






Article

Semantic Traffic Sensor Data: The TRAFAIR Experience

Federico Desimoni ^{1,†}, Sergio Ilarri ^{2,†}, Laura Po ^{1,*,†}, Federica Rollo ^{1,†} and Raquel Trillo-Lado ^{2,*,†}

¹ “Enzo Ferrari” Engineering Department, University of Modena and Reggio Emilia, 41121 Modena, Italy; federico.desimoni@unimore.it (F.D.); federica.rollo@unimore.it (F.R.)

² Department of Computer Science and Systems Engineering, I3A, University of Zaragoza, 50018 Zaragoza, Spain; silarri@unizar.es

* Correspondence: laura.po@unimore.it (L.P.); raqueltrl@unizar.es (R.T.-L.)

† All the authors contributed equally to this work and are shown in alphabetic order.

Received: 18 June 2020; Accepted: 20 August 2020; Published: 25 August 2020



Abstract: Modern cities face pressing problems with transportation systems including, but not limited to, traffic congestion, safety, health, and pollution. To tackle them, public administrations have implemented roadside infrastructures such as cameras and sensors to collect data about environmental and traffic conditions. In the case of traffic sensor data not only the real-time data are essential, but also historical values need to be preserved and published. When real-time and historical data of smart cities become available, everyone can join an evidence-based debate on the city’s future evolution. The TRAFAIR (Understanding Traffic Flows to Improve Air Quality) project seeks to understand how traffic affects urban air quality. The project develops a platform to provide real-time and predicted values on air quality in several cities in Europe, encompassing tasks such as the deployment of low-cost air quality sensors, data collection and integration, modeling and prediction, the publication of open data, and the development of applications for end-users and public administrations. This paper explicitly focuses on the modeling and semantic annotation of traffic data. We present the tools and techniques used in the project and validate our strategies for data modeling and its semantic enrichment over two cities: Modena (Italy) and Zaragoza (Spain). An experimental evaluation shows that our approach to publish Linked Data is effective.

Keywords: data management; semantics; sensor data; data integration; data annotation; traffic in smart cities

1. Introduction

Public administrations handle large amounts of data concerning their internal processes as well as to the services that they offer to citizens. Following the “open by default” worldwide recognized principle [1], a lot of public-sector information is increasingly published as open data in standard formats, to enhance interoperability and efficiency in data reuse. Among all public data, open transport data is one of the most frequently re-used data domains in the European Data Portal (EDP) [2] and has been identified as highly impactful. Indeed, 7600 datasets related to transport are published on the EDP, which provide information about bike-sharing and bicycle hiring systems, seasonal traffic conditions, and road construction. These datasets are accessible via the EDP, which is harvesting metadata from national open data portals. In turn, the national open data portals publish the data or harvest themselves the data from different institutions within their countries, such as federal, regional, and local portals, national, regional and local government bodies, and research institutions. Sharing traffic data in an anonymized form can lead to innovative products, such as services that enable

users to find available parking slots and provide the best route to reach those locations, apps able to make predictions about traffic conditions based on past data, as well as assistance agents that provide real-time traffic information for users and relevant government units.

On the other hand, the analysis of traffic data is crucial not only for the development of smart traffic management systems (e.g., see [3]) but also for making the cities more safe, healthy, and sustainable [4]. Several European countries have adopted the Vision Zero and Safe System approach [5], to eliminate deaths and grave injuries on European roads. Furthermore, global efforts focus on achieving the transition to sustainable mobility of freight and people and meeting the sustainable development goals of the 2030 Agenda for Sustainable Development [6]. Moreover, since traffic is considered to be responsible for a large portion of urban pollutants released into the atmosphere (e.g., see [7–9]), understanding traffic flows can help in mitigating urban pollution, that is a major source of health problems (e.g., see [10,11]).

Motivated by the importance of sharing these data, this paper tackles the modeling of traffic-related data, and particularly the conversion of the data about traffic sensors' locations and measurements into Linked Data and their publication as Open Data. This is a small part of a bigger project called "TRAF AIR—Understanding Traffic Flows to Improve Air Quality". The TRAF AIR project is concerned about the study of how traffic impacts on the urban atmospheric pollution. Given that the traffic generated by motor vehicles is a principal source of air pollution, information related to traffic is highly valuable.

This paper focuses on traffic data managed within TRAF AIR. Specifically, our experience with the modeling and semantic annotation of traffic data, as well as regarding the analysis and application of different tools and techniques used for those tasks, is explained. The structure of the rest of this paper is as follows. Firstly, some related work is discussed in Section 2. Secondly, in Section 3, the background of this work is presented, including a brief description of the goals and motivations of the TRAF AIR project and the modeling choices adopted for handling traffic data. Thirdly, in Section 4, our approach for annotating and publishing traffic data is described. Fourthly, in Section 5, the evaluation and validation of our proposal is tackled. Finally, in Section 6, conclusions and future work are sketched.

2. Related Work

Publishing open data has become an increasingly pressing need within government bodies and public administrations. The principles of sharing public information have been defined by the International Open Data Charter [1] and are the following: (1) Open By Default, (2) Timely and Comprehensive, (3) Accessible and Usable, (4) Comparable and Interoperable, (5) For Improved Governance and Citizen Engagement, and (6) For Inclusive Development and Innovation. Opening up data often happens in an ad-hoc manner, and in many cases thousands of datasets are published without adhering to commonly-agreed standards and without reusing common identifiers. Hence, finding, reusing, and integrating data from different sources is a real challenge. Linked Data can respond to these challenges and can lead to smarter and more efficient government services and applications. Therefore, a crucial aspect when sharing open data is to follow the Linked Data principles [12].

Moreover, to publish high-quality, semantically annotated Open Data, it is crucial to identify the ontologies that better describe the domain of interest [13]. Ontologies provide a formal representation of the domain of interest and constitute the component with which the Linked Open Data (LOD) consumers (both humans and software programs) interact. Then, also the mapping between ontologies and data is important, since it is used to translate the operations on the ontology in terms of concrete actions on the data.

In the following subsections, related approaches for publishing smart city traffic data are described and different types of ontologies related to traffic data are analyzed. The identification of the most relevant ontologies and concepts and the mapping between data and the selected ontologies, instead, is described in Section 4.1.

2.1. Sharing Smart City Traffic Data

Several works have been published to define how to structure and share data produced in a smart city [14]. Smart urban traffic ecosystems are identified in [15] as an example of a “big service”, “evolved from the collection of collaborating, interrelated services for handling and dealing with big data”. By collecting suitable sensor data and defining appropriate data exploitation strategies, it is possible to empower both citizens and decision-makers to improve our quality of life. However, for this dream to come true, the development of suitable data management strategies that can provide citizens and administrations with the information they need is a key issue. Thus, for example, according to [16], informational interventions are vital to encourage changes in attitudes and perceptions of people. Along the same lines, the work presented in [17] emphasizes that “open data can impact positively on citizens in particular and society in general”.

Traffic data can help in detecting traffic congestion, providing traffic flow prediction, and identifying traffic accidents. For this reason, several projects have published traffic information as open data. To lower the barrier for open data consumers to reuse traffic information, an Open Traffic Lights ontology has been proposed in [18]. That paper also reports a specification to publish historical and live data with Linked Data Fragments and a method to preserve the published data in the long-term. In [19], for measuring the urban road congestion degree, that is one of the major issues in most metropolises, the estimation of a Traffic Congestion Index (TCI) of every road segment at every time slot has been proposed. As a final example, an open traffic data platform has been presented in [20] and used as a sensor data provider for different management applications.

The increased interest in smart city data sharing for the public interest can be assessed by the number of datasets shared on the open data portals. When we searched in the European Data Portal for “traffic”, we obtained 8949 datasets (search done on 5 August 2020: <https://www.europeandataportal.eu/data/datasets?locale=en&query=traffic&page=1>). Even if transport data only cover a 2.25% of the total datasets (see the European Data Portal statistics per category at <https://www.europeandataportal.eu/catalogue-statistics/CurrentState>), a positive trend can be observed. Mainly, traffic open data are statistics that show the number of registered cars in different countries, and they are usually provided in the form of high-level vehicle fleet data in a city. On the other hand, data concerning the average daily traffic volumes on different roads and the specific traffic volume on different days and at different hours on specific road segments are not always shared with citizens. We argue that having these data would be remarkably useful, as their exploitation could enable politicians of our cities to make more informed decisions, and the public could also be better informed about the traffic situation and use real data to promote health and environmental protection. The publication of the data as Linked Data enables a suitable and interoperable sharing of data that can facilitate the development of applications and services for smart cities [21,22].

2.2. Analysis of Traffic-Related Ontologies

Different types of existing ontologies related to the traffic of vehicles can be considered. We can highlight the following ones:

- The Vocabulary to Represent Data About Traffic Ontology [23], developed by Óscar Corcho (a member of the Ontology Engineering Group at the Polytechnic University of Madrid) has been proposed for the representation of the situation of traffic in a city. It extends the Sensor Network Ontology (SSN) [24–26] to represent the intensity of traffic on the different road segments of a city. It represents road segments (concept *escjr:TramoVia*), traffic observations (concept *estrf:TrafficObservation*, which for the moment is specialized only in the subconcept *estrf:TrafficIntensityObservation*, but other subconcepts could be added in the future to represent other types of traffic observations), the sensor or sensing system used to obtain a given measurement (concept *estrf:TrafficIntensitySensor*, which is considered optional), the result of an observation (concept *TrafficIntensitySensorOutput*, which has a value-concept

estrf:TrafficIntensityObservationValue, linked to *TrafficIntensitySensorOutput* through the property *ssn:hasValue* and is produced by a specific sensor or sensing system identified by a specific URI and linked to *TrafficIntensitySensorOutput* through the property *ssn:isProducedBy*), and finally an instance *estrf:TrafficIntensity* that represents the type of property being measured (in this case, the intensity of the traffic).

This vocabulary is still work in progress, developed in the context of the working group on transport of AENOR [27]. The authors recommend using this vocabulary in conjunction with the vocabulary proposed to represent city road maps (particularly, road segments) [28]. This proposal does not currently contemplate the modeling of traffic properties other than traffic intensities (*estrf:TrafficIntensityObservation*), but they can be easily added by extending *estrf:TrafficObservation*.

- The work presented in [29] presents an ontology-driven architecture that enables performing several automatic tasks to increase traffic safety and improve the comfort of the drivers. The ontology layer is described as composed of three groups of interrelated concepts: concepts related to vehicles, concepts related to roads, and concepts related to sensors. The concepts related to vehicles describe a taxonomy of vehicles of different types, including commercial vehicles, public vehicles (buses and taxis), private vehicles (cars, bicycles, and motorbikes) and priority vehicles (ambulances, police cars, and fire trucks), and also allow representing information about their routes and locations. The concepts related to the infrastructure include a taxonomy of different types of roads (local roads, prefectural roads, national highways, and national expressways), as well as the representation of other parts of the infrastructure, such as the road segments, traffic lights and traffic signs, lanes, road markings (e.g., painted arrows), and other infrastructure elements (tunnels, parkings, roundabouts, bridges, gas stations, and toll stations). Finally, the concepts related to sensors are based on the use of the SSN ontology. Besides, a mapping schema is proposed to map the sensor data to semantic data, as in [30], in such a way that the sensor data can be automatically represented as instances of the SSN ontology; the property observed is *Car_flow* property.

This is a relevant work that proposes an ontological layer covering different aspects of traffic. Still, it mainly focuses on the development of an architecture that exploits such a layer to perform various actions through an agent layer. Some use case scenarios are presented: regulating the air conditioning of a car, traffic light adjustment based on the traffic flow and the weather conditions, and traffic congestion control for GPS navigators. Regarding the representation of traffic sensor data, the focus is only on the traffic flow, and, rather than proposing a new ontology or extending an existing one, the SSN ontology is directly adopted.

- The Open511 specification [31] has been proposed as an open format for publishing road event data. Information about the road events can be provided by publishing an XML file or by allowing access to the data through a dynamic API. It supports representing elements such as events and geographic areas (places represented in GeoNames [32,33]); examples of events are constructions, special events (such as the celebration of a sport event), incidents (including accidents and other unexpected events), weather conditions, and road conditions (such as snow, ice, or fire on the road).

This work currently covers event data rather than traffic information. Nevertheless, some additional resources have also been proposed (currently as drafts that may be included in the Open511 specification in the future) to represent average historical speeds and the current speed of road segments.

- The Road Accident Ontology [34] focuses on the representation of information about accidents (vehicles affected, location of the accident) and the parties involved (persons involved in the accident and their insurance companies). This proposal is a draft, submitted by Daniel Dardailler for the W3C Geek Week celebrated in July 2012.

This ontology does not represent traffic, but we have included it because accidents can affect traffic and even lead to traffic jams.

- As another work focusing on accidents, the work in [35] proposes a lightweight Car Accident Ontology for VANETs (CAOVA), that includes information about vehicles, accidents, occupants and the environment. The goal is to facilitate information about an accident to emergency vehicles.
- It is also relevant to mention the Transportation Planning Suite of Ontologies (TPSO) [36], which is a set of ontologies proposed for transportation planning. More specifically, eight ontologies are proposed to cover concepts related to time, meteorology, spatial locations, units of measure, changes, activities, recurring events, resources, and observations. Among these, we can highlight here the Observation Ontology [37], which reuses the SSN Ontology to capture the concepts related to sensors, but also extends it by adding a few classes and properties for the organization of terms. Specific traffic properties (such as the traffic flow or speed) are not explicitly modeled in the proposed ontology.
- The KM4City [38] is an ontology for smart cities developed by the University of Florence (Italy) as a support for a platform that collects and integrates data related to the Tuscany region in Italy. It includes concepts regarding streets (Road, Node, RoadElement, AdministrativeRoad, Milestone, StreetNumber, RoadLink, Junction, Entry, EntryRule, Maneuver, Lanes, and Restriction), local public transportation (Ride, Route, RouteSection, BusStop, etc.), and sensors of traffic and different types of events (e.g., SensorSite, TrafficObservation, TrafficSpeed, TrafficConcentration, TrafficHeadway, etc.).
- Finally, some ontologies support modeling energy consumption data. Although they are not explicitly focused on traffic, they could be used as an input for traffic estimation. On the one hand, the Smart Appliances REference (SAREF) ontology [39] allows the representation of information related to devices (e.g., a washing machine, a temperature sensor, etc.) in a smart appliances domain as well as their functions and profiles (e.g., for energy optimization). On the other hand, the FIEMSER ontology [40] models the organization of building spaces (using concepts such as Building, BuildingPartition, BuildingSpace, and BuildingZone) and the devices used in the building (defining concepts such as Device, HomeEquipment, ControlledDevice, and also more specific types such as Boiler and Radiator). Based on data provided by smart appliances, it could be possible to estimate the occupancy levels in households and buildings and thus indirectly estimate information about the traffic of vehicles outside (e.g., expected traffic variations along the day).

Summing up, as far as we know, for the moment, there is no comprehensive working traffic ontology extensively being applied. However, the Vocabulary to Represent Data About Traffic Ontology [23] commented above is very promising and can be easily extended to include all the elements that may be needed for traffic monitoring, mainly when used in conjunction with other ontologies, such as an ontology for road maps and a weather ontology to represent the weather conditions affecting the traffic observed. We have found the KM4City ontology particularly relevant for our purposes. Our specific approach used for the semantic representation of traffic data is described in Section 4.1.

3. Traffic Modelling in TRAFAIR

In this section, the context of this work is provided. Firstly, Section 3.1, summarizes the motivation and goals of the TRAFAIR project. Then, Section 3.2 focuses on the description of traffic data, which is the subject of this paper, and its modeling.

3.1. Scope and Purpose of the TRAFAIR Project

Pollution is the primary environmental cause of premature death in Europe. Thus, according to a European Union (EU) report published by the European Environmental Agency [41], poor air

quality caused 412,000 premature deaths in Europe in 2016. Intending to improve air quality, the European Commission is carrying out several policies with their respective legislative measures. However, nowadays, the situation is still critical in some member states that cannot reach the goals fixed by Europe. Indeed, in February 2017, the European Commission warned five countries (Spain and Italy, among them) about continuous violations of rules established regarding atmospheric pollution. These countries are having difficulties in controlling the levels of NO_2 emissions within the allowable ranges, that are mainly produced by the traffic of vehicles. Therefore, the European Commission requires its member states to perform actions to guarantee the quality of air and safeguard public health. In this context, public administrations and citizens lack a complete set of tools to allow the estimation of the level of pollution at an urban scale, which depends on the variable traffic conditions, which would lead to an optimization of the control strategies and an increase of the air quality awareness.

Motivated by the problems mentioned above, the overall goal of the TRAF AIR (Understanding Traffic Flows to Improve Air Quality) project [42,43] is to develop a service that will allow citizens and municipalities to estimate and predict urban air quality in six cities in Europe. The prediction estimation, in particular, is produced by considering the 3-D shape of city buildings, meteorological conditions, and traffic flows. As part of the project, datasets representing urban air quality maps will be published in catalogs of data collected by the European Data Portal [2]. Besides, different use cases will be considered, including the development of mobile apps for final users. More specifically, the main goals are the following:

1. The definition of a standard set of metadata, extending the ones adopted at the European level and defined by FAIRMODE [44], able to represent urban air quality maps [45].
2. The provision of real-time estimations of air pollution in a city on an urban scale. For this purpose, low-cost air quality sensors are deployed, combining their measures with measures provided by official air quality stations to build informative maps of the different levels of pollution in different urban areas.
3. The development of a service to predict the urban air quality based on meteorological prediction and traffic flow, using High-Performance Computing (HPC) technologies to estimate the dissemination of pollutants. A traffic flow model is used to simulate new circulation hypothesis (e.g., changes regarding the types of vehicles and their proportions in the float of vehicles in the city, increments in the number of low-emission vehicles used, the definition of areas with restricted circulation in a city, etc.) and their impact on the air quality.
4. The publication, in catalogs collected by the European Data Portal, of open datasets describing urban air quality maps of six European cities of diverse size where the service will be deployed: Zaragoza (Spain), with about 600,000 inhabitants, Florence (Italy), with about 382,000 inhabitants, Modena (Italy), with about 185,000 inhabitants, Livorno (Italy), with about 160,000 inhabitants, Santiago de Compostela (Spain), with about 95,000 inhabitants, and Pisa (Italy), with about 90,000 inhabitants.

As mentioned previously, the transport sector is responsible for a large proportion of urban air pollution. Therefore, this paper focuses on traffic data and presents our data modeling, data integration, and data publication strategy followed for traffic data in the context of the TRAF AIR project.

3.2. Modeling of Data Provided by Traffic Sensors

Traffic data can be measured by using different types of sensors, such as detectors located along the roadside, which use various technologies to detect the presence of vehicles [46,47]. More specifically, traffic count technologies can be split into two categories: intrusive methods and non-intrusive methods. On the one hand, intrusive methods usually consist of a data recorder and a sensor placed on the road like pneumatic road tubes or induction loops. On the other hand, non-intrusive techniques are based on remote observations, such as manual counting, microwave radars, or video image detection. These techniques allow the detection of different types of data, such as the volume of traffic (counts of

the numbers of vehicles on different road segments in a city), travel speeds, and in some cases even the specific types of vehicles (cars, motorbikes, buses, vans, pickup trucks, trailer trucks, large trucks, articulated lorries, etc.), occupancy rates, etc.

In the rest of this section, key ideas about the modeling of data provided by traffic sensors are explained. Firstly, Section 3.2.1 presents some examples of sensors that provide traffic data for the TRAFAIR project. Then, Section 3.2.2 describes our database model for traffic data.

3.2.1. Traffic Sensors in Two Representative Cities

For illustration, the traffic sensors used in two representative cities within the context of the TRAFAIR project (Modena and Zaragoza) are mentioned in this section.

The traffic sensors used in Modena are induction loops that are insulated electrically-conducting loops installed under the road surface. A lead-in cable connects the loop to the detector, which is an electronic unit that detects the presence of vehicles above the loop. In these sensors, the vehicle passing over the sensor is registered by an increase in inductance. An induction loop can be located in a specific lane of a street to count the number of vehicles passing over it in a specific direction. In Modena there are 400 induction loops (see Figure 1). Three hundred forty-six induction loops are managed by the City Council (the blue markers in Figure 1) and are distributed around the city center, on almost every road of junctions with traffic lights. In contrast, 54 induction loops are placed on regional and municipal roads under the control of the Emilia Romagna Region (the magenta markers in Figure 1). Sensors provide, along with the identifier of the sensor, the number of vehicles passing over the sensor during a time interval, the timestamp of the beginning of the time interval, and the average speed.

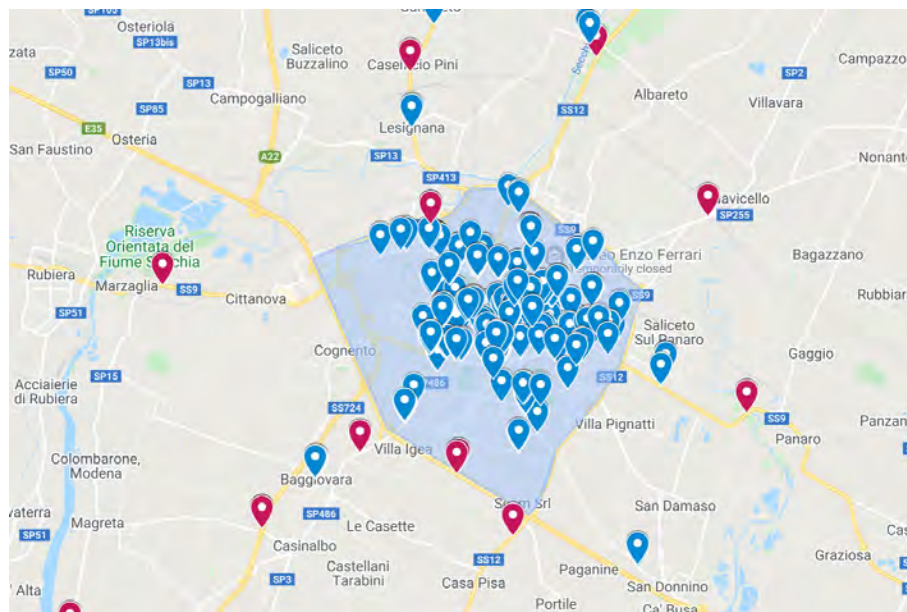


Figure 1. Induction loops in Modena (the blue and magenta markers indicate the position of sensors managed by the City Council and the Emilia Romagna Region, respectively). Map data: Google, 2020.

In the case of Zaragoza, the Zaragoza Council has Bluetooth antennas distributed around the city. Besides, several “links” have been defined as specific routes from one antenna to another antenna: the average speed of the vehicles that went through a link within a specific time interval (5 min) is computed by considering the distance between the antennas and the time needed by the vehicles to traverse that link. After that, according to that average speed, a color is assigned to each specific route to show the data on a map. Besides, the Zaragoza Traffic Control Center also provides us with some historical data obtained by both static devices and mobile devices measuring the traffic flow of different road segments in the city:

- Traffic static devices, which are 46 devices installed in different positions of the city of Zaragoza. More specifically, they are inductive coils located under the asphalt. These devices provide data about the traffic for 24 h a day for all the days in a year. Usually, there are two devices on the same road, one for each direction of circulation. However, in a few exceptions (specifically, for two cases), there is only one device measuring the traffic in just one direction. In Figure 2, a representation of the positions of these sensors is provided (shown with green markers).
- Traffic mobile devices, which are mobile traffic-detecting devices installed in 594 different points of the city throughout the year. Usually, there are also two devices on the same traffic road (one for each direction of circulation), as it is also the case for static devices. With these devices, data about the traffic measured during 24 h can be obtained (usually during only one or two days in a year, as these devices are located at fixed positions only for a few days).



Figure 2. Static traffic sensors in Zaragoza (shown with green markers). Map data: Google, Instituto Geográfico Nacional, 2020.

The traffic sensors, described in this section, are equipment belonging to the cities of Modena and Zaragoza. These sensors were already in place when the TRAFair project started (they were previously installed by the corresponding city councils), and the respective city councils collaborate with the TRAFair project by providing and facilitating access to those data.

3.2.2. Database Model for Traffic Data

Since smart cities collect and make decisions based on data coming from sensors installed in the city, a platform where all the sensor-related information can be stored is needed.

A unified data platform was created to collect the measurements coming from the sensors by using automatic processes. The data platform is a PostgreSQL object-relational database [48] with around 30 tables. Besides, the open-source PostGIS extension [49], which adds support for spatial and geographic objects and enables location queries in SQL, is used.

OpenStreetMap (OSM) [50], which relies on Volunteered Geographical Information (VGI), collected by contributors, to offer free map data, has been used as a source of road data. Alternatives to OSM include proprietary solutions such as Google Maps [51], Apple Maps [52], HERE maps [53], and TomTom maps [54], to cite some examples. To ensure a better sustainability and maintenance of our project, we have chosen OSM, as it is the only solution that is completely freely available for downloading in a format that can be easily stored and exploited in a database. In the cities of Zaragoza and Modena, the roadmap data provided by OSM are satisfactory for our purposes. Indeed, according to existing studies, usually cities are expected to be well represented in OSM

(e.g., see [55,56]). According to [55], “VGI can reach very good spatial data quality”; some works have analyzed the quality of OSM (e.g., recent studies of OSM datasets have been presented for Spain [57] and the Lombardy region in the north of Italy [58]). Furthermore, since OpenStreetMap is based on the contribution of volunteers, it is easy to correct some information. For the city of Modena, our modifications were related to the number of lanes of some roads and whether the road is one-way or not, which are crucial data to geolocalize traffic sensors. Besides, according to [59], OSM datasets are a great source of open data and can contribute to a more sustainable and transparent modelling. Nevertheless, if more complete and accurate data are required in a project, other roadmap data sources can be used, and this would not have a major impact on our approach, as our model is generic and can accommodate other data sources.

The two entity types used to model the information related to traffic sensors are illustrated in the entity/relationship (E/R) diagram shown in Figure 3, based on Chen’s notation [60]. The corresponding “SENSOR_TRAFFIC” table stores the identifier of the sensor (ID), its type (SENSOR_TYPE), its position as a point data type of PostGIS (GEOM), the identifier of the street in OpenStreetMap (OSM) in which the sensor is located (ROAD_SECTION), the sequential number of the specific piece of a street (segment) with the sensor (NUM_SEGMENT), the OSM node which is the closest one to the sensor (NEAREST_NODE), the direction of the vehicles counted by the sensor (DIRECTION, which is true if it is the same specified by the order of the nodes mapped on the street in OSM and false otherwise), and the sequential number of the lane in which the sensor is located (LANE, where the value zero indicates the rightmost lane in that direction). The measurements of the sensors are stored in the “SENSOR_TRAFFIC_OBSERVATION” table. In particular, the identifier of the sensor (ID), the beginning of the sampling interval of the observation (DATETIME), and the type of vehicles the measurements are related to (VEHICLE_TYPE), all compose the primary key of the table. For the sensors that are not able to categorize the type of vehicle, the value of VEHICLE_TYPE is “unknown”. The other attributes are the number of vehicles counted by the sensor (FLOW), the average speed (SPEED), and an optional occupancy rate attribute (OCCUPANCY, which is an estimation of the time a vehicle is above the sensor). The observation rates can have different values, according to the model of the sensors and their configuration. “SENSOR_TRAFFIC_OBSERVATION” is a weak entity type (its total participation in the relationship R is shown in Figure 3 using the notation proposed by Elmasri and Navathe [61]) that depends on “SENSOR_TRAFFIC”; its primary key is composed by the attributes DATETIME, VEHICLE_TYPE and also the ID of the corresponding SENSOR_TRAFFIC. Notice that Figure 3 only represents a small fragment of the TRAFAIR database (the part related to traffic sensor observations).

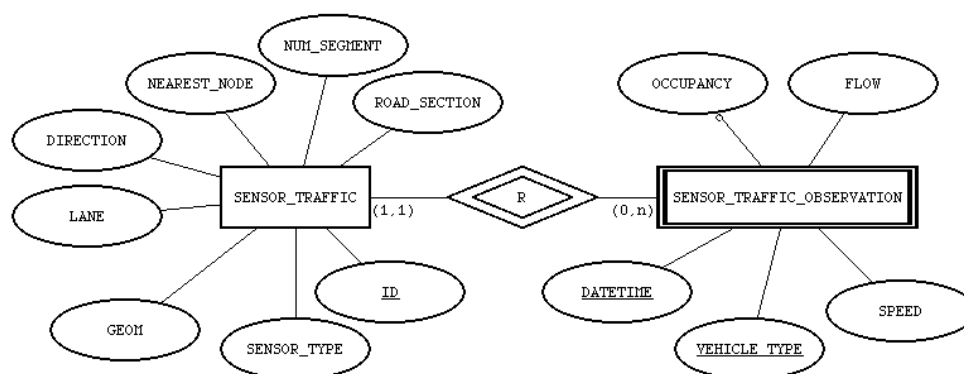


Figure 3. E/R schema with the entity types of the TRAFAIR data platform used to collect the measurements of the traffic sensors.

4. Data Annotation and Publishing

The conversion of the data about traffic sensors and the measurements they take over a long period into Linked Data is within the scope of the TRAFAIR project. The approach implemented for this purpose is shown in Figure 4 and will be detailed in this section. Data related to the general information about the sensors and their measurements are stored into the TRAFAIR database. The tool Karma [62,63] takes this data as input in CSV format and transforms it into Linked Data by using appropriate ontologies and by exploiting the Linked Geo Data ontology [64] for mapping the information of OpenStreetMap. The Linked Data produced by Karma is automatically loaded into a SPARQL endpoint, which is queried by the visualization tool called Lodview [65].

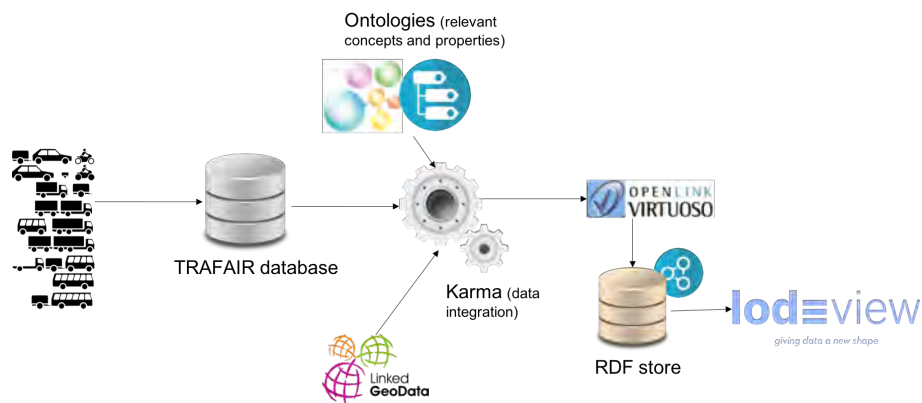


Figure 4. Overview of the mappings and tools used for data annotation and publishing. The credit of some images used has to be accredited to the creators of the software mentioned and used here for illustration purposes, others are freely available images extracted from Pixabay. The logo of Virtuoso has been extracted from <https://commons.wikimedia.org/wiki/File:Virtuoso-logo-sm.png>, contributed by Deirdre Gerhardt.

In this section, the process that starts with the data storing into the operational database and progresses until the production of the Linked Open Data is described. Firstly, Section 4.1 focuses on the identification of the relevant concepts and properties. Secondly, Section 4.2 describes the process followed for data integration. Thirdly, Section 4.3 explains the implementation of a SPARQL endpoint by using Virtuoso [66,67] and how Lodview [65] has been used to make the data available online. In the end, Section 4.4 summarizes the technological choices made and other potential alternatives.

4.1. Identification of Relevant Concepts and Properties

As reported in Section 2.2 an extensive research looking for already-existing traffic-related ontologies and vocabularies has been undertaken. At the end of this process, some of these ontologies have been selected to annotate the traffic concepts. Since no single ontology fits our needs perfectly, a combination of concepts defined in different ontologies has been used. Moreover, in some cases, it was necessary to create new classes and properties, since the available definitions were not suitable.

For the “sensor_traffic” entity type, described in Section 3.2.2 and depicted in Figure 3, some definitions of the Km4City ontology have been used to annotate the content of this table. In particular, the class *km4c:SensorSite* (Traffic Sensor) is used to identify the sensor capable of observing the traffic and the speed of the vehicles, the property *km4c:hasGeometry* to specify the point where the sensor is located, the class *km4c:Road* with the property *km4c:placedOnRoad* to refer to the name of the road where the sensor is located, the property *km4c:type* to identify the type of the traffic sensor, and the property *km4c:direction* for the direction of the vehicles counted by the sensor. It is important to specify that, to better link our data, all the streets present in our database are transformed into instances of the class *km4c:Road* giving them a URI that is, simply, the concatenation of the strings <https://trafair.eu/> and the street name. In addition, the Basic Geo (WGS84 lat/long) Vocabulary [68] has

been used to represent the latitude (*geo:lat*) and the longitude (*geo:long*) of the above mentioned point. The Basic Geo Vocabulary is a basic RDF vocabulary that provides the Semantic Web community with a namespace for representing the latitude, longitude, and other information about spatially-located entities, using WGS84 as the reference datum.

Furthermore, the Linked Geo Data ontology has been exploited to transform the “road_section” attribute of the “sensor_traffic” table into linked data, since this attribute contains the identification number of a way in OSM and the Linked Geo Data ontology makes the information collected by the OSM project available as an RDF knowledge base according to the Linked Data principles [12]. In Linked Geo Data, the ways of OSM are dereferenceable objects at the link <http://linkedgeodata.org/triplify/wayOSMID>, where *OSMID* is the identifier of the way.

The content of the “road_section” attribute has been concatenated to the link <http://linkedgeodata.org/triplify/way>; therefore, if the value of the “road_section” attribute is “387989963”, then it becomes <http://linkedgeodata.org/triplify/way387989963>, which is a dereferenceable object in the Linked Geo Data Knowledge Base [64]. The same approach has been used for the attribute “nearest_node”, adding the link <http://linkedgeodata.org/triplify/node> to the identifier of the node, which is another element of OSM. Two new properties have been defined to connect a sensor to its way and its nearest node, called *trafair:isLocatedInOSMWay* and *trafair:hasNearestOSMNode*, respectively (see Figure 5). Moreover, the name of the city where the sensor is located has been added. The city has been identified by exploiting the class *dbo:Place* of the DBpedia Ontology [69], and the property *dbo:Location* to link the instance of the class *dbo:Place* to the traffic sensor.

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#> .

<http://trafair.eu/ontology/hasNearestOSMNode>
a rdf:Property, owl:ObjectProperty ;
rdfs:domain geo:SpatialThing ;
rdfs:range <http://linkedgeodata.org/meta/Node> .

<http://trafair.eu/ontology/isLocatedInOSMWay>
a rdf:Property, owl:ObjectProperty ;
rdfs:domain geo:SpatialThing ;
rdfs:range <http://linkedgeodata.org/meta/Way> .
```

Figure 5. Definition of the “isLocatedInOSMWay” and the “hasNearestOSMNode” properties.

Concerning the “sensor_traffic_observation” entity type in Figure 3, the Km4City ontology has been exploited. Each instance of this entity type represents an observation made by one sensor. An instance of the class *km4c:Observation* identifies the observation. Since the primary key of the corresponding “sensor_traffic_observation” table is composed of three attributes (identifier of the sensor, timestamp indicating the beginning of the observation, and type of vehicles observed), the URI of each observation has been created as the concatenation of the values of these attributes. The property *km4c:measuredBySensor* has been used to connect the observation to the sensor, while the properties *km4c:vehicleFlow* and *km4c:averageSpeed* have been exploited to indicate the number of vehicles and their average speed, respectively. The property *rdfs:label* has been used to represent the type of vehicles, which is enough for representing the names of the types of vehicles; as an alternative, an existing ontology of vehicles, like the Vehicle Ontology or the Vehicle Sales Ontology, could have been extended (it should be noticed that the Vehicle Ontology available at <https://enterpriseintegrationlab.github.io/icity/Vehicle/doc/index-en.html> does not currently define subclasses of the concept “Vehicle” and that the Vehicle Sales Ontology available at <http://www.heppnetz.de/ontologies/vso/ns> does not cover all the types of vehicles managed in TRAF AIR; however, new concepts can be added as needed).

The last two attributes hold the start and the end of the time interval of the observations. Different kinds of sensors send measurements with different time intervals, while defining how to aggregate the original data when sharing this information is up to the public administration that owns the data. The second attribute is a new attribute calculated over the first attribute adding the time interval the observation refers to. These attributes have been mapped using the *prov:startedAtTime* and *prov:endedAtTime* properties of the PROV-O ontology [70]. Figure 6 shows the triples generated for a sample traffic observation.

```

@prefix km4c: <http://www.disit.org/km4city/schema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<https://trafair.eu/observation/modena/LP1/moto/2019-03-03T01:15:00>
km4c:measuredBySensor <https://trafair.eu/sensor/modena/LP1> ;
rdfs:label "moto" ;
a km4c:TrafficObservation ;
prov:startedAtTime "2019-03-03T01:00:00"^^xsd:dateTime ;
prov:endedAtTime "2019-03-03T01:15:00"^^xsd:dateTime ;
km4c:averageSpeed "94.2"^^xsd:string ;
km4c:vehicleFlow "2"^^xsd:string .

```

Figure 6. Triples generated for a sample traffic observation in Turtle syntax.

In Appendix A, the data model, based on the identification of the relevant concepts and properties explained in this section, is shown.

4.2. Data Integration

Karma [62,63] has been selected as the tool for representing the traffic data provided by the sensors in Linked Data. The goal is to map data stored in the TRAF AIR database by using the selected classes and properties described in Section 4.1. Karma is a data integration tool developed by the University of Southern California. It is an Extract, Transform, Load (ETL) tool [71,72] which is capable of (1) retrieving data from different data sources such as files, some Relational Database Management Systems (RDBMS) such as MySQL, Microsoft SQL Server, Oracle, and PostgreSQL with PostGIS, and various API services, (2) applying several kinds of transformations to the data, such as adding columns to the dataset and renaming columns, and then (3) providing the output file containing data transformed into the RDF format. This last operation is the most time-consuming one because the user has to select and load the ontologies he/she wants to use and then map each attribute of the dataset to the most appropriate class/property of the selected ontologies. This assumes the identification of the ontologies of interest according to the type of data to map. After the mapping, it is possible to download the R2RML model, which has been created, and the RDF file. R2RML [73] is a standard language proposed by the W3C RDB2RDF Working Group for expressing customized mappings from relational databases to RDF datasets. The R2RML model contains the mappings as RDF graphs which are written down in Turtle syntax. The R2RML model can be used to execute Karma in batch mode to generate RDF for large datasets and automate the transformation process. In this way, the user is not required to map the ontologies over the data every time he/she wants to make some transformation. Furthermore, in our scenario, this was the approach adopted.

The graphical user interface of Karma has been used to create two models: one for each table. In Figure 7, a graphical representation of the model used to map the attributes of the “sensor_traffic” table is provided, while the mapping in Figure 8 is related to the “sensor_traffic_observation” table. Karma requires some input data to create the model; however, it is not necessary to develop the model importing the whole amount of data from the table. For this reason, initially, some exemplar tuples from the corresponding tables of the TRAF AIR database have been imported by configuring

the connection to the database and using the appropriate query. Then, the selected ontologies have been uploaded and each attribute has been manually mapped to the suitable class/property. Once the mapping is concluded, the R2RML models that can be applied to a larger dataset were downloaded by using the Karma RDF Generation Service [74]. This service allows generating RDF data and publishing it on a SPARQL Endpoint. The information required for the transformation is the path of the file containing the data to be transformed (allowed formats are CSV, JSON, XML, and Excel), the URI of the R2RML model, the SPARQL endpoint, and the graph URI where the RDF data will be published.

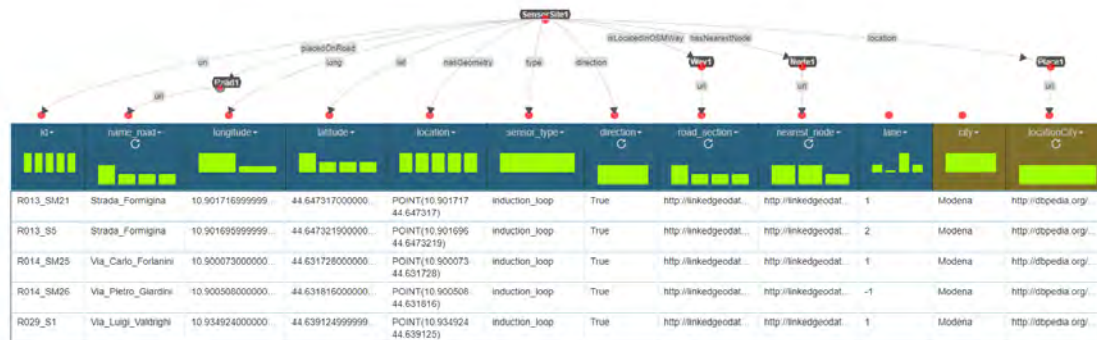


Figure 7. Karma model implemented to transform data from the “sensor_traffic” table (Figure 3) into Linked Data.

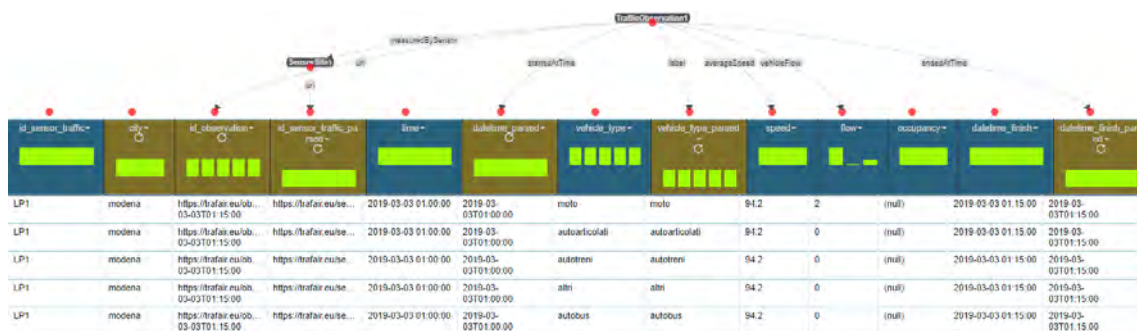


Figure 8. Karma model implemented to transform data from the “sensor_traffic_observation” table (Figure 3) into Linked Data.

4.3. Data Publication and Exploitation

Once the data transformation process is over, the open data are ready to be published. There are two main ways of publishing Linked Data on the Web: through a data dump or on a SPARQL endpoint.

On the one hand, a data dump places all dataset triples in one or more archive files. Dumps need to be downloaded entirely before they can be queried. This might be a problem since dump files can have large sizes (e.g., in our context, one year of sensor observations at 1-h granularity takes about 3.5 GB). For this to be manageable, some policies must be established to define a suitable period granularity of each data dump (e.g., one per month, one per year, etc.). Moreover, keeping the data in each data dump up-to-date requires effort. With a solution based on data dumps, the users can download the entire dump after every update or download and apply incremental patches.

On the other hand, a SPARQL endpoint lets clients evaluate any desired (read-only) query on a server. This gives clients direct access to (only) the data they are interested in. Thus, only very little bandwidth is required, and the data is always up-to-date and can be flexibly queried.

The choice of publishing data on a SPARQL endpoint has been selected as the most convenient one. The possibility of providing data dumps was discarded because of the difficulty to foresee the needs of the users, which would be needed to set the appropriate granularity and criteria that should be considered to generate and keep the available data dumps up-to-date. Nevertheless, data dumps

could also be provided in cases where specific types of needs are foreseen, in order to cover those specific needs, along with the SPARQL endpoint for more flexible querying; complementing the SPARQL endpoint with some data dumps to cover the expected popular needs of users could be useful particularly for non-technical users with no knowledge of SPARQL.

A SPARQL endpoint is a service that can be queried in real-time and allow further data processing, for example, for traffic management systems. The RDF repository can be explored through SPARQL queries, which allow the users to express their data needs in a precise way. In this way, the user can generate his/her own data dumps (by submitting queries), adapted to his/her specific needs, rather than being forced to select among a predefined list of data dumps generated according to specific granularities and criteria set in advanced. The SPARQL endpoint is available at <https://trafair.eu/sparql>; since traffic data are not published by the City of Modena, data shared through the SPARQL endpoint are currently example data.

For the implementation of the RDF repository, Virtuoso [66,67] has been chosen because it combines the functionality of a triple store and a SPARQL endpoint and it offers a user interface for querying the underlying data store. Indeed, every user can query the dataset in the way he/she needs, looking for the required information by using the SPARQL interface and thanks to the expressiveness of the SPARQL query language. After the installation of the tool, a new named graph called “trafair” has been created. It is possible to upload the files containing the data transformed by Karma through the graphical interface of Virtuoso. However, the Karma RDF Generation Service has been used since it allows automating the transformation and publication of the RDF data to the “trafair” graph.

A limitation of Virtuoso is that it is not able to visualize the information of a particular subject unless this is done through an ad-hoc query. This task can be particularly hard for non-skilled users. To overcome this limitation, a visual tool was installed alongside Virtuoso, called LodView [65]. LodView is a JSP web application able to offer a W3C standard compliant IRI dereferenciation. It is a Linked Data visualization tool that shows, in a tabular layout, the information of a resource, given its URI. LodView improves the end user’s experience in accessing HTML-based representations of RDF resources. In particular, it splits the information into four sections: (1) a header, containing associated images (if any), audios, videos, and a map if the resource has latitude and longitude properties; (2) the main section, that contains all the information related to the resource; (3) a section that contains all the inverse relations that LodView was able to gather from the underlying endpoint; and (4) a section that includes data related to the instances connected to the resource through the *owl:sameAs* property.

LodView, in conjunction with a SPARQL endpoint, allows publishing RDF data according to all the defined standards for Linked Open Data. Once the user clicks on an RDF resource (for example, a URI extracted from a SPARQL query), LodView queries the dataset looking for all the information related to that specific resource and displays the data to the user. Our installation of Lodview is available at <https://trafair.eu/lodview>.

An example of the use of LodView is shown in Figure 9. Here the information of the traffic sensor “R001_SM3” is described: the position of the sensor is visualized on a map while other information is displayed as property-object pairs. At the bottom of the figure, the inverse relationships are listed. Figure 10 shows one of these relationships, which represents the observation made by the sensor “R001_SM3” on the 6th of August 2019 from 19:00 to 20:00.

In Figure 11, a simple example of a query performed on Virtuoso is shown. This query shows the data related to one traffic sensor and contains the same data shown in Figure 9.

As of Virtuoso 7.1, several improvements have been made to integrate the support of geospatial queries. Indeed, Virtuoso can understand representations for several types of geometric objects (points, linestrings, multilinestrings, polygons, multipolygons, and geometry collections). Besides, it supports several functions with geospatial objects, increasing compliance with GeoSPARQL and OGC (Open Geospatial Consortium) standards. Figure 12 shows an example of a GeoSPARQL query, which selects the number of vehicles counted by the sensors placed in the town square of Modena on 8 January 2019.

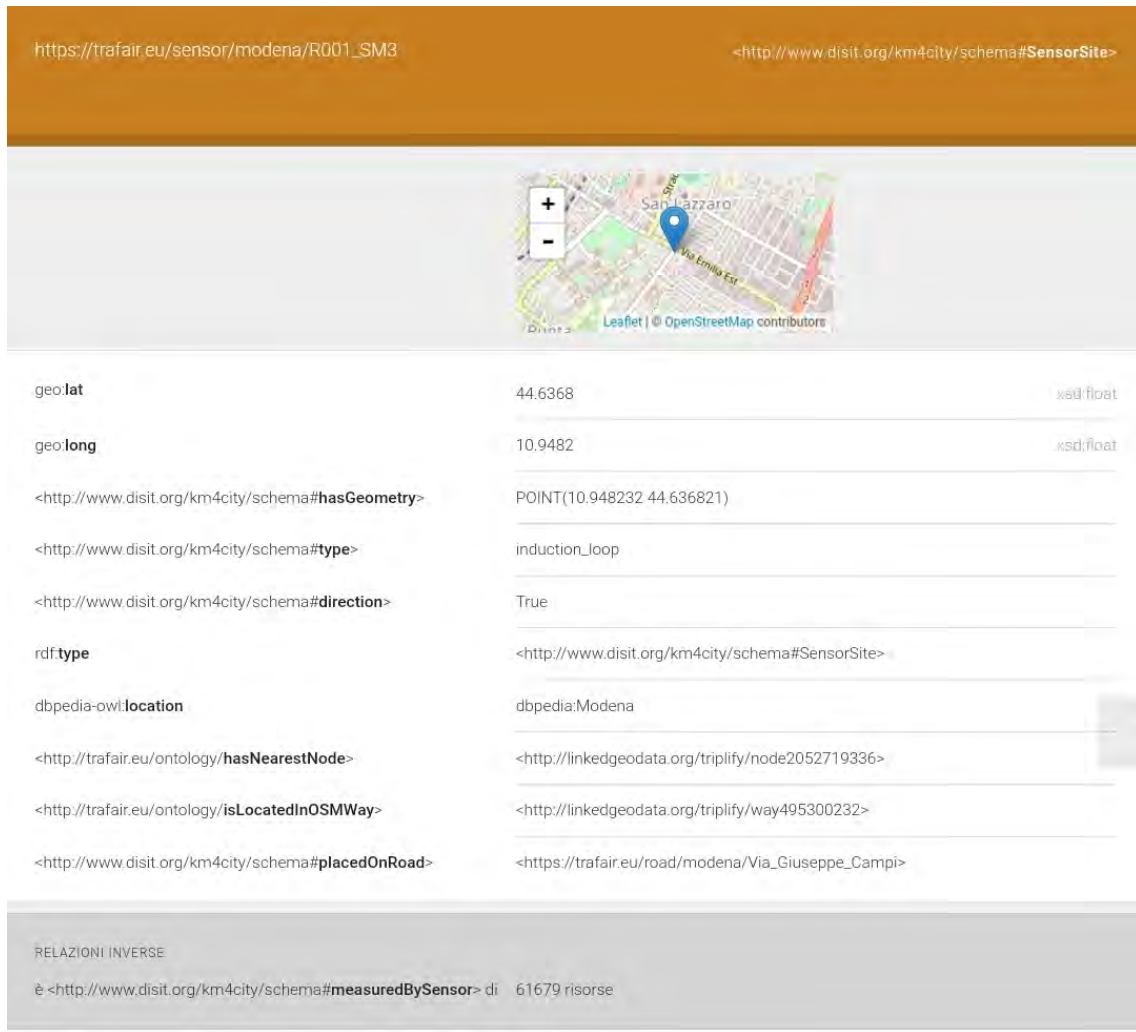


Figure 9. LodView representation of the station information of the traffic sensor “R001_SM3”.

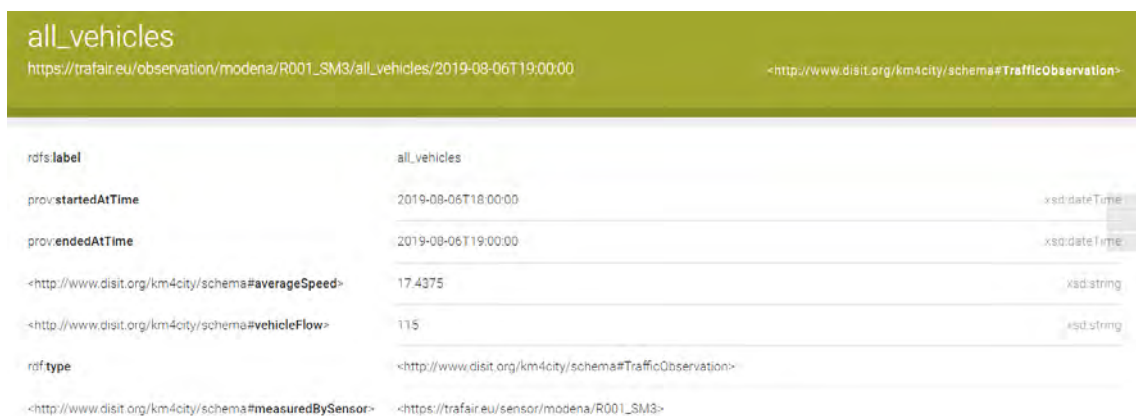


Figure 10. LodView representation of one observation made by the traffic sensor “R001_SM3”.

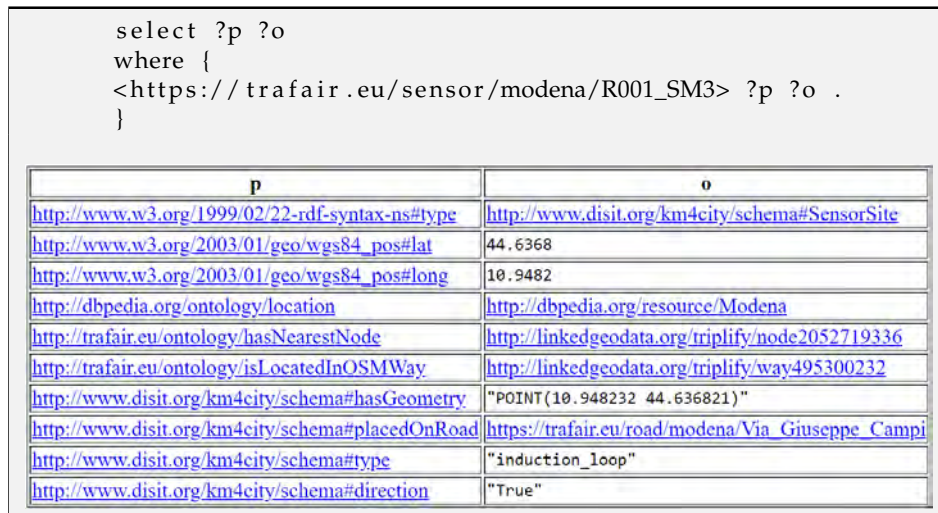


Figure 11. SPARQL query showing information related to sensor “R001_SM3”.

The reader can also find other examples of possible SPARQL queries in Appendix B. In particular, the query in Figure A2 counts how many sensors in our data store are located in the two cities, and the one in Figure A3 shows the number of vehicles counted by every sensor in our data store during a specific day. It is also possible to filter the sensors considered according to the street where they are placed. An example of this filter is shown in the query in Figure A4, where the results are ordered by the position of the sensors (longitude and latitude). Figure A5 shows an example of a GeoSPARQL query that selects how many sensors in Modena are located within the area delimited by the ring road. The last two queries mentioned show the number of vehicles counted by the sensors on 8 January 2019.

4.4. Technological Choices

In this section, the reasons for our choice of technological solutions (Karma, Virtuoso, and LodView) are justified.

As explained in Section 4.2, Karma [62,63] has been used to represent the traffic data provided by the sensors in Linked Data. Karma is a useful ETL tool that supports the publication of data in RDF format. Besides Karma, other alternatives could have been chosen to achieve this goal [75]. Thus, a variety of tools and languages can be used to convert from different data formats to RDF-like data formats, such as OpenRefine [76], RML [77], ShExML [78], YARRML [79], and SPARQL-Generate [79]. Any of these could have been used instead of Karma, without much impact on the work presented in this paper. The main reasons why Karma was chosen for this project are the following:

- Karma allows to import data from a variety of sources other than a PostgreSQL database, and therefore our approach can be exploited even if the input data are available in other types of sources.
- Karma allows to export the data model in R2RML format, which can be applied to transform a huge amount of data in RDF. Besides, the model can be easily shared with other researchers interested in our mapping to make the same transformation; the model is independent of the data sources. In [75], Karma is compared to other tools and it is the only one that supports exporting models in R2RML.
- Karma enables importing multiple ontologies in the same project. This feature is crucial in our case since a unique ontology which includes all the classes and properties needed was not available.
- Karma offers a batch mode procedure that can be exploited for automating the conversion process given the R2RML model and a set of similar data sources. Furthermore, it is able to interact with

a Virtuoso instance and directly load the RDF data into the Virtuoso instance instead of using RDF files.

```

SELECT ?s xsd:double(?long) as ?long xsd:double(?lat) as ?lat
sum(xsd:integer(?count)) as ?vehicle_count
WHERE {
?s a <http://www.disit.org/km4city/schema#SensorSite> ;
geo:lat ?lat ;
geo:long ?long .
?o <http://www.disit.org/km4city/schema#measuredBySensor> ?s ;
<http://www.disit.org/km4city/schema#vehicleFlow> ?count ;
<http://www.w3.org/ns/prov#startedAtTime> ?date ;
<http://www.w3.org/ns/prov#endedAtTime> ?enddate .
FILTER
( bif:st_intersects
( bif:st_geomfromtext
("POLYGON(
(10.919688 44.649097,
10.925297 44.650970,
10.934204 44.648047,
10.929151 44.639855,
10.918631 44.642557)
)"
), bif:st_point (xsd:double(?long), xsd:double(?lat))
)
)
FILTER (xsd:date(?date) = "2019-01-08"^^xsd:date) .
}
group by ?s xsd:double(?long) xsd:double(?lat)
order by DESC(?vehicle_count)

```

s	long	lat	vehicle_count
https://trafair.eu/sensor/modena/R118_SM117	10.9293	44.6419	107765
https://trafair.eu/sensor/modena/R118_SM118	10.9293	44.6419	107764
https://trafair.eu/sensor/modena/R018_SM33	10.9191	44.6439	10185
https://trafair.eu/sensor/modena/R023_SM41	10.9235	44.6421	9085
https://trafair.eu/sensor/modena/R017_SM32	10.919	44.6427	8377
https://trafair.eu/sensor/modena/R017_SM31	10.9193	44.643	8023
https://trafair.eu/sensor/modena/R026_S2	10.9291	44.6403	6163
https://trafair.eu/sensor/modena/R020_SM35	10.9204	44.6422	5911

Figure 12. GeoSPARQL query showing the vehicle count of some sensors located in the town square of Modena on 8 January 2019.

Similarly, Virtuoso [66,67] is being used as a SPARQL endpoint (see Section 4.3). Alternatives that could be considered include AllegroGraph [80] or RDF4J [81], as well as graph databases such as Neo4J [82], Titan [83], GraphDB [84], or Stardog [85], to cite some examples. A benchmarking framework to evaluate and compare different data management solutions for RDF and property graph data models, called LITMUS, has been proposed in [86]. Virtuoso is very popular in academia, which motivated our use in the project. Besides, according to the preliminary experimental evaluation presented in [86], where several solutions are compared (4Store, Jena, Neo4J, OrientDB, RDF3X, Sparksee, Tinker, and Virtuoso), Virtuoso achieves the best results overall (the best performance in terms of the loading time as well as regarding the cold cache execution time, and the second best one concerning the warm cache execution time). Some relevant benefits of Virtuoso include the following:

- Virtuoso is a popular tool that exposes a SPARQL endpoint for performing SPARQL queries, thus covering our fundamental need.
- Karma provides functionalities for operating with instances of Virtuoso. So, these two tools complement each other and can be easily used in conjunction.
- Virtuoso provides an open source version that is constantly being updated and improved.
- It features a backend authentication system which supports setting different privileges for different users. In this way, it is possible, for example, to block potential DELETE statements that can be sent from the Internet.

In the end, as visualization tool for RDF, in this work LodView [65] has been used (see Section 4.3). Other possible alternatives include Rhizomer [87,88], LODMilla [89,90], and LODGVis [91,92], to cite some examples. LodView was chosen because it provides a dereferenciation system that allows the users to easily explore the relation between different instances. Some relevant benefits of LodView include the following:

- It is open-source and can be easily customized.
- It provides a simple and tabular visualization that is easy to understand.
- It is able to navigate and display the resources connected through the *owl:sameAs* relation.
- It is able to navigate and display inverse relations.
- It provides a connection with LodLive [93]. Therefore, our resources can also be visualized through the online version of LodLive, since it is able to explore the resources of a remote SPARQL endpoint. By exploiting the online version of LodLive, it was not necessary to set up a personalized instance.

Notice that it is not our purpose to provide a complete survey of existing technological solutions, as this is out of the scope of this paper, but to justify and motivate our technological choices and show some potential alternatives. Other works have focused on comparing different approaches. For example, the analysis of approaches to generate RDF data from relational data has been the subject of several studies. Several surveys on RDF data storage/management approaches and technologies have been presented [94–98]. Finally, several surveys on visualization tools for RDF data have been published [99–102]; from these, the most recent survey is [102], where 77 linked data visualization tools have been analyzed.

5. Experimental Evaluation

The sensors and observations R2RML models, created with Karma, have been successfully applied over traffic data of Modena and Zaragoza. The following statistics refer to a Debian 9 machine with 32 Intel(R) Xeon(R) Silver 4108 CPU at 1.80GHz and 64GB of RAM.

For the city of Zaragoza, in these experiments, the data of the 46 static devices described in Section 3.2.1 have been considered: the information relating to those devices resulted in 506 triples. The process for generating and loading the triples took less than 1 s. Sensors' data for the city of Modena corresponded to data about 400 traffic sensors, which resulted in 4400 triples. The process for generating and loading the triples took about 5 s. A summary of the performance of the loading process of the information about sensors is provided in Table 1.

Table 1. Sensor data performance evaluation: loading time.

City	#Sensors	#Triples	Loading Time
Zaragoza	46	506	~0.75 s
Modena	400	4400	~5 s

Statistics related to the publication of traffic observations in the two cities are displayed in Table 2. Each traffic observation is transformed into a set of seven RDF triples following the approach described

in Section 4.1. The loading process is the entire process that loads the data into Virtuoso through Karma: a query is executed on the database, the extracted data is stored in a CSV file, and the file is processed by Karma, that loads the data in Virtuoso.

Each Zaragoza's traffic sensor generates one observation every hour, and therefore a total of 24 observations in a day. Considering that all the sensors are working correctly, more than 1000 observations are gathered daily. From January 2019 to December 2019, the number of observations arises to 383 thousands for a total of 2.5 million triples. The triple generation and loading process took about 1.5 min. This information is reported in the first row of Table 2.

Table 2. Observation data performance evaluation: loading time of hourly observations from January to December 2019

City	#Sensors	Period	#Observations	#Triples	Loading Time
Zaragoza	46	1 January 2019–31 December 2019	383 K	2.5 M	1.5 min
Modena	400	1 January 2019–31 December 2019	6.5 M	46 M	1 h

The situation in Modena is quite different due to a higher number of sensors and observations' rate. Indeed, in Modena, 1-min observations, which means about 1440 measurements for each sensor in a day, are gathered. In order to compare the statistics of Modena and Zaragoza, the data have been aggregated hourly. Some of the sensors in Modena collect more fine-grained data, since they can recognize up to 10 types of passing vehicles; on the other hand, not all of the sensors provide one measure per min. Therefore, about 17,500 hourly observations are gathered daily. From January to December 2019, the total number of observations arises to 6.5 million observations for a total of over 46 million of triples. This information is reported in the second row of Table 2.

Some tests on the loading process, to understand the capabilities of Virtuoso and Karma, the scalability of the loading process, and the variation of loading time in diverse configurations, have been performed. Traffic data generated in Modena, that are fine-grained and can be further aggregated, have been exploited. The loading process has been tested using different granularities (1-h aggregated observations, 15-min aggregated observations, and 1-min observations) and different window lengths, i.e., the time period of data to load in one iteration (1 day, 12 h and 3 h). 1-h granularity data is usual in some open data initiatives (e.g., the City Council of Zaragoza currently publishes pollution data with 1-h granularity and traffic and mobility indicators with granularity not smaller than 1 h); however, tests with lower granularity data have been performed in order to stress the system. The aim of these tests is to compare the loading time and performance. Selecting suitable granularities and window lengths for the traffic observations that have to be shared as Linked Open Data belongs to the public administration which owns the data. Currently, traffic data are not published by the city of Modena; therefore, these tests are executed on real data, while the data that are shared through the SPARQL endpoint are example data (random data).

As reported in Table 3, in a generic day, in Modena, about 17,500 1-h observations (first row, forth column in the table), 70,000 15-min observations (second row, forth column in the table) and 430,000 1-min observations (forth row, forth column in the table) are collected. A procedure for transforming and loading data has been deployed. This procedure divides the data in windows of 1-day length so that 365 iterations (loading stages) are needed to load the data of the whole year. In this context, hourly data were successfully handled and the time needed to upload the observations of the whole year was about 1 h ($\sim 10 \text{ s} \times 365 \text{ iterations}$), as shown in the table. Due to scalability issues with the tool used, 15-min data and 1-min data cannot be loaded by following this approach. In Table 3, a loading process is denoted by "failure" in case one iteration does not end successfully within 3 h.

The table also shows that, if we reduce the length of the temporal window to 12 h, 15-min aggregated data of the whole year can be loaded in approximately 6 h ($\sim 30 \text{ s} \times 730 \text{ iterations}$); every iteration loaded approximately 35,000 observations. Due to scalability issues, 1-min aggregated data cannot be handled when a 12-h interval is adopted; nevertheless, by considering a window length

of 3 h, every iteration has to handle approximately 54,000 observations and 1-min data can be loaded in about 36 h ($\sim 45 \text{ s} \times 2920$ iterations). For more details, please see Table 3, which summarises the loading time of each option.

Table 3. Loading process performance for 1-year data under different granularity and window length conditions.

Granularity of Data	Window Length	#Iterations Required	#Observations for Each Iteration	#Generated Triples	Loading Time of a Single Iteration	Total Time	Result
1-h data	1 day	365	17,500	122.5K	14 s (avg)	1.25 h	success
15-min data	1 day	365	70,000	490K	-	-	failure
15-min data	12 h	730	35,000	245K	30 s (avg)	6 h	success
1-min data	1 day	365	430,000	3M	-	-	failure
1-min data	12 h	730	215,000	1.5M	-	-	failure
1-min data	3 h	2920	54,000	378K	45 s (avg)	36 h	success
1-min data	1 h	8760	18,000	126K	14 s (avg)	34 h	success
1-min data	1 min	525,600	200	1400	0.375 s (avg)	55 h	success

After these tests, it can be noticed that the loading process was not able to manage more than 54,000 observations in a single step due to scalability issues. However, this fact should not be considered a big issue. In fact, in case a city council decides to publish fine-grained data (1-min or 15-min data), it is likely that this data will be shared in semi-real time, therefore with a window length even less than 3 h, so loading problems will not occur. Regarding the publication of historical data (e.g., data of the last year, or last month, every year or every month), carrying out the process with a time window of 3 h is not a limitation.

In the end, it could also be interesting to analyse the response time of the queries. Different SPARQL queries have been performed during the test phase and very fast responses have been obtained, even given the high number of observations stored in the endpoint. In particular, Table 4 contains some statistics related to the queries presented in Section 4.3 and the queries reported in Appendix B.

Table 4. SPARQL queries response time.

Query	Short Description	Response Time	Notes
Query Figure 11	Data of the sensor "R001_SM3"	300 ms	
Query Figure 12	Number of vehicles counted by sensors in Modena's square	2.6 s	GeoSpatial
Query Figure A2	Number of sensors in each city	750 ms	
Query Figure A3	Number of vehicles counted by each sensor in the datastore	26.4 s	
Query Figure A4	Number of vehicles counted by sensors on a street	1.86 s	
Query Figure A5	Number of sensors within the ring road in Modena	650 ms	GeoSpatial

This approach has proven to be very effective in publishing Linked Data. The robustness provided by OpenLink Virtuoso (open source edition) allows efficiently managing large quantities of triples. Furthermore, its ability to manage geospatial data makes it a valuable tool for our purposes. To the best of our knowledge, Virtuoso open source edition does not support the creation of RDF views over external databases, so the Karma tool has been used in our pipeline to convert relational data into RDF triples. Moreover, an instance of LodView has been adopted to obtain graphical representations of the data hosted in the Virtuoso endpoint. In the end, this completely open-source approach is well suited for handling a large amount of geospatial data and will be the base for further improvements.

6. Conclusions and Future Work

Open data have the capacity to let citizens have a better understanding of what politicians are doing; on the other hand, they also stimulate the economy by encouraging companies that use open data in their business activities. This transparency can improve public services and spur inclusive

economic development. For example, greater access to traffic data can be used to tackle sustainable mobility needs. In this paper, we have presented the mappings and tools that we have used within the context of the TRAFAIR project to model, integrate, enhance semantically, and exploit traffic data. Besides, we have evaluated the feasibility and benefits of our approach. We believe that this work represents a relevant and compelling use case concerning the collection and exploitation of semantic sensor data in real-world scenarios. Moreover, the validity of the proposed approach is not limited to the traffic sphere, as it belongs to a more generic one such as the publication of any kind of sensor data. The approach can be easily adapted and applied in different fields, especially in a smart city context. Within the TRAFAIR project, this approach will be applied also for the publication of air quality data.

Regarding the exploitation of traffic data, in the TRAFAIR project, the collected traffic is used as part of a more complex process defined to estimate and predict pollutants along different areas and road segments of a city. For this purpose, additional tools such as the Graz Lagrangian Model (GRAL) [103] and other data sources (meteorological data, data about the presence of buildings in a city, air quality data provided by official monitoring stations, etc.) are used. This represents a more advanced and complex use of traffic data, and it is out of the scope of this paper. Nevertheless, the more direct exploitation techniques described in this paper are representative of the usefulness of the availability of traffic data and its semantic annotation.

Several related future lines of research could be pursued. Specifically, we are currently tackling some challenges related to other types of sensor data relevant to TRAFAIR. In particular, we are focusing on the final output of the system that provides estimations and predictions of the concentration of pollutants in different road segments. These output data, resulting in the application of the TRAFAIR models over the different types of sensor input data collected, should be properly annotated to be published in repositories that will be collected by the European Data Portal. For this purpose, we are using the Comprehensive Knowledge Archive Network (CKAN) [104]. By using the CKAN Quality Assurance extension, we will be able to grade our CKAN site according to the five stars of openness proposed by Tim Berners Lee [105,106]. Finally, stream reasoning approaches [107] could be useful techniques for the exploitation of published real-time data and we would like to explore this in more detail.

Author Contributions: Experimentation and paper writing, F.D.; state of the art, challenge identification, paper writing and funding acquisition, S.I.; state of the art, challenge identification, paper writing and funding acquisition, L.P.; experimentation and paper writing, F.R.; state of the art, challenge identification, paper writing and funding acquisition, R.T.-L. All authors have read and agreed to the published version of the manuscript

Funding: This research has been supported by the TRAFAIR project 2017-EU-IA-0167, co-financed by the Connecting Europe Facility of the European Union. We also thank the support of the projects TIN2016-78011-C4-3-R (AEI/FEDER, UE) and the Government of Aragon (Group Reference T64_20R, COSMOS research group).

Acknowledgments: The authors appreciate the fruitful cooperation with the City of Modena, Lepida S.c.p.A., and the Council of Zaragoza (Ayuntamiento de Zaragoza), which kindly provides historical and real-time traffic sensor data. The contents of this publication are the sole responsibility of its authors and do not necessarily reflect the opinion of the European Union. A special thanks go to Veronica Molinari, who originally started experimentation on traffic sensor data publication.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AENOR	Spanish Association for Standardization and Certification
API	Application programming interface
CAOVA	Car Accident Ontology for VANETs
CKAN	Comprehensive Knowledge Archive Network
CPU	Central Processing Unit
CSV	Comma Separated Values

E/R	Entity/Relationship
EDP	European Data Portal
ETL	Extract, Transform, Load
FAIRMODE	Forum for Air quality Modelling
GPS	Global Positioning System
GRAL	Graz Lagrangian Model
HPC	High-Performance Computing
HTML	Hypertext Markup Language
IRI	Internationalized Resource Identifier
JSON	JavaScript object notation
JSP	JavaServer Pages
LOD	Linked Open Data
OGC	Open Geospatial Consortium
OSM	OpenStreetMap
PROV-O	PROV Ontology
R2RML	RDB to RDF Mapping Language
RAM	Random-Access Memory
RDB2RDF	Relational Database to RDF
RDBMS	Relational Database Management Systems
RDF	Resource Description Framework
SAREF	Smart Appliances REference
ShEx	Shape Expressions
ShExC	Shape Expressions Compact Syntax
SPARQL	SPARQL Protocol and RDF Query Language
SQL	Structured Query Language
SSN	Sensor Network Ontology
TCI	Traffic Congestion Index
TPSO	Transportation Planning Suite of Ontologies
TRAFAIR	Understanding Traffic Flows to Improve Air quality
URI	Uniform Resource Identifier
VGI	Volunteered Geographical Information
WGS84	World Geodetic System 1984
XML	Extensible Markup Language

Appendix A. Data Model

In this appendix, the data model is presented in its entirety. First, in Appendix A.1, the model using Shape Expressions (ShEx) is presented. Then, in Appendix A.2, the structure of the URIs used is described.

Appendix A.1. ShEx Data Model

The definition of the data model introduced in Section 4.1 is depicted in its entirety in Figure A1 and can be downloaded from <https://trafair.eu/trafair-model>. The ShEx schema is serialized using Shape Expressions Compact Syntax or ShExC (this is one of the possible serialization of ShEx; please, see <https://shex.io/shex-primer/> for additional information).

```

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX prov: <http://www.w3.org/ns/prov#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX km4c: <http://www.disit.org/km4city/schema#>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
PREFIX trafair: <https://trafair.eu/ontology/>

<Sensor> {
  km4c:type xsd:string ,
  geo:lat xsd:float ,
  geo:long xsd:float ,
  km4c:hasGeometry xsd:string ,
  km4c:direction xsd:boolean ,
  rdf:type km4c:SensorSite ,
  dbo:location IRI ,
  km4c:placeOnRoad @<Road> ,
  trafair:hasNearestOSMNode IRI ,
  trafair:isLocatedInOSMWay IRI
}

<Observation> {
  rdf:type km4c:TrafficObservation ,
  km4c:measuredBySensor @<Sensor> ,
  rdfs:label xsd:string ,
  prov:startedAtTime xsd:dateTime ,
  prov:endedAtTime xsd:dateTime ,
  km4c:averageSpeed xsd:string ,
  km4c:vehicleFlow xsd:string
}

<Road> {
  rdf:type km4c:Road
}

```

Figure A1. Data Model described through Shape Expressions.

Appendix A.2. Structure of the URIs Employed

The structure of the URIs implemented is the following:

- Instances of the class *km4c:Road* have the following URI structure: https://trafair.eu/road/<<city>>/<<road_name>>. It is the concatenation of the strings <https://trafair.eu/road>, the name of the city, and the road name (e.g., https://trafair.eu/road/modena/Viale_Italia).
- Instances of the class *km4c:SensorSite* have the following URI structure: https://trafair.eu/sensor/<<city>>/<<sensor_code>>. It is the concatenation of the strings <https://trafair.eu/sensor>, the name of the city, and the identifier of the sensor (e.g., <https://trafair.eu/sensor/modena/LP1>).
- Instances of the class *km4c:TrafficObservation* have the following URI structure: https://trafair.eu/observation/<<city>>/<<sensor_code>>/<<vehicle_type>>/<<end_date_of_the_observation>>. It is composed by the concatenation of the following items: the string <https://trafair.eu/observation>, the name of the city, the identifier of the sensor, the type of vehicles observed, and the timestamp indicating the ending of the observation (e.g., <https://trafair.eu/lodview/observation/modena/LP1/autobus/2019-03-04T15:00:00>).

Appendix B. Additional SPARQL Queries

In this appendix, some additional examples of SPARQL queries that illustrate how the RDF data generated can be exploited are shown.

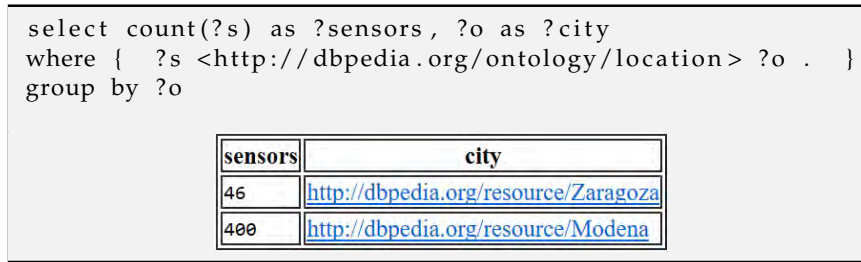


Figure A2. SPARQL query showing the number of sensors in Modena and in Zaragoza.

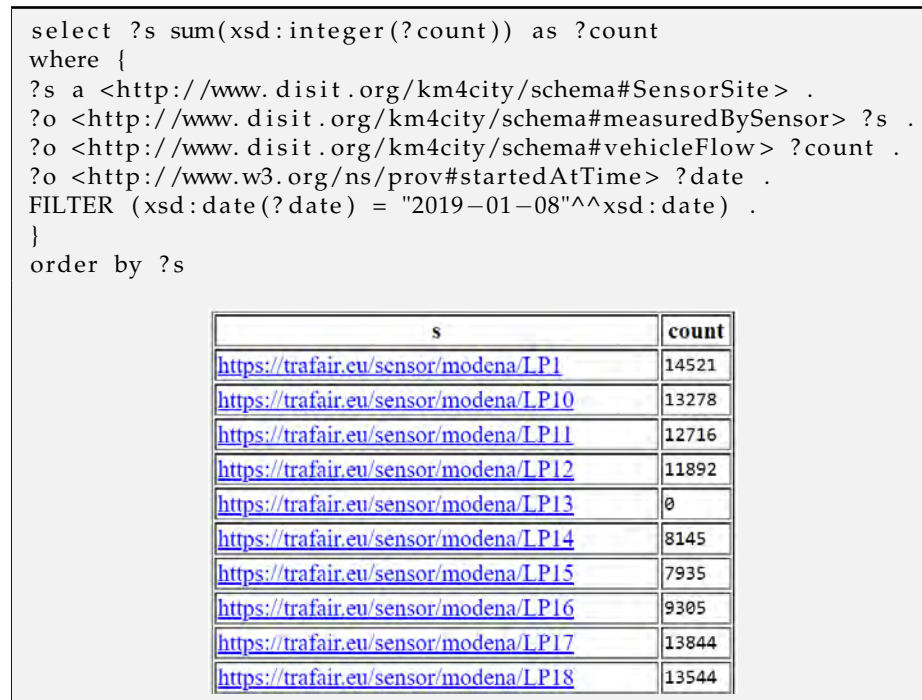


Figure A3. SPARQL query showing the number of vehicles counted by each sensor on 8 January 2019 (only a fragment of the answer is shown).

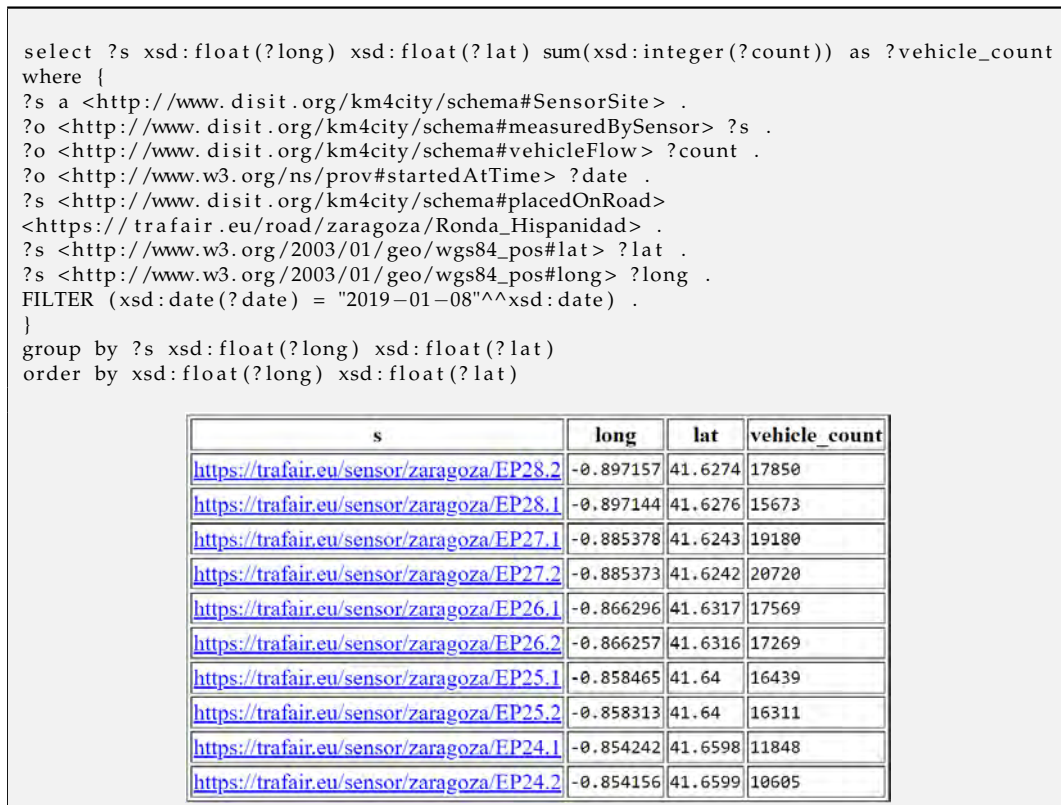


Figure A4. SPARQL query showing the list of sensors located on the street named “Ronda Hispanidad” and the number of vehicles counted by these sensors on 8 January 2019.

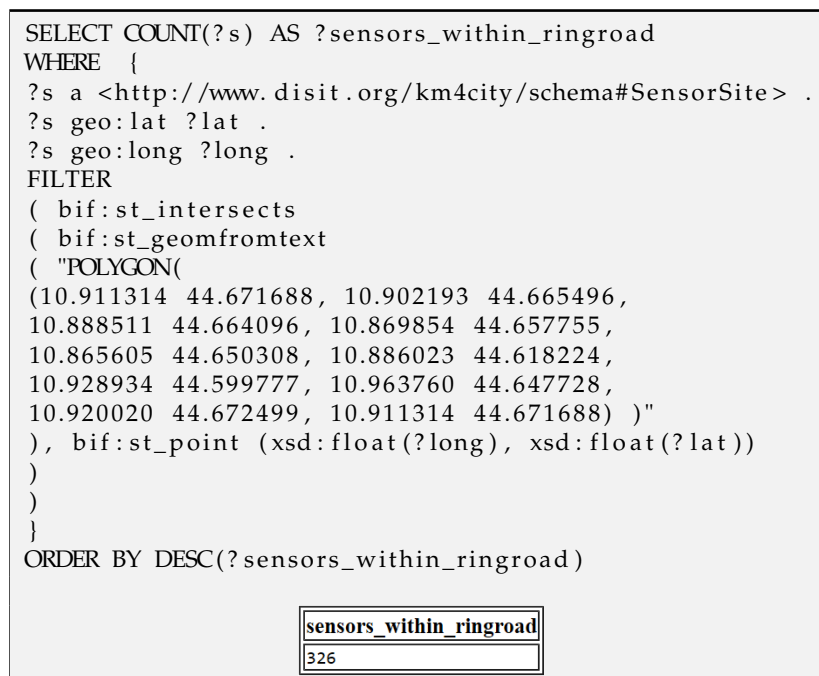


Figure A5. GeoSPARQL query showing the number of sensors in Modena located inside the area delimited by Modena’s ring road.

References

1. Open Data Charter Principles—International Open Data Charter. Available online: <https://opendatacharter.net/principles> (accessed on 2 August 2020).
2. European Union. European Data Portal. Available online: <https://www.europeandataportal.eu> (accessed on 6 June 2020).
3. Sharif, A.; Li, J.; Khalil, M.; Kumar, R.; Sharif, M.I.; Sharif, A. Internet of Things — Smart traffic management system for smart cities using Big Data analytics. In Proceedings of the 14th International Computer Conference on Wavelet Active Media Technology and Information Processing, ICCWAMTIP 2017, Chengdu, China, 15–17 December 2017; pp. 281–284. [CrossRef]
4. Colacino, V.G.; Po, L. Managing road safety through the use of linked data and heat maps. In Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics, WIMS 2017, Amantea, Italy, 19–22 June 2017; Akerkar, R., Cuzzocrea, A., Cao, J., Hacid, M., Eds.; ACM: New York, NY, USA, 2017; pp. 18:1–18:8. [CrossRef]
5. European Commission (2019) EU Road Safety Policy Framework 2021–2030. Available online: <https://ec.europa.eu/transport/sites/transport/files/legislation/swd20190283-roadsafety-vision-zero.pdf> (accessed on 6 June 2020).
6. The 2030 Agenda for Sustainable Development. Available online: <https://sustainabledevelopment.un.org/post2015/transformingourworld> (accessed on 6 June 2020).
7. Mayer, H. Air pollution in cities. *Atmos. Environ.* **1999**, *33*, 4029–4037. [CrossRef]
8. Samet, J.M. Traffic, Air Pollution, and Health. *Inhal. Toxicol.* **2007**, *19*, 1021–1027. [CrossRef] [PubMed]
9. Laña, I.; Ser, J.D.; Padró, A.; Vélez, M.; Casanova-Mateo, C. The role of local urban traffic and meteorological conditions in air pollution: A data-based case study in Madrid, Spain. *Atmos. Environ.* **2016**, *145*, 424–438. [CrossRef]
10. Curtis, L.; Rea, W.; Smith-Willis, P.; Fenyves, E.; Pan, Y. Adverse health effects of outdoor air pollutants. *Environ. Int.* **2006**, *32*, 815–830. [CrossRef] [PubMed]
11. Anenberg, S.C.; Henze, D.K.; Tinney, V.; Kinney, P.L.; Raich, W.; Fann, N.; Malley, C.S.; Roman, H.; Lamsal, L.; Duncan, B.; et al. Estimates of the Global Burden of Ambient PM_{2.5}, Ozone, and NO₂ on Asthma Incidence and Emergency Room Visits. *Environ. Health Perspect.* **2018**, *126*, 107004–1–107004–14. [CrossRef]
12. Bizer, C.; Heath, T.; Berners-Lee, T. Linked Data: Principles and State of the Art. Talk at the 17th International World Wide Web Conference W3C Track, at the WWW 2008. Available online: <https://www.w3.org/2008/Talks/WWW2008-W3CTrack-LOD.pdf> (accessed on 12 June 2020).
13. Poggi, A.; Lembo, D.; Calvanese, D.; Giacomo, G.D.; Lenzerini, M.; Rosati, R. Linking Data to Ontologies. *J. Data Semant.* **2008**, *10*, 133–173. [CrossRef]
14. Meléndez, J.A.R.; de Vyvere, B.V.; Gevaert, A.; Taelman, R.; Colpaert, P.; Verborgh, R. A Preliminary Open Data Publishing Strategy for Live Data in Flanders. In Proceedings of the Web Conference 2018, WWW 2018, Lyon, France, 23–27 April 2018; Champin, P., Gandon, F.L., Lalmas, M., Ipeirotis, P.G., Eds.; ACM: Geneva, Switzerland, 2018; pp. 1847–1853. [CrossRef]
15. Xu, X.; Sheng, Q.Z.; Zhang, L.J.; Fan, Y.; Dustdar, S. From Big Data to Big Service. *Computer* **2015**, *48*, 80–83. [CrossRef]
16. Ahmed, S.; Adnan, M.; Janssens, D.; Brattich, E.; ul Haque Yasar, A.; Kumar, P.; di Sabatino, S.; Shakshuki, E.M. Estimating pro-environmental potential for the development of mobility-based informational intervention: A data-driven algorithm. *Pers. Ubiquitous Comput.* **2018**, *23*, 653–668. [CrossRef]
17. Soriano, F.R.; Samper-Zapater, J.J.; Martinez-Dura, J.J.; Cirilo-Gimeno, R.V.; Plume, J.M. Smart Mobility Trends: Open Data and Other Tools. *IEEE Intel. Transport. Syst. Magaz.* **2018**, *10*, 6–16. [CrossRef]
18. de Vyvere, B.V.; Colpaert, P.; Mannens, E.; Verborgh, R. Open traffic lights: A strategy for publishing and preserving traffic lights data. In Proceedings of the Web Conference 2019, WWW 2019, San Francisco, CA, USA, 13–17 May 2019; Amer-Yahia, S., Mahdian, M., Goel, A., Houben, G., Lerman, K., McAuley, J.J., Baeza-Yates, R., Zia, L., Eds.; ACM: New York, NY, USA, 2019; pp. 966–971. [CrossRef]
19. Lv, M.; Chen, T.; Li, Y.; Li, Y. Urban Traffic Congestion Index Estimation With Open Ubiquitous Data. *J. Inf. Sci. Eng.* **2018**, *34*, 781–799.

20. Pollhammer, K.; Novak, T.; Raich, P.; Kastner, W.; Treytl, A.; Kovacs, G. Open traffic data platform for scenario-based control. In Proceedings of the 42nd Annual Conference of the IEEE Industrial Electronics Society, IECON 2016, Florence, Italy, 23–26 October 2016; pp. 4677–4682. [CrossRef]
21. Consoli, S.; Presutti, V.; Recupero, D.R.; Nuzzolese, A.G.; Peroni, S.; Mongiovi, M.; Gangemi, A. Producing Linked Data for Smart Cities: The Case of Catania. *Big Data Res.* **2017**, *7*, 1–15. [CrossRef]
22. Janssen, M.; Matheus, R.; Zuiderwijk, A. Big and Open Linked Data (BOLD) to Create Smart Cities and Citizens: Insights from Smart Energy and Mobility Cases. In Proceedings of the International Conference on Electronic Government, EGOV 2015, Thessaloniki, Greece, 30 August–2 September 2015; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2015; Volume 9248, pp. 79–90. [CrossRef]
23. Óscar Corcho (Ontology Engineering Group Universidad Politécnica de Madrid, L. Vocabulary to Represent Data about Traffic (Vocabulario para la representación de datos sobre tráfico). Available online: <http://vocab.linkeddata.es/datosabiertos/def/transporte/trafico> (accessed on 6 June 2020).
24. Semantic Sensor Network Ontology. W3C Recommendation, 19 October 2017. Available online: <https://www.w3.org/TR/vocab-ssn> (accessed on 12 June 2020).
25. Compton, M.; Barnaghi, P.; Bermudez, L.; García-Castro, R.; Corcho, O.; Cox, S.; Graybeal, J.; Hauswirth, M.; Henson, C.; Herzog, A.; et al. The SSN ontology of the W3C Semantic Sensor Network incubator group. *J. Web Semant.* **2012**, *17*, 25–32. [CrossRef]
26. Janowicz, K.; Haller, A.; Cox, S.J.; Phuoc, D.L.; Lefrançois, M. SOSA: A lightweight ontology for sensors, observations, samples, and actuators. *J. Web Semant.* **2019**, *56*, 1–10. [CrossRef]
27. AENOR (Spanish Association for Normalization). Available online: <https://www.aenor.com> (accessed on 6 June 2020).
28. Vera, J.; Tobarra, M.; Fernández, M.J.; Corcho, Ó.; Morlán, V. Vocabulary to Represent Data of a City Roadmap (Vocabulario para la representación de datos de un callejero). Available online: <http://vocab.linkeddata.es/datosabiertos/def/urbanismo-infraestructuras/callejero> (accessed on 6 June 2020).
29. Fernandez, S.; Hadfi, R.; Ito, T.; Marsa-Maestre, I.; Velasco, J. Ontology-Based Architecture for Intelligent Transportation Systems Using a Traffic Sensor Network. *Sensors* **2016**, *16*, 1287. [CrossRef]
30. Zhang, X.; Zhao, Y.; Liu, W. A Method for Mapping Sensor Data to SSN Ontology. *Int. J. e-Service Sci. Technol.* **2015**, *8*, 303–316. [CrossRef]
31. Open North. Open511 Specification. Available online: <http://www.open511.org> (accessed on 6 June 2020).
32. GeoNames. Available online: <https://www.geonames.org> (accessed on 6 June 2020).
33. Ahlers, D. Assessment of the Accuracy of GeoNames Gazetteer Data. In Proceedings of the Seventh Workshop on Geographic Information Retrieval, GIR 2013, Orlando, FL, USA, 5 November 2013; Association for Computing Machinery: New York, NY, USA, 2013; pp. 74–81.
34. Dardailler, D. Road Accident Ontology—Draft. Available online: <https://www.w3.org/2012/06/rao.html> (accessed on 6 June 2020).
35. Barrachina, J.; Garrido, P.; Fogue, M.; Martinez, F.J.; Cano, J.C.; Calafate, C.T.; Manzoni, P. CAOVA: A Car Accident Ontology for VANETs. In Proceedings of the IEEE Wireless Communications and Networking Conference, WCNC 2012, Paris, France, 1–4 April 2012; pp. 1864–1869.
36. Katsumi, M.; Fox, M. An Ontology-Based Standard for Transportation Planning. In Proceedings of the Joint Ontology Workshops, JOWO 2019, Graz, Austria, 23–25 September 2019; CEUR Workshop Proceedings: Aachen, Germany, 2019; Volume 2518.
37. Enterprise Integration Lab, University of Toronto. Observations Ontology. Available online: <http://ontology.eil.utoronto.ca/icity/Observations/1.0> (accessed on 6 June 2020).
38. Bellini, P.; Nesi, P.; Soderi, M. Km4City—The Knowledge Model 4 the City Smart City Ontology. 2018. Available online: <http://www.disit.org/5606>, <http://www.disit.org/km4city/schema> (accessed on 6 June 2020).
39. Villalón, M.P.; García-Castro, R. Smart Appliances REference (SAREF). Available online: <https://ontology.tno.nl/saref> (accessed on 6 June 2020).
40. Daniele, L. FIEMSER Ontology. Available online: <https://sites.google.com/site/smartappliancesproject/ontologies/fiemser-ontology> (accessed on 6 June 2020).
41. European Environmental Agency. *Air Quality in Europe—2019 Report*; Technical Report; European Environmental Agency: Copenhagen, Denmark, 2019. [CrossRef]

42. Website of TRAFair—Understanding Traffic Flows to Improve Air Quality. INEA CEF-TELECOM Project co-funded by European Union. Grant Agreement n. INEA/CEF/ICT/A2017/1566782 of 7 August 2018. Available online: <https://trafair.eu> (accessed on 6 June 2020).
43. Po, L.; Rollo, F.; Viqueira, J.R.R.; Lado, R.T.; Bigi, A.; López, J.C.; Paolucci, M.; Nesi, P. TRAFair: Understanding Traffic Flow to Improve Air Quality. In Proceedings of the 2019 IEEE International Smart Cities Conference, ISC2 2019, Casablanca, Morocco, 14–17 October 2019; pp. 36–43. [CrossRef]
44. Joint Research Centre (JRC) of the European Commission. FAIRMODE—The Forum for Air Quality Modelling in Europe. Available online: <https://fairmode.jrc.ec.europa.eu> (accessed on 6 June 2020).
45. Viqueira, J.R.R.; Villarroja, S.; Mera, D.; Taboada, J.A. Smart Environmental Data Infrastructures: Bridging the Gap between Earth Sciences and Citizens. *Appl. Sci.* **2020**, *10*, 856. [CrossRef]
46. Coleri, S.; Cheung, S.Y.; Varaiya, P. Sensor networks for monitoring traffic. In Proceedings of the Allerton Conference on Communication, Control and Computing, Monticello, IL, USA, 29 September–1 October 2004; pp. 32–40.
47. Ilari, S.; Wolfson, O.; Delot, T. Collaborative Sensing for Urban Transportation. *IEEE Data Eng. Bull.* **2014**, *37*, 3–14.
48. The PostgreSQL Global Development Group. PostgreSQL. Available online: <https://www.postgresql.org> (accessed on 6 June 2020).
49. PostGIS Team. PostGIS. Available online: <https://postgis.net> (accessed on 6 June 2020).
50. OpenStreetMap Foundation (OSMF). OpenStreetMap. Available online: <https://www.openstreetmap.org> (accessed on 22 July 2020).
51. Google. Google Maps. Available online: <http://maps.google.com/> (accessed on 22 July 2020).
52. Apple. Apple Maps. Available online: <https://maps.apple.com/> (accessed on 22 July 2020).
53. HERE Technologies. HERE. Available online: <https://www.here.com/> (accessed on 22 July 2020).
54. TomTom International BV. TomTom. Available online: <https://www.tomtom.com> (accessed on 22 July 2020).
55. Haklay, M. How Good is Volunteered Geographical Information? A Comparative Study of OpenStreetMap and Ordnance Survey Datasets. *Environ. Plan. B Plan. Des.* **2010**, *37*, 682–703. [CrossRef]
56. Camboim, S.; Bravo, J.; Sluter, C. An Investigation into the Completeness of, and the Updates to, OpenStreetMap Data in a Heterogeneous Area in Brazil. *ISPRS Int. J. Geo Inf.* **2015**, *4*, 1366–1388. [CrossRef]
57. Almendros-Jiménez, J.; Becerra-Terón, A. Analyzing the Tagging Quality of the Spanish OpenStreetMap. *ISPRS Int. J. Geo Inf.* **2018**, *7*, 323. [CrossRef]
58. Brovelli, M.; Zamboni, G. A New Method for the Assessment of Spatial Accuracy and Completeness of OpenStreetMap Building Footprints. *ISPRS Int. J. Geo Inf.* **2018**, *7*, 289. [CrossRef]
59. Alhamwi, A.; Medjroubi, W.; Vogt, T.; Agert, C. OpenStreetMap data in modelling the urban energy infrastructure: a first assessment and analysis. *Energy Procedia* **2017**, *142*, 1968–1976. [CrossRef]
60. Chen, P.P.S. The Entity-Relationship Model—Toward a Unified View of Data. *ACM Trans. Database Syst.* **1976**, *1*, 9–36. [CrossRef]
61. Elmasri, R.; Navathe, S.B. *Fundamentals of Database Systems*, 7th ed.; Pearson: London, UK, 2015.
62. University of Southern California (USC). Karma—A Data Integration Tool. Available online: <https://usc-isi-i2.github.io/karma> (accessed on 6 June 2020).
63. Gupta, S.; Szekely, P.; Knoblock, C.A.; Goel, A.; Taheriyani, M.; Muslea, M. Karma: A System for Mapping Structured Sources into the Semantic Web. In *The Semantic Web: ESWC 2012 Satellite Events*; Simperl, E., Norton, B., Mladenic, D., Della Valle, E., Fundulaki, I., Passant, A., Troncy, R., Eds.; Springer: Berlin/Heidelberg, Germany, 2015; pp. 430–434.
64. Agile Knowledge Engineering and Semantic Web (AKSW) Research Group—University of Leipzig, Institute for Applied Informatics (InfAI). The Linked GeoData Knowledge Base. Available online: <http://linkedgeodata.org> (accessed on 22 August 2020).
65. LodLive Team. LodView. Available online: <https://lodview.it/>, <https://github.com/LodLive/LodView> (accessed on 6 June 2020).
66. OpenLink Software. Virtuoso. Available online: <https://virtuoso.openlinksw.com> (accessed on 6 June 2020).
67. Erling, O.; Mikhailov, I. RDF Support in the Virtuoso DBMS. In *Studies in Computational Intelligence*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 7–24. [CrossRef]
68. W3C Semantic Web Interest Group. Basic Geo (WGS84 lat/long) Vocabulary. Available online: <https://www.w3.org/2003/01/geo> (accessed on 6 June 2020).

69. DBpedia. The DBpedia Ontology. Available online: <https://wiki.dbpedia.org/services-resources/ontology> (accessed on 6 June 2020).
70. W3C. PROV-O: The PROV Ontology. W3C Recommendation, 30 April 2013. Available online: <https://www.w3.org/TR/prov-o> (accessed on 12 June 2020).
71. Vassiliadis, P. A Survey of Extract–Transform–Load Technology. *Int. J. Data Warehous. Min.* **2009**, *5*, 1–27. [CrossRef]
72. Vassiliadis, P.; Simitsis, A.; Baikousi, E. A Taxonomy of ETL Activities. In Proceedings of the ACM Twelfth International Workshop on Data Warehousing and OLAP, DOLAP 2009, Hong Kong, China, 6 November 2009; ACM: New York, NY, USA, 2009; pp. 25–32. [CrossRef]
73. W3C. R2RML: RDB to RDF Mapping Language. W3C Recommendation. 27 September 2012. Available online: <https://www.w3.org/TR/r2rml> (accessed on 12 June 2020).
74. University of Southern California (USC). Karma RDF Generation Service. Available online: <https://github.com/usc-isi-i2/Web-Karma/tree/master/karma-web-services/web-services-rdf> (accessed on 6 June 2020).
75. Kokolaki, A.; Tzitzikas, Y. Facetize: An Interactive Tool for Cleaning and Transforming Datasets for Facilitating Exploratory Search. *arXiv* **2018**, arXiv:1812.10734.
76. Metaweb Technologies, Inc. OpenRefine. Created by Metaweb Technologies, Inc. and Originally Written and Conceived by David Huynh, OpenRefine Is Now an Open Source Project with Several Contributors. Available online: <https://openrefine.org> (accessed on 21 July 2020).
77. Anastasia Dimou and Miel Vander Sande. RDF Mapping Language (RML). W3C, Unofficial Draft 15 July 2020. Ghent University–iMinds–Multimedia Lab. Available online: <https://rml.io/specs/rml> (accessed on 21 July 2020).
78. García, H. ShExML. WESO Research Group, University of Oviedo. Available online: <http://shexml.herminio Garcia.com> (accessed on 21 July 2020).
79. Heyvaert, P.; Meester, B.D.; Dimou, A. YARRML. imec—Ghent University—IDLab. Available online: <https://rml.io/yarrml> (accessed on 21 July 2020).
80. Franz Inc. AllegroGraph. Available online: <https://allegrograph.com/products/allegrograph> (accessed on 21 July 2020).
81. Eclipse Foundation, Inc. RDF4J. Available online: <https://rdf4j.org> (accessed on 21 July 2020).
82. Neo4j, Inc. Neo4J. Available online: <https://neo4j.com> (accessed on 21 July 2020).
83. DataStax. Titan. Available online: <https://titan.thinkaurelius.com/> (accessed on 21 July 2020).
84. Ontotext. GraphDB. Available online: <http://graphdb.ontotext.com> (accessed on 21 July 2020).
85. Stardog Union. Stardog. Available online: <https://www.stardog.com> (accessed on 21 July 2020).
86. Thakkar, H.; Keswani, Y.; Dubey, M.; Lehmann, J.; Auer, S. Trying Not to Die Benchmarking: Orchestrating RDF and Graph Data Management Solution Benchmarks Using LITMUS. In Proceedings of the 13th International Conference on Semantic Systems, Semantics 2017, Amsterdam, The Netherlands, 11–14 September 2017; ACM: New York, NY, USA, 2017; pp. 120–127. [CrossRef]
87. de Lleida, U. Rhizomer. Rhizomik Initiative, GRIHO (Human–Computer Interaction and Data Integration) Research Group. Available online: <http://rhizomik.net/html/rhizomer/> (accessed on 21 July 2020).
88. Brunetti, J.M.; García, R.; Auer, S. From Overview to Facets and Pivoting for Interactive Exploration of Semantic Web Data. *Int. J. Semant. Web Inf. Syst.* **2013**, *9*, 1–20. [CrossRef]
89. Micsik, A. LODMilla. 2016. Available online: <https://github.com/dsd-sztaki-hu/LODmilla-frontend> (accessed on 22 August 2020).
90. Micsik, A.; Tóth, Z.; Turbucz, S. LODmilla: Shared Visualization of Linked Open Data. In *Theory and Practice of Digital Libraries (TPDL)—Selected Workshops*; Springer: Berlin, Germany, 2014; Volume 416, pp. 89–100. [CrossRef]
91. José Negrão. LODGVis. Available online: <https://github.com/joseolimpio/LODBrowser> (accessed on 6 June 2020).
92. Coimbra, D.B.; Negrão, J.O.M.; Durão, F.A. LODGVis: An Interactive Visualization for Linked Open Data Navigation. In Proceedings of the 25th Brazillian Symposium on Multimedia and the Web, WebMedia 2019, Rio de Janeiro, Brazil, 29 October–1 November 2019; ACM: New York, NY, USA, 2019; pp. 433–440. [CrossRef]
93. Camarda, D.V.; Mazzini, S.; Antonuccio, A. LodLive, exploring the web of data. In Proceedings of the 8th International Conference on Semantic Systems, I-SEMANTICS 2012, Graz, Austria, 5–7 September 2012.

94. Faye, D.C.; Curé, O.; Blin, G. A survey of RDF storage approaches. *Rev. Afr. Rech. Inform. Math. Appl.* **2012**, *15*, 11–35.
95. Modoni, G.E.; Sacco, M.; Terkaj, W. A survey of RDF store solutions. In Proceedings of the 2014 International Conference on Engineering, Technology and Innovation (ICE), Bergamo, Italy, 23–25 June 2014; pp. 1–7.
96. Ma, Z.; Capretz, M.A.M.; Yan, L. Storing massive Resource Description Framework (RDF) data: A survey. *Knowl. Eng. Rev.* **2016**, *31*, 391–413. [[CrossRef](#)]
97. Nitta, K.; Savnik, I. Survey of RDF Storage Managers. In Proceedings of the Sixth International Conference on Advances in Databases, Knowledge, and Data Applications (DBKDA), Chamonix, France, 20–25 April 2014; pp. 148–153.
98. Özsü, M.T. A survey of RDF data management systems. *Front. Comp. Sci.* **2016**, *10*, 418–432, [[CrossRef](#)]
99. Dadzie, A.S.; Rowe, M. Approaches to Visualising Linked Data: A Survey. *Semant. Web* **2011**, *2*, 89–124. [[CrossRef](#)]
100. Jacksi, K.; Dimililer, N.; Zeebaree, S.R.M. State of the Art Exploration Systems for Linked Data: A Review. *Int. J. Adv. Comp. Sci. Appl.* **2016**, *7*, 155–164. [[CrossRef](#)]
101. Antoniazzi, F.; Viola, F. RDF Graph Visualization Tools: A Survey. In Proceedings of the 23rd Conference of Open Innovations Association (FRUCT), Bologna, Italy, 13–16 November 2018; pp. 25–36.
102. Desimoni, F.; Po, L. Empirical evaluation of Linked Data visualization tools. *Future Gener. Comput. Syst.* **2020**, *112*, 258–282. [[CrossRef](#)]
103. Graz University of Technology. GRAL (Graz Lagrangian Model). Available online: <http://lampz.tugraz.at/~gral> (accessed on 6 June 2020).
104. CKAN Association. CKAN (Comprehensive Knowledge Archive Network). Available online: <https://ckan.org> (accessed on 6 June 2020).
105. Bauer, F.; Kaltenböck, M. *Linked Open Data: The Essentials*; Edition Mono/Monochrom; DGS: Vienna, Austria, 2012.
106. Martin, S.; Foulonneau, M.; Turki, S. 1-5 Stars: Metadata on the Openness Level of Open Data Sets in Europe. In *Communications in Computer and Information Science*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 234–245. [[CrossRef](#)]
107. Dell’Aglia, D.; Della Valle, E.; van Harmelen, F.; Bernstein, A. Stream reasoning: A survey and outlook. *Data Sci.* **2017**, *1*, 59–83. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).