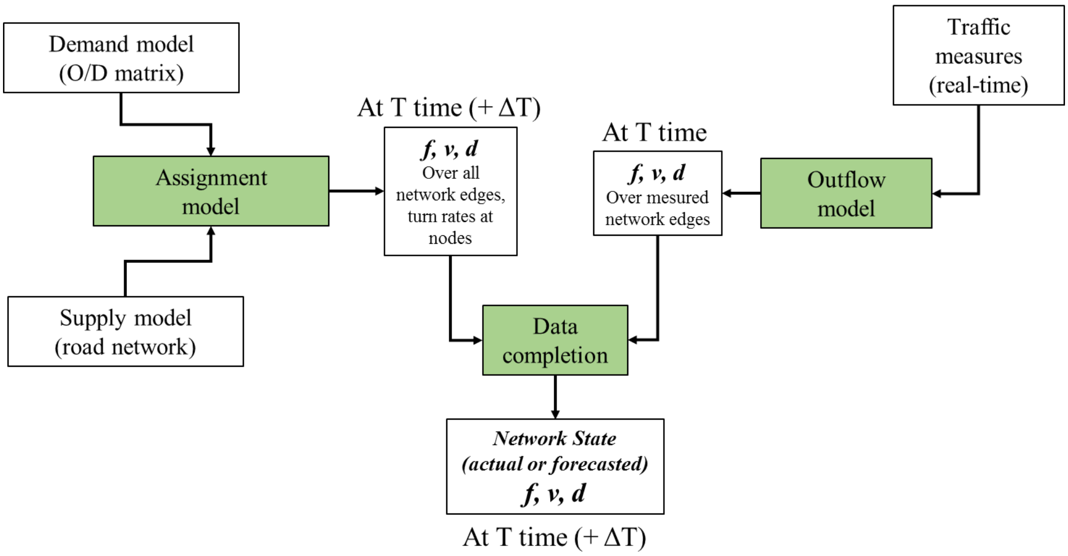


Figure 63: Overview diagram of the integration between traffic model and real measure to define the traffic state.

The short-term forecast is carried out on the basis of the measures acquired at the current time, propagated at the following instants according to the turn rate at the nodes, provided by the assignment model. Historical measures are used for model calibration.

The two supervisor used in 5T have many differences as can be noted in Table 13.

The **Metropolitan Supervisor** through mathematical models of supply, demand and allocation of traffic, is able to simulate traffic conditions on the roads in terms of traffic flow, travel times, vehicle speed and density.

The O/D matrix used is periodically updated using census socioeconomic data and surveys mobility data, and categorised for day type and reporting period. The supply model consists of a logical representation of arcs and nodes of the main roads of the study area. In Figure 64, centroids and arcs are shown.

Table 13: Comparison between regional and metropolitan supervisors.

	Metropolitan Supervisor (SVM)	**Regional Supervisor (SVR)**
Area	Turin, Chisone and Susa Valley	Piedmont Region
Zoning	367 zones (166 in Turin)	2009 zones
Demand	Hourly O/D matrix (for day types)	O/D matrix with start profiling (for day types)
Supply	Defined by 5T (5458 edges, 4100 km)	NAVSTREETS Streets Data
Assignment	Static	Dynamic

The model results are corrected with traffic measures coming from the field (flux, speed). In particular, the observations allows to detect congestions, and to produce Level of Service events, which are spread through DATEX protocol in messages for variable message panels and company website. Two operation modes cooperate with each other through a continuous exchange of data: the strategic mode and the tactical mode. The first uses information, develop strategies, and simulates the traffic flow for the long-term (60 minutes) and is repeated cyclically every hour. The second uses measured traffic data and traffic events, in order to obtain short-term estimation (5 minutes), repeating the cycle every 5 minutes.

The **Regional Supervisor** is built on proprietary software (PTV Visum and Optima), which combines the offline transport modelling method with real-time data and algorithms. In particular, the transport model is created in Visum, based on an O/D matrix created for day type. The supply model is a selection of main roads of the NAVSTREETS Streets Data (Figure 61), on which Visum applies some changes in data structure. Then the Optima module, calculating the time-related traffic volume and turning movements in networks based on travel demand, performs the dynamic traffic assignment. The real-time date are used to adjust capacity, speeds and volumes derived Visum base model. In particular, real-time data used are: flux and speed from fixed detectors, travel times from FCD, traffic lights cycles and green rates from UTC systems, data from parking facilities and traffic events (via DATEX or manually entered by operators).

The traffic state is simulated also over the roads without detectors and takes into account the impact of traffic events. The prediction of traffic behaviour is performed using a rolling approach: the simulation is based on a 60-minute time frame, producing scenario evolutions valid for 15, 30 and 45 minutes, and automatically recalculated every 5 minutes.

The scenarios produced by the software system are spread through S.I.MO.NE protocol in messages and alerts for variable message panels and company website.

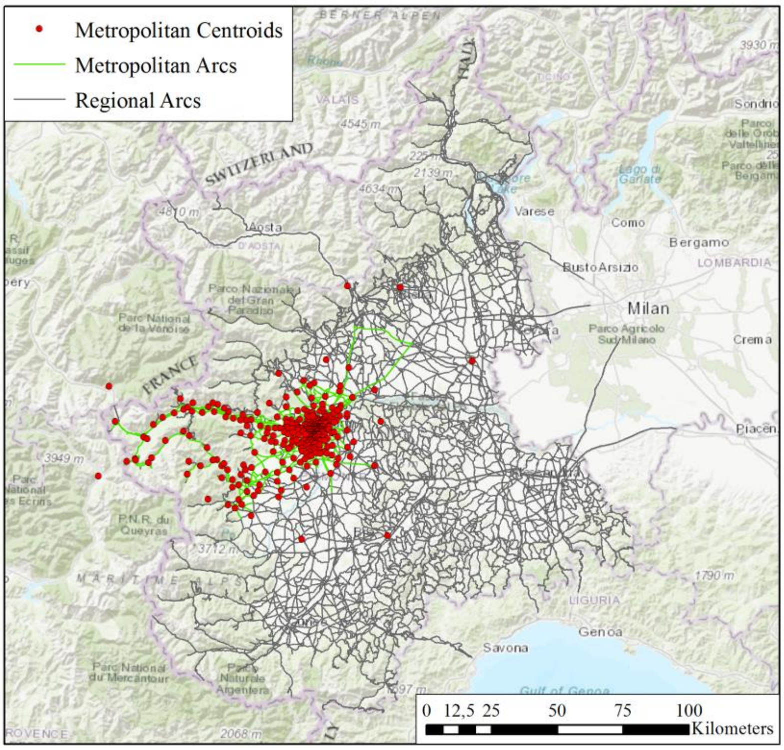


Figure 64: 5T metropolitan road network and centroids (SVM supply and demand), and SVR supply (NAVSTREETS Streets Data).

5.3 Info-mobility services

One of the main task of 5T agency is to provide info-mobility services to the public. Raw data gathered on field and processed by the supervisors become the source of information to mobility users.

Between services, one is related to private traffic information spread through Variable Message Signs (VMS): these are panels, located above roads that display real-time information to public. In particular, there are four types of VMS (Figure 65):

- VMS – T (26 in Turin, 18 in the suburban area), which display information and warnings about private traffic events, as road works, road closures and exceptional weather conditions, suggesting alternative routings.
- VMS – P (20 in Turin), which display information about parking lots availability in real-time.
- VMS – Z (36 in the proximity of the Restricted Access Area gates), which display information setting out the conditions for access (allowed or prohibited) and related schedules.
- VMS – V (2 for the Velox of Moncalieri Avenue), which display information about allowed speed.

Figure 65: Types of VMS devices managed by 5T.

The other mean used for information dissemination is the website. Websites not only provide real-time information, but also routing applications that take into account real-time information, based on a customized version of Open Trip Planner, an open source platform for multi-modal and multi-agency journey planning. Actually three websites are managed by the agency:

- "Muoversi in Piemonte" (MiP - https://www.muoversinpiemonte.it/): the website offers the real-time display of traffic status over the whole Piedmont Region and of the traffic events (from SVR). It also have a routing application that allows planning trips by different transport modes (car, bus, bike and walking).
- "Muoversi a Torino" (MaTo - https://www.muoversiatorino.it/): the website offers real-time news about traffic events, information about public transport stop arrival (scheduled and in real-time), bus tracking, availability of bike sharing (ToBike service), car sharing and taxis information, parking availability information, display of traffic status over Turin, general information about train and airport (arrivals and departures). It also have a routing application that allows planning trips by different transport modes (car, bus, bike and walking).
- "Bunet" (https://www.bunet.torino.it/): the website is designed for bike trip planning, mainly in the Turin area. The routing application take into account three parameters in order to define the most suitable bike trip: slope, security and speed. Information of the availability of ToBike bike sharing service is also available.

In addition, 10 radio traffic bulletins are broadcasted on local and regional radio stations every day.

Some other information services, mainly related to public transport, are no more managed by 5T (now in charge of GTT). In particular:

- Panels at bus stops (VIA panels) that display scheduled and in real-time arrivals;
- On-board displays into buses, with information on bus position (next stop);
- Arrivals at stop via SMS;
- Google Transit Feed provision.

5.3 5T data catalogue

In this section, data available in 5T are described. For practical purposes, only data effectively used in this research has been described. In the final chapter a discussion about possible other data to be added in further development is deepened.

The data catalogue is organised in two main part: fixed spatial information, representing road network data and ancillary traffic elements, which allow the traffic management, described in Table 14, and measures information, which comprise a set of private traffic measures, raw and estimated, and traffic events, described in Table 15.

Each element is characterised by a source, which is the description of the database and table from which data can be derived [DB.table]. Data managed in this research take into account data available in three main server instances: the instance C and D managed through Microsoft SQLServer and mainly used by SVM system, and the PostgreSQL instance, on which reside data used by the SVR system (OPTIMA database).

The description of spatial data take into account: the method of acquisition (how data has been obtained); the spatial representation, which comprise also the geographic reference system; the positional accuracy, generally expressed as a qualitative parameter, using the NAVSTREETS Streets Data as reference; the attribute accuracy, considering completeness and consistency of values and other relevant information; the spatial coverage. If available also the last update of the data is given.

Table 14: Data catalogue of spatial elements available at 5T.

UTC loops detector [2946 elements]	
Source	UTC.t_anagrafica_spire_geo_ref (instance C)
Method of acquisition	Field surveys [5T coordination].
Spatial representation	WGS84 Lat/Lon coordinates [attribute information].
Positional accuracy	High (46% of data in 3 m buffer from NAVSTREETS Streets Data, 70% in 5 m buffer).
Attribute accuracy	Complete and consistent, additional attributes about reference arcs can be found in ancillary tables, status of the devices is derived from measures.
Spatial coverage	City of Turin

UTC SPOT stations [332 elements]

Source	UTC.t_spot (instance C)
Method of acquisition	Field surveys [5T coordination].
Spatial representation	WGS84 Lat/Lon coordinates [attribute information].
Positional accuracy	High (55% of data in 3 m buffer from NAVSTREETS Streets Data, 74% in 5 m buffer).
Attribute accuracy	Complete and consistent, additional attributes about reference arcs can be found in ancillary tables, status of the devices is derived from measures.
Spatial coverage	City of Turin

PASTA traffic detectors [409 elements]

Source	SV.anagrafica_PASTA (instance C)
Method of acquisition	Field surveys [5T coordination + information given by owners].
Spatial representation	WGS84 Lat/Lon coordinates [attribute information].
Positional accuracy	Low (26% of data in 3 m buffer from NAVSTREETS Streets Data, 44% in 5 m buffer).
Attribute accuracy	Inconsistencies in attributes are related mainly to devices owner. In particular, the reference to the road can be given through: - reference to 5T arcs (for some devices in the metropolitan area); - reference to TMC (LCD_1, road_LCD, offset and direction [positive or negative]) for sensors outside the metropolitan area, or located on major road; - reference to a road name (extended or road number). Inconsistencies are also related to the monitored direction (when it is not referred to TMC): sometimes the destination of the road is used (either locality or street name), sometimes cardinal direction (‘N’ or ‘NORD’).
Spatial coverage	Piedmont Area

Cameras [172 elements]

Source	OPTIMA.webcam
Method of acquisition	Field surveys [5T coordination + information given by owners].
Spatial representation	WGS84 Lat/Lon coordinates [attribute information].
Positional accuracy	Low (14% of data in 3 m buffer from NAVSTREETS Streets Data, 23% in 5 m buffer).
Attribute accuracy	Comprises urban and suburban traffic detection cameras, Pilomat and Velox cameras. Inconsistencies are related to reference road: - Road name and km linear reference for suburban cameras; - Verbose description, including orientation (cardinal direction or name of road) for urban cameras.
Spatial coverage	Turin Metropolitan Area + Haute Alpes Département

Parking facilities [27 elements]	
Source	OPTIMA.park
Method of acquisition	Field surveys [5T coordination].
Spatial representation	WGS84 Lat/Lon coordinates [attribute information]
Positional accuracy	Low (3% of data in 3 m buffer from NAVSTREETS Streets Data, 22% in 5 m buffer).
Attribute accuracy	Name of the parking area and information about owner and capacity.
Spatial coverage	City of Turin

Pilomat [8 elements]	
Source	OPTIMA.pilomat
Method of acquisition	Field surveys [5T coordination].
Spatial representation	WGS84 Lat/Lon coordinates [attribute information].
Positional accuracy	High (50% of data in 3 m buffer from NAVSTREETS Streets Data, 75% in 5 m buffer).
Attribute accuracy	Name of the pilomat, group (Centro or Murazzi) and status (up or down).
Spatial coverage	City of Turin

VMS [172 elements]	
Source	OPTIMA.vms_v
Method of acquisition	Field surveys [5T coordination + information given by owners].
Spatial representation	WGS84 Lat/Lon coordinates [attribute information].
Positional accuracy	Medium (31% of data in 3 m buffer from NAVSTREETS Streets Data, 50% in 5 m buffer), between types, VMS related to RAA have higher accuracy.
Attribute accuracy	Consistent and complete in owner information, panel dimension and type. Inconsistencies are related to road reference: - VMS – P have no information; - VMS – Z, VMS – V and VMS – T have reference to road name (extended or road number) and direction (locality destination or cardinal direction), a km offset is given only for suburban VMS -T.
Spatial coverage	Turin Metropolitan Area

Weather Stations [60 elements]	
Source	OPTIMA.weather_station
Method of acquisition	Field surveys [information given by owners].
Spatial representation	WGS84 Lat/Lon coordinates [attribute information].
Positional accuracy	For ones located in the Metropolitan area [8] accuracy is low (12% of data in 3 m buffer from NAVSTREETS Streets Data, 3% in 5 m buffer).
Attribute accuracy	Road reference is available only for the France stations (road name, direction as destination locality and km offset).

Spatial coverage	Turin Metropolitan Area + Haute Alpes Département
Last Update	2017 (no recent measures available)

5T Arcs [5458 elements]

Source	SV.S_ARC (instance C)
Method of acquisition	-
Spatial representation	Only logical information.
Positional accuracy	-
Attribute accuracy	Attributes relevant for arcs description as measured length, number of lanes, width, even if not complete for all elements.
Spatial coverage	Turin Metropolitan Area

5T Node [2338 elements]

Source	SV.S_NODE (instance C)
Method of acquisition	5T visual interpretation from satellite imagery.
Spatial representation	WGS84 Lat/Lon coordinates [attribute information].
Positional accuracy	High (50% of data in 3 m buffer from NAVSTREETS Streets Data, 71% in 5 m buffer).
Attribute accuracy	Only identifiers and position.
Spatial coverage	Turin Metropolitan Area

5T Path [442 elements]

Source	SV.S_PATH (instance C)
Method of acquisition	Aggregation of 5T arcs.
Spatial representation	-
Positional accuracy	-
Attribute accuracy	Complete reference to the Arcs which compose the single path, and order of the arc for single path.
Spatial coverage	City of Turin

SVR Link [800126 elements]

Source	OPTIMA.strt
Method of acquisition	HERE visual interpretation from satellite imagery + additional edges manually added by 5T.
Spatial representation	WGS84
Positional accuracy	Declared by HERE documentation (5 m absolute, 1 m relative).
Attribute accuracy	Complete, with high number of attributes. Many attributes are different from the original data source (changed by OPTIMA software). The most relevant change is related to the edges duplication: the new "ref" attribute (1 or -1) allow to understand if the edge is or not traversable.
Spatial coverage	Piedmont area (plus 40 km of Level 1 Func. Class outside the region).

Last Update	2014

HERE NAVSTREETS Streets Data [465771 elements]

Source	Original shapefile provided by HERE
Method of acquisition	Visual interpretation by HERE.
Spatial representation	WGS84
Positional accuracy	Declared by HERE documentation (5 m absolute, 1 m relative).
Attribute accuracy	Complete, with high number of attributes.
Spatial coverage	Piedmont area (plus 40 km of Level 1 Functional Class outside the region). NB. The original shapefile covers the Piedmont, Aosta Valley, Lombardy and Liguria Region.
Last Update	2016

The description of measures considers the temporal aggregation of data and the temporal availability, the spatial elements to which they are related, the measures available in tables and comments on how data are periodically stored for historical purposes. Between measures described two main group can be distinguished: raw data (or bin[17]), usually refereed to sensors, and data estimated by traffic supervisors (SVM and SVR). A list is given in Table 15.

Table 15: Data catalogue of measures available at 5T.

UTC flux and speed

Source	UTC.t_flussi_bin (instance C)
Temporal Aggregation	5 minutes
Temporal Availability	From the beginning of current year to current day.
Spatial element of reference	UTC loops detector, at LINK level (no differentiation for lanes with the same turning behavior).
Measures	Mean flux [veh/h] and mean speed [km/h] on 5 minutes interval, accuracy of the measure [%].
Comments	Historical data for year are available in the t_flussi_bin_yyyy table.

UTC stationary vehicles

Source	UTC.t_code_bin (instance C)
Temporal Aggregation	5 minutes

---

[17] An automated traffic recorder monitors traffic conditions at a designated site. Depending on technology, a traffic recorder can count vehicles, detect speed of the vehicle, and distinguish between heavy and light vehicles. Traffic counts may be conducted for two-way traffic, by direction of travel, or by lane. Subtotals may be reported by various combination of time intervals; in case of UTC sensors, this interval is usually set to 5 minutes. Each of this subtotal reported is called a bin. A single bin holds the volume counted during the interval for that sensor.

Temporal Availability	From the beginning of current year to current day.
Spatial element of reference	UTC loops detector, at LINK level (no differentiation for lanes with the same turning behavior).
Measures	Mean number of vehicles [n] in a queue on 5 minutes interval, accuracy of the measure [%].
Comments	Historical data are available in the t_code_bin_yyyy table.

UTC traffic light phases duration

Source	UTC.t_fasi_bin (instance C)
Temporal Aggregation	30 minutes
Temporal Availability	From the beginning of current year to current day.
Spatial element of reference	SPOT stations
Measures	Mean duration [s] of each traffic light phase of an intersection managed by UTC system.
Comments	Historical data for year are available in the t_fasi_bin_yyyy table.

5T Arcs Measures

Source	SV.t_toc_traffico_osservato_new (instance C)
Temporal Aggregation	5 minutes
Temporal Availability	From the beginning of current year to current day.
Spatial element of reference	5T Arcs
Measures	Mean flux [veh/h], mean travel time [s], mean speed [km/h], mean density [veh/km], congestion level [%], accuracy of the measure [%].
Comments	Values estimated by the SVM based on measures (flow and speed) from traffic detectors. Historical data for year are available in the t_toc_traffico_osservato_yyyy table.

5T Paths Measures

Source	SV.t_traf_obs_macro_5 (instance C)
Temporal Aggregation	5 minutes
Temporal Availability	From the beginning of current year to current day.
Spatial element of reference	5T Paths
Measures	Mean travel time [s], mean speed [km/h], mean density [veh/km], free flow travel time [s], free flow speed [km/h], criticality level [%], congestion level [%], vehicles for km [veh*km], vehicles for hour [veh*h].
Comments	Values estimated by the SVM based on measures (flow and speed) from traffic detectors. Historical data for year are available in the t_traf_obs_macro_5_yyyy table.

SVR Measures

Source	OPTIMA.trafficstate_* (*FCD, PASTA, UTC, FAMAS)
Temporal Aggregation	5 minutes
Temporal Availability	Beginning of the day to current time.
Spatial element of reference	SVR link (strt table) identified by fields "idno", "fromNode", "toNode".
Measures	Mean flux [veh/h], mean speed [km/h], mean density [veh/km], accuracy of the measure [%].
Comments	Values estimated by the SVR based on measures from traffic detectors (UTC, PASTA sensors, FAMAS and FCD). Schema of the tables is the same, but depending on the type of sensor, different measures are available (e.g. FCD do not provide flow and density, UTC do not provide density…).

Parking Measures

Source	OPTIMA.parking_information
Temporal Aggregation	5 minutes
Temporal Availability	Last 5 minutes
Spatial element of reference	Parking access positions.
Measures	Number of free parking lots, capacity of the parking facilities.
Comments	Records are stored at every update in the parking_information_history table, which contain information starting from current year.

PASTA Measures

Source	http://opendata.5t.torino.it/get_fdt
Temporal Aggregation	5 minutes
Temporal Availability	Last 5 minutes
Spatial element of reference	PASTA traffic detectors
Measures	Mean flux [veh/h] and mean speed [km/h] on 5 minutes interval.
Comments	Data are available in S.IMO.NE XML format, which is used to expose data both for SVR (internal) and as open data (external). Additional information are related to the TMC location (LCD, Road_LCD, offset, direction, road name) and WGS84 Lat/Lon position.

Traffic events

Source	tcm_sistema.events_yyyy_mm, tcm_sistema.events_descriptions_yyyy_mm, tcm_sistema.events_locations_linear_yyyy_mm, tcm_sistema.events_locations_point_yyyy_mm (instance D)
Temporal Aggregation	-
Temporal Availability	Month

Spatial element of reference	TMC network, reported on the SVR network.
Measures	DATEXI dictionary categorisation of events.
Comments	-

Traffic events have been included into the general group of measures, in reason of their dynamic nature. Traffic events data management in 5T deserves some additional consideration. The agency manages traffic events through two databases:

- MISTIC database (DATEX I node component);
- TCM_sistema database (SVR component).

MISTIC database is part of the component DATEX I Node, which acquires information about traffic from automated and manual channels and inserts it into the MISTIC database. The location system is based on TMC standard. The component currently implements the DATEX Data Dictionary 3.1a and the version 3.3 of the TMC location table.

The TCM_sistema database (Traffic Count Manager) belongs to the SVR component and interacts with it for traffic event management and spreading. The component manages and elaborates events formalized following the DATEX I standard and implements the version 3.3 of the TMC location table. In this source, events are already materialized over the SVR road network, thanks to internal algorithm of the SVR component (Arneodo, Botta and Gagliardi, 2012; Arneodo, Foti and Cocozza 2009). The TCM_sistema database contains historical data, aggregated by month. The main table “events_yyyy_Mm”, can be related to “events_descriptions_yyyy_Mm”, which integrates a verbose description of the event.

Between the two sources this latter has been chosen as main source, firstly because the already implemented spatial materialisation has allowed to verify the scripts for TMC location referencing, and secondly because the set of available events is greater than in MISTIC (which have a limited support for historicising).

Chapter 6

Data model design and implementation

In this chapter, the core activity of the research is described. Starting from the requirements definition, the classical data model phases are deepened, stepping from the conceptual model to the 5T tailored physical data model. Then a description of conflation, data transformation and queries used to extract data is given.

6.1 5T Requirements

As described in the last chapter, the activities of 5T Agency span over different fields. The overall data management system is complex and huge, and it is difficult for a single user to have a global and exhaustive perspective of how various systems work and interact each other. In addition, the evolution of the agency tasks has been conducted through additions of new systems on top of old ones, putting first functionality, but generating overlays between applications and services (as for the metropolitan and regional supervisor), and adding complexity to the general management situation.

Probably a more extensive activity of rationalization of the systems must be implemented, but in the operational activity, this is difficult to achieve without a huge effort in terms of money, time, people and knowledge. This research wants to give a start in terms of a general integration between systems.

As already stated, the application logic is not deepened in this work: in general only the different outputs coming from software systems and applications are took into consideration for this study.

Starting from these considerations a list of requirements can be defined, on which building the conceptual data model.

The first requirement is focused on **customised spatial data representation**, which is also the main focus of the research activity. The agency has a well-established system (VI.DA, and MO.VI.DA for public transport[18]) that allows viewing real-time and current day traffic situation, down to single device detail. However, extract, manipulate and reuse those data for other purposes (as on demand analysis from public administration) is not a straightforward process.

A spatial data model that includes devices, network and measures available at 5T, combined with a set of custom scripts to perform on demand measures extraction, may ease and speed up the production of on-demand analysis. To fulfil this requirement a solid spatial geometry of the road network related devices must be build (as from the first aim of this research), assuring topology consistency between elements, and then a flexible management system for measures extraction must be associated.

Taking into account lessons learned from transport standard comparison activity, the data model produced is based on a set of well-known terms and concepts, allowing it reuse for data sharing by other TOCs.

Spatial data integration has been identified as fundamental requirement. Deepening the 5T management system has been highlighted the overlay between Regional and Metropolitan Supervisor, two systems that produce a large set of estimated measures over the same area. Measures coming out from those systems are not comparable as they uses different data as supply and demand model to estimate traffic behaviour. If the Regional Supervisor works on a more updated road network and has an higher processing power, on the other hand the Metropolitan one is more focused on Turin city area and strictly tied to the UTC system, which has been recently updated in order to improve public transport

---

[18] These applications are substantially a group of kml layers, visible through Google Earth. With VI.DA is possible to see single sensors state and data, and the general traffic state in the Metropolitan Area. MO.VI.DA allow visualising a single public transport line and relative vehicles tracking in real-time.

priority at intersection. Nevertheless, 5T defined arcs and paths that compose the supply model of the SVM have a poor accuracy, as they are formerly created as logical ones. A similar issue is related to public transport digitized paths, which have not reference on any of these transport networks.

A first improvement for a future system integration is a conflation process of the 5T arcs (and eventually of the public transport paths) on the NAVSTREETS Streets Data used by the SVR. This process involves data transformation and conflation techniques, which are further described in this chapter.

The research process to reach these objectives has not been straightforward: data extraction and cleaning, and data modelling activities have been developed in parallel, learning from errors from both sides, and finally reaching a suitable solution.

6.2 Data modelling process

6.2.1 Used technology

The use of specific software for UML management has been evaluated in order to reduce the database design effort. Indeed, the production of visual schema have benefits in the development of the project, in the communication phases and in the implementation phases, allowing a complete traceability of the design process.

As ArcGIS software is used in 5T agency, the software choice for UML design has been driven by the need to produce a geodatabase in ESRI format in the most simple and straightforward way. ESRI suggests using Microsoft Visio or Sparrx Systems Enterprise Architect (EA) for data modelling, as they provide a built-in support for modelling ArcGIS geodatabase. In particular both software allow the export of UML design schemas in an XML document, which be directly imported in ArcGIS, where data can be than loaded manually or with automated ETL scripts. Both are proprietary software with a comparable cost, but finally EA has been chosen. As matter of fact, most of the standard documentation analysed in Chapter 3 can be found only in .html or EA format (in particular INSPIRE, TRANSMODEL and DATEX documentation). The choice of EA software has made possible to acquire expertise exploring technical documentation and then to reuse the acquired knowledge in the design process.

Between features of the software, the version control system has been particularly useful in the design process, easing the traceability of working steps.

In addition, the software allows to generate different views of a generic schema, in order to create specific domain data models, showing and hiding properties, rules, and fields useful for a specific application. It also allow reverse engineer importing existing geodatabases and converting them in UML schemas. The support for ArcGIS classes is quite complete, and fields related to general ArcGIS data management (as OBJECTID, SHAPE and SHAPE_Length, spatial and attribute indexes) are automatically added at feature creation. Finally, EA make available a validation tool that allow checking errors before exporting the model.

6.2.2 Conceptual data model

First activity to develop a conceptual data model regards the clear and unambiguous definition of general elements and relationships that compose the model.

An overview of the conceptual data model is given in Figure 66. Firstly, a set of abstract classes have been defined (red in Figure 66), which can be used as template for more specific ones.

A **LINK** is a linear and oriented object, which describe the connection between two points. This is a general concept commonly used in most transport standards. A LINK can be view both as a spatial object with its own geometry, representing a physical element as a carriageway, and as a conceptual object like public transport path or a linear traffic event, with or without a geometry associated.

A **POINT** is an object used both for the spatial description of the network, and for the description of elements located the network. Therefore, it can be used to represent physical infrastructure objects like intersections, ancillary traffic elements as traffic lights, or more abstract concepts as point traffic events.

A **TRANSPORT PROPERTY** is used to define a set of characteristics that describe a transport object. These characteristics can change over time, however the update time is usually low. TRANSPORT PROPERTY is directly related to LINK and POINT classes to which they belong. This configuration is taken from INSPIRE Transport Network approach.

An **EVENT** represents a data directly referred to a LINK or a POINT object. In this class, objects changes rapidly over time, differently from TRANSPORT PROPERTY class. It is used mainly to represent raw and estimated traffic measures and traffic events. As from the FGDC Framework, the event abstract class represents the central class for the definition of an event model, separated from the segmentation one.

From these general elements, aggregations and specializations associations defines more specifics objects, and between them specific relationships are defined.

The **NetworkLink** class derives from LINK, and is used to describe the physical infrastructure and its basic characteristics. A NetworkLink can be specialized in *RoadLink, RailLink* and *TrackLink*. The RailLink describes iron links used by vehicles as trains (but eventually also trams); the TrackLink describes links used by pedestrian and soft mobility vehicles (as cycle paths). This functional separation is taken from INSPIRE and FGDC Framework.

NetworkLink objects can be grouped in *LinkSequence* and *LinkSet*. A LinkSequence is an ordered list of NetworkLink objects defining a continuous path without branches through the network, as from INSPIRE definition. Two child classes define the possible use of a LinkSequence: the *MonitoringArc* class represents a group of links mainly used for traffic monitoring purposes, and the *PTRoute* class represents the linear path covered by a public transport vehicle in a specific direction of travel. In a LinkSet, in contrast, grouped links do not have specific ordering and can be eventually unconnected, but usually share a common property: the child class defined is the *PTLine*, which groups two PTRoute objects (composite relationship) in the opposite direction, known to the public by a common name or number. Therefore, it is evident that a LinkSet can be composed by LinkSequence objects.

The POINT abstract class is specialized in **NetworkNode** and **PointOnLink** classes. These two elements differ in the relationships with the NetworkLink class.

A **NetworkNode** is a point spatial object used for describe connectivity between links: they must exist at either ends of a link. Two associations relate this class to the NetworkLink one: each NetworkNode can be the start of one or more NetworkLink and, reciprocally each NetworkNode can be the end of one or more NetworkLink. The relationships are bidirectional: given a NetworkLink object is possible to know which are the NetworkNode objects that define it, and given a NetworkNode object is possible to know which is the NetworkLink defined.

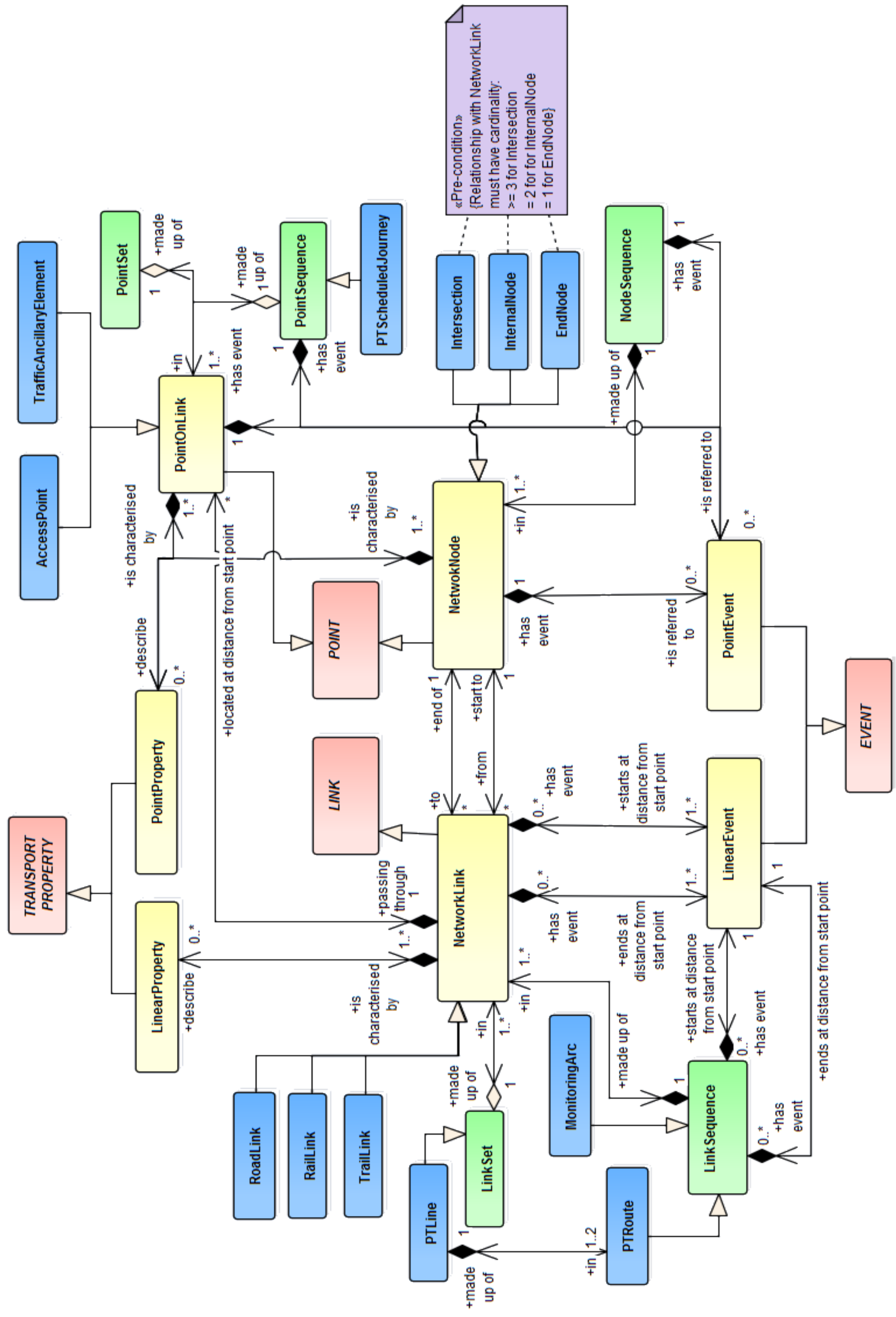


Figure 66: The conceptual data model.

The NetworkNode class is further specialised in *Intersection*, *InternalNode* and *EndNode* classes. A NetworkNode can be instantiated as an Intersection only when the cardinality of the relationship with NetworkLink is grater or equal to three, as an InternalNode if the cardinality is equal to two and as an EndNode if cardinality is equal to one. This configuration defines a more strict condition on topology rules and connectivity behaviour.

A **PointOnLink** is a point object not used for links definition but for the representation of network ancillary elements as traffic lights, traffic monitoring devices and traffic information devices, or other objects that can be located along a link as access points or public transport stops. The association between the NetworkLink and PointOnLink classes exploits the concept of linear referencing: a NetworkLink object can pass through zero, one or multiple PointOnLink object, and reciprocally and a PointOnLink object must be located on a NetworkLink object, at a certain distance from its start point. It has to be noted that this latter relationship is a composite aggregation: if a NetworkLink object is deleted, PointOnLink objects located along it are also deleted.

PointOnLink child classes are: *TrafficAncillaryElement*, which defines devices useful for traffic management, and *AccessPoint* that describes elements as parking access, access to point of interest or public transport stops.

PointOnLink class and NetworkNode class can be respectively aggregated in ordered lists as *PointSequence* and *NodeSequence* classes. A PointSequence can represent a *PTScheduledJourney* (ordered stop lists of a vehicle in a route); this latter association is a simple aggregation, so if a point part of the sequence is deleted, the PointSequence object continue to exist. In addition, a *PointSet* class is defined as aggregation of PointOnLink objects: this class can represent objects as group of access to the same public transport stop or point of interest, or group of traffic detectors.

The TRANSPORT PROPERTY abstract class is specialised in **LinearProperty** and **PointProperty** child classes. Those two classes are related with a composition association to their relevant spatial classes.

Finally, the EVENT abstract class is specialised in **LinearEvent** and **PointEvent** classes, and related to their pertinent spatial object with a composition association, which can be single objects as NetworkNode, PointOnLink and NetworkLink, or grouping objects as LinkSequence and

PointSequence. In addition, the LinearMeasure relationship with spatial element uses linear referencing method, which allows specifying a valid measure also along a partial stretch of the linear element.

The conceptual model takes into account several aspects: a basic geometry definition and rules for topology definition from the segmentation point of view, and a basic semantic level, which comprises also ancillary traffic elements, usually not considered in other standards. In addition it try to integrate the general road network with elements of the public transport network and, through the event modelling, temporal measures related to private traffic as public transport monitoring.

6.2.3 Logical data model

The logical data model deepen the details of the conceptual data model in an application independent way, defining attributes and specifying relationships. In addition, also enumerations are added to model. In this phase, specific objects that belong to 5T environment are described, focusing on private traffic management.

As a first step, a common set of attributes that characterises all elements of the model have been defined. In particular, the field “InspireID” is global unique identifier, useful both for external publication purposes, and for version management (the type “Identifier” indicates that it is a global unique identifer); fields “validFrom”, “validTo”, and “VersionID” complete the set of fields to be used for version management. It has to be noted that the “InspireID” is a candidate primary key in the model, however other identifier fields, specific of the defined classes, are instead used as primary key. Even if most of classes have been defined as “FeatureType”, a geometry field is explicitly added only for the class that represent linear road network, as all other elements can be referenced to it, and may optional have a spatial representation. As a rule, candidate primary keys are identified in diagrams using {id} notation.

Starting from the abstract classes described in the conceptual model, the logical model adds concrete classes and general attributes. A diagram of classes involved in road segmentation model is shown in Figure 67, and enumerations involved in this diagram are shown in Figure 68.

The abstract classes **LineFeature** and **PointFeature** contain the general fields for version management and data publication that will be inherited by the derived classes.

The **NetworkLink** class represents the linear road network: the “LinkID” field is the primary key of the table, used as foreign key for the defined relationships. The NetworkLink class contains a set of attributes with associated enumeration values, which are mainly derived from INSPIRE Transport Network standard.

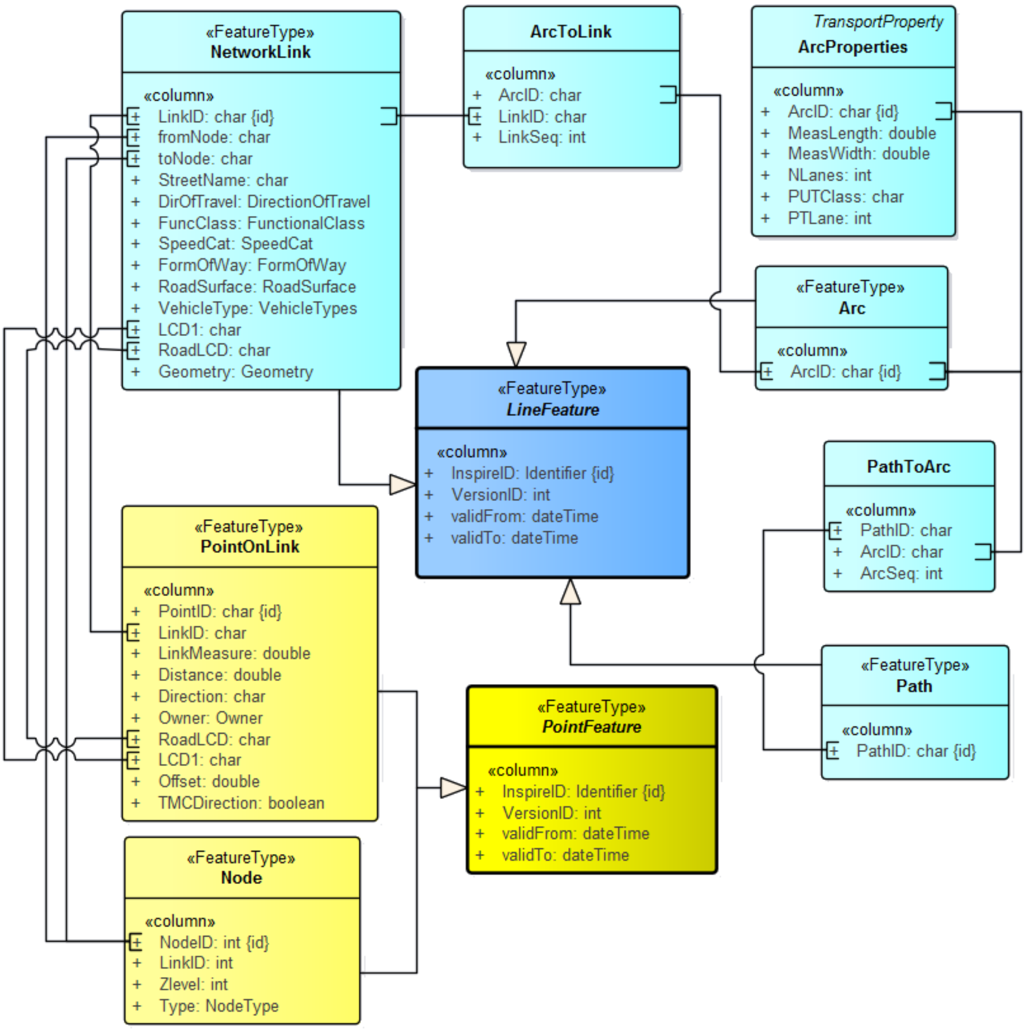


Figure 67: Diagram of road network elements.

Arc also is a child of LineFeature class, representing an example of the LinkSequence/MonitoredArc class defined in the conceptual model. It is characterised by a set of properties defined by the external table ArcProperties (related through the field “ArcID”). The feature type **Path** represents another example of LinkSequence/ MonitoredArc, which groups together a series of Arc objects. Path is a specific element used in 5T, mainly for traffic measures representation. In order to concretize the relationship between NetworkLink objects and LinkSequence, relationship tables have been defined: the **ArcToLink** table allows relating NetworkLink objects with Arc ones through the reference to identifiers, and, in the same way, the **PathToArc** table explicates the relationship between Arc and Path objects.

Looking at PointFeature child classes, the **Node** class represents elements used for line network definition. In particular, the field “Zlevel” is directly derived from NAVSTREETS Streets Data, and describes the relative vertical position of the Node respect to the ground level, allowing to identify tunnels and bridges. The field “Type” is used discriminate between intersections, internal and end nodes. The Node class is related to the NetworkLink class using the fields “fromNode” and “toNode” as foreign keys; additionally, Node class contains also the foreign key “LinkID”, which identifies the reference link. This is enough to assure connectivity and define the direction of travel of the link.

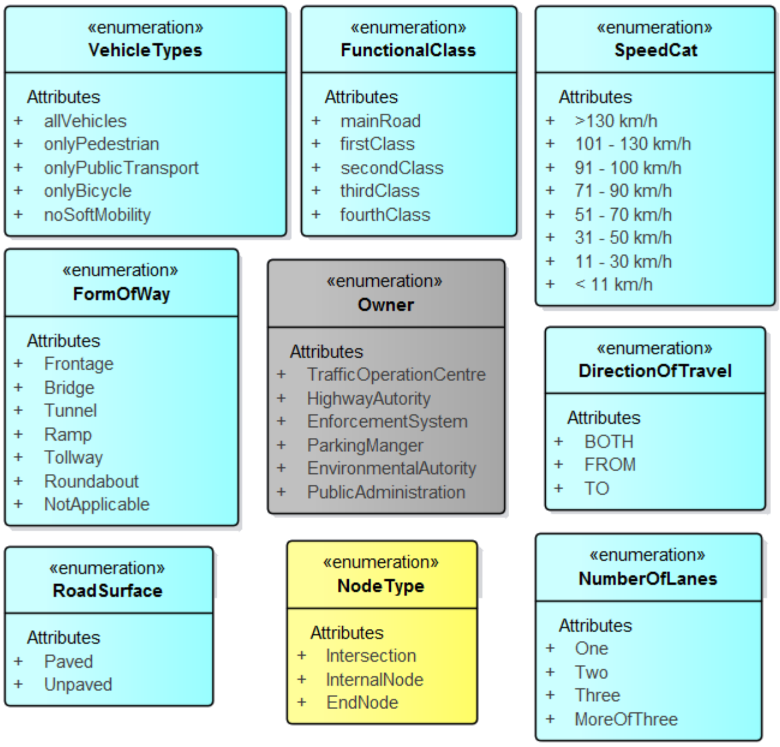


Figure 68: Enumerations involved in road network definition.

PointOnLink is the other child class of PointFeature and represents an object located along the linear network. Two linear location systems have been implemented: through the reference to the foreign key “LinkID” and the use of “LinkMeasure” and “Distance” fields is possible to locate the point along the link (as instance using ArcGIS linear referencing Toolbox). In addition, fields representing the TMC linear location systems can be used, implementing a custom linear referencing script. Both implementations have been taken into account as the TMC referencing is the most used for traffic events information spreading (e.g. through DATEX protocol), whereas others types of referencing allow to locate objects and events also on the lower hierarchy road network.

The PointOnLink class is further specialised in several classes, which represents the set of ancillary traffic elements that must be taken into account for traffic management. In Figure 69, a diagram shows these classes, and in Figure 70 related enumerations are shown.

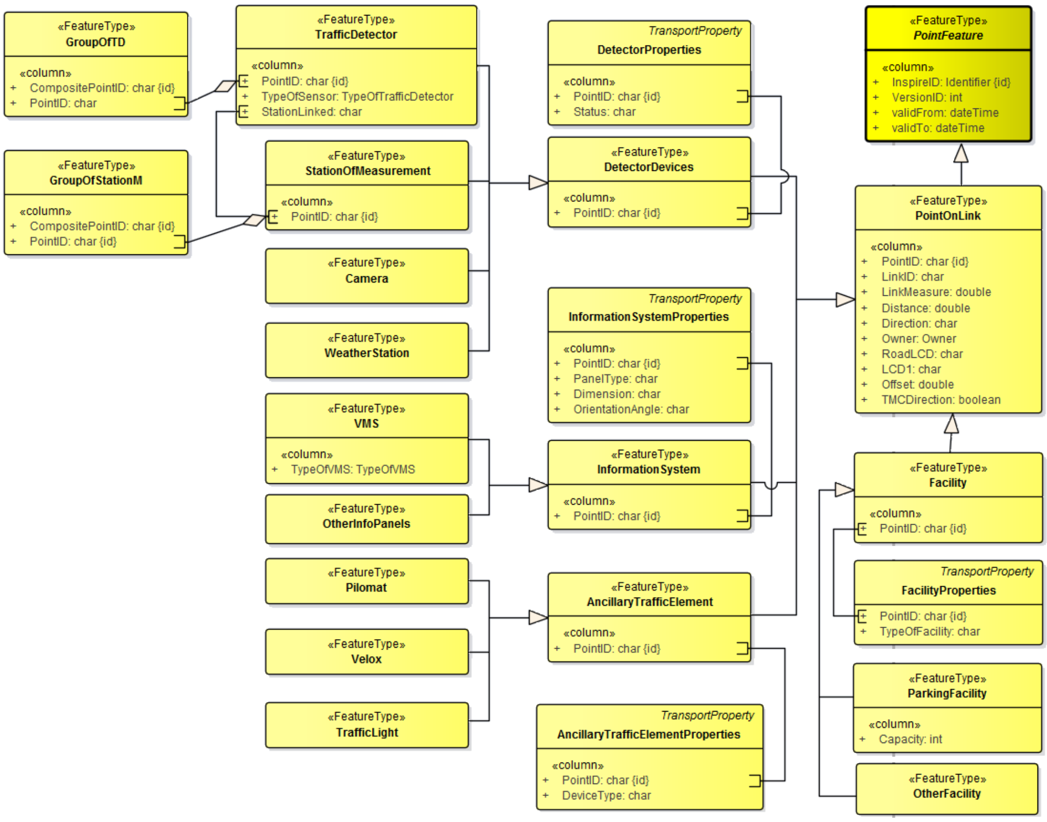


Figure 69: Diagram of ancillary traffic elements.

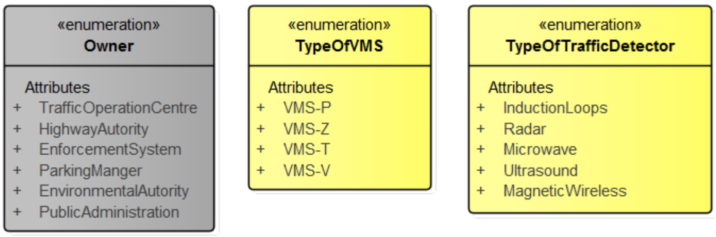


Figure 70: Enumerations involved in ancillary traffic elements definition.

Finally, a set of elements derived from the Event abstract class is used for specifying traffic measures and events. A diagram showing classes and their relationships is shown in Figure 71, and related enumerations are shown in Figure 72.

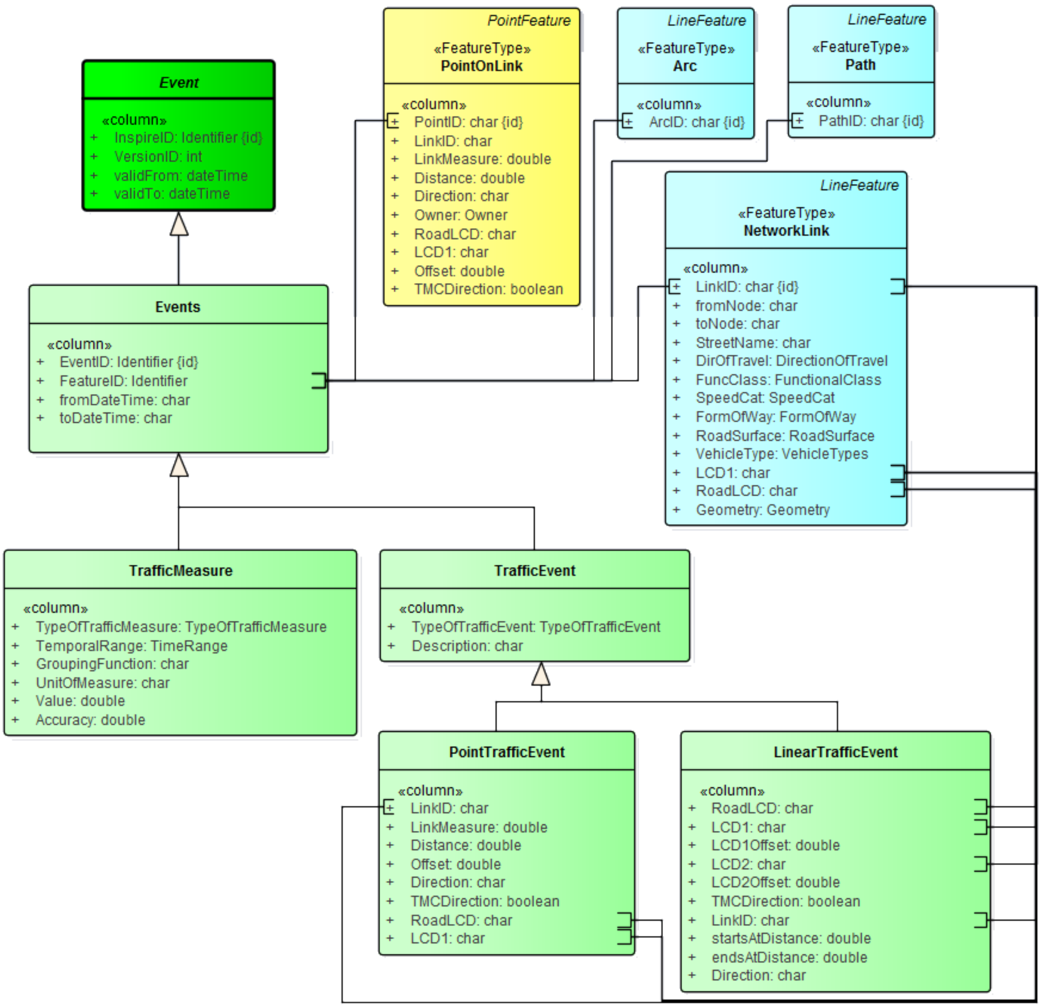


Figure 71: Diagram of events and relationships with network elements.

The general **Events** class contains attributes useful to link data to spatial element (“FeatureID”) and to define its temporal extent. In contrast with the previous classes, Events is always a table and needs to be linked to a spatial element or located through linear referencing in order to be visualised.

The child class **TrafficMeasure** has a set of attributes that allow to describe and characterise the type of measure. This attributes are the ones derived from the standard “Observations and Measurements” described in section 3.2.1.

The other child class **TrafficEvent** is further specialised in order to define a point or linear event. In this case, in addition to the “FeatureID” attribute, also fields useful for TMC referencing method are defined. Indeed, a TrafficMeasure can be represented both as line or point, but it is usually referred to a defined PointOnLink, NetworkLink or groups of them. This is not true for traffic events, which can involves stretch of roads or located along a link (or link sequences): the TMC referencing enables this kind of visualisation. The associated enumeration TypeOfTrafficEvent allow distinguishing between several events, and is based on the DATEXI dictionary used in 5T. The “Description” field can be used to further describe the traffic event.

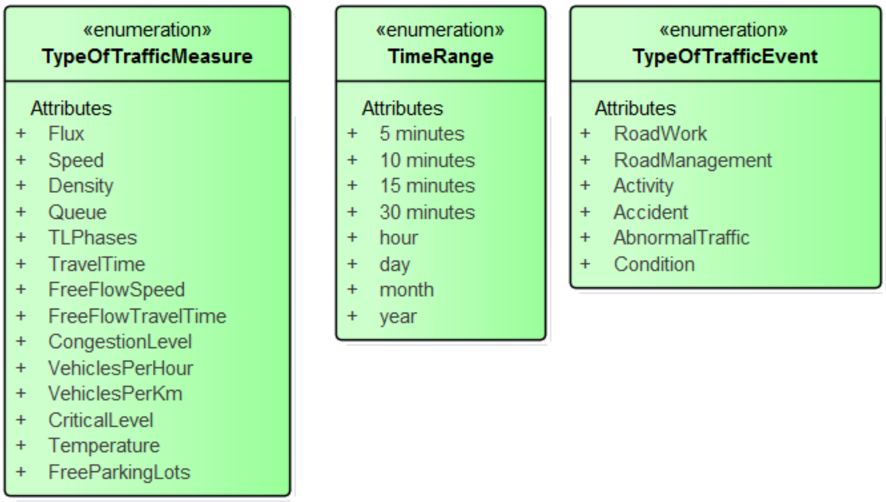


Figure 72: Enumerations involved in event definition.

6.2.4 Physical implementation

The physical data model is designed using ArcGIS class concept and it is tailored on data available at 5T. This section documents through diagrams and tables the entire structure.

Compared to logical data model, enumerations have been converted in ArcGIS domains and associated to relevant fields. The geodatabase is composed by one feature dataset called **RoadNetwork**, which contains the polyline feature class **Street**, a customisation of the NAVSTREETS Streets Data, and the relationships for which it has a source role; all other elements are tables and so, located outside of the feature dataset. In general, the identifiers of the objects are of type “String”: most of these values are derived from the 5T data and sometimes multiple fields have been concatenated in order to obtain a unique key for relationships. In Table 16, a description of most common fields is given.

The first diagram in Figure 73 illustrates the structure of linear and point elements that define the topology structure of the network, and associated sequences of links; relevant domains for this group of elements are shown in Figure 74. An exhaustive description of the elements and fields depicted is found in Table 17. In Table 18, a description of domains and relevant NAVSTREETS Streets Data field conversion is given. Table 19 describes the relationships defined between road network elements.

A diagram of the ancillary traffic elements and relevant relationships is depicted in Figure 75, and relative domains in Figure 76. Relationships between ancillary traffic elements and road network elements are illustrated in Figure 77. In Table 20, a description of relevant elements and fields is given, and in

Table **21**, the description of domains can be found. Table 22 describes the relevant relationships for this group.

Last group includes measures and events, and their relationships with spatial elements. The diagram in Figure 79 shows the TrafficMeasure table and its subtypes, which define the different types of traffic measures taken into consideration in this research. For each subtype, in addition, a set of default values is specified (as instance, the unit of measure). In order to define these relationships, two abstract classes have been created, depicted in Figure 78: a **LineFeature** class, as a generalisation of StrtSVR, Arc and Path; and a **PointFeature** class, as a generalisation of ancillary traffic elements. These two abstract classes allow to create a template for the relationship classes **AncillaryTrafficElementHasMeasures** and **LinearTrafficElementHasMeasure,** which will be used by the customised script for measures extraction, in order to define specific relationships (e.g. AncillaryTrafficElementHasMeasures template will generate

LoopUTCGroupHasMeasures relationship class). A detailed explanation of the process is further given in section 6.3.

Looking at traffic events table (Figure 80), no explicit relationships have been created to locate events along edges, as it is not possible through a simple ArcGIS relationship class: in this case, a custom script has been developed in order to uses the TMC location referencing. In this case, a subtype has been used to differentiate point traffic event from linear ones, with associated default values. A list of tables and fields description of measure group is given in Table 23 , and the description of the relationship is given in Table 25. Finally, domains relative to measures and events are illustrated and descripted in Figure 81 and Table 24.

Table 16: List of common fields used in various tables of the data model.

General common fields

Field Name	Description
GUID	Global Unique Identifier for version management and data publication (INSPIRE).
VersionID	Identifier of the version (INSPIRE).
validFrom	Date and time from which the version of the data is valid (INSPIRE).
validTo	Date and time to which the version of the data is valid (INSPIRE).
LCD1	Location Code of the TMC point to which the event ends.
LCD2	Location Code of the TMC point from which the event starts.
RoadLCD	Reference to the road in the TMC network
LCD1Offset	Offset distance from primary location point.
LCD2Offset	Offset distance from secondary location point.
TMCdirection	TMC direction, positive or negative.
CodName	Verbose identifier of the device
Distance	Distance offset (positive or negative) from Link, used for linear referencing.
KMOffset	Distance in km from the start of the road, as linear reference for device location.
LinkID (in traffic ancillary elements)	Identifier of the link on which the device is located, used for linear referencing.
LinkMeasure	Measure along the Link, used for linear referencing.
Locality	Name of the locality on which the device is located
Owner	Identifier of the owner/manager of the device
RoadLoc	Name of the road on which the camera is located
ArcID (in traffic ancillary elements)	Concatenation of NODEA, NODEB and ArcType fields in order to have a unique foreign key allowing connection with ARC spatial element.
fromDateTime	The date and time from which the observation is valid.
toDateTime	The date and time to which the observation is valid.

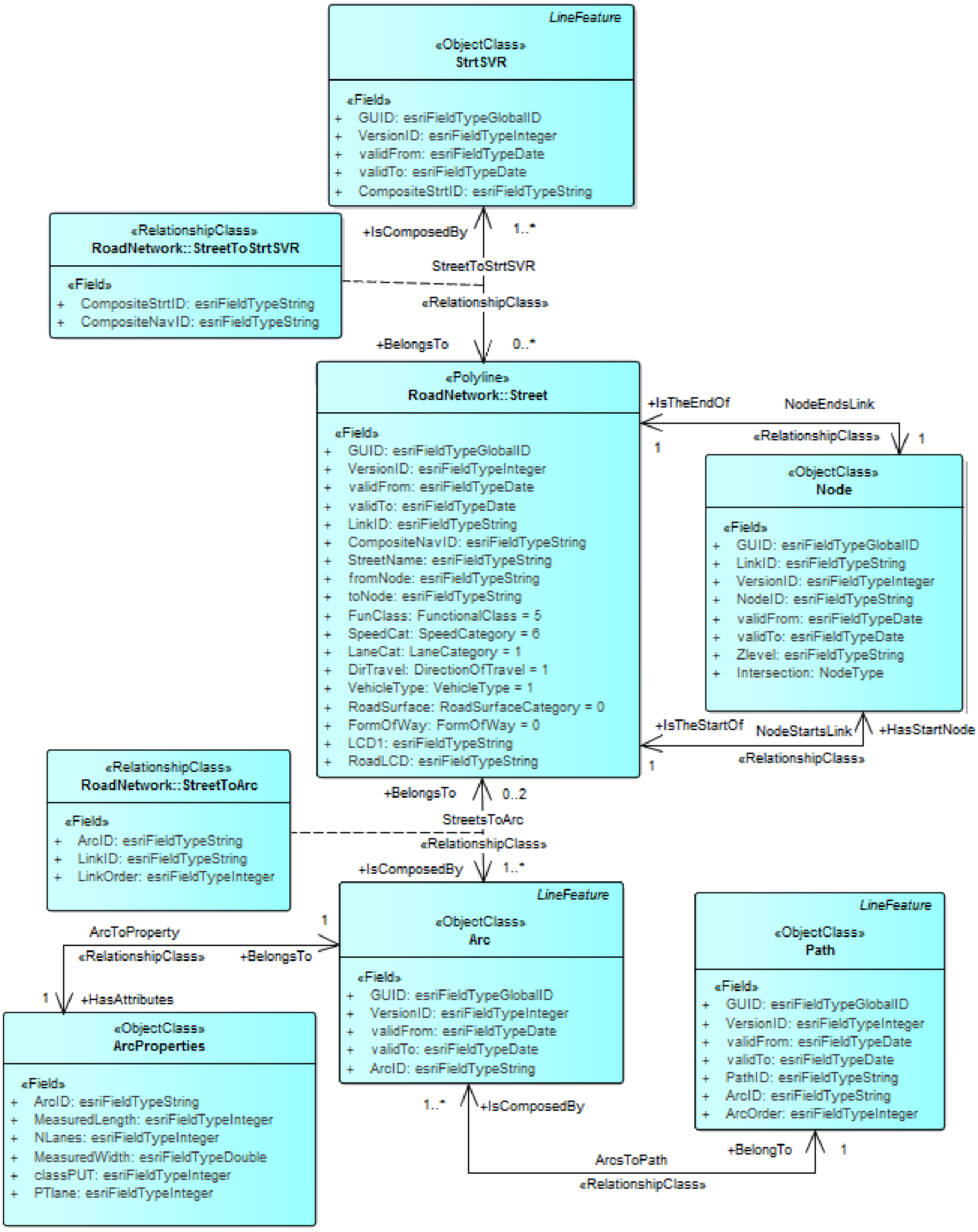


Figure 73: Diagram of the road network elements, and relevant relationships.

Table 17: Fields description of elements that compose the road network.

Streets **[NAVSTREETS Streets Data of 2016] - Polyline**

Field Name	**Description**
CompositeNavID	Concatenation of LinkID, fromNode and toNode field of Streets for relationship with strt data (SVR).
DirTravel	Direction of Travel identifies legal travel directions for a navigable link. Values mapped in order to fit the fromNode and toNode.
FormOfWay	Classification based on the physical properties of the road element. The coded domain is derived from INSPIRE FormOfWay code list, and it map a series of boolean attributes of the NAVSTREETS.
fromNode	Identifier of the Node from which the link is digitized.
FunClass	Functional Class defines a hierarchical network used to determine a logical and efficient route for a traveller. Original Functional class values are mapped in order to fit INSPIRE FunctionalRoadClassValue enumeration.
LaneCat	Lane Category classifies a road based on the number of lanes in each direction. Original attribute of NAVSTREETS.
LinkID	Unique identifier for the link. Is used to identify each link in the database. Link IDs are unique.
RoadSurface	Describes roads that are made of materials which create a solid surface. Domain associated is taken from INSPIRE RoadSurfaceCategoryValue code list.
SpeedCat	Speed Category classifies the general speed trend of a road based on posted or legal speed. Values represent the combination of several factors besides legal speed limit. Therefore Speed Category values can differ from Speed Limit values, which represent the legal speed limit only.
StreetName	The Street Name is a combination of Feature Name, Street Type, Name Prefix and Name Suffix.
toNode	Identifier of the Node to which the link is digitized.
VehicleType	Identifies the types of traffic allowed on a link. Original boolean fields of NAVSTREETS has been remapped in order to fit the INSPIRE VehicleType code list.

Node **[Zlevel of NAVSTREETS Streets Data of 2016] - Table**

Field Name	**Description**
NodeID	Identifier of the intersection node (otherwise 0).
Intersection	Indicates if the node represents an intersection of roads or an internal node of the link.
Zlevel	Gain or loss in elevation.
LinkID	Identifier of the link to which the node belongs.

StrtSVR **[NAVSTREETS Streets Data of 2014, from OPTIMA SVR] - Table**

Field Name	**Description**
CompositeStrtID	Concatenation of original fields idno, tail, head of the SVR strt for relationship with NAVSTREETS.

Arc **[Arcs defined by 5T] - Table**

Field Name	Description
ArcID	Arc identifier.

Path **[Path defined by 5T - group of arcs] - Table**

Field Name	Description
PathID	Identifier of the path.
ArcID	Identifier of the arc which compose the path.
ArcOrder	Serial number identifying the order of the arc (direction of travel of the path).

ArcProperty **[Characteristics of arcs] - Table**

Field Name	Description
ArcID	Arc identifier.
MeasuredLength	Measured length of the arc.
MeasuredWidth	Measured width of the arc.
Nlanes	Number of lanes of the arc.
classPUT	Class of the "Piano Urbano del Traffico" to which the arc belong.
PTlane	Describe if there is or not a lane reserved for the public transport vehicles.

Table 18: Domain description for elements that compose the road network.

Domain Name	Description
DirectionOfTravel	Describe the direction of travel.
FormOfWay	Describe the type of road, it take into account INSPIRE code list FormOfWay, with some customization: Bridge: BRIDGE equal to Y; EnclosedTrafficArea: UNDEFTRAF equal to Y; Frontage: FRONTAGE equal to Y; Private: PRIVATE equal to Y; Ramp: RAMP equal to Y; Roundabout: ROUNDABOUT equal to Y; Tollway: TOLLWAY equal to Y; TrafficSquare: INTERINTER equal to Y; Tunnel: TUNNEL equal to Y.
FunctionalClass	Classification based on the importance of the role that the road performs in the road network, readapted from INSPIRE Transport Network.
LaneCategory	Lane Category classifies a road based on the number of lanes in each direction.
RoadSurfaceCategory	Describe the type of surface of the road, using the INSPIRE code list RoadSurfaceCategory. Paved: PAVED equal to Y; Unpaved: PAVED equal to N.
SpeedCategory	Speed Category classifies the general speed trend of a road based on posted or legal speed. Speed Category values represent the combination

	of several factors besides legal speed limit (e.g., physical restrictions or access characteristics). Therefore, Speed Category values can differ from Speed Limit values, which represent the legal speed limit only.
VehicleType	Describe the allowed vehicle for the road, it take into account INSPIRE code list VehicleType, with some customization. Original fields of NAVSTREETS have been mapped in order to fit these new categories: allVehicles: AR_AUTO, AR_BUS, AR_TAXIS, AR_CARPOOL, AR_TRUCKS, AR_TRAFF, AR_DELIV, AR_EMERVEH, AR_MOTOR equal to Y, AR_PEDEST equal to N; bicycle: not mapped; noRestriction: AR_AUTO, AR_BUS, AR_TAXIS, AR_CARPOOL, AR_PEDEST, AR_TRUCKS, AR_TRAFF, AR_DELIV, AR_EMERVEH, AR_MOTOR equal to Y; pedestrian: AR_PEDEST AR_DELIV, AR_EMERVEH equal to Y, all other equal to N; publicTransport: AR_BUS, AR_TAXIS, AR_CARPOOL, AR_EMERVEH equal to Y, all other equal to N;
NodeType	Domain for Node definition (internal, end node or intersection).

Figure 74: Relevant domains for road network elements.

Table 19: Relationships for elements that compose the road network.

Relationship Classes

Name	Characteristics	
StreetToArc	Direction: Bi-Directional, Type: Simple Additional attributes: ArcID, LinkID, LinkOrder OriginPrimaryKey: Street.LinkID OriginForeignKey: StreetToArc.LinkID DestinationPrimaryKey: Arc.ArcID DestinationForeignKey: StreetToArc.ArcID	
	Source	*Target*
	Name: Street Role: BelongsTo Cardinality: [0..2]	Name: Arc Role: IsComposedBy Cardinality: [1…*]
StreetToStrtSVR	Direction: Bi-Directional, Type: Simple Additional attributes: CompositeStrtID, CompositeNavID OriginPrimaryKey: Street.CompositeNavID OriginForeignKey: StreetToStrtSVR.CompositeNavID DestinationPrimaryKey: StrtSVR.CompositeStrtID DestinationForeignKey: StreetToStrtSVR.CompositeStrtID	
	Source	*Target*
	Name: Street Role: BelongsTo Cardinality: [0..*]	Name: StrtSVR Role: IsComposedBy Cardinality: [1…*]
NodeEndsLink	Direction: Bi-Directional, Type: Composite OriginPrimaryKey: Node.NodeID OriginForeignKey: Street.toNode	
	Source	*Target*
	Name: Node Role: HasEndNode Cardinality: [1]	Name: Streets Role: IsTheEndOf Cardinality: [1]
NodeStartsLink	Direction: Bi-Directional, Type: Composite OriginPrimaryKey: Node.NodeID OriginForeignKey: Street.fromNode	
	Source	*Target*
	Name: Node Role: HasStartNode Cardinality: [1]	Name: Streets Role: IsTheStartOf Cardinality: [1]
ArcToPath	Direction: Bi-Directional, Type: Simple OriginPrimaryKey: Arc.ArcID OriginForeignKey: Path.ArcID	
	Source	*Target*
	Name: Arc Role: IsComposedBy Cardinality: [1..*]	Name: Path Role: BelongTo Cardinality: [1]

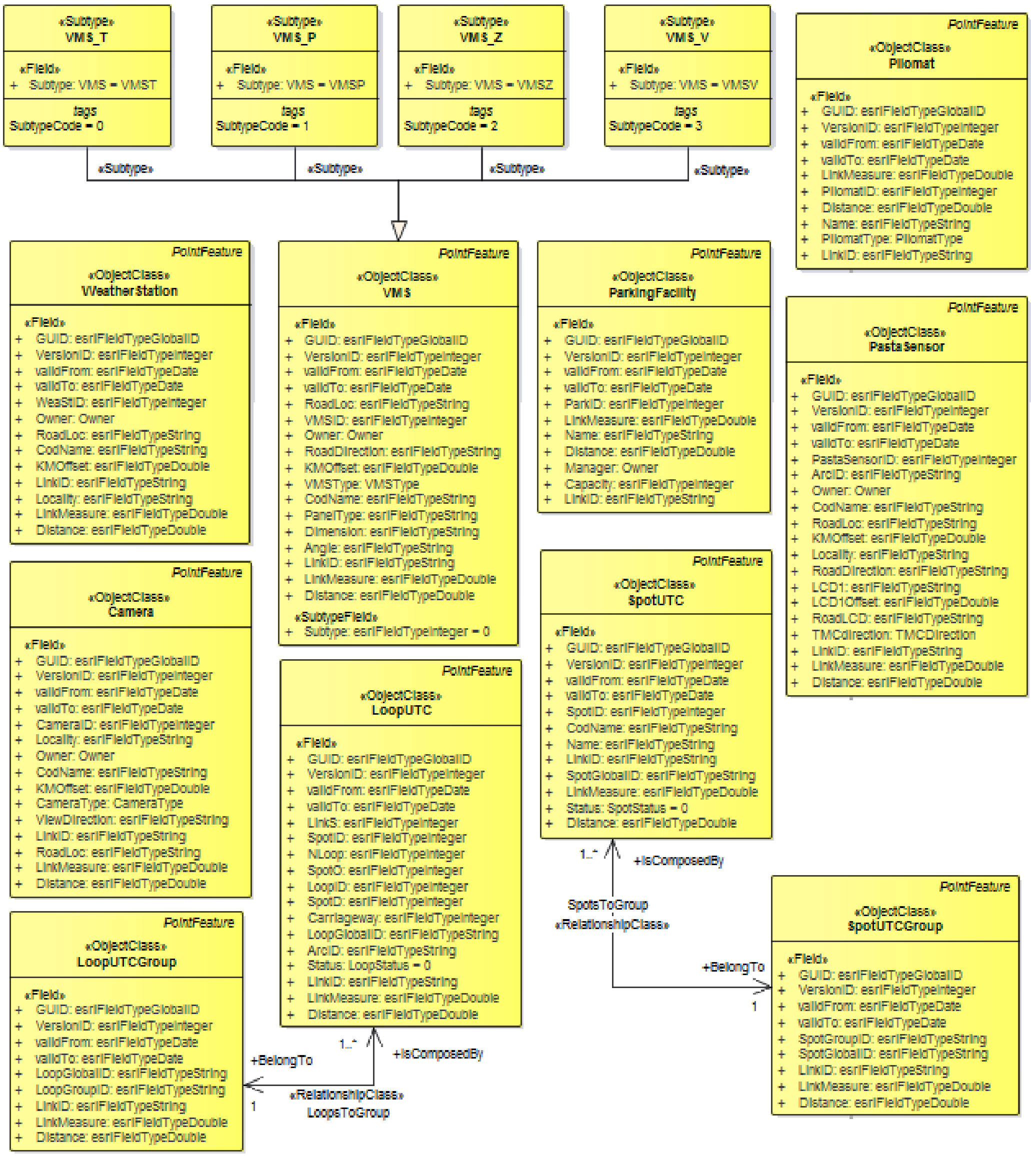


Figure 75: Diagram of the ancillary traffic elements, and relevant relationships.

Table 20: Tables and fields description of ancillary traffic elements.

***Cameras* [Cameras for traffic monitoring] - Table**

Field Name	**Description**
CameraID	Identifier of the camera.
CameraType	Type of camera based on purpose [domain].
ViewDirection	Direction of flow viewed by the camera.

***LoopUTC* [Single sensor loops of UTC system] - Table**

Field Name	**Description**
LoopID	Internal identifier of the loop detector, for internal purposes of 5T Agency.
SpotID	SPOT station ID to which the sensor is connected.
SpotO	Identifier of the origin SPOT station.
SpotD	Identifier of the destination SPOT station.
Carriageway	Identifier of the carriageway monitored.
LinkS	Identifier of the lane or group of lanes, characterized by the same turn behaviour.
Nloop	Identifier (serial number) of the loops in a group of lanes (LINK).
LoopGlobalID	Concatenation of SPOT_O, SPOT_D, Carriageway and LINK fields in order to have a unique primary key to identifies loops, and allowing connection with bin measures.
Status	Status of the loop [domain].

***LoopUTCGroup* [Aggregation of loops at Link level for measures visualization purpose] - Table**

Field Name	**Description**
LoopGroupID	Identifier of the group of loops.
LoopGlobalID	Concatenation of SPOT_O, SPOT_D, Carriageway and LINK fields in order to have a unique primary key to identifies loops, and allowing connection with bin measures.

***ParkingFacility* [Position of parking areas for which 5T gives information] - Table**

Field Name	**Description**
ParkID	Identifier of the parking area.
Name	Name of the parking area.
Capacity	Capacity of the parking area.

***PastaSensor* [Traffic detector of group PASTA] - Table**

Field Name	**Description**
PastaSensorID	Identifier of the traffic detector.
RoadDirection	Direction of the travel detected by the sensor (verbose).

***Pilomat* [Pilomat (bollard) position] - Table**

Field Name	**Description**
PilomatID	Identifier of the pilomat.
Name	Name of the pilomat.
PilomatType	Group of pilomat (Murazzi or Centro) [Domain].

***SpotUTC* [Spot stations of UTC system] - Table**

Field Name	**Description**
SpotID	Identifier of the SPOT station.
Name	Extended name of the station (names of the streets on which the device is located).
CodName	Acronym of Name.

SpotGlobalID	Concatenation of SPOT_ID and COD_NAM, as multiple SPOT_ID may interest the same intersection.
Status	Describe the status of the SPOT station, for maintenance purposes [Domain].

***SpotUTCGroup* [Aggregation of SPOT at intersection level (SPOT_ID) for measures visualization purpose] - Table**

Field Name	**Description**
SpotGroupID	Identifier of the group of spots.
SpotGlobalID	Concatenation of SPOT_ID and COD_NAM, as multiple SPOT_ID may interest the same intersection.

***VMS* [Variable Message Signs] - Table**

Field Name	**Description**
VMSID	Identifier of the VMS.
VMSType	Identify the type of the VMS (solar, tripod, flag...) [Domain].
Subtype	Subtype identifying the type of VMS (Traffic, Parking, Restricted Access Area, Velox) [Domain].
RoadDirection	Direction of the road on which the panel is view.
PanelType	Characteristic size of panel.
Dimension	Height and width of the panel.
Angle	Orientation angle for the panel (valid only for VMS-P type).

***WeatherStation* [Position of weather stations] – Table**

Field Name	**Description**
WeaStID	Identifier of the weather station.

Table 21: Domains description for the ancillary traffic elements.

Domain Name	**Description**
CameraType	Describe the purpose of the camera.
LoopStatus	Describe the status of the loop devices, for maintenance purposes.
Owner	Owner or manager code list, for several devices of 5T.
PilomatType	Description of group of pilomat.
SpotStatus	Describe the status of the SPOT stations, for maintenance purposes.
TMCDirection	Describe the direction in TMC referencing (positive or negative).
VMS	Identify the VMS purpose: traffic, parking, velox and CTZ information [used for subtyping].
VMSType	Description of VMS characteristics (flag, tripod...).

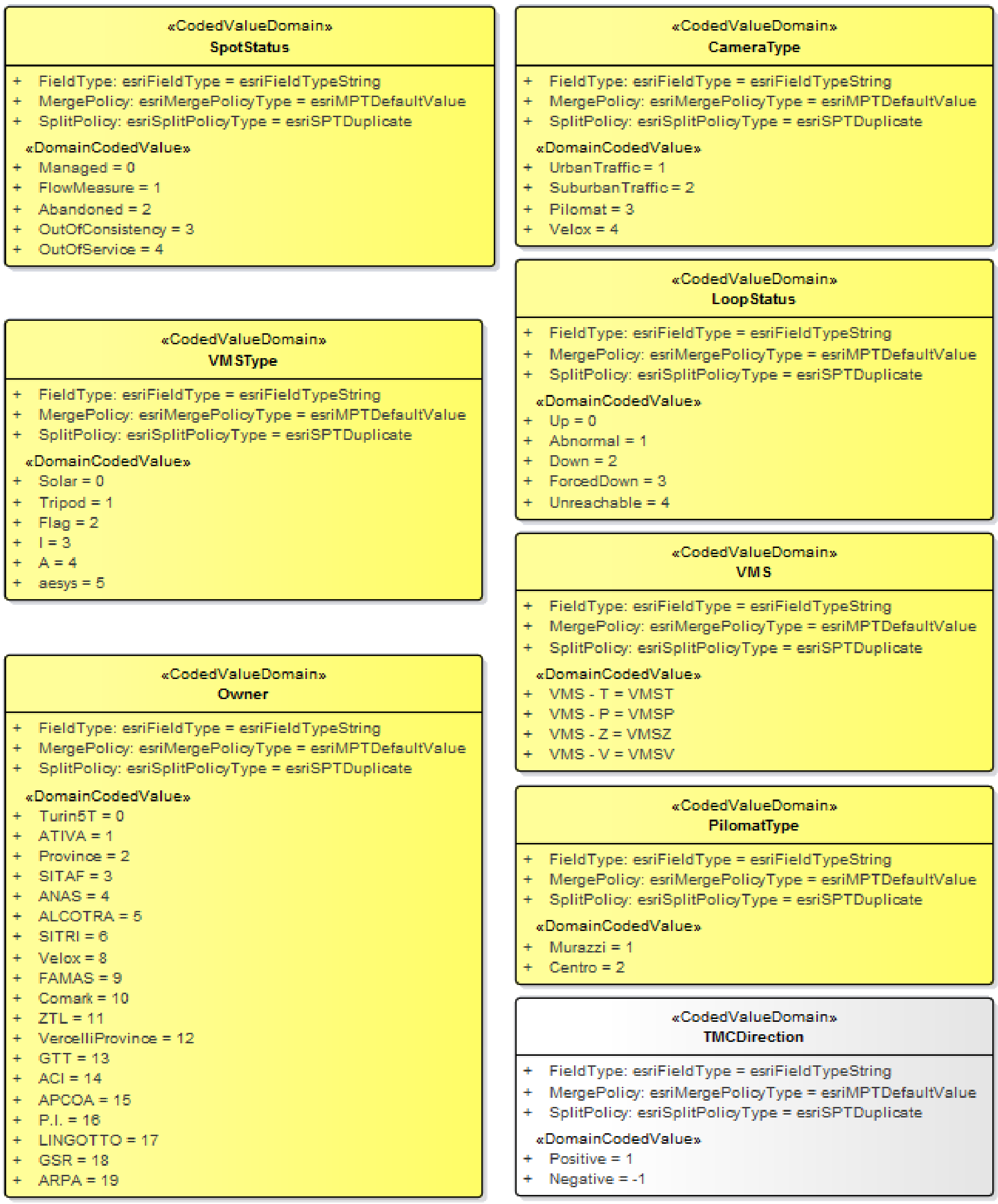


Figure 76: Domains for the ancillary traffic elements.

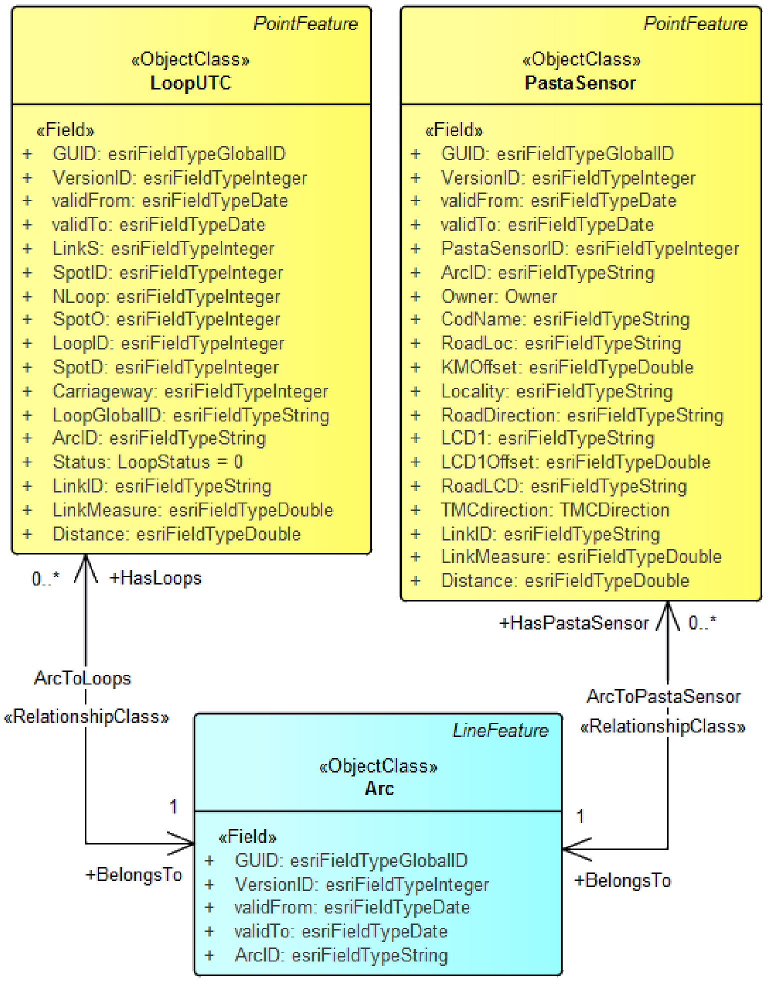


Figure 77: Diagram of the ancillary traffic elements, and relevant relationships with road network element (Arc).

Table 22: Relationships for ancillary traffic elements.

Relationship Classes

Name	Characteristics	
ArcToLoops	Direction: Bi-Directional, Type: Simple OriginPrimaryKey: Arc.ArcID OriginForeignKey: LoopUTC.ArcID	
	Source	Target
	Name: Arc Role: BelongsTo Cardinality: [1]	Name: LoopUTC Role: HasLoops Cardinality: [0…*]
ArcToPastaSensor	Direction: Bi-Directional, Type: Simple OriginPrimaryKey: Arc.ArcID OriginForeignKey: PastaSensor.ArcID	

	Source	Target
	Name: Arc Role: BelongsTo Cardinality: [1]	Name: PastaSensor Role: HasPastaSensor Cardinality: [0…*]
LoopsToGroup	Direction: Bi-Directional, Type: Simple OriginPrimaryKey: LoopUTC.LoopGlobalID OriginForeignKey: LoopUTCGroup.LoopGlobalID	
	Source	Target
	Name: LoopUTC Role: IsComposedBy Cardinality: [1..*]	Name: LoopUTCGroup Role: BelongTo Cardinality: [1]
SpotsToGroup	Direction: Bi-Directional, Type: Simple OriginPrimaryKey: SpotUTC.SpotGlobalID OriginForeignKey: SpotUTCGroup.SpotGlobalID	
	Source	Target
	Name: SpotUTC Role: IsComposedBy Cardinality: [1..*]	Name: SpotUTCGroup Role: BelongTo Cardinality: [1]

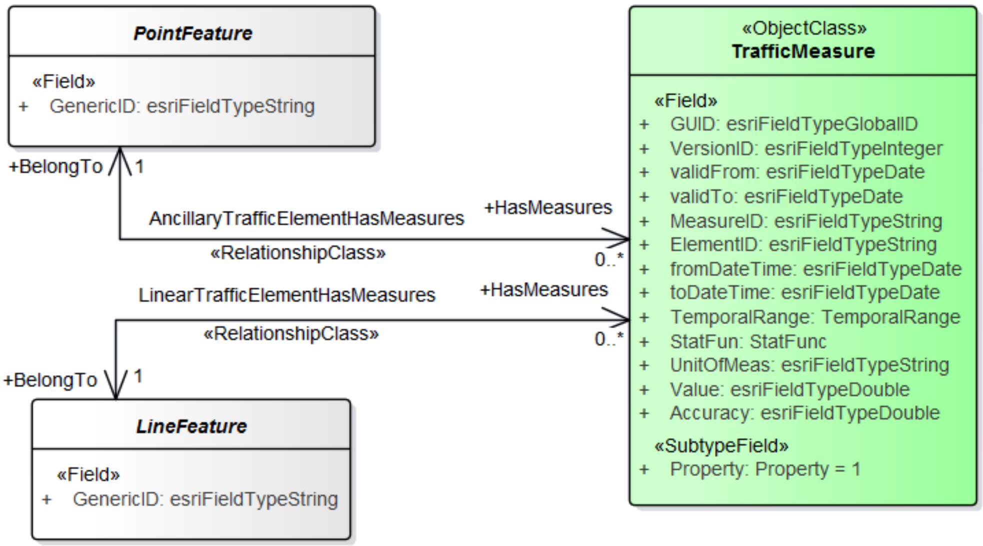


Figure 78: Diagram of the relationships involving TrafficMeasure table.

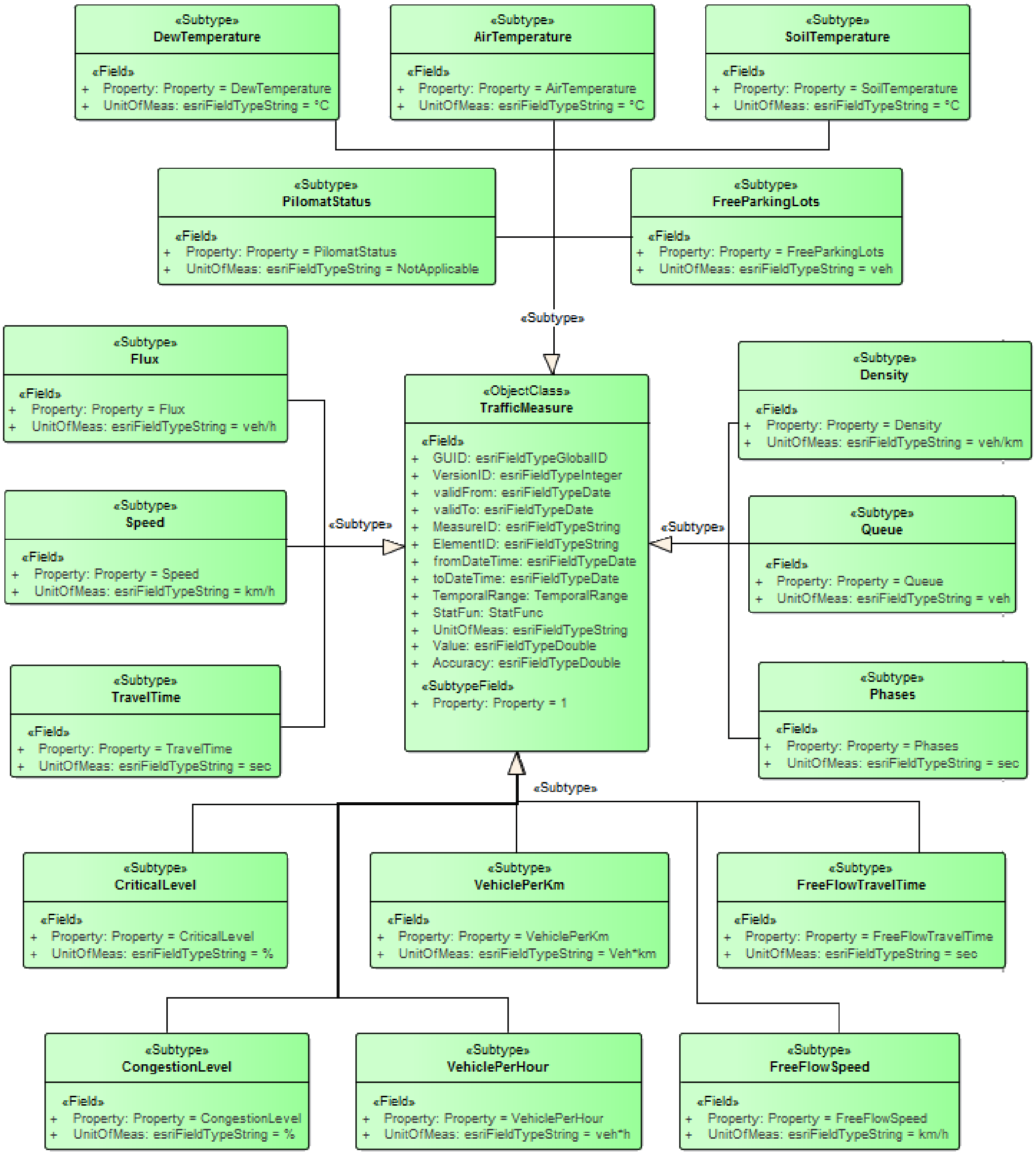


Figure 79: Diagram of the TrafficMeasure table and associated subtypes.

Table 23: Tables and fields description of measures tables.

***TrafficEvent* [Describe a traffic event] - Table**

Field Name	Description
TrafficEventID	Identifier of the traffic event.
TrafficEvent	Subtype identifying the type of traffic event [Domain].
EventType	Describe the category of traffic event (DATEXI) [Domain].
Description	Description of traffic event.

***TrafficMeasure* [Describe traffic measure] - Table**

Field Name	Description
MeasureID	Identifier of the measure.
ElementID	Spatial element on which the measure belong.
Property	Type of observed phenomenon, used for subtyping [Domain].
StatFun	Statistical function for aggregation.
TemporalRange	Aggregation time period.
UnitOfMeas	Unit of measure.
Value	Value of the observation.
Accuracy	Accuracy of the observation (rate).

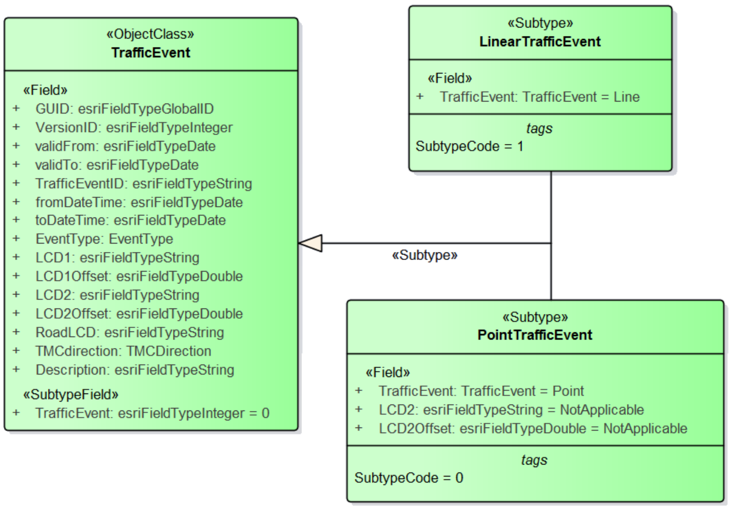


Figure 80: Diagram of the TrafficEvent table and associated subtypes.

Table 24: Domains description for measures tables.

Domain Name	Description
EventType	Describe the category of traffic event (DATEXI).
Property	Identify the observed property [used for subtyping].
StatFunc	Describe the statistical function used to aggregate traffic measure.
TemporalRange	Describe the possible temporal aggregation of a traffic measure.
TrafficEvent	Identify if a traffic event is linear or point type [used for subtypying].

Table 25: Relationships for TrafficMeasure table.

Relationship Classes

Name	**Characteristics**	
LinearTrafficElementHasMeasures	Direction: Bi-Directional, Type: Simple OriginPrimaryKey: LineFeature.GenericID OriginForeignKey: TrafficMeasure.ElementID	
	Source	Target
	Name: LineFeature Role: BelongTo Cardinality: [1]	Name: TrafficMeasure Role: HasMeasures Cardinality: [0..*]
AncillaryTrafficElementHasMeasures	Direction: Bi-Directional, Type: Simple OriginPrimaryKey: PointFeature.GenericID OriginForeignKey: TrafficMeasure.ElementID	
	Source	Target
	Name: PointFeature Role: BelongTo Cardinality: [1]	Name: TrafficMeasure Role: HasMeasures Cardinality: [0..*]

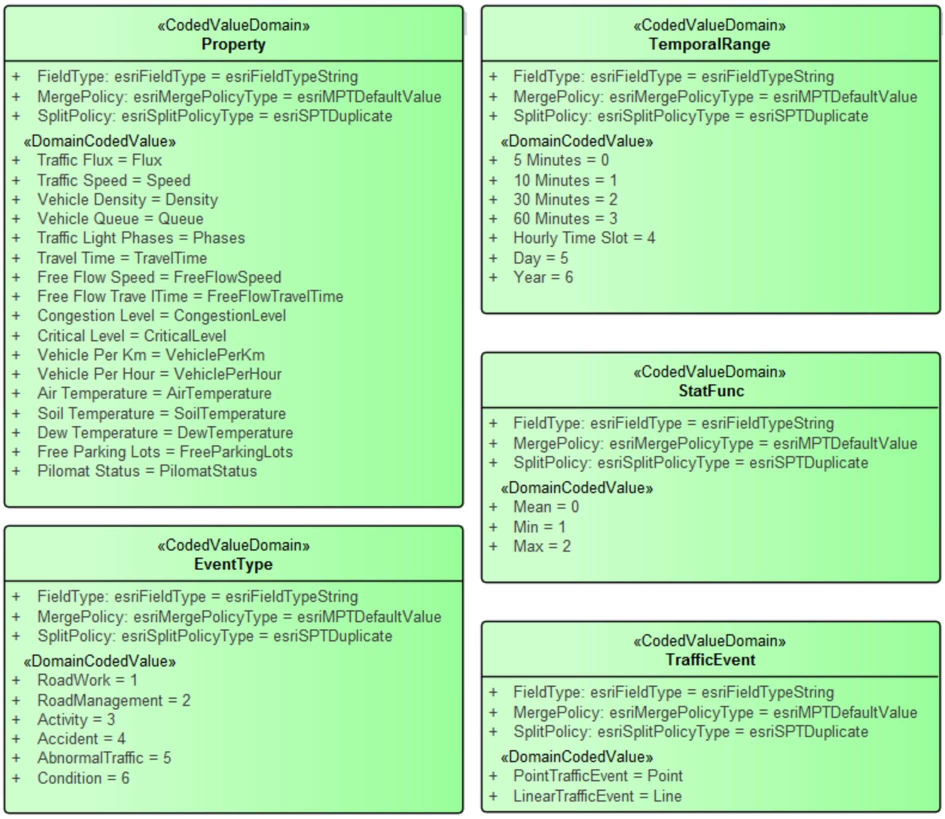


Figure 81: Diagram of the domains related to measures and events.

6.3 Data extraction transformation and load

In order to correctly load data into the data model previously described, a broad activity of data transformation has been applied. Firstly, data relative to road network have been modified in order to find the matching between the different sources available in 5T, as stated in the requirements section. Secondly, attributes have been mapped into new fields of the data model. Finally, a linear referencing activity has been performed in order to reduce the processing load when the geodatabase is used. Only after these steps, data have been loaded into the new geodatabase.

As stated in the requirement section, one of the goal of this activity was to integrate the different sources of linear network data used in 5T. Indeed, for the city of Turin, Arc and Path are available as source for the supply model and for displaying estimated measures. This set of data is coherent in itself, but do not have any correspondence with the source of data available over the Region. The latter, is derived from NAVSTREETS Streets Data of 2014, conveniently modified in order to be ingested by the OPTIMA software that manages the SVR.

As the NAVSTREETS Streets Data of 2016 has been purchased by the Agency, in the perspective of an update of the network data for the OPTIMA component, it has been chosen to use this updated network source as reference on which conflate data sources already in use.

In order to perform this activity, the “Conflation Toolset” of ArcGIS has been used. In particular, the workflow proposed in the Technical Workshop at Esri UC (2014) “Conflation – Getting Started and Understanding Workflows” has been applied.

Conflation processes can be highly automated, but they require in all cases a manual quality assessment activity. In addition, sometimes is not possible to find a “right” solution, and choices have to be made in order to find an optimal solution. As regard the network dataset of OPTIMA (SVRLinks), the process has been applied only over a subset, as the number of feature to check was too high. Arcs, indeed have been totally conflated, but some missing matchings persist, in particular over complex intersections. Further activities of data cleaning can be eventually performed by the agency, but are now out of the scope of this research.

The first step has involved the NAVSTREETS Streets Data of 2016. Indeed, the direction of travel is based on the concept of reference and non-reference node, distinguished by their relative latitude and longitude position. This method is not coherent with the one used by the SVRLinks, and even if it is possible to read it correctly in ArcGIS setting an expression, it has been decided to apply a different logic, also in order to simplify the conflation process. In addition, the “Conflation Toolset” of ArcGIS does not allow to use external primary key, like the “link_id” (which is permanent between different versions of the NAVSTREETS), but performs the comparison on the base of the OBJECTID field. However, to overcome this issue the conflation workflow suggests to extract and classify vertices, depending on the number of connected links, and compare them. Therefore, it became important to make comparable start and end nodes between SVRLinks and the NAVSTREETS Streets Data of 2016.

In particular, the direction of travel in ArcGIS is defined through the support of start point and end-point of the digitized polyline. In geometry definition indeed, a polyline is defined as a sequence of unique point (not repeated). The start point is the first one and the last point is the end-point. When a polyline is manually digitized, the end-point is the last of the task. These points are recognized by ArcGIS engine, as can be seen in Figure 82.

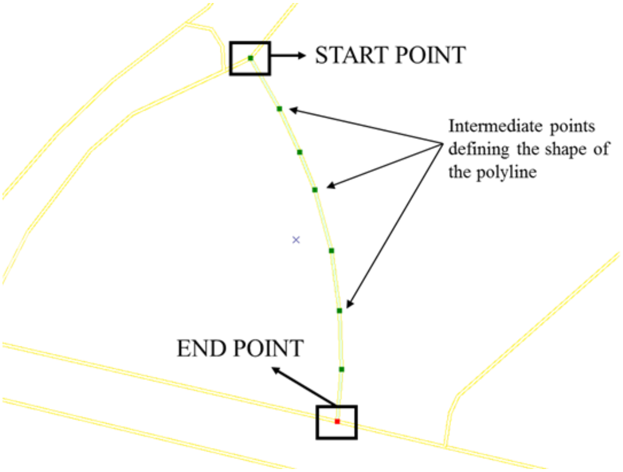


Figure 82: A polyline in an ArcGIS editing session.

The SVRLink dataset also is based on this approach, and the values of “tail” and “head” fields effectively correspond to the start and end node of the digitized polyline.

Therefore, an ArcPy script has been developed in order to check and eventually switch the digitized direction of the NAVSTREETS Streets Data. Below the general steps of the script:

1. For each link, coordinates of the end point has been extracted;
2. For the selected link coordinates of the non-reference node and reference node has been extracted and compared with the one of the end-point.
3. Then, if the non-reference point has the same coordinates of the end point, the direction of the link is evaluated:
    a. If the direction is “F” or “B” the link is left as it is;
    b. If the direction is “T” a flip (switch start/end-node of the polyline) operation is performed on the link.
4. Conversely, if the reference node point have the same coordinates of the end point, the direction of the link is evaluated:
    a. If direction is “T” or “B” the link is left as it is;
    b. If the direction is “F” a flip operation is performed on the link.

After this processing, was easier to compare SVRLink and NAVSTREETS.

Second step has involved a data cleaning of the SVRLink, removing all features where the field “ref” was equal to -1, in order to obtain only navigable edges.

Then the conflation workflow procedure has been applied over a subset of the two data:

1. Extraction of end-point vertices and classification of them by their role in connectivity (looking at the number of connected link) [StartNode, EndNode, RegularVertex, Dangle, PseudoNode, T-Node, XNode, Other].
2. Then a “Detect Feature Change” is performed, obtaining a matching table of the two dataset and a classification of the changes [NotChanged, Attributes, Spatial, Spatial and Attributes, New, Deleted]. After this operation over the 65 % of the data was not changed (but the 28% of these was flagged as potentially incorrect).
3. Using the “QualityAssessment Toolbar” (provided in the workshop) links have been revised: the toolbar allows to add a “revision_flag”, which can be used in next steps. It is important to evaluate possible incorrect matching between the two dataset.
4. Once the revision is ended, a join with the original input and the new checked class is performed. Two join can be implemented: a join with unmatched features can be used as a new input to run again the “Detect Feature Change” tool, changing some parameters in order to repeat the Quality Assessment phase and increase the quality of the results. A join

with correctly matched features (in case the QA phase is evaluated as sufficient) create the new input to be used for the next steps.

5. The “GenerateRubberLink” is performed on the sub set of data correctly matched. It create a set of links (rubber links) that define the direction on which the conflation must be done.
6. Again using the “QualityAssessment Toolbar” the results of “GenerateRubberLink” must be checked (and edited). Now vertices previously extracted can helps in the check. It is important to verify the presence of intersecting rubber links, rubber links that have different vertex types (between source and target), rubber links that are particularly long (maybe a matching with a wrong feature) and missing rubber links.
7. Finally, the “Rubbersheeting” tool is run in order to match the two data. Eventually, after this operation the “GenerateRubberLink” and quality assessment procedure can be repeated in an iterative way, reducing errors from time to time.

This procedure was applied also for the 5T Arc dataset, previously pre-processed. In particular, as the “Conflation ToolSet” tries to identify 1 to 1 matching, the Arcs data (which have a longer extent respect to the NAVSTREETS Streets data) have been split at the intersections with buffer area of 15 m, generated around each single feature of the NAVSTREETS Streets Data. This operation has led to increase the number of features from 5’458 to 48’673.

As regard the **attribute mapping activity**, several operations have been conducted in order to convert legacy attributes in the new ones. In particular, some fields have changed in data type [string to integer or the opposite], and in other cases new code lists have been prepared. Map matching between old and new attributes is documented in the attribute description contained in tables in section 6.2.4.

One of the main problem in converting data structured in a DBMS to an ArcGIS Geodatabase schema was related to candidate primary key for joining data: indeed several data in 5T are based on composite candidate primary keys, not supported by ArcGIS environment. New composited identifier fields have been build (preserving the in most cases the original disaggregated information for further uses) through the concatenation of fields.

In addition, looking at the UTC system, depending on the type of measure, different keys can be involved in the relationships with bin data. Most of the measure detected are at the link aggregation level (see section 5.1), but the turn rate, for instance, needs an additional level of disaggregation.

In this case, the choice has been to maintain the maximum level of disaggregation in the LoopUTC table and create a set of detectors for the aggregated level in the LoopUTCGroup table. In order to better visualise measures, this level of data aggregation has been achieved not only at field level (concatenation of fields) but also at spatial level. The original disaggregated spatial level has been dissolved over the new concatenated field, creating a multipoint class, and then using the “Feature To Point” ArcGIS tool a median point of the multipoint feature has been generated. This operation has been performed both for loop detectors and spot stations.

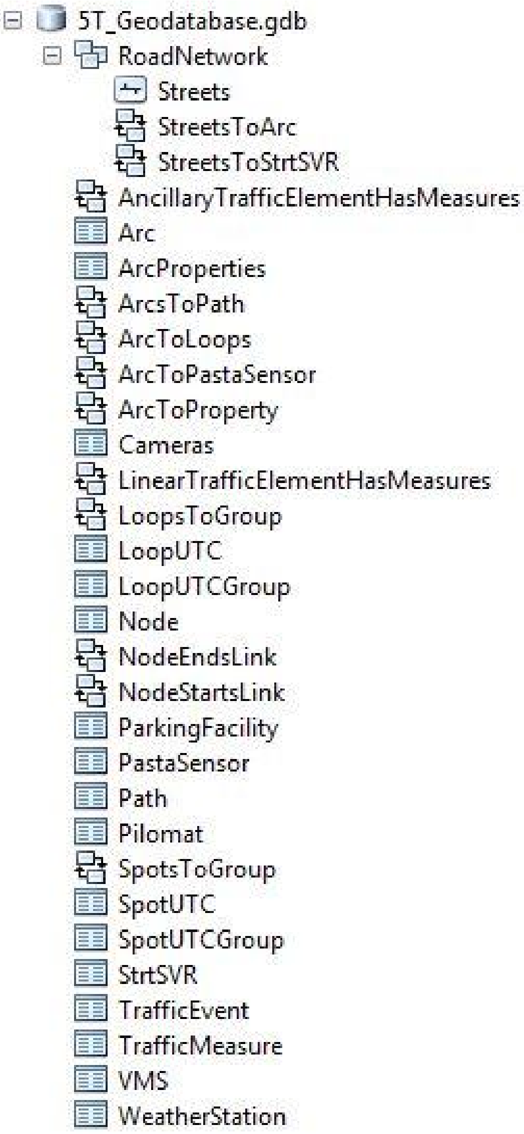
As 5T has a its disposal only one machine on which ArcGIS is installed, in order to not overload the system it has been chosen to use only the NAVSTREETS as geometry feature and maintain all the other information as tables. The features visualisation will be applied on the fly.

Point data in particular, are actually stored in 5T database with Lat/Lon coordinates fields. ArcGIS allows to visualise these tables, taking advantage of the coordinates fields. However, it has been chosen to use the **linear referencing** toolbox, in order to build a more consistent way of data visualisation: indeed, in this way not only data can be visualised on the fly, but also in addition they have a new reference to the linear object to which they belong. As example, it has to be considered that UTC systems is connected through attribute to a reference Arc: using the linear referencing toolbox, it has been possible to reference it to a single NAVSTREETS Streets Data edge.

Tables are firstly visualised as X,Y data and then exported as feature classes. A route has been build over the NAVSTREETS Street Data, using the “LinkID” field. Creating routes essentially provides an M (measured) value on the geometry attribute. Then, using the “Locate Features Along Route”, a new table is created, preserving original fields and adding the relevant one needed to locate the point. This procedure can lead to some errors in link assignment: a check of the features with a value relative high of the “LinkDistance” has been performed to assure a better quality.

Once static data are cleaned and relative attributes are set to fit the new data model, a custom script which use the “Load data” function of ArcGIS can be run.

In order to run the script, the XML of the data model must have been already exported and available in a selected working folder. Then the script checks for the presence of the XML and create the geodatabase schema using it. Once tables and feature are created, the script search for the list of data to be uploaded: if data are found it run the "Load data" function in association with a set of predefined strings, which allow matching features and attributes of the source with the one of the Target. In Figure 83, a view of the final geodatabase from ArcCatalog, is shown.



- ⊟ 5T_Geodatabase.gdb
  - ⊟ RoadNetwork
    - Streets
    - StreetsToArc
    - StreetsToStrtSVR
  - AncillaryTrafficElementHasMeasures
  - Arc
  - ArcProperties
  - ArcsToPath
  - ArcToLoops
  - ArcToPastaSensor
  - ArcToProperty
  - Cameras
  - LinearTrafficElementHasMeasures
  - LoopsToGroup
  - LoopUTC
  - LoopUTCGroup
  - Node
  - NodeEndsLink
  - NodeStartsLink
  - ParkingFacility
  - PastaSensor
  - Path
  - Pilomat
  - SpotsToGroup
  - SpotUTC
  - SpotUTCGroup
  - StrtSVR
  - TrafficEvent
  - TrafficMeasure
  - VMS
  - WeatherStation

Figure 83: A view from ArcCatalog of the geodatabase.

Last activity performed was the creation of **custom scripts to extract and group measures and traffic events**. The scripts for measures extraction perform basic queries (essentially a GROUP BY), giving as input the measure type, the start and end date time of interest, and optionally a predefined temporal range over which aggregate data, with a statistical function associated. This approach has been considered more efficient than storing predefined set of data that may overload the geodatabase. In this way, the computing power of the server is used for group by operation and fields transformation, whereas the machine has only to create the customised object class and the relationship class and load data into. Several scripts has been created in order to fulfil a set of possibility. A more object-oriented approach has to be developed in order to allow major customisation in data extraction.

Chapter 7

Applications

In this Chapter some of the possible applications enabled by the new data model are described. As already stated, the geodatabase is designed mainly for visualisation purpose.

7.1 Network fixed element visualisation

As a first result, the geodatabase enable 5T user to visualise the set of sensors and other traffic management element with the possibility to query and display only the needed subset with a customised symbology. This was previously quite complex in the past as the knowledge of data source location (separated in several servers) and the skills to extract data were not so accessible.

In order to visualise point element position from the available tables, data need to be loaded in ArcMap and then using the "Display Route Events" interface data are visualised on the fly. Data can be exported as feature class or used as it with a customised symbolisation. In Figure 84, is possible to see the position of the main point elements mapped in the geodatabase.

Visualising Arcs and Path instead needs a further step: starting from the Street feature class, the "Join" interface must be used, selecting the option "Join data based on a pre-defined relationship class". In this way, fields of the related table are reported in the Street feature class and can be used for thematic visualisation. An example of Arc and Path visualisation is given in Figure 85.

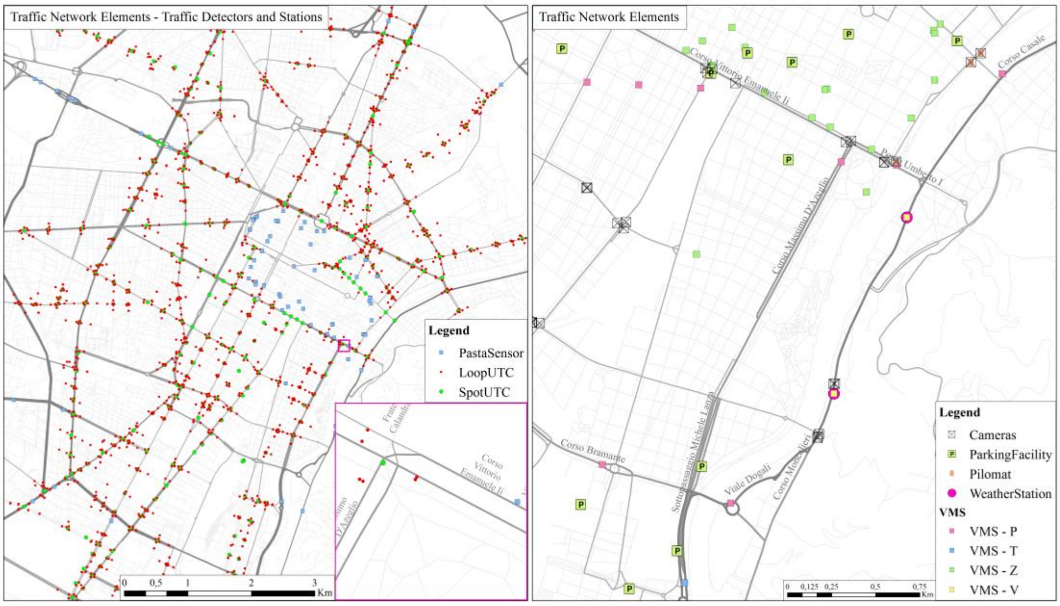


Figure 84: Position of UTC system elements and other ancillary traffic elements in the area of Turin.

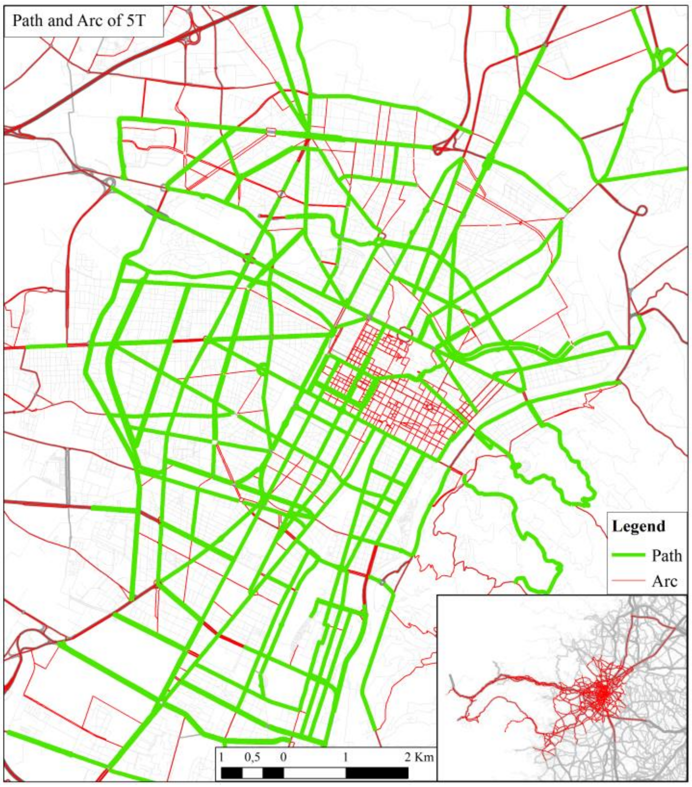


Figure 85: Arc and Path visualisation in the area of Turin.

7.2 Measures visualisation

The geodatabase schema defined allows the extraction of single measure on demand. In order to give an idea of the potential uses of this structure some extraction are shown in this section.

In particular, the scripts developed for data extraction create two objects: an object table of type “TrafficMeasure_«Property_subtype»” and a relationship class named “«SpatialReferenceElement»HasMeasure”.

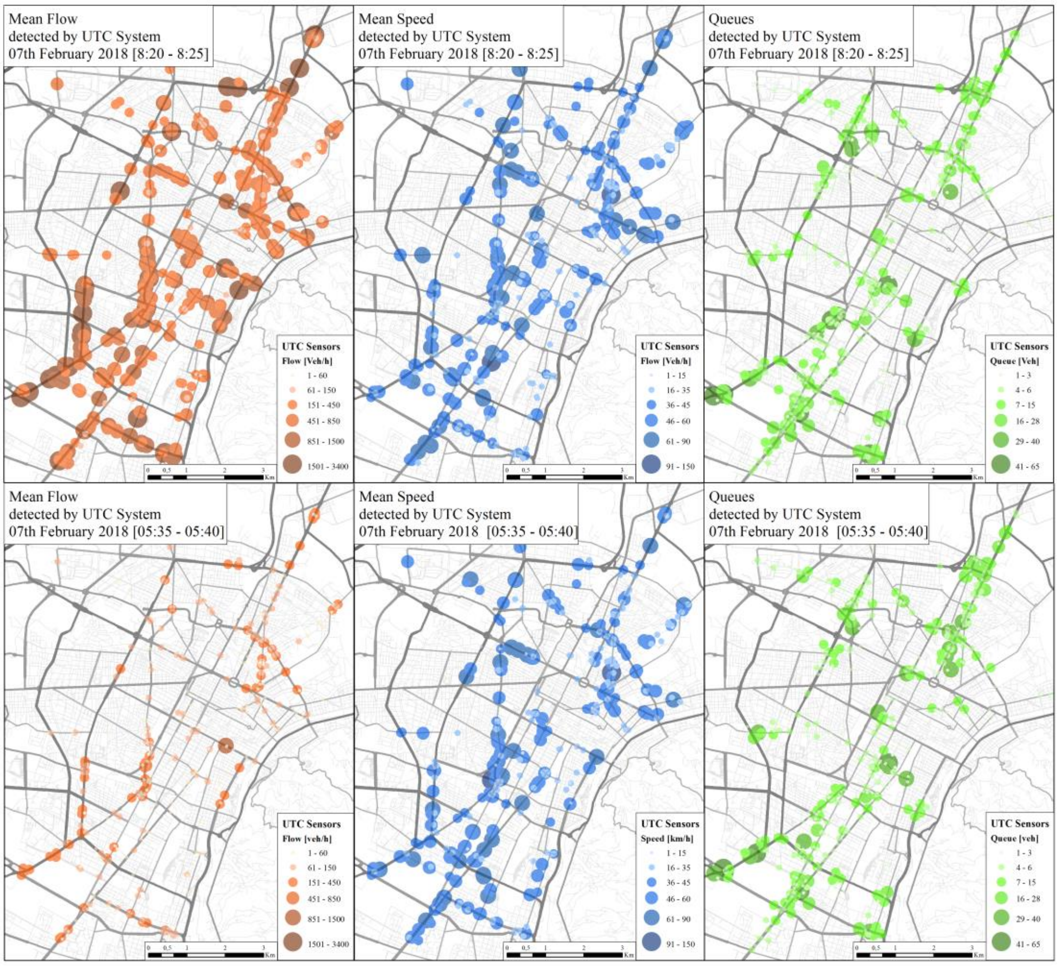


Figure 86: Traffic measures detected by UTC loops (flow, speed, queues), comparison of a single bin detection.

As for the Arc and Path representation shown in the previous chapter, once the feature table is load in ArcMap and data are visualised as Route Event, a join with the option “Join data based on a pre-defined relationship class” have to be applied, linking data with measures. In Figure 86, a comparison of a set of measures detected by the UTC loops is given: first row represents the situation at a peak hour (h. 8:30), the second row represents the situation at a non-peak hour (h. 5:35). In this case, the measure is given as it is from sensors: no aggregation has been applied in data extraction procedure.

In Figure 87, an aggregation over the whole day has been applied. The figure compare the situation in a typical working day (Wednesday) and in a non-working day (Sunday).

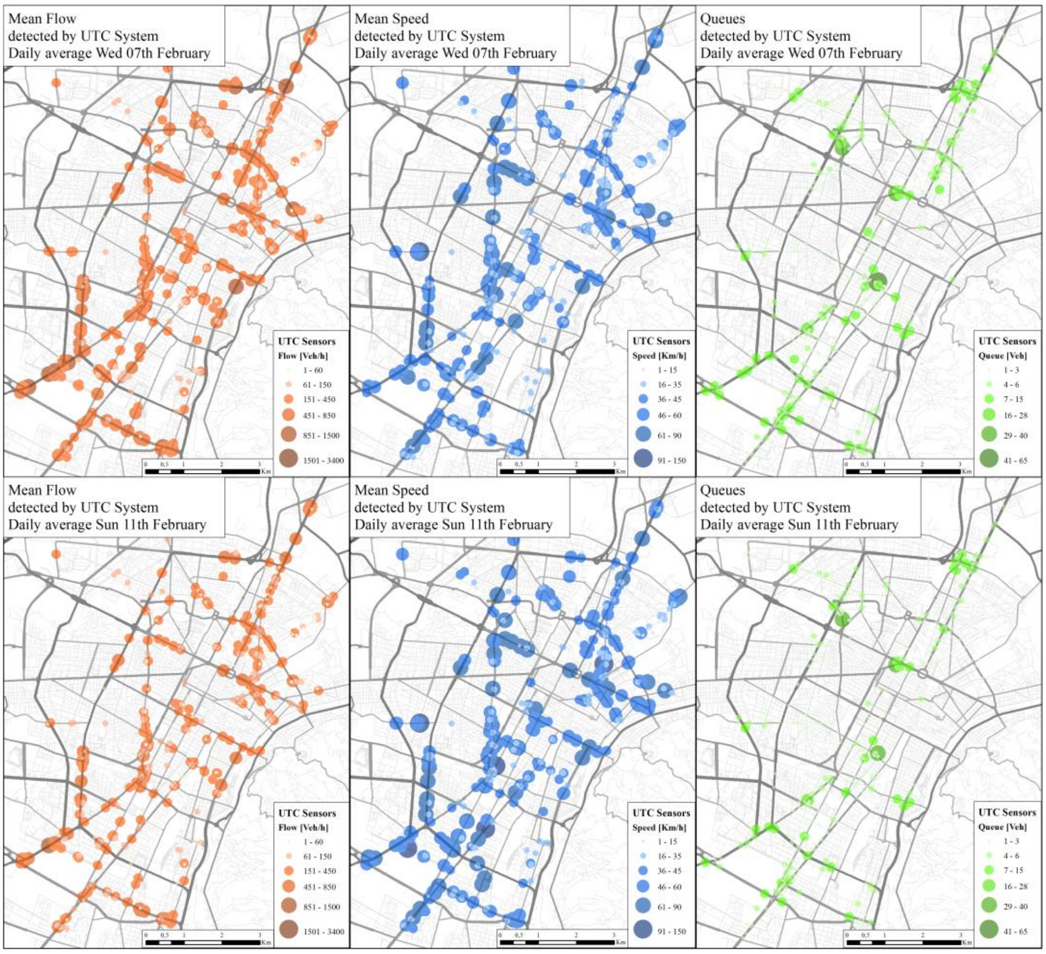


Figure 87: Traffic measures detected by UTC loops (flow, speed, queues), comparison of a daily averages.

In a similar way, also measures relatives to Arc and Path can be visualised: in Figure 88, is shown the mean travel time measure (normalised on the free flow travel time measure), estimated by the SVM, and reported over Path features. As

Arc (and consequently Path) can be referred to the same bidirectional link, the output visualisation shows Path elements overlay, with different measures referred to it. More appropriate rules of representation can be applied in order to not allow the overlay between measures, but in this case is quite effective to show the differences in traffic flow in opposite flow direction.

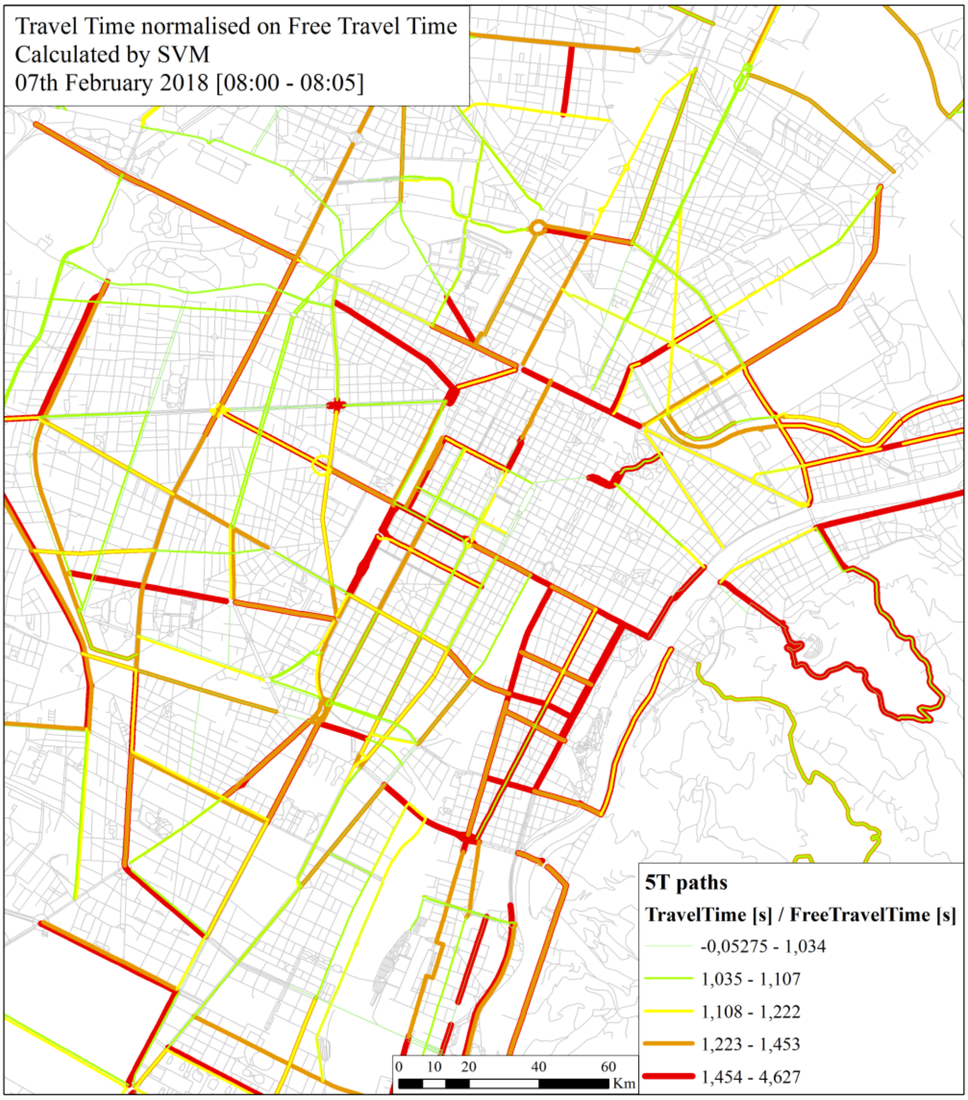


Figure 88: Travel time, normalised on free flow travel time, estimated by SVM.

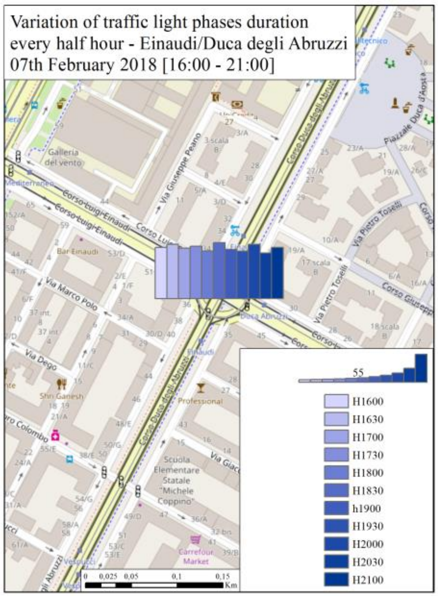


Figure 89: Variation of traffic light phases duration in a time window (h. 16:00 – 21:00).

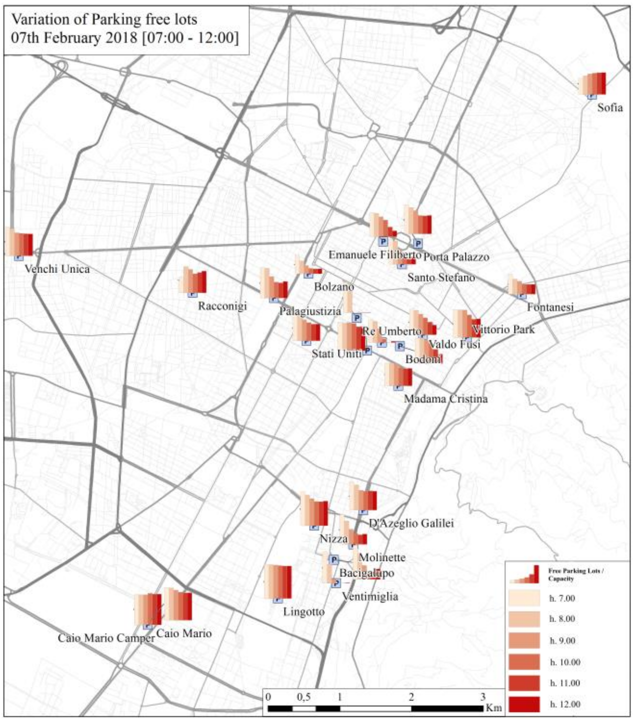


Figure 90: Variation of the availability of free lots in parking area, normalised on the capacity of the parking area, in a time window (h. 7:00 – 12:00)

Also short time series can be represented though graphs. An example is given in Figure 89 and in Figure 90. In this case, in order to obtain the series, multiple extraction of the same measure has been done in sequential time ranges. Tables created are then joined together in order to obtain a single table to link with the relative spatial object. Then an appropriate symbolisation has been set.

Finally, measures can also be integrated in a Network Dataset and used as real impedances. In Figure 91, the network dataset has been build on an export of Arc features, over which a table of measures has been previously joined. The figure shows the differences in routing choices made by the ArcGIS routing engine, setting different arc measures as impedance.

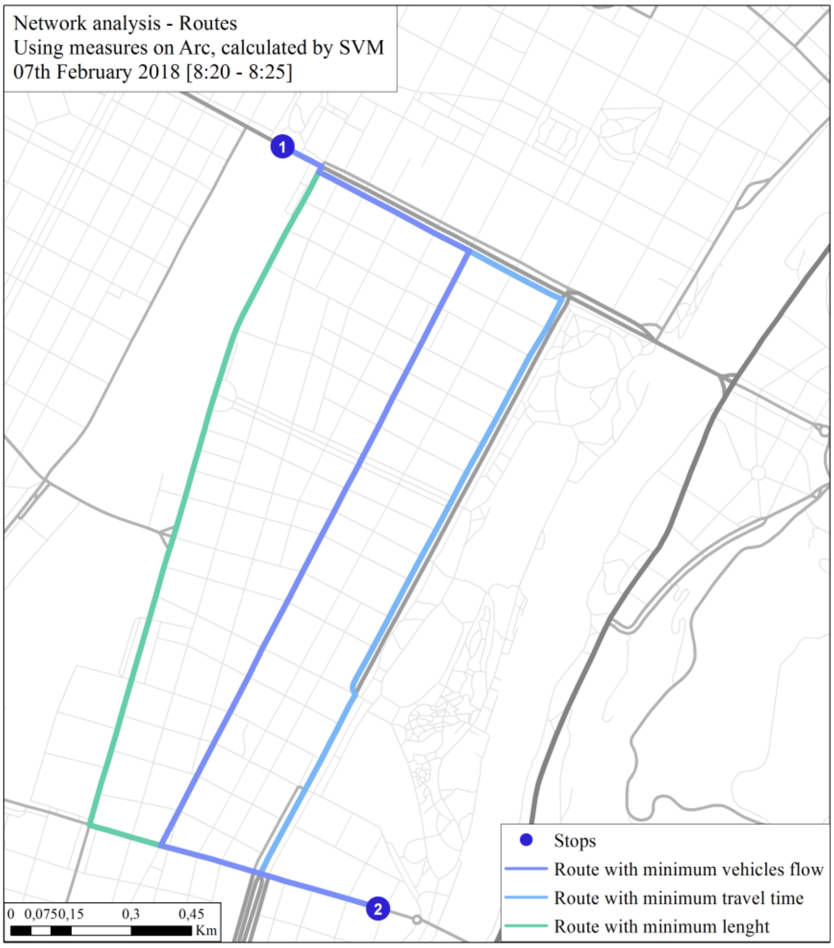


Figure 91: Differences in Route Analysis outputs using different measures of Arc as impedances.

7.3 A traffic event visualization application

One of the applications that has allowed also to evaluate the data and to test extraction and linear referencing scripts has regarded the evaluation of a possible framework that integrates the traffic events information flow provided by Traffic Information Centers (TICs) with the Copernicus Emergency Management Service. This activity has led to the submission of a paper (Arco et al, 2017. “An operational framework to integrate traffic message channel (TMC) in emergency mapping services (EMS)”), from which this section is partially readapted.

The Copernicus Emergency Management Service - Mapping module (EMS-Mapping) is an on-demand service that provides support to all actors involved in the fields of crisis management, humanitarian aid as well as disaster risk reduction, preparedness and prevention. Copernicus EMS-Mapping covers the entire process from the satellite tasking, image acquisition, processing and analysis of satellite imagery and other geo-spatial raster and vector data sources until the production and delivery of maps and vectors, in standard formats, to the user who requested the service and delivery to the public. The service offers three map types, one pre-event map and two post-event maps, each of which performs specific functions relevant to crisis management. In particular, the two post-event map types, known as *delineation* and *grading* maps, are produced from post-disaster satellite images. In the fastest map production mode, the service provides both post-event maps within 12 h after image data reception and quality acceptance (Ajmar et al., 2017). Between emergent exigencies expressed by users of the service, there is the possibility to have interactive maps that allow real-time situation monitoring (Wania et al., 2016), and in general to reduce the delay in satellite imagery delivery (Voigt et al., 2016).

The availability of traffic events information in an integrated way to emergency response services, such as Copernicus EMS-Mapping, could have high benefits. Traffic information could be used during the production activity, helping in spotting and reporting damages to infrastructures and in reporting roadblocks. Additionally, in the pre-activation phase and in case of a large amount of traffic events (likes roads and bridge closures) located over a specific territory area, an automatic triggered alert can be delivered to EMS – Mapping managers, who can therefore decide to pre-task satellite platforms and consequently reduce imagery delivery times. Furthermore, the position of traffic events can be used as ancillary information to better define the areas of interest that could be most affected.

If during the production activity traffic event usage can provide auxiliary information that can support in damages search and in the delineation of the emergency events, an information flow in the other direction may also be relevant. Map producers can highlight new damages not still acquired by TICs and so directly share the information with them and, through traffic message channels (TMCs), with the public.

Figure 92 shows the proposed general workflow for the integration of the traffic events, usually spread by TICs, within the operating procedures of the Copernicus EMS-Mapping. The European TICs through their DATEX II or I node send information to the Copernicus real-time platform. In order to be able to use these data, firstly the DATEX format is decoded and data are filtered according to the traffic event category (excluding, e.g. accidents, limitations due to social events), and therefore, the TMC location is decoded to allow a precise localization on the portal's base map. Following these pre-processing steps, the traffic events are displayed on the portal and can be used by Copernicus service production sites to improve damage assessment and delineation. The delineation and grading data produced by Copernicus are therefore appropriately filtered (e.g. considering only impact on roads and bridges) and converted into a DATEX format in order to be properly received by the European TICs DATEX node affected by the Copernicus activation.

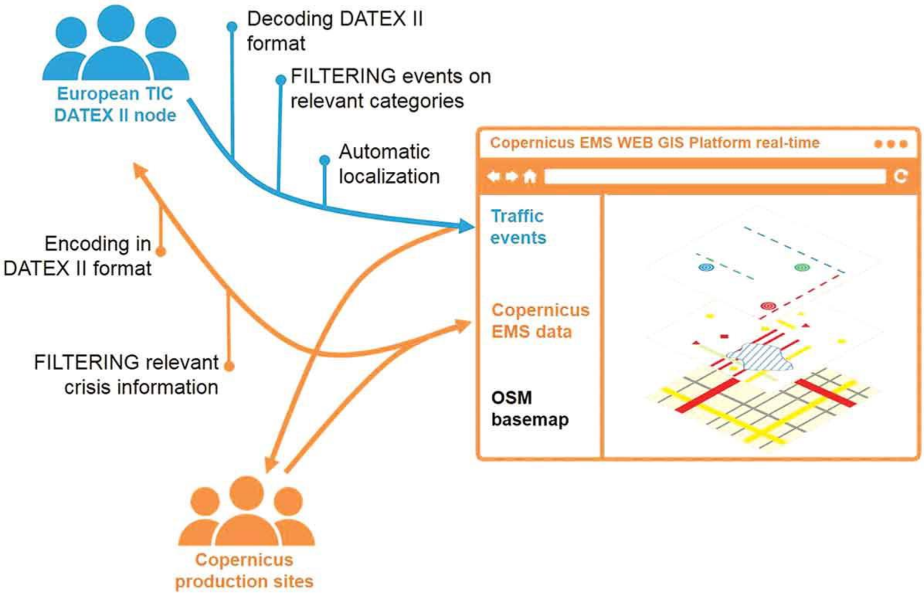


Figure 92: General workflow for integration of real-time traffic events and Copernicus EMS-Mapping.

7.2.1 Methodology

This activity has led to a deepen exploration of available source of traffic information in 5T. As already stated in section 5.3, in 5T Company traffic events are mainly managed by two databases:

- MISTIC database (DATEX I node component)
- TCM_sistema database (SVR component).

In particular, the MISTIC database contains two tables useful for this analysis:

- the STORY_ELE table that contains relevant timestamp and other information related to the event and
- the STORY_LOC table that contains the location reference of the event through the TMC.

The fields version (VER) and situation define the primary key of the table and represent the fields through which the element table is joined with the location table. The event description is given by the combination of the two codes PHR and DOB, derived by DATEX Data Dictionary. PHR has 509 entries and DOB 23. The dictionary derived from the combination of these two codes generates a total of 564 types of traffic events.

The TCM_sistema database indeed contains historical data, aggregated by month. The main table "events_yyyy_Mm", can be related to "events_descriptions_yyyy_Mm", which integrates a verbose description of the event.

In case of particular emergency situation to be managed in 5T, when information about temporary road closures from authorities comes rapidly and has several updates, some events are not managed through internal operating procedures through MISTIC. An ad hoc page on the "MuoversiInPiemonte" portal, the informative portal about traffic condition over the Piedmont Region, has been developed, in order to offer to the citizens a complete and updated information.

The first set of traffic events used for this analysis, comes from the MISTIC database: records occurred in the period interested by the flood event case study have been extracted from "STORY_ELE" and "STORY_LOC" tables. The traffic events from MISTIC needed to be geocoded over the road network, exploiting source points (first and second location codes) and offset information: for this a function has been designed in order to locate a traffic event over the TMC network referenced to the NAVSTREETS Streets Data, using PostgreSQL and its spatial extension PostGIS.

The function iterates over the table of the selected events from MISTIC database and looks for first and second location codes. If only primary location code is present, the event is processed as punctual event, in order to obtain a point geometry attribute for the situation analyzed, otherwise the event is processed as linear event. The function searches for the edge of the NAVSTREETS associated with the TMC road LCD and then locates the primary LCD point. Then it evaluates the offset distance, if present: the two distances (pri_distance, sec_distance) are measured along the selected edge (with linear referencing functions toolset), using always the snapped primary LCD point as start point.

Between the materialized events, a second filtering operation has been performed in order to select only those events that can be related to a flood emergency. Using a filter over the PHR and DOB fields, only the categories MISTIC event categories “Meteorological information”, “Traffic restrictions – Road section closure” and “Obstructions – flooding/landslide” have been considered useful for the analysis.

The second set of traffic events is derived from TMC_sistema database, considering the table with the historical data of November 2016 (events_2016_M11). Data extracted from this table (with a convenient transformation of geometry from SQL Server dialect to PostGIS one) can be directly imported in ArcGIS, the software chosen for the analysis, performing:

- a preventive selection between point events and linear events (line and multiline types);
- a selection on “fdat” field (start time) in order to consider the events that start from the 22 November 2016 to the end of the month and
- a filter over the category type (“etyp”) in order to select only the events that can be related to a flood emergency.

In this table, there are only seven categories: abnormal traffic, roadworks, condition, obstruction, accident, road_management, activity. Only condition and obstruction events have been chosen to perform the analysis.

The third source of traffic events considered is the “MuoversiInPiemonte” web site. Between the 22 November 2016 and the 02 December 2016, there were 276 different versions of the page. The table structure contains the version id, the date time creation and a field that contains the complete HTML page. For each version the field containing HTML page was parsed, creating a new table where each row contains version id of the page, date and time of the version created, area of reference for the event, road interested by the event and descriptions, when available. Then an algorithm has been developed in order to compare which roads in a version of the page are contained or not in the following version, assigning a start and stop time using the date time field of the page version when a new road comes into the web page and when the road is not more present in the page. As only the road name was available as information, firstly a join on road name has been performed over the set of events extracted, in order to select only relevant roads. A second selection has been performed on the set of over 160 roads affected previously extracted, in order to reduce significantly the number of traffic events to be manually located through the verbose description. A spatial intersection between the roads affected and the areas of interest defined by the EMS activation, has decreased the number of relevant events to 40. After those selections, looking at the descriptions, events have been manually located over the NAVSTREETS data set. Nineteen linear events and twenty-one point events have been added to the event layer.

From 200 traffic events extracted from the three sources, 152 spatially intersect the Area Of Interests (AOIs). A split operation of traffic events over the AOIs has been performed in order to separately work on each AOI: after this operation, events that spatially intersect the AOIs rise up to 274 (events crossing multiples AOIs were duplicated).

The final phase of the analysis in order to validate the information provided by Copernicus EMSMapping has been performed with the use of ArcGIS software and the Time Slide Tool. In particular, the idea was to simulate what can happen if the EMS operators can access the TMC event information during the map production. Two scenarios have been evaluated: the first one assumes that traffic events are available only in real-time; the second one implies also a storage system that enables the capacity of retrieving historical data. In fact, a traffic event, even if resolved at the time of AOI production, could still be useful for identifying possible damages.

Firstly, for each AOI defined by Copernicus EMS, a time window has been defined: the Copernicus protocol requires a maximum of 12 h to deliver a delineation or grading maps, so the date time of the delivery has been considered the stop time, and “stop time – 12 h” has been chosen as start time. For the first scenario, the time enablement on the traffic event layers uses two dates: the timestamp of the start of the event and the timestamp of the end of the event. For the second scenario, only the timestamp of the start of the event has been considered in time enablement.

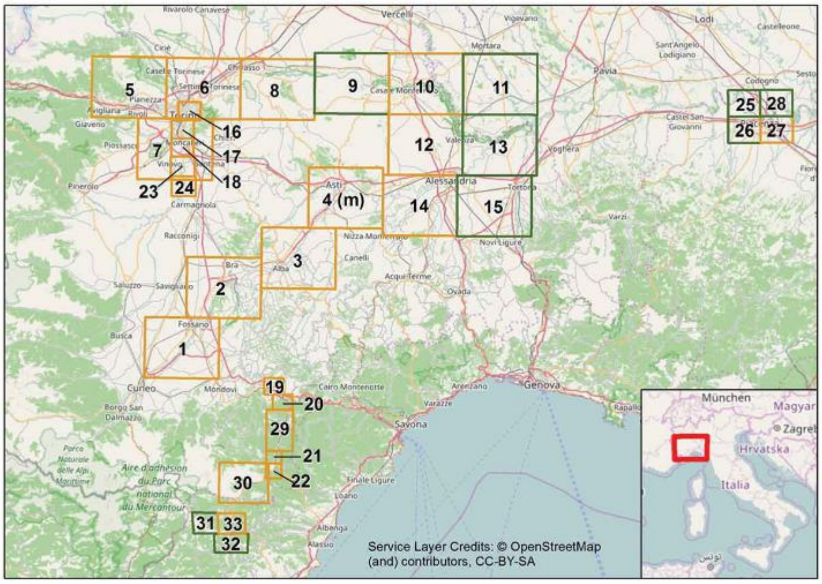
In this way, using the Time Slider, it is possible to slide over a time window and analyse which AOIs and traffic events occur during time. The resulting maps have been manually evaluated.

7.2.2 The case study

Copernicus EMS-Mapping was activated on 24 November 2016 and produced a total of 16 delineation maps and 18 grading maps (Figure 93). In this section traffic events and Copernicus EMS-Mapping products are compared in order to make consideration on if the integration of the two services may have improved the quality.

In the chart depicted in Figure 94, it is possible to see for each AOI the count of traffic events analysed in the two scenarios (only real-time or considering historical data). It is frequent that traffic events were present in AOI but did not overlap with the time extent of the map production. Considering the second scenario, the number of traffic events that can be used for EMS-Mapping analysis increases. From the analysis of traffic events, it has become necessary to categorize the utility for the mapped traffic events. In particular, there are traffic events validated by the Copernicus image analysis (“Validated”). For instance, in the case of a road closure it is verified that there are no cars crossing the road. In such cases, the road affected by the traffic events was often not inserted in the damaged roads detected by Copernicus, but in a perspective of integration, the road section could be classified as potentially affected. The second category defined is traffic events potentially useful to deepen the analysis (“Potentially Validated”): particularly in delineation cases, the location of closed roads and bridges could help in concentrate the analysis around those traffic events. The third and last categories are those traffic events that do not seem to be validated by the Copernicus image analysis (“Not Validated”), like traffic events far from the most affected areas or involving several kilometres of road crossing multiple

AOIs. In the chart (Figure 95) for each source of traffic events and for each scenario, there is the count of traffic for category of utility.



Service Layer Credits: © OpenStreetMap (and) contributors, CC-BY-SA

Figure 93: Areas covered by Copernicus EMS-Mapping during the late 2016 flood event in northern Italy.

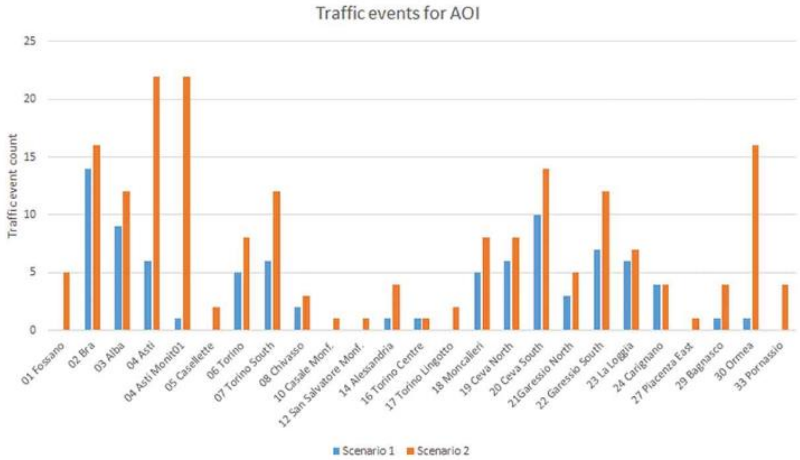


Figure 94: Distribution of the traffic events for each AOI (AOIs without traffic events have been excluded).

In the first scenario, from over 274 total events, only 88 are spatially and temporally overlaid with AOIs (32%). Of these, however, those that actually contribute to improve the damage detection activity of Copernicus (“Validated” and “Potentially Validated” events) are 38, about 13% of total events.

In the second scenario instead, from over 274 total events, 194 events are spatially and temporally overlaid with AOIs (71%) and those that contribute to improve the damage detection activity of Copernicus are 96, about 35% of total events.

In the first scenario, TCM_sistema is the richest source of located events (34), but only 35% of these were information that enriched the analysis (12 events). WordPress is the source that has proven to be the most useful, with 14 of the 24 events actually used to enrich the analysis. This is probably because, unlike the other two sources, these events are also located outside the TMC network.

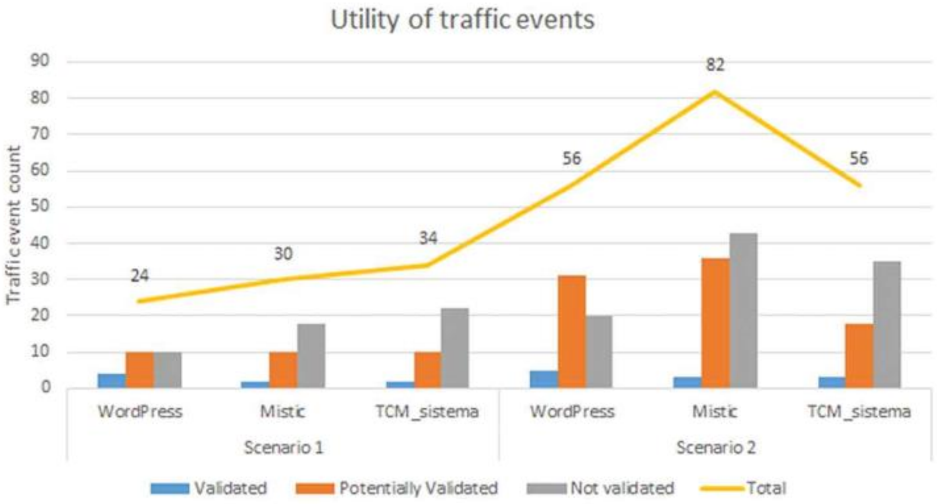


Figure 95: Distribution of the traffic events for category of utility.

In the second scenario, again WordPress is the most useful source, with 64% of the events used to enrich the analysis. MISTIC database, even if it is the source with the highest number of located events (82), has several events “Not Validated” (53%). During the analysis, a series of duplications between events have to be handled. In particular, among the traffic events from MISTIC database, there are duplicated points with the same identification code and same information about the event, except for the direction of travel concerned. In case of linear events, duplications (on the same time extent) have concerned road section of several kilometres, with a long temporal extension (about 3 days) along which there are other traffic events that affect limited portions of the affected section, with different types of events or temporal extension (usually 1 day). On TCM_sistema events, duplicate identification code appears to be more numerous. In this case, many of the duplicated events vary by the date and time of update, but there is no change in the version number of the identification code (as should be done following the procedures). Finally, between sources, MISTIC and TCM_sistema share many events (same identification code) that sometimes differ slightly in geometry, probably due to differences in the algorithm that calculates the geometry. Rarely, the event is duplicated in all three sources.

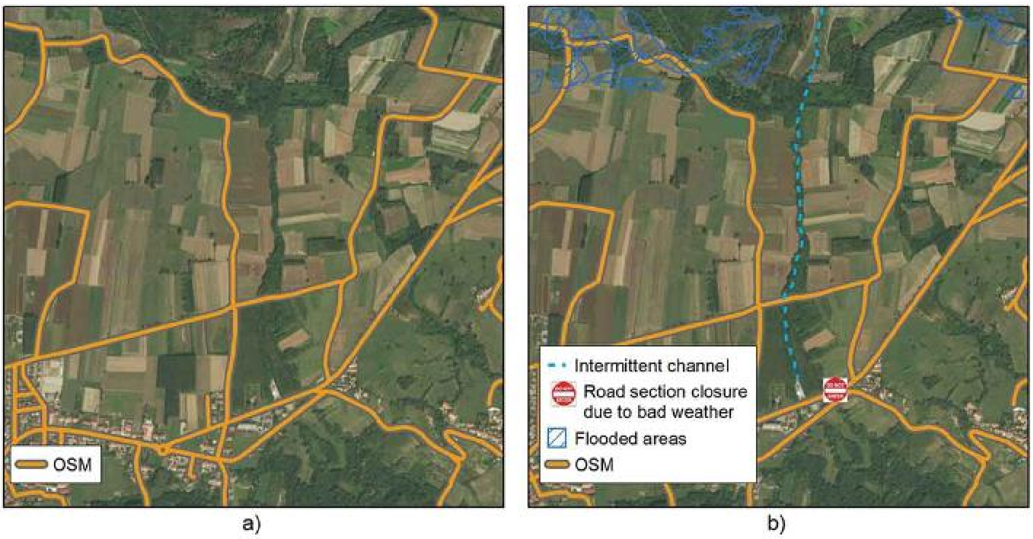


Figure 96: Enhancement of reference data through traffic events. (a) shows the reference data: no channel has been detected on this area. In (b) knowing the traffic events, the intermittent channel could have been mapped (dashed blue line in (b)).

In general, during the manual assessment of traffic events, those duplications have not affected the analysis, as duplications do not increase the time needed to perform the visual interpretation. Delineation maps produced are built on SAR imagery, which allow a semi-automatic extraction of the flooded areas. For those maps, in general, the presence of traffic events like temporary bridge or roads closure could have brought to a more accurate manual delineation of the flooded areas. Several traffic events in fact are close to detected flooded areas. In some cases, the presence of traffic events could have led to improved reference data: for example, in the Chivasso AOI, there was a road closure in an area not affected by flood according to EMS-Mapping analysis, but close to a seasonal stream, not detected in reference data produced by Copernicus EMS-Mapping, that presumably overflowed (see Figure 96). In this particular case, as the traffic event comes from the WordPress source, the point is located outside the TMC network data, unlike for MISTIC and TCM_sistema sources, thanks to the verbose description of the location. Knowing the traffic event, a more accurate investigation could be done, identifying the stream affected by the event, which was not easily identifiable from the imagery on which reference data are based.

Another particular case is the traffic event that reports the closure of the underpass near the Po River in San Mauro (North of Torino). This event had a big impact on traffic as the road is particularly important for the general traffic flow from Torino north, but such kind of event cannot be detected by any imagery. In case of grading maps, TMC location referencing shows limits related to the TMC coverage: most of the roads flagged as damaged by Copernicus EMS-Mapping are local and rural roads that are not monitored by the TMC system (Figure 97).

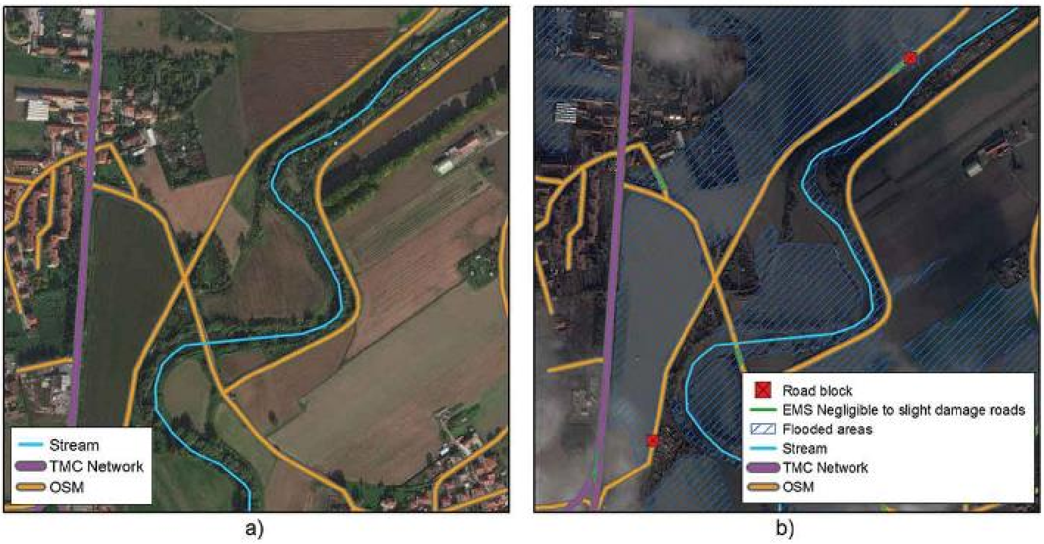
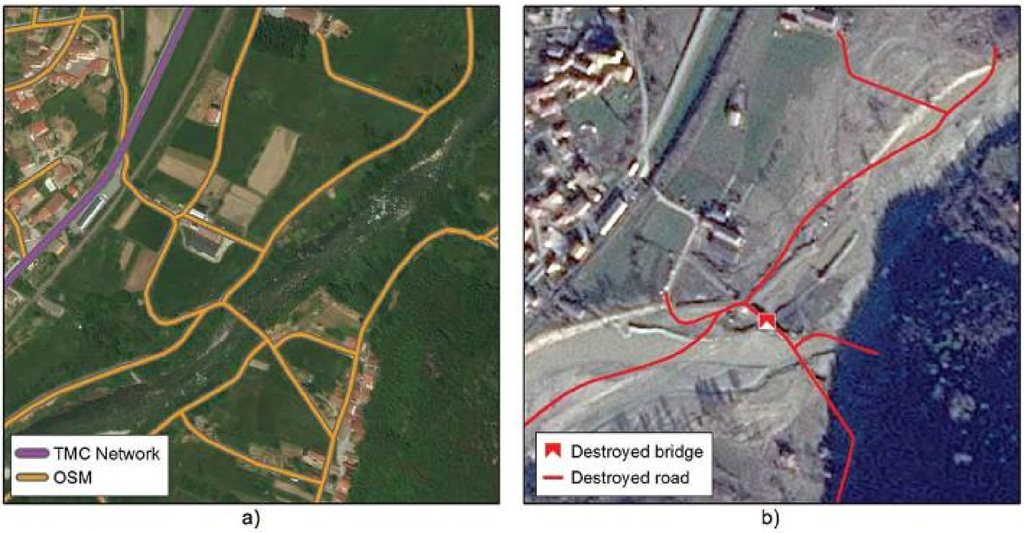


Figure 97: Road blocks on Chisone left bank. (a) shows the pre-event situation and the different coverage between the TMC and the OpenStreetMap road network coverage. (b) shows the road blocks on low functional class network (unpaved road) detected by Copernicus EMS-Mapping.

Similarly, in the area near to Bagnasco, a small village along the Tanaro river, several damages to the road infrastructure were identified according to Copernicus EMS-Mapping analysis: a bridge and portion of the road network were destroyed (Figure 98). The event was not reported by TMC because the affected roads were in a low-functional class, currently not covered by the service.

As a criterion to determine if TMC-based traffic event can be used as source for EMS-Mapping damage detection, the presence of cars on a road reported as closed by TMC-based traffic event has been evaluated. There are several cases where there was the evidence of road closure but not damages where visible from the imagery: this can be considered as a new special case where the EMS-Mapping operator can declare the road as possibly affected, specifying the TMC-based traffic event as source. For instance, on Ceva South area a road closure traffic event (SP353) is supported by the evidence that no cars are crossing the road in the imagery and other similar cases regard the SP300 to Valdinferno in Garessio South area, the SP143 from Vinovo to Carignano (La Loggia area) and the SP143 near Carignano. In those cases, the road does not seem to be particularly affected by flood or mudflow and damages are not visible, so probably the closure was decided for security reason.



a) b)

Figure 98: Damages to the road network near Bagnasco. (a) shows the pre-event situation and the different coverage between the TMC and the OpenStreetMap road network coverage. (b) shows the damages on low functional class network (bridge and roads) detected by Copernicus EMS-Mapping.

In several areas, relevant erosion phenomena associated with the flood event (i.e. not present in the prevent imagery) affected areas in proximity to road segments where TMC reported blockages, floodings and road closures. Figure 99 displays an area where Tanaro river left bank was clearly eroded: TMC reported an obstruction of the road network, and EMS-Mapping reported a complete destruction of a portion of the rail network next to the obstructed road. The availability of TMC event during Copernicus mapping activity may have led to report some degree of damages to the road network too.

In other areas, flood traces can be detected using satellite imagery on portion of the road network mentioned as disrupted by TMC. During the EMSMapping production phase, those traces were not noticed and therefore no damage reported on the road network. As in the previous case, the availability of the TMC event could have led to report some damages to the road network in the EMSMapping products.

Figure 99: Damages to the transport network. (a) shows the pre-event situation. (b) shows how EMS-Mapping reported damages only to the railway network while TMC reported an obstruction event to the nearby road too.

Turin Municipality, with a couple of decrees issued on the 12 and 19 December 2016, closed several portions of the cycle tracks along the Po River for lack of security requirements: damages were so relevant that most of those limitations were still active in June 2017 (Città Metropolitana di Torino, 2016). Damages on these infrastructures were detected by Copernicus EMS-Mapping and reported in products released on 28 November 2016 (Figure 100). The availability of a communication channel from Copernicus service to TMC would have allowed to timely disseminate this information to the population.

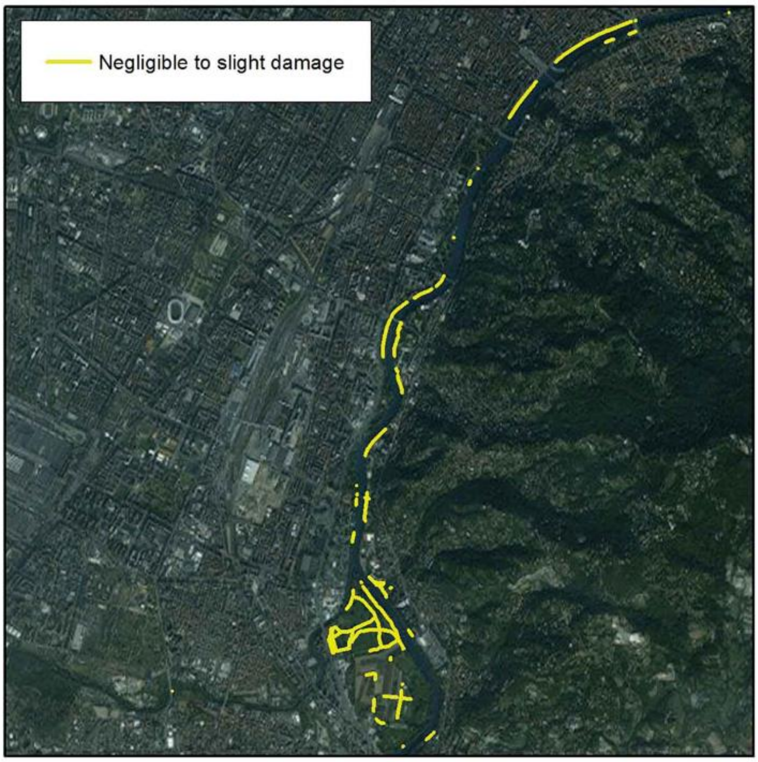


Figure 100: Damages on cycle tracks along the Po River in Turin municipality detected by Copernicus EMS-Mapping.

7.2.3 Consideration on the case study

The analysis conducted and documented in this section confirms the benefit of integrated TMC traffic events with emergency mapping services.

In particular, traffic events sent using TMC are certified and are a kind information that can also have a high level of detail, allowing for instance, improving the reference data, and above all, guiding the analysis of post-event imagery, in order to find damages otherwise not easily interpretable. It would be therefore an additional help to photo interpretation, which also benefits the fact of being certified and therefore highly reliable.

Looking at the distribution of AOIs and traffic events, it is evident that by integrating the two services, it was possible to agree on a better definition of the AOIs with the authorized user, justifying them also due to the presence of significant traffic events.

With the aim of developing real-time services and interactive maps for the Copernicus services, information generated by TICs becomes important during both activation and pre-activation phases.

The availability of traffic information in real-time for rapid mapping also includes the TICs in Europe, as they are the ones that can share a "certified" information. This is different from EMS-Mapping procedures, where damages are only interpreted and validated only afterwards. Therefore, one of the main efforts is to unify and standardize the TICs operational workflow and data. At the same time, the flow of the information can be reverted. As Copernicus EMS-Mapping usually maps detailed area and spot damages, it may be useful to provide that information, even if related to local or rural roads, to TICs, which can also eventually validate the event in a reasonable time.

However, there are still some issues that can be addressed by future studies. In particular, the difference between managing traffic events for public information and their use for emergency management has been highlighted. As far as the function of the TICs is concerned, the aim is to inform the public and especially drivers: this type of information is structured so that it can be transmitted via radio and GPS to enabled vehicles, allowing to update routing based on traffic status. For this type of objective, managing the event as a linear element, involving an even larger portion of the road segment actually affected, allows the information to be potentially transmitted to a greater number of users, even if to the detriment of information detail. Conversely, in the RM emergency service, the aim is to provide detailed information about damage, and in order to use traffic event information, it is better if it is provided by point location, which are less used in TICs.

The resolution of the TMC network is poor compared to the operational scale of RM services: from this study, it has emerged that some of the local roads involved with floods are not reported in TMC-based traffic events.

In addition, during the analysis some duplicated traffic events were found. This issue is mainly due to the internal workflow management for traffic events in 5T company and is related to the different aim of the application: for 5T the main purpose is to inform, lower attention is paid to the consistency of the information and redundancy is preferable.

The progressive implementation of the DATEX II standard may refine the localization methods: in fact, the standard provides three different possibilities for locating an event: WGS84 coordinates, TMC location codes and Open LR, a new protocol map-agnostic with a higher level of detail compared to TMC location system.

As previously stated the location referencing is the main problem in those kinds of applications, in particular if automation of encoding and decoding location of traffic events is expected. In Germany, a tentative to integrate TMC location code in OSM has started: the research institute BAST has been contacted in order to provided TMC location codes and authorize the process (OpenStreetMap Wiki contributors, 2009; OpenStreetMap Wiki contributors, 2016). Unfortunately, the project has been stopped since 2010. If all European TMC location will be pre-coded on OSM data, the automatic processing of traffic event data will be possible and will ease the process of creation of real-time services. With the spread of the OpenLR methods, a similar initiative can be conducted on OSM data.

Apart from increasing the automation for managing and disseminating traffic alerts, the adoption of a common and open road network data set, such as OSM, would be important for integrating crowdsourced information acquired by volunteers by means of mobile applications: that would support the completeness of the traffic events collected, including smaller ones but capable to cause, or participate in causing, congestions and disruptions of the road network. Social media intelligence can also be applied in order to capture events from textual information and geocode them on OSM. The main drawback of volunteer-acquired data is their accuracy and reliability, requiring the setup of validation processes.

Finally, during the simulation some doubts have emerged about the category assigned to the traffic events. In particular, there seems to be an overlap between categories and it is unclear, probably also for the operators, in which cases use one or other category (e.g. “Weather information – road closure due to bad weather” and “Traffic restrictions – road closure due to bad weather”). This issue will be probably solved by the adoption of the new Dictionary of DATEX II that among other things tries to eliminate the problem of overlap between categories.

Chapter 8

Conclusions and further developments

In this research, various aspects of mobility management have been deepened. The main goal of the current research was to develop a geospatial data model including main elements used in Traffic Operation Centre. In particular, one of the main purposes was to enable spatial data reuse for different purposes, as customised spatial analysis. The solution proposed achieve this goal, allowing the reuse of data and measures for custom purposes, without the need to deeply know the entire ITS environment system.

The mobility management field crosses a wide range of activities and thanks to new technologies, tends to be ever more complex and interconnected. In particular, mobility tasks todays are widely carry out using ITS solutions, which are highly efficient but hard to develop and sometimes blinded respect to spatial data. This approach indeed makes difficult to extract and reuse spatial mobility data in solutions “out of the box”. In addition, each Traffic Operation Centre is characterised by a different variety of tasks, and needs to merge a various range of solutions in order to covers all the activities. The nature of TOCs, which can be private or public companies, add complexity in data sharing and integration.

This research has made evident some considerations: first of all the difficult of use ITS data in GIS desktop solution in a straightforward way. The proposed solution try to overcome the peculiarity of ITS platforms and concentrates on data available and mechanism to reuse ITS data in a GIS environment.

From a spatial point of view, standard solution exists to manage data, but are sometimes too general, or related to a specific field of action. From the standard analysis has also emerged the difficult in creating a common dictionary of the terms involving the mobility: lot of terms are used for referring to the same concepts leading confusion when data integration is needed between TOCs.

In particular, the proposed solution of this research take into account various standard. INSPIRE Transport Network has been used for the segmentation and topology approach in the definition of the road network, but also several code list of the standard has been used to define network characteristics. The INSPIRE implementation allows also to eventually publish and share the road network with other administrations.

The O&M standard, as INSPIRE implementation, has been used as reference for the implementation of the traffic measures and for defining relationships with the spatial objects to which they are referred. The DATEX I standard has been used as reference to defined code lists for traffic events categorisation.

The analysed standards do not consider the representation of ancillary traffic elements as sensors, group of sensors, cameras, velox and traffic lights: the proposed data model indeed take into account the representation of those elements and define their spatial relationships with the road network. In addition, it try to develop general code lists (in the logical data model), implementing a first vocabulary activity which can be reused by other TOCs.

Looking at the logical data model proposed, as the solutions is not technology dependent, it can be used also by other TOCs, implementing it as whole or only partially and using the DBMS technology preferred. The proposed model indeed can be used also in an open source environment as PostgreSQL/PostGIS.

The proposed ArcGIS geodatabase solution is designed in order to be performant also in a limited power-computing environment. A weak point of the ArcGIS geodatabase solution proposed is that is highly dependent on the platform structure from which takes data. For instance, a change in field definitions in the original database server can affects the scripts developed for data extraction, compromising its functioning.

During the research, considerations about open data available as road network have been deepened. From this first analysis, even if spatial and attribute characteristics seems to be lacking for TOCs purposes, some of these data can be reused by TOC: this is particularly true for bicycle paths, not available in the commercial datasets but quite complete as open data. Operations needed to integrate those datasets can be deepened in further developments. Indeed, the logical data model proposed already contains elements for managing bicycles paths.

However, this solution adopted represents a first step in the outcomes of this research: indeed several aspects has been deepened in theory but not developed in practice.

In particular, from the point of view of measures, only a subset has been chosen. Other measures as delays, turn rates, VMS messages, cameras outputs, categorisations of flow in terms of light and heavy vehicles and traffic states have not been evaluated. In addition, real-time measures have not been addressed in this research.

The modelling activity of traffic event has not taken into account the DATEX II dictionary, which is wider and more complex respect to the code list used. Moreover, a more DATEX II compliant schema can be developed for traffic event, taking into account the situation concept, for grouping spatial events.

Looking at spatial aspect, many of the original fields of the NAVSTREETS Streets Data can be opportunely added to the model, enabling better routing and geocoding (as for instance the management of alternative street names). Other aspects can be further added in the model: in particular, data related to bicycles paths and related services as bike sharing and the public transport component.

References

Ambler, S., 2003. *Agile Database Techniques: Effective Strategies for the Agile Software Developer*. John Wiley & Sons, Inc., New York, NY, pp. 23 – 50.

Arco E., Ajmar A., Arneodo F., Boccardo P., 2017. An operational framework to integrate traffic message channel (TMC) in emergency mapping services (EMS). In: *European Journal of Remote Sensing,* vol. 50-1, pp. 478-495. ISSN 2279-7254. DOI: 10.1080/22797254.2017.1361306.

Ajmar, A., Boccardo, P., Broglia, M., Kucera, J., Giulio-Tonolo, F., Wania, A., 2017. Response to flood events: The role of satellite-based emergency mapping and the experience of the Copernicus emergency management service. In: Flood damage survey and assessment: New insights from research and practice, pp. 213–228, Geophysical Monograph Series 228, , ISBN: 978-1-119-21792-3. DOI:10.1002/9781119217930.ch14.

Arneodo, F., Cocozza, M., Gastaldi, D., Aguilar, R., Gagliardi, G., Taddei, A., Morello, E., Di Corpo, F., 2014. Progetto S.I.MO.NE. Definizione del protocollo di comunicazione. Version 4.0. [online] Available at: http://simone.5t.torino.it/index.php?option=com_jdownloads&Itemid=27&view=viewcategory&catid=9 [Last access: 20 September 2017]

Arneodo, F., Foti, G., Cocozza, M., 2009. S.I.MO.NE. Innovative system for metropolitan area mobility management. In: *Proceedings of the 16th ITS World Congress*, 21-25 September 2009, Stockholm, Sweden.

Arneodo, F., Gagliardi, G., 2010. *Progetto S.I.MO.NE. - Studio di fattibilità per l'estensione dei servizi di infomobilità, attraverso l'utilizzo del TMC.* Unpublished manuscript.

Berendsen, M. E., Harmerlinck, J.D., Lanning, S.G., 2010. *Developing NSDI Framework Data: ANSI Framework Data Content Standards. Guidance Document.* Prepared for the Federal Geographic Data Committee, by the Wyoming Geographic Information Science Centre, University of Wyoming, Laramie, Wyoming USA.

Booch, G., 1994. *Object-Oriented Analysis and Design with Applications*. 2nd edn, Santa Clara CA: Benjamin/Cummings. P. 39.

Bourée, K., Arneodo, F., Bjersing, U., Duquesne, C., Knowles, N., Tibaut, A. and Tijmensen, J., 2015. A European Standard for Modelling and Exchanging Fares. Presentation in: *ITS World Congress*, Bordeaux, France, 5 – 9 October 2015. Available at: http://www.transmodel-cen.eu/project/its-worldcongress-2015/ [Last access: 20 September 2017]

Buckley, D.J., 1997. *The GIS Primer. An Introduction to Geographic Information Systems*. Innovative GIS Solutions, Inc. Fort Collins, Colorado US. Chapter G.3. Available at: http://planet.botany.uwc.ac.za/nisl/GIS/GIS_primer/page_40.htm [Last access: 20 September 2017]

Butler, J.A., 2008. *Designing geodatabase for transportation.* Environmental Systems Research Institute, Inc., Redlands, CA, USA.

Cadkin, J., 2002 a. Dynamic Segmentation in ArcGIS. *ArcUser*, v.7, pp. 52 – 54. Available at: http://www.esri.com/news/arcuser/0702/files/dynseg.pdf [Last access: 20 September 2017]

Cadkin, J., 2002 b. Understanding Dynamic Segmentation Working With Events in ArcGIS 8.2. *ArcUser*, v.10, pp. 40 – 43. Available at: http://www.esri.com/news/ArcUser/1002/files/dynseg_2.pdf [Last access: 20 September 2017]

CEN/TC278, 2017. *CEN/TC 278 Intelligent transport systems.* [online] Available at: http://www.itsstandards.eu/tc278 [Last access: 20 September 2017]

Chorley, R.R.E., 1987. *Handling Geographic Information*. Report of the Committee of Enquiry chaired by Lord Chorley, Department of the Environment. London: HMSO.

Città Metropolitana di Torino, 2016. Relazione sintetica sull’evento alluvionale del 23 – 26 Novembre 2016. Available at: http://www.cittametropolitana.torino.it/cms/risorse/protciv/dwd/notizie-comunicati/anno-2016/maltempo-24-11-2016/cmto_rapporto_preliminare_evento_2016.pdf [Last access: 10 June 2017]

Council Directive 2010/40/EU of 7 July 2010 on the framework for the deployment of Intelligent Transport Systems in the field of road transport and for interfaces with other modes of transport. Available at: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32010L0040&from=EN [Last access: 20 September 2017]

Council of the European Union, 2006. *Review of the EU Sustainable Development Strategy (EU SDS) - Renewed Strategy*. Available at: http://register.consilium.europa.eu/doc/srv?l=EN&f=ST%2010117%202006%20INIT [Last access: 20 September 2017]

Curtin K., Noronham V., Goodchild M.F., Grise S., 2003. *ArcGIS Transportation GIS Model.* Environmental Systems Research Institute, Inc., Redlands, CA, USA.

Goodchild M.F., 1998. Geographic Information Systems and disaggregate transportation modelling. *Geographical Systems,* v. 5, n. 2, pp. 19–44.

Date, C.J., 2000. *An Introduction to Database Systems*. Seventh ed., Addison-Wesley. Reading, MA, USA.

DATEX II, 2006. *DATEX II V1.0 USER GUIDE.* Available at: http://www.datex2.eu/sites/www.datex2.eu/files/sites/test.datex2.eu/files/DATEXIIv1.0-UserGuide_v1.0.pdf [Last access: 5 May 2017]

Dölger, R., Geißler, T., 2011. DATEX II – The standard for ITS on European Roads. [online] Available at: http://www.datex2.eu/sites/www.datex2.eu/files/Datex_Brochure_2011.pdf [Last access: 5 May 2017]

ESRI, 1990. *Understanding GIS: The ARC/INFO Method*. Environmental Systems Research Institute, Inc., Redlands, CA, USA. Pp. 1 – 2.

ESRI, 1998. *ESRI Shapefile Technical Description - An ESRI White Paper.* Environmental Systems Research Institute, Inc., Redlands, CA, USA. Available at: https://www.esri.com/library/whitepapers/pdfs/shapefile.pdf [Last access: 20 September 2017]

ESRI, 2001. *Linear Referencing and Dynamic Segmentation in ArcGIS™ 8.1. - An ESRI White Paper.* Environmental Systems Research Institute, Inc., Redlands, CA, USA. Available at: http://www.unigis.ac.at/fernstudien/unigis_professional/Materialien/module/modul2/dynamic_segmentation_arcgis.pdf [Last access: 20 September 2017]

ESRI, 2017 a. *ESRI GIS Dictionary*. [online] Available at: http://support.esri.com/en/other-resources/gis-dictionary [Last access: 20 September 2017]

ESRI, 2017 b. *What is Network Analyst?* [online] Available at: http://desktop.arcgis.com/en/arcmap/latest/extensions/network-analyst/ [Last access: 20 September 2017]

European Commission, 2011. *WHITE PAPER Roadmap to a Single European Transport Area – Towards a competitive and resource efficient transport system*. Available at: http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:52011DC0144 [Last access: 20 September 2017]

Federal Geographic Data Committee, 2008 a. *Geographic Information Framework Data Content Standard. Part 7: Transportation Base*. Federal Geographic Data Committee Secretariat, U.S. Geological Survey, Virginia, USA. [online] Available at: https://www.fgdc.gov/standards/projects/FGDC-standards-projects/framework-data-standard/GI_FrameworkDataStandard_Part7_Transportation_Base.pdf [Last access: 20 September 2017]

Federal Geographic Data Committee, 2008 b. *Geographic Information Framework Data Content Standard. Part 7c: Transportation – Roads*. Federal Geographic Data Committee Secretariat, U.S. Geological Survey, Virginia, USA. [online] Available at: https://www.fgdc.gov/standards/projects/FGDC-standards-projects/framework-data-standard/GI_FrameworkDataStandard_Part7c_Transportation_Road.pdf [Last access: 20 September 2017]

Federal Geographic Data Committee, 2008 c. *Geographic Information Framework Data Content Standard. Part 7d: Transportation - Transit*. Federal Geographic Data Committee Secretariat, U.S. Geological Survey, Virginia, USA. [online] Available at: https://www.fgdc.gov/standards/projects/FGDC-standards-projects/framework-data-standard/GI_FrameworkDataStandard_Part7d_Transportation_Transit.pdf [Last access: 20 September 2017]

Fu, P., Sun, J., 2010. *Web GIS: Principles and Applications*. Environmental Systems Research Institute, Inc., Redlands, CA, USA.

Gabay, Y., Doytsher, Y., 1994. Automatic adjustment of line maps. In: *Proceedings of GIS/LIS'94 Annual Convention*, Phoenix, Arizona, USA, 25-27 October, pp. 333 – 341.

Goodchild M.F., 2007. Citizens as sensors: the world of volunteered geography. *GeoJournal*, v.69, n.4, pp. 211 – 221.

Goodchild M.F., Glennon, J.A., 2010. Crowdsourcing geographic information for disaster response: a research frontier. *International Journal of Digital Earth*, v. 3, n. 3, pp. 231 – 241.

Goodwin, C., Latham, F. E., Siegel, D., Gordon, S., 1998. Intelligent Transportation Systems Datum Public Sector Requirements. *Workshop Report in Spatial Data Interoperability Protocol for ITS Project*, Oak Ridge National Laboratory, Oak Ridge, Tennessee.

Guttman, A., 1984. R-tree: A Dynamic Index Structure for Spatial Searching. *Proceedings of the 1984 ACM SIGMOD International Conference on Management of data*, ACM New York, NY, USA, v. 14, n. 2, pp. 47 – 57.

Huxhold, W.E., 1991. An Introduction to Urban Geographic Information Systems. Oxford University Press, New York. Pp. 27.

INSPIRE, 2013 a. *D2.10.1 INSPIRE Data Specifications – Base Models – Generic Network Model.* Drafting Team "Data Specifications". [online] Available at: https://inspire.ec.europa.eu/file/1542/download?token=_7tV1nx7 [Last access: 20 September 2017]

INSPIRE, 2013 b. *Guidelines for the use of Observations & Measurements.* INSPIRE Cross Thematic Working Group on Observations & Measurements.

[online] Available at: http://inspire.ec.europa.eu/documents/Data_Specifications/D2.9_O&M_Guidelines_v2.0rc3.pdf [Last access: 20 September 2017]

INSPIRE, 2014. *D2.8.I.7 Data Specification on Transport Networks – Technical Guidelines.* European Commission Join Research Centre. [online] Available at: https://inspire.ec.europa.eu/file/1542/download?token=_7tV1nx7 [Last access: 20 September 2017]

Iovanella, A., 2009. *Dispense per il corso di GRAFI E RETI DI FLUSSO*. Università degli Studi di Roma "Tor Vergata". Chapter 1 and 3. Available at: http://www.uniroma2.it/didattica/ro1 [Last access: 20 September 2017]

ISO, 2005. *ISO/IEC 19501:2005 Information technology - Open Distributed Processing - Unified Modeling Language (UML) Version 1.4.2*". Available at: https://www.iso.org/standard/32620.html [Last access: 20 September 2017]

ISO, 2017. *ISO in brief. Great Things Happen when the world agrees.* [online] Available at: https://www.iso.org/files/live/sites/isoorg/files/archive/pdf/en/isoinbrief_2015.pdf [Last access: 20 September 2017]

ISO/TC211, 2017 a. *ISO/TC211 – Geographic Information/Geomatics.* [online] Available at: https://committee.iso.org/home/tc211 [Last access: 20 September 2017]

ISO/TC211, 2017 b. C*o-operative agreement between ISO/TC 211 Geographic Information/Geomatics and the Open GIS Consortium, Inc. (OGC).* [online] Available at: http://www.isotc211.org/Agreements/Agreement_OGC.pdf [Last access: 20 September 2017]

Jedlička K., Hájek, P., Cada, V., Martolos, J., Šťastný, J., Beran, D., Kolovský, F. and Kozhukh D., 2016. *Open transport map — Routable OpenStreetMap.* In: IST-Africa 2016 Conference Proceedings, Paul Cunningham and Miriam Cunningham (Eds), IIMC International Information Management Corporation. Durban, South Africa, 11 - 13 May, 2016. ISBN: 978-1-905824-54-0

Kamalski, T. , 2005. European experience with TMC: Supplying value. *TMC workshop*, Brussels.

Knowles, N., Miller, P., 2008. *A Transmodel based XML schema for the Google Transit Feed Specification – With a GTFS / Transmodel comparison*. Kizoom Ltd 109-123 Clifton Street, London. [online] Available at: http://naptan.dft.gov.uk/transmodel/schema/doc/GoogleTransit/TransmodelForGoogle-09.pdf [Last access: 20 September 2017]

Lee, D., Yang, W., Ahmed, N., 2014. Conflation in Geoprocessing Framework – Case Studies. Recent development and looking ahead. In: *GEOProcessing 2014.* Barcelona, Spain, March 23 – 27, 2014.

Litwin, L., Rossa, M., 2011. *Geoinformation Metadata in INSPIRE and SDI.* Lecture Notes in Geoinformation and Cartography, Springer Ed.

Longhorn, R. A., 2005. Geospatial Standards, Interoperability, Metadata Semantics and Spatial Data Infrastructure. Background paper for *NIEeS Workshop on Activating Metadata*, 6-7 July 2005, Cambridge, UK.

Ludwig, I., Voss, A., Krause-Traudes, M., 2011. *A Comparison of the Street Networks of Navteq and OSM in Germany.* Advancing Geoinformation Science for a Changing World. Editors: Geertman, S., Reinhardt, W., Toppen, F.. In: Lecture Notes in Geoinformation and Cartography. Ed. Springer-Verlag Berlin Heidelberg. , v. 1, pp: 65 – 84. DOI: 10.1007/978-3-642-19789-5_4

Mathis, P., 2007. *Graphs and Networks: Multilevel Modeling.* 2nd Ed., London, UK: ISTE Ltd. Pp. xv-xxiii. Available at: http://onlinelibrary.wiley.com/doi/10.1002/9781118595473.fmatter/pdf [Last access: 20 September 2017]

Maron, M., 2015. How complete is OpenStreetMap?. *Point of Interest, the Official Mapbox blog.* [online] Available at: https://www.mapbox.com/blog/how-complete-is-openstreetmap/ [Last access: 25 May 2017]

McDermott, J., 2013. Google takes its tracking into the real world. *Digiday UK.* [online] Available at: https://digiday.com/media/google-tracking/ [Last access: 20 September 2017]

Nystuen, J.D., Frank, A.I., Frank, L., 1997. Assessing topological similarity of spatial networks. In: *Proceedings of International Conference on Interoperating Geographic Information Systems*, Santa Barbara, CA, USA.

OGC, 2005. *Topic 0: Abstract Specification Overview.* In: The OGC^{TM} abstract Specification, Open Geospatial Consortium, Inc. Available at: http://portal.opengeospatial.org/files/?artifact_id=7560. [Last access: 20 September 2017]

OGC, 2012. OGC City Geography Markup Language (CityGML) Encoding Standard. [online] Available at: https://www.citygml.org/ [Last access: 20 April 2018]

OGC, 2017. *The Open Geospatial Consortium*. [online] Available at: http://www.opengeospatial.org/ [Last access: 20 September 2017]

OpenStreetMap Wiki contributors, 2009. Kommunikation/BASt/LCL. *OpenStreetMap Wiki.* [online] Available at: http://wiki.openstreetmap.org/wiki/DE:Kommunikation/BASt/LCL [Last access: 26 May 2016].

OpenStreetMap Wiki contributors, 2016. TMC_Import_Germany. *OpenStreetMap Wiki.* [online] Available at: http://wiki.openstreetmap.org/wiki/TMC/TMC_Import_Germany [Last access: 20 February 2017].

OpenStreetMap Wiki contributors, 2017 a. Way. *OpenStreetMap Wiki.* [online] Available at: http://wiki.openstreetmap.org/w/index.php?title=Way&oldid=1520779 [Last access: 20 February 2017].

OpenStreetMap Wiki contributors, 2017 b. Key:highway. *OpenStreetMap Wiki.* [online] Available at: http://wiki.openstreetmap.org/w/index.php?title=Key:highway&oldid=1498196 [Last access: 20 February 2018].

OpenStreetMap Wiki contributors, 2018 a. Forward & backward, left & right. *OpenStreetMap Wiki.* [online] Available at: http://wiki.openstreetmap.org/w/index.php?title=Forward_%26_backward,_left_%26_right&oldid=1565451 [Last access: 20 February 2018].

OpenStreetMap Wiki contributors, 2018 b. Relation. *OpenStreetMap Wiki*. [online] Available at: http://wiki.openstreetmap.org/w/index.php?title=Relation&oldid=1550853 [Last access: 20 February 2018].

OpenStreetMap Wiki contributors, 2018 c. Accuracy. *OpenStreetMap Wiki*. [online] Available at: http://wiki.openstreetmap.org/w/index.php?title=Accuracy&oldid=1551047 [Last access: 20 February 2018].

OpenStreetMap Wiki contributors, 2018 d. Completness. *OpenStreetMap Wiki*. [online] Available at: http://wiki.openstreetmap.org/w/index.php?title=Completeness&oldid=1551362 [Last access: 20 February 2018].

Oracle® Spatial, 2008. *Topology and Network Data Models Developer's Guide.* Available at: https://docs.oracle.com/cd/B28359_01/appdev.111/b28399.pdf [Last access: 20 September 2017]

Qumer, A., Henderson-Sellers, B., 2008. An evaluation of the degree of agility in six agile methods and its applicability for method engineering. *Information and software technology*, v. 50, n. 4, pp. 280 – 295.

Rodrigue, J.P., 2017. *The Geography of Transport Systems*. Hofstra University, Dept. of Global Studies & Geography, New York, USA. Available at: http://people.hofstra.edu/geotrans. [Last access: 20 September 2017]

Roush, W., 2012. Welcome to Google transit: How (and why) the search giant is remapping public transportation. *Community Transportation* v.3, pp. 20 – 29. [online] Available at: http://web1.ctaa.org/webmodules/webarticles/articlefiles/Spring_12_DigitalCT_Google_Transit.pdf [Last access: 20 September 2017]

Ruiz-Alarcon-Quintero, C., 2016. Harmonization of transport data sources according to INSPIRE data specification on transport networks. *XII Conference on Transport Engineering*, CIT 2016, 7-9 June 2016, Valencia, Spain. In: Transportation Research Procedia, v.18, pp. 320 – 327. DOI: 10.1016/j.trpro.2016.12.043

Saalfield, A., 1988. Conflation: Automated Map Compilation. *International Journal of Geographic Information Systems*, v.2, n.3, pp. 217 – 228.

Scarponcini, P., 1999. Generalised model for linear referencing. In: *Proceedings of the Seventh International Symposium on Advances in Geographic Information Systems*, Kansas City, MO, USA, November 5-6 1999, pp. 53 – 59.

Snyder, A., "Encapsulation and Inheritance in Object-Oriented Languages", Proc. Object-Oriented Programming Systems Languages and Applications Conf., pp. 38-45, Nov. 1986.

Thill, J.C., 2000. Geographic information systems for transportation in perspective. *Transportation Research Part C - Emerging Technologies,* v.8, pp. 3 – 12.

Tibaut, A., Arneodo, F., Bjersing, U., Bourée, K., Duquesne, C., Knowles, N. and Reynolds, S., 2017. Importance of data standards for European passenger mobility. Presentation in: *25th International Symposium on Electronics in Transport (ISEP)*, Ljubljana, Slovenia, 27 - 28 March 2017. Available at: http://www.transmodel-cen.eu/project/isep-2017-congress/ [Last access: 20 September 2017]

TMC Forum, 2003. *TMC Handbook - Location Coding Handbook*. Version 07.

Tomlin, C.D., 1990. *Geographic Information Systems and Cartographic Modelling*. Prentice Hall, Englewood Cliffs, New Jersey, pp. XI.

Tomlinson, R., Toomey, M. A.G., 1999. GIS and Lis in Canada. In: G. McGrath, L. Sebert, ed. 1999. *Mapping a Northen Land: the Survey of Canada 1947-1994.* McGill-Queen's University Press, Toronto. Ch. 15. Available at: http://www.esri.com/news/arcnews/fall12articles/origins-of-the-canada-geographic-information-system.html [Last access: 20 September 2017]

TRANSMODEL, 2015. Data model in Enterprise Architect format. [online] Available at: http://www.transmodel-cen.eu/wp-content/uploads/sites/2/Trm6-Part-1-2-3-Sept2015.eap_.zip [Last access: 20 September 2017]

Voigt, S., Giulio-Tonolo, F., Lyons, J., Kucera, J., Jones, B., Schneiderhan, T., Guha-Sapir, D., 2016. Global trends in satellite-based emergency mapping. *Science*, vol. 353, n.6296, pp. 247–252. DOI:10.1126/science.aad8728

Walter, V., Fritsch, D., 1999. *Matching spatial data sets: a statistical approach.* International Journal of Geographical Information and Science, vol. 5, n.13, pp. 445 – 473.

Wania, A., Broglia, M., Kucera, J., Spruyt, P., Rossi, M., Spinoni, J., Louvrier, C., 2016. 2016 user workshop of the Copernicus emergency management service – Summary report. *Publications Office of the European Union*. ISBN: 978-92-79-63450-5, DOI:10.2788/2672

Wikipedia contributors, 2017 a. Traffic message channel. *Wikipedia, The Free Encyclopedia*. [online] Available at: https://en.wikipedia.org/w/index.php?title=Traffic_message_channel&oldid=825695154 [Last access: 20 February 2018]

Wikipedia contributors, 2017 b. Navteq. *Wikipedia, The Free Encyclopedia*. [online] Available at: https://en.wikipedia.org/w/index.php?title=Navteq&oldid=827324995 [Last access: 20 February 2018]

Wikipedia contributors, 2018. OpenStreetMap. *Wikipedia, The Free Encyclopedia.* [online] Available at: https://en.wikipedia.org/w/index.php?title=OpenStreetMap&oldid=824772218 [Last access: 20 February 2018]

Wroclawski, S., 2014. Why the world needs OpenStreetMap. *Guardian News*. [online] Available at: https://www.theguardian.com/technology/2014/jan/14/why-the-world-needs-openstreetmap [Last access: 20 September 2017]

Wroclawski, S., 2018. Why OpenStreetMap is in serious trouble. *Emacsen's Blog*. [online] Available at: https://blog.emacsen.net/blog/2018/02/16/osm-is-in-trouble/ [Last access: 20 September 2017]