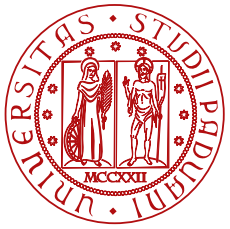


Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Master Thesis in Computer Engineering

Open Data for Italian Municipalities: Ontology, Data and WebApps

Master Candidate

Luca Martinelli

Student ID 2005837

Supervisor

Prof. Gianmaria Silvello

University of Padova

Academic Year
2021/2022

To my parents
and friends

Abstract

Sommario

Contents

List of Figures

List of Tables

List of Algorithms

List of Code Snippets

List of Acronyms

AgID Agency for Digital Italy

API Application Programming Interface

ASCII American Standard Code for Information Interchange

CSV Comma Separated Values

DBMS DataBase Management System

HTML HyperText Markup Language

HTTP HyperText Transfer Protocol

IRI International Resource Identifier

JSON JavaScript Object Notation

LOD Linked Open Data

OntoIM Ontology for Italian Municipalities

OWL Web Ontology Language

RDF Resource Description Framework

RDFa RDF in Attributes

RDFS RDF Schema

SPARQL SPARQL Protocol and RDF Query Language

SQL Structured Query Language

TSV Tab Separated Values

URI Uniform Resource Identifier

URL Uniform Resource Locator

W3C World Wide Web Consortium

XML eXtensible Markup Language

1 Introduction

1.1 The OntoIM Ontology

Ontology for Italian Municipalities (OntoIM)

1.2 Scope and organization of the thesis

2 Background

2.1 The Web of Data

The World Wide Web was originally designed to be a space where documents are connected by links without semantic value, and most of these documents are designed for humans to read, not for machines to process. For this reason, Tim Berners-Lee in 2001 introduced the idea of the Semantic Web. In particular, the Semantic Web is an enhancement of the current Web that aims to create a web of data, in which information has a well-defined meaning and can be easily read and processed by programs [BHL01].

Due to this machine-comprehensible capacity, the Semantic Web has enormous potential to automate daily tasks in our lives and is helping to advance scientific and health care fields [Fei+07], such as drug discovery and clinical research, but also in the automotive industry, in the enhancement of cultural heritage, etc. . . .[1] In this context, ontologies play a fundamental role in supporting interoperability and common understanding between different web applications and services, solving the problem of semantic heterogeneity [Tay10].

Although there are different definitions of "ontology" [Tay10], in computer science, an ontology is defined as an "explicit and formal specification of a shared conceptualization" [Gru95], where conceptualization means a simplified view of

[1] https://www.w3.org/2001/sw/sweo/public/UseCases/

the world we wish to represent. An ontology is made up of four main types of components, which are (1) *classes* (or *concepts*), which describe concepts in the domain; (2) *instances* of classes, which represent specific objects or elements of a class; (3) *properties* (or *slots*), which are used to express relationships between a first concept in the domain and a second concept in the range; (4) *axioms* (or *role restrictions*), which are used to impose constraints on the values of instances and classes [Tay10; NM+01]. In addition to the interoperability problem, ontologies are also used to satisfy the following needs:

- To share common understanding of the structure of information among people or software agents;
- To enable reuse of domain knowledge;
- To make domain assumptions explicit;
- To separate domain knowledge from the operational knowledge
- To analyze domain knowledge [NM+01].

Along with ontologies, controlled vocabularies, taxonomies, and thesauri are other resources used in different domains, in particular the medical one [IB14]. A controlled vocabulary is a closed list of named subjects, called *terms*, which is usually used for classification. A taxonomy is a subject-based classification that organizes terms in a controlled vocabulary into a hierarchy. Finally, a thesaurus extends taxonomies, allowing making other statements about the subjects and providing a much richer vocabulary [IB14].

In 2006, Tim Berners-Lee used for the first time the term Linked Data to describe the structured and interlinked data that populate the Semantic Web [Ber06]. He also introduced a set of rules, also known as the Linked Data Principles, to provide some best practices for publishing and connecting data on the Web [BHB11]. These principles, published by W3C, are the following:

1. Use URIs as names for things;
2. Use HTTP URIs so that people can look up those names;
3. When someone looks up URIs, provide useful information, using standards such as RDF and SPARQL;
4. Include links to other URIs so that they can discover more things.

The two main fundamental technologies for Linked Data are Uniform Resource Identifiers (URIs) and HyperText Transfer Protocol (HTTP). In particular, URIs are used to identify any entity that exists in the world, while HTTP provides a simple and universal mechanism for retrieving the resources to which they refer. These two technologies are integrated in RDF, which provides a graph-based data model to structure and link data that describe entities in the world [BHB11]. Using HTTPs, URIs, and RDF, Linked Data builds on the architecture of the Web, called the Web of Data. This means that the Web of Data shares many properties with the traditional Web, which are:

- Web of Data can contain any type of data;
- Anyone can publish data on the Web of Data;
- Publishers are not restricted in the choice of vocabularies used to represent the data;
- Entities are connected by RDF links [BHB11].

However, in addition to those of the traditional Web, the Web of Data also has the following characteristics:

- Data are separated from formatting and presentational aspects;
- Data is self-describing;
- Data access is simplified by the use of the HTTP and RDF standards;
- Web of Data is open, and new data sources can be discovered at run-time by following RDF links [BHB11].

Semantic Web and Linked Data are empowered by technologies developed by the World Wide Web Consortium such as RDF, OWL, serialization formats (Section 2.3), and SPARQL (Section 2.4).[2]

2.2 The five stars of Open Data

Linked Data does not have to be open and can be used internally, such as for personal data. When Linked Data is released under an open license that does not

[2] https://www.w3.org/2001/sw/wiki/Main_Page

impede its reuse for free, such as Creative Commons CC-BY[3] or the Italian Open Data License[4], we can use the term Linked Open Data (LOD) [Ber06]. In 2010 Tim Berners-Lee developed a star rating system to define and classify Linked Open Data, "in order to encourage people, especially government data owners, along the road to good linked data" [Ber06]. The star rating system assigns a star if the information is publicly available under an open license, even if the information is a photo or an image scan of a table. The more stars the information gets, the easier it will be for people (and machines) to use it [Ber06].

★ Available on the Web (any format) but with an open license to be Open Data

★★ Available as machine-readable structured data (e.g., Excel instead of an image scan of a table)

★★★ Available in a non-proprietary format (e.g., CSV instead of Excel)

★★★★ Use URIs to identify things, so that people can point at your stuff

★★★★★ Data are linked to other people's data to provide context

However, as the information receives a greater number of stars, both the benefits for consumers and the costs for the publisher increase. In particular, a five-stars data let consumers discover new data of interest, access to the data schema, reuse parts of the data, and link it to other places. They also do not have to pay for tools in order to read the data (e.g., Excel), and they can download and export the data into other formats and process them. On the other hand, to make these data available, publishers must invest time and resources in slicing and organizing the data, assigning URIs to the data items, thinking about how to represent them, linking the data with other data on the Web and making them discoverable [BK11].

2.3 RDF, OWL, AND SERIALIZATION FORMATS

The Resource Description Framework (RDF) is a W3C graph-based standard model to represent information about resources on the Web (including documents,

[3] https://creativecommons.org/
[4] https://www.dati.gov.it/content/italian-open-data-license-v20

people, physical objects, and abstract concepts). Using RDF, machines can process information on the Web using common parsers and processing tools, and information can be exchanged between different applications without losing meaning [Con+14b]. In particular, in recent years RDF has become the *de-facto* standard for publishing Linked Data on the Web. The core structure of the RDF syntax is a set of statements, called *triples*, because they consist of three elements: a *subject*, a *predicate*, and an *object*, following the structure `<subject> <predicate> <object>`, which can be visually represented in Figure 2.1 [Con+14a].

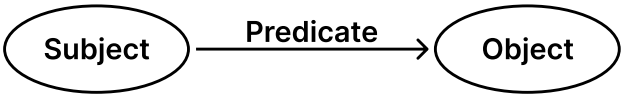


Figure 2.1: The structure of a triple, with two nodes and a predicate connecting them.

The subject and the object represent the two resources being related. The relationship that goes from the subject to the object is called *property*, and its nature is represented by the predicate. A set of statements generate a direct graph, called RDF graph, where subjects and objects are the nodes of the graph, and the predicates form the arcs. For example, the set of triples below produces the graph shown in Figure 2.2 [Con+14b].

```
<Bob> <is a> <person>.
<Bob> <is a friend of> <Alice>.
<Bob> <is born on> <the 4th of July 1990>.
<Bob> <is interested in> <the Mona Lisa>.
<the Mona Lisa> <was created by> <Leonardo da Vinci>.
<the video 'La Joconde à Washington'> <is about> <the Mona Lisa>
```

In an RDF graph, resources may be represented using an International Resource Identifier (IRI), a *literal value* or a *blank node*. An IRI is a generalization of URI, where non-ASCII characters are allowed in the IRI character string. IRIs identify resources, and can appear in all three positions of a triple. In the example above, the IRI for Leonardo Da Vinci in DBpedia[5] is `http://dbpedia.org/resource/Leonardo_da_Vinci`.

[5] `https://www.dbpedia.org/`

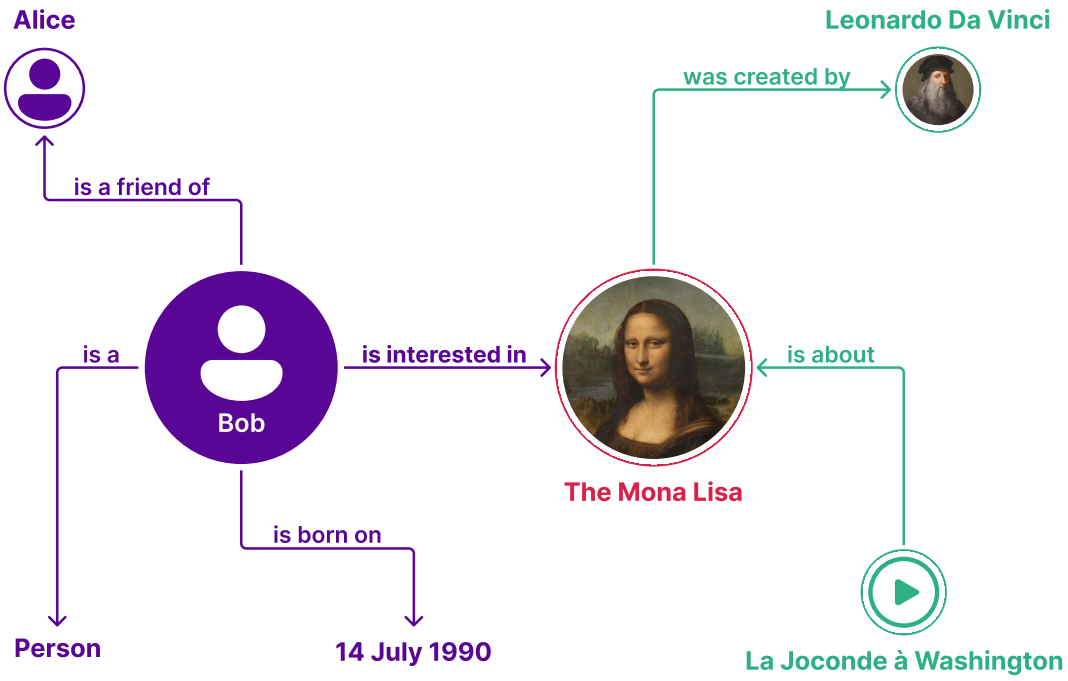


Figure 2.2: The example of the RDF graph presented by W3C.

Literals are basic values such as strings, dates, and numbers. In the RDF graph literals can only be used as objects, and consists of two or three elements, which are: (1) the value itself; (2) an IRI that identifies the *datatype* (string, number, date, etc. . .); (3) if and only if the datatype is a `rdf:langString`,[6] a *language tag* (such as `en`, `it`, `fr`, etc. . .) [Con+14a].

Finally, blank nodes can appear in the subject and object position of a triple and are used to represent resources without using a IRI [Con+14b].

In Section 2.1 ontologies and vocabularies are presented as a core element for creating the Semantic Web. The RDF data model does not provide semantic information about the resources. For this reason, RDF provides the RDF Schema (RDFS) language, that allows to define semantic characteristics of data. RDF Schema uses the notion of *class* to classify resources, while uses the *type* property to define a relation between an instance and its class. RDF Schema also allows defining type restrictions on subject and objects of particular triples through *domain* and *range* restrictions. Finally, with RDF Schema it is also possible to define hierarchies of classes and properties, using *subClassOf* and *subPropertyOf*

[6] `http://www.w3.org/1999/02/22-rdf-syntax-ns#langString`

predicates [Con+14b]. All of these modeling constructs provided by RDF Schema are summarized in Table 2.1.

Construct	Syntactic form	Description
Class	C rdf:type rdfs:Class	C is an RDF class
Property	P rdf:type rdf:Property	P is an RDF property
type	I rdf:type C	I is an instance of C
subClassOf	C1 rdfs:subClassOf C2	C1 is a subclass of C2
subPropertyOf	P1 rdfs:subPropertyOf P2	P1 is a sub-property of P2
domain	P rdfs:domain C	domain of P is C
range	P rdfs:range C	range of P is C

Table 2.1: The main modeling constructs provided by RDF Schema.

However, in 2004 the World Wide Web Consortium presented Web Ontology Language (OWL), a more complete language for publishing and sharing ontologies on the Web [Bec+04], and replaced in 2009 and then in 2012 by OWL 2. OWL 2 is a Semantic Web language to represent rich and complex knowledge about things, groups of things, and relations between things. In addition, since OWL is a computational logic-based language, the knowledge expressed in OWL can be reasoned with by computer programs either to verify the consistency of that knowledge or to make implicit knowledge explicit. A OWL document, called *ontology*, can be published in the World Wide Web and may refer to or be referred from other OWL ontologies [Hit+09]. In OWL 2 knowledge is represented by statements, called *axioms*. Axioms normally refer to objects of the world and describe them by putting them into categories or saying something about their relation. In OWL 2 objects, categories and relations are called *entities*, and in particular objects are denoted as *individuals*, categories as *classes* and relations as *properties*. Moreover, properties are further subdivided into (1) *object properties* that relate objects to objects; (2) *datatype properties* that assign data values to objects; (3) *annotation properties* that encode information about the ontology itself. Finally, names of entities can be combined into *expressions* using *constructors* to form complex descriptions from basic ones [Hit+09].

In order to publish RDF data on the Web, the RDF graphs need to be serialized. Today there are several serialization formats, but the most famous one are: N-Triples, Turtle, RDF/XML, RDFa, and JSON-LD. These formats are briefly described below, reporting as example of small excerpt of DBpedia[7] is reported.

N-Triples[8] It's one of the simplest formats, formed by sequences of RDF triples. Each statement is formed by the subject, predicate, object, and a ".", that are separated by white space.

```
<http://dbpedia.org/page/Jotaro_Kujo>
  <http://dbpedia.org/ontology/relative>
    <http://dbpedia.org/page/Joseph_Joestar> .
```

Turtle[9] It's a common data format for serializing RDF graphs that introduces some features to N-Triples language. In particular, it introduces the use of `@base` IRI and relative IRIs, `@prefix` and prefixed names, predicate lists separated by ";", object lists separated by ",", and the representation of `rdfs:type` with the token `a`.

```
@prefix dbr: <http://dbpedia.org/page/> .
@prefix dbo: <http://dbpedia.org/ontology/> .

dbr:Jotaro_Kujo dbo:relative dbr:Joseph_Joestar .
```

RDF/XML[10] Expresses RDF graphs as an XML document. The nodes and predicates are represented in XML terms: element names, attribute names, element contents and attribute values.

```
<rdf:RDF xmlns:dbr="http://dbpedia.org/page/"
  xmlns:dbo="http://dbpedia.org/ontology/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xml:base="http://www.ldf.fi/service/rdf-serializer/">
  <rdf:Description
    rdf:about="http://dbpedia.org/page/Jotaro_Kujo">
```

[7] https://www.dbpedia.org/
[8] https://www.w3.org/TR/n-triples/
[9] https://www.w3.org/TR/turtle/
[10] https://www.w3.org/TR/rdf-syntax-grammar/

```
    <dbo:relative
      rdf:resource="http://dbpedia.org/page/Joseph_Joestar"/>
  </rdf:Description>
</rdf:RDF>
```

RDF in Attributes (RDFa)[11] Provides a set of markup attributes to HTML pages to augment the visual information on the Web with machine-readable hints.

```
<body
  prefix="dbr: http://dbpedia.org/page/
    dbo: http://dbpedia.org/ontology/">
  <div about="dbr:Jotaro_Kujo">
    <div
      rel="dbo:relative"
      resource="dbr:Joseph_Joestar">
    </div>
  </div>
</body>
```

JSON-LD[12] Serializes RDF graphs into JavaScript Object Notation (JSON). The syntax is designed to easily integrate into deployed systems that already use JSON. It's intended to be a way to use Lined Data in Web-based programming environments, to build interoperable Web services, and to store Linked Data in JSON-based storage engines.

```
[
  {
    "@id": "http://dbpedia.org/page/Joseph_Joestar"
  },
  {
    "@id": "http://dbpedia.org/page/Jotaro_Kujo",
    "http://dbpedia.org/ontology/relative": [
      {
        "@id": "http://dbpedia.org/page/Joseph_Joestar"
```

[11] https://www.w3.org/TR/rdfa-primer/
[12] https://www.w3.org/TR/json-ld/

```
      }
    ]
  }
]
```

2.4 SPARQL

SPARQL is a query language developed by W3C retrieve and manipulate RDF graph content on the Web or in an RDF store. A SPARQL query contains a set of triple patterns called *basic graph pattern*. These patterns are like RDF triples except that subjects, predicates and objects may be replaced by variables. The basic graph pattern matches a subgraph of the RDF data and returns a new RDF graph in which the variables are replaced with the matched data. Queries are usually processed by an HTTP service, called *SPARQL endpoint*. [Con+13]. The example below shows a SPARQL query on DBpedia SPARQL endpoint,[13] while Table 2.2 shows its result.

```
PREFIX dbr: <http://dbpedia.org/page/>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbp: <http://dbpedia.org/property/name/>

SELECT ?relative ?name WHERE {
    dbr:Jotaro_Kujo dbo:relative ?relative .
    ?relative dbp:name ?name .
}
```

relative	name
dbr:Dio_Brando	"Dio Brando"@en
dbr:Joseph_Joestar	"Joseph Joestar"@en
dbr:Jonathan_Joestar	"Jonathan Joestar"@en

Table 2.2: A query result example from DBpedia.

SPARQL queries supports features like union of patterns, nesting queries,

[13] https://dbpedia.org/sparql

optional patterns or filtering values. Once the RDF subgraph is computed, it's also possible to modify it by ordering, limiting and grouping the values.

Another important feature of SPARQL is the possibility to perform federated queries, which explicitly delegates certain subqueries to different SPARQL endpoints, allowing to navigate through the Web of Data.

Finally, to return a more machine-readable form, SPARQL supports four common exchange formats, which are: eXtensible Markup Language (XML), JSON, Comma Separated Values (CSV), and Tab Separated Values (TSV) [Con+13].

2.5 Protégé

Protégé[14] is the most popular free and open source[15] ontology development environment.[16] The first version was developed by Mark Musen in 1987, and has been so far by a team at Stanford University [Gen+03]. The latest version, 5.5.0, has been released in March 2019, and it is written in Java, making it a cross-platform tool. In recent years, in addition to the desktop version, a web version, called WebProtégé[17] is also being developed, focused on collaborative viewing and editing.

Protégé supports creation and editing of one or more ontologies, providing a customizable graphic user interface. Among the several features available in Protégé, the most relevant are the possibility to create, rename and delete entities, add notations, merge ontologies, and more. It also includes a visualization tool for interactive navigation of ontology relationships and different reasoners.

The two most important sections for creating and editing an ontology are the "Active ontology" and "Entities" tabs. The first, that is opened by default, is designed to view and edit the information of the ontology, such as its IRI, its annotations and the imported ontologies. On the right there is also a panel that reports some metrics about the ontology, such as the total number of axioms, classes, properties, and more. Figure 2.3 shows an example of the "Active ontology" tab.

The "Entities" tab is the most important section for creating an ontology. Indeed, in this tab it is possible to manage the classes, the properties (object proper-

[14] https://protege.stanford.edu/
[15] https://github.com/protegeproject/protege
[16] https://protege.stanford.edu/shortcourse/
[17] https://webprotege.stanford.edu/

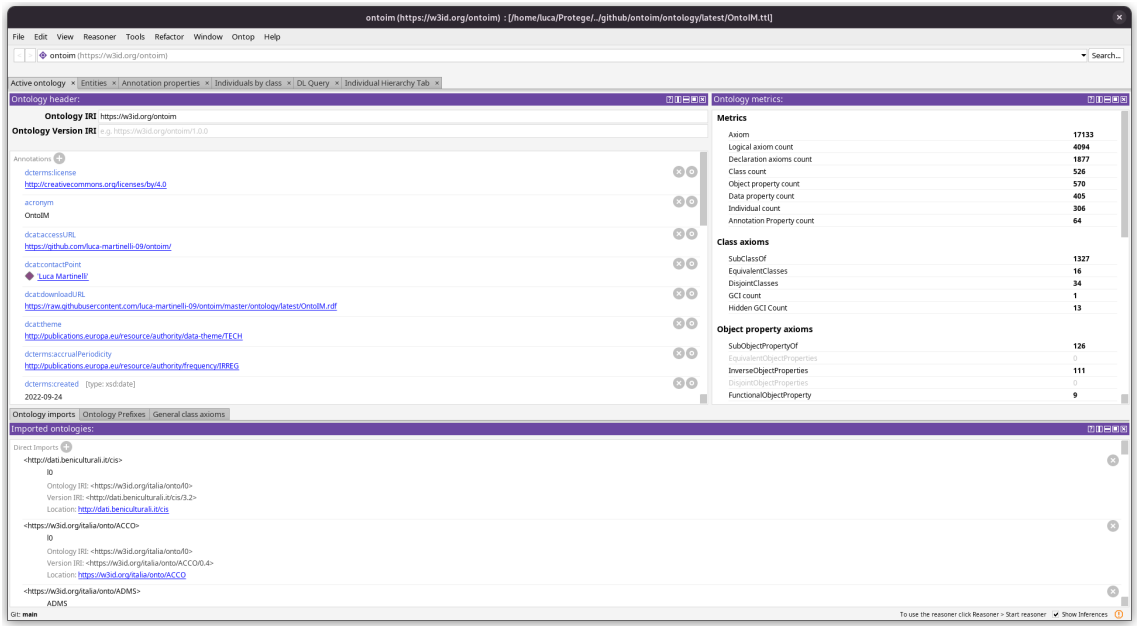


Figure 2.3: A snapshot of the Protégé "Active ontology" tab.

ties, data properties, and annotation properties), datatype and individuals. The left part provides a navigation tool to select, add and deletes entities, while the right part is focused on viewing and editing the selected entity by adding properties and axioms. Figure 2.4 shows an example of the "Entities" tab.

Of course, these were only the most relevant tools of Protégé, whose full documentation is available at `http://protegeproject.github.io/protege/`.

2.6 Virtuoso

Virtuoso Universal Server,[18] often called just Virtuoso, or OpenLink Virtuoso, at core is a high-performance object-relational SQL database. It was born in 1998 when OpenLink Software wanted to merge in a single solution its Universal Data Access Middleware and Kubl DBMS.[19]

Besides the database, Virtuoso has a built-in web server with support to Virtuoso's Web Language (VSP), and the most popular scripting languages such as PHP or ASP.NET. This same web server provides SOAP and REST access to Virtuoso stored procedures, supporting a broad set of WS* protocols. Virtuoso has also a

[18] `https://virtuoso.openlinksw.com/`
[19] `https://vos.openlinksw.com/owiki/wiki/VOS/VOSHistory`

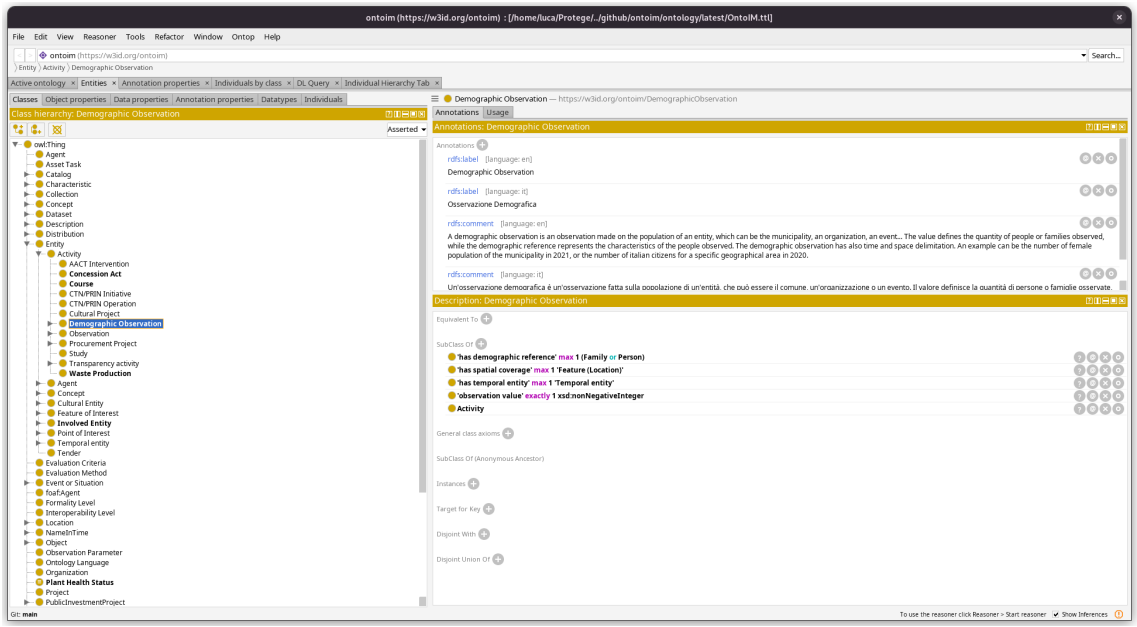


Figure 2.4: A snapshot of the Protégé "Entities" tab.

built-in WebDAV repository to host static and dynamic web content and provide versioning, making it a convenient and secure place for keeping files on the net.[20]

Since 2005, Virtuoso supports SPARQL for querying RDF data stored in its Quad Store database. In particular, it supports the HTTP-based SPARQL Protocol, SPARQL federated queries, different exchange formats such as HTML, CSV, TSV, JSON, RDF/XML, Turtle, N-Triples, and more. For this reasons Virtuoso has become the most popular and efficient tool for serving a SPARQL endpoint, which is usually located at `http://{host}/sparql`. Figure 2.5 shows an example of how the endpoint looks like.

All the aspects of a Virtuoso instance can be managed through the Virtuoso Conductor, that is located at `http://{host}/conductor`. For example, from "Linked Data" tab it is possible to add and remove RDF graphs, import schemas, declare persistent namespaces, generate statistics such as the number of classes, triples, subjects, etc. . . .

There are many methods to insert an RDF resource into the Virtuoso Quad Store. Some of them are:

Virtuoso Conductor Using Virtuoso Conductor web interface, under "Linked Data" and then "Quad Store Upload" tab it is possible to upload a RDF

[20] `https://vos.openlinksw.com/owiki/wiki/VOS/VOSIntro`

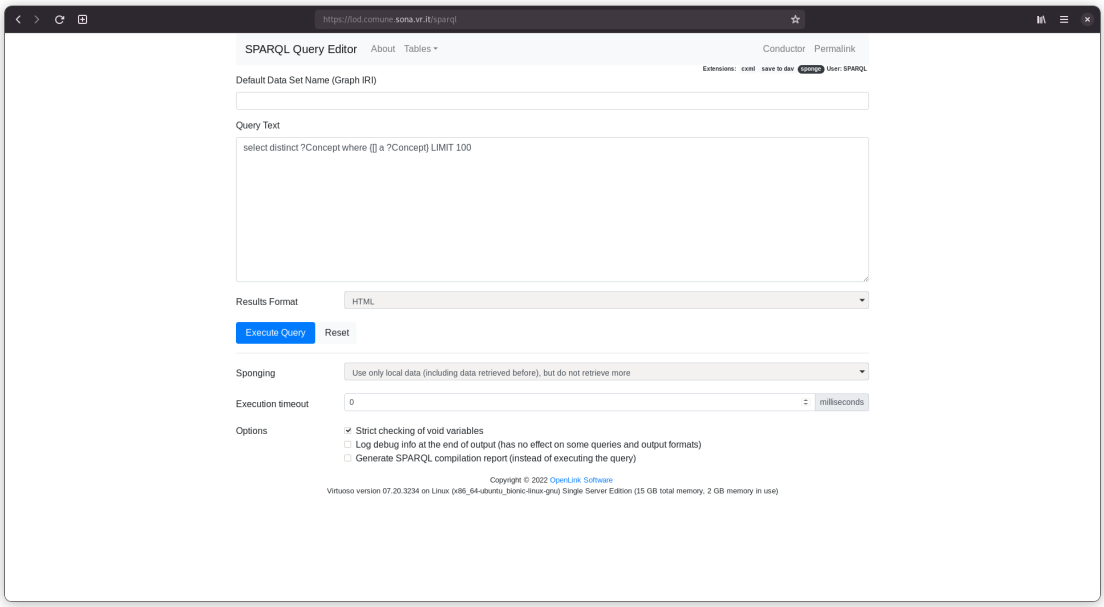


Figure 2.5: A snapshot of the Virtuoso SPARQL endpoint.

resource directly into the Virtuoso Quad Store. It is also possible to assign a graph IRI where to upload the resource. A snapshot of this feature is shown in Figure 2.6.

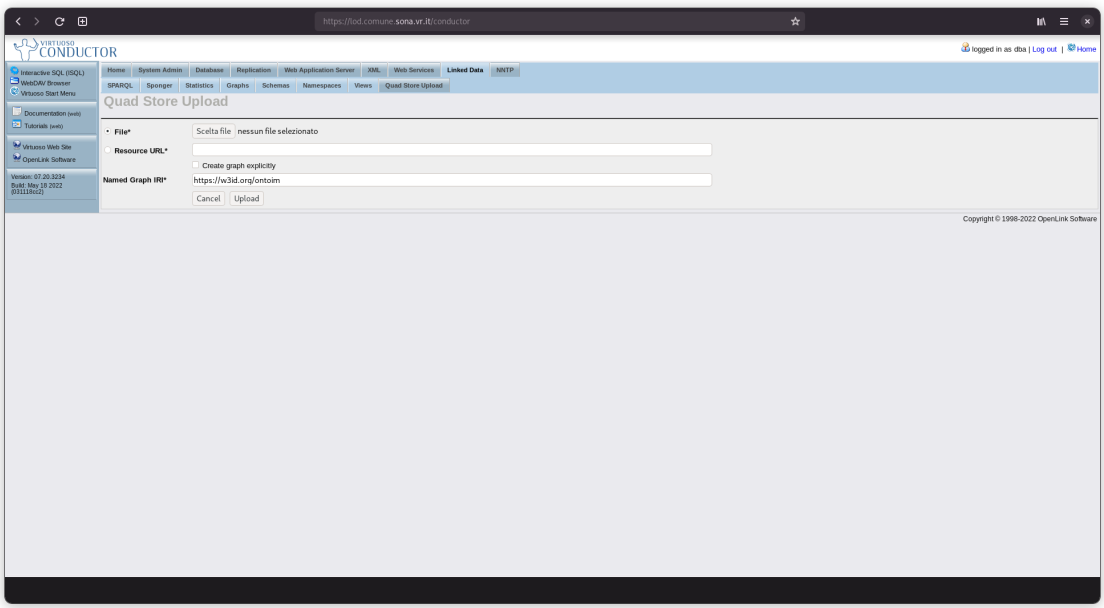


Figure 2.6: A snapshot of the "Quad Store Upload" tab.

RDF Sink Folder WebDAV supports a special folder called `rdf_sink`. This folder

can be used to upload RDF files from any WebDAV client, which are automatically uploaded to the Virtuoso Quad Store.

HTTP PUT RDF files can be uploaded to a `rdf_sink` folder through the HTTP PUT method. Using cURL, an example is:

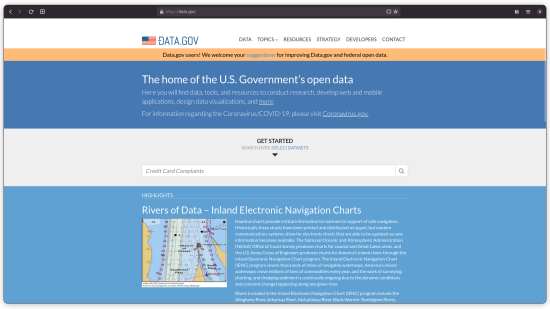
```
curl -T foaf.rdf
  http://localhost:8890/DAV/home/dba/rdf_sink/foaf.rdf
  -u dba:dba
```

HTTP POST Virtuoso supports HTTP POST method to execute SPARQL/Update language using `Content-Type: application/sparql-query` in the HTTP request headers. Using cURL, an example is:

```
curl -i -d "INSERT {
  <http://w3id.org/people/lucamartinelli>
  <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
  <http://xmlns.com/foaf/0.1/User>
}" -u "dba:dba"
-H "Content-Type: application/sparql-query"
http://localhost:8890/DAV/home/xx/yy
```

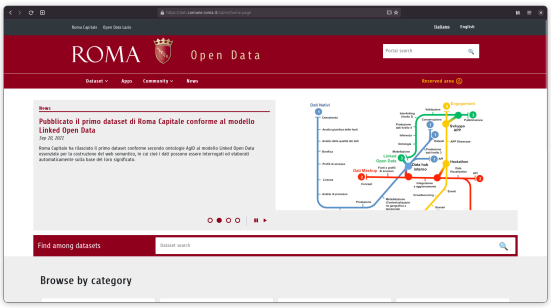
SPARQL endpoint If the user has the permission to insert graphs directly from the SPARQL endpoint, using the SPARQL/Update language, as in the example above.

These were just some features provided by Virtuoso Universal Server in order to use it as a SPARQL endpoint and RDF data store system. The full documentation on how to use Virtuoso is available at `https://vos.openlinksw.com/owiki/wiki/VOS`.



(a) U.S. Government's open data portal.[a]

[a] https://data.gov/



(b) Roma Capitale's open data portal.[a]

[a] https://dati.comune.roma.it/

Figure 2.7: Examples of CKAN open data governments portals.

2.7 CKAN

The Comprehensive Knowledge Archive Network, or CKAN[21], is an open source[22] data management system. In particular, CKAN is the world's leading tool for making open data websites, helping to manage and publish collections of data. It is mainly used by national and local governments, research institutions and other organizations who collect data. Two examples are the U.S. Government's open data portal, shown in Figure 2.7a or the Roma Capitale's open data portal, shown in Figure 2.7b.

In CKAN data is published in units called *datasets*. Each dataset is owned by an *organization* and contains information about the data like the title, publisher, data or the license; and one or more resources which are the data itself. For example, a dataset can contain different files, like the data for different years, or the same data in different formats. Any user can view, download, and search for public datasets, but there is also the possibility to restrict the access of some datasets only for registered and authorized users.

Despite the core version of CKAN has only few basic features, one of the strengths of this tool is the possibility to add different plugins which extend its functionalities and customize the user interface. The most popular plugins, developed and maintained by CKAN itself, are: (1) different tools to visualize data directly on the web page, such as tables, plots or maps; (2) DataStore extension

[21] https://ckan.org/
[22] https://github.com/ckan/ckan

that provides an *ad hoc* database for storage of structured data from resources and integrates them into CKAN API to return data in JSON format; (3) DCAT extension that includes RDF serialization of datasets and harvesters to import RDF resources into CKAN. An example of this feature can be seen in the Italian Open Data portal[23], that include the datasets from all the local governments.

2.8 OntoPiA

The only ontology OntoIM imports is OntoPiA. OntoPiA[24] is a network of ontologies and controlled vocabularies developed in 2017 by the Agency for Digital Italy (AgID)[25] and the Italian Digital Transformation Team[26] with the collaboration of research entities (CNR) and other Italian public administrations (ISTAT, Agenzia delle Entrate, Ministero della Cultura, etc. . .). OntoPiA aims to facilitate the process of data exchange between public administrations, standardize government data, and create the knowledge graph of Italian Public Administration [Dig17b; Dig17a]. Actually the network is composed by 28 ontologies and 39 controlled vocabularies. The OntoPiA ontological stack, shown in Figure 2.8, consists of the following levels:

Foundation Level It's composed by the top-level ontology *L0*, which allows all the ontologies to be linked, enabling the network of ontologies. This ontology defines a few general concepts, such as *Entity*, *Location*, *Activity*, etc. . . , which are used by the ontologies of the upper levels;

Core Level It comprehends the core ontologies, which describes concepts used by different datasets. In particular, the core level describes people, organizations, and locations;

Supporting Level The third level is composed by supporting ontologies, which describe concepts used in the other ontologies. These concepts are: time, roles, measurement units, access conditions, tickets, social media and languages;

[23] https://www.dati.gov.it/
[24] https://github.com/italia/daf-ontologie-vocabolari-controllati/
[25] https://www.agid.gov.it/
[26] https://teamdigitale.governo.it/

Domain Level The final level comprehends all the ontologies that describe specific domains such as accommodation facilities, events, public contracts, etc. . . .

In addiction, there are two metadata ontologies: (1) *DCAT-AP_IT*, an extension of DCAT,[27] and DCAT-AP[28] ontologies, that aims at facilitating the interoperability between Italian data catalogs; (2) *ADMS-AP_IT*, based on ADMS[29], it is used to add metadata to all ontologies in the OntoPiA network. Table 2.3 describes all the ontologies that are part of OntoPiA, with their URIs and prefixes.

Finally, in order to facilitate the interoperability of the data, and let ontology-based application to work properly [EMS08], `CPV-AP_IT`, `CLV-AP_IT`, `L0-AP_IT`, `POI-AP_IT`, `ACCO-AP_IT`, `Lang-AP_IT`, and `COV-AP_IT` ontologies are aligned with some common ontologies, such as FOAF[30], Org[31] or GeoSPARQL[32].

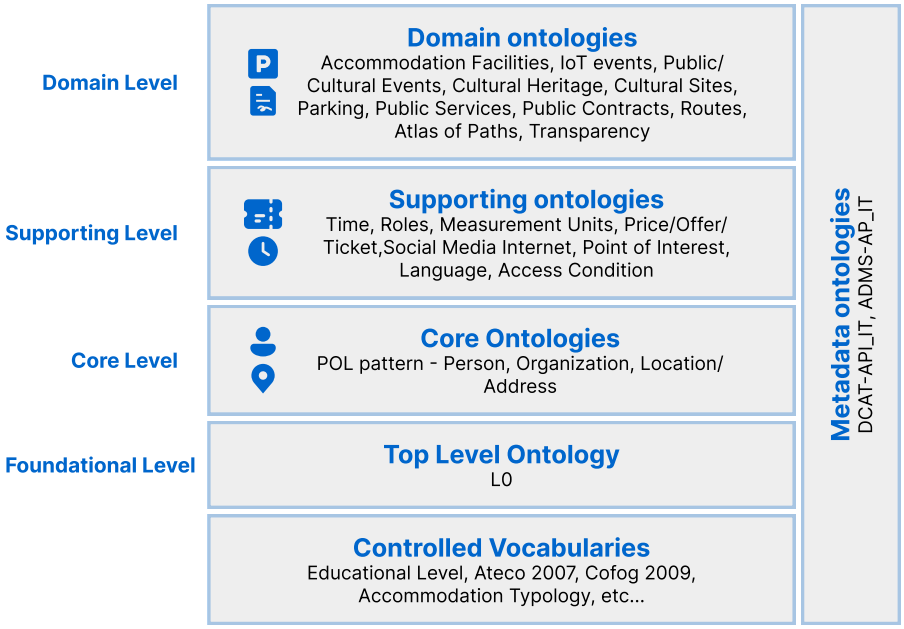


Figure 2.8: The OntoPiA ontological stack.

[27] `https://www.w3.org/TR/vocab-dcat-2/`
[28] `http://data.europa.eu/r5r/`
[29] `https://www.w3.org/TR/vocab-adms/`
[30] `http://xmlns.com/foaf/0.1`
[31] `http://www.w3.org/ns/org#`
[32] `http://www.opengis.net/ont/geosparql`

Prefix	URI	Name
Foundation Level		
l0	https://w3id.org/italia/onto/l0	Level-0
Core Level		
clvapit	https://w3id.org/italia/onto/CLV	Address (Location)
covapit	https://w3id.org/italia/onto/COV	Organization (Public or Private)
cpvapit	https://w3id.org/italia/onto/CPV	Person
Supporting Level		
acapit	https://w3id.org/italia/onto/AccessCondition	Access Conditions
langapit	https://w3id.org/italia/onto/Language	Language
muapit	https://w3id.org/italia/onto/MU	Value and Measurement Unit
poiapit	https://w3id.org/italia/onto/POI	Points of Interest
potapit	https://w3id.org/italia/onto/POT	Price/Offer/Ticket
roapit	https://w3id.org/italia/onto/RO	Role
smapit	https://w3id.org/italia/onto/SM	Social Media/Contact and Internet
tiapit	https://w3id.org/italia/onto/TI	Time
Domain Level		
accoapit	https://w3id.org/italia/onto/ACCO	Accommodation Facilities
aopapit	https://w3id.org/italia/onto/AtlasOfPaths	Atlas of Paths
chapit	https://w3id.org/italia/onto/CulturalHeritage	Cultural Heritage

Prefix	URI	Name
cis	http://dati.beniculturali.it/cis	Cultural Institute/Site and Cultural Event
cpevapit	https://w3id.org/italia/onto/CPEV	Public Events
cpsvapit	https://w3id.org/italia/onto/CPSV	Public Services
herapit	https://w3id.org/italia/onto/HER	Higher Education and Research
indicator	https://w3id.org/italia/onto/Indicator	Indicator
iotapit	https://w3id.org/italia/onto/IoT	IoT event
parkapit	https://w3id.org/italia/onto/PARK	Parking
pcapit	https://w3id.org/italia/onto/PublicContract	Public Contracts
prjapit	https://w3id.org/italia/onto/Project	Project
rtapit	https://w3id.org/italia/onto/Route	Routes
trapit	https://w3id.org/italia/onto/Transparency	Transparency Obligations
Metadata		
admsapit	https://w3id.org/italia/onto/ADMS	Asset Description Metadata Schema
dcatapit	https://w3id.org/italia/onto/DCAT	Data Catalog Vocabulary

Table 2.3: Ontologies part of the OntoPiA network.

3

Related works

As said in Chapter 1, the purpose of this thesis is to design an ontology for Italian municipalities, facilitate the publication on the Web of Linked Open Data, and develop a web application that makes this data easier for people to comprehend and visualize. It is therefore interesting to understand how major Italian, European, and global cities publish their data on the Web, and what data they publish.

The next sections will show the information collected by Italian, European and global cities about their Open Data portal, and in particular: (1) the number of available datasets; (2) the most common data file types; (3) a score from 1 to 5 based on the five stars classification presented in Section 2.2. Since the score is assigned to the entire data catalog and not to a single resource, only the types of files most present in the portal were considered.

3.1 Italian cities

For what concerns Italian cities, has been analyzed the most economically and culturally relevant cities in northern, central, and southern Italy: Bologna, Firenze, Genova, Milano, Napoli, Roma, Torino, and Venezia. The results, collected during April 2022, are shown in Table 3.1. All the cities scored three stars, since data are mostly published in non-proprietary format, in particular CSV, JSON, and Shapefile. Firenze and Bologna use API that serves the resources in different formats. Firenze's data can be accessed in JSON format or downloaded as a ZIP

archive containing the CSV file and a metadata file. Bologna, on the other hand, lets export resources in different formats, including RDF/XML, JSON-LD, N-Triples, and Turtle. However, these resources are not accessible through SPARQL, there are no semantic information, and they're not linked each other. For these reasons this catalog also obtained three stars.

City	# Datasets	File type	Score	Software
Firenze	1902	Uses API	3	Drupal + CKAN
Bologna	425	Uses API	3	OpenDataSoft
Milano	1618	CSV (1540)	3	CKAN
Torino	1954	CSV (1460)	3	CKAN
Roma	319	CSV (230)	3	CKAN
Venezia	248	CSV (179)	3	Drupal
Genova	138	CSV (111)	3	DKAN
Napoli	62	CSV (35)	3	Custom

Table 3.1: Analysis of Italian cities' Open Data Portals. The data reported in this table was collected during April 2022.

All the Italian cities analyzed, except Napoli, follows the *Linee guida nazionali per la valorizzazione del patrimonio informativo pubblico*.[1] Indeed, they provide their entire catalog as Linked Open Data using the `DCAT_AP-IT` ontology for resource metadata, like the access and download URL, the name and the file type of the resource, the owner of the dataset, the frequency of updating the data, the theme, and more. This approach aims to maintain the ease of publishing data (e.g. using the CKAN portal), but at the same time allows resources to be more accessible, provide additional information about the nature of the data, and enables the ability to access resources from regional, national,[2] and European[3] portals.

Moving on, an example of four-star data comes from Roma, which provided the list and the information of accommodation facilities[4] using the OntoPiA ontology described in Section 2.8.

[1] `https://docs.italia.it/italia/daf/lg-patrimonio-pubblico/`
[2] `https://dati.gov.it`
[3] `https://data.europa.eu`
[4] `https://dati.comune.roma.it/catalog/dataset/suar2021`

Finally, some notable attempts to publish data as Linked Open Data come from Milano and Bologna, which have respectively two portals (Roma[5], and Bologna[6]) dedicated to Linked Open Data. Milano developed a custom ontology called *OntoMI*[7] that partially extends OntoPiA, which is described in Section 2.8. In particular, the ontology describes six subject areas that represent a part of the services offered by the City of Milano: libraries, administrative acts, kindergartens, consumer price detection, sports facilities, and Area C entry detection. However, the SPARQL endpoint is no longer available, and the data can no longer be accessed. For what concerns Bologna, it also developed a custom ontology, called *Onto Municipality*[8], that describes districts, areas, streets, squares and other circulation areas, civic numbering, places and people of interest, schools, and demographic statistics. For the latter, Bologna uses an ontology developed by ISTAT as part of the 2011 census[9] that is no longer maintained and accessible. Despite the SPARQL endpoint, and the data are still accessible, the project has not been maintained since 2016, making it currently useless.

3.2 European and global cities

As for Italian cities, it is interesting to analyze the approach to Open Data (and Linked Open Data) of European and global cities. In particular, has been analyzed the political and economic capitals of major European states, the United States, Canada and Australia. The results, collected during April 2022, are shown in Table 3.2. All the cities except for Amsterdam and London scored three stars, since data are mostly published in non-proprietary format, in particular CSV, JSON, and GeoJSON. On the contrary, Amsterdam and London, despite publishing data under an open license, most resources are available only in the proprietary Excel format. Notice that Berlin, Brussels, Paris, The Hague, New York, Los Angeles, Washington DC, Melbourne, and Sydney use APIs and let export the data in different formats. Brussels, and Paris, which use the same software as Bologna, and New York, Los Angeles, and Melbourne, also allow resources to be exported in RDF format, but without a SPARQL endpoint, without semantic information,

[5] https://dati.comune.milano.it/sparql/home.html
[6] http://linkeddata.comune.bologna.it
[7] https://dati.comune.milano.it/sparql/onthdoc.html
[8] http://linkeddata.comune.bologna.it/ontologies/2014/04/onto-municipality/
[9] https://www.istat.it/it/archivio/160039

and without links between data, so the same considerations about Bologna made in Section 3.1 apply.

City	# Datasets	File type	Score	Software
European cities				
Berlin	2470	Uses API	3	Drupal + CKAN
London	1047	XLSX (644)	2	DataPress
Zurich	683	CSV (463)	3	Custom
Vienna	560	CSV (477)	3	CKAN
Brussels	550	Uses API	3	OpenDataSoft
Barcelona	525	CSV (471)	3	CKAN
Lisbon	359	GeoJSON (206)	3	CKAN
Prague	354	CSV (194)	3	CKAN
Amsterdam	327	XLSX (n.d.)	2	Custom
Paris	321	Uses API	3	OpenDataSoft
The Hague	308	Uses API	3	Dataplatform
Madrid	195	JSON (138)	3	CKAN
Munich	176	CSV (175)	3	CKAN
Global cities				
New York	3541	Uses API	3	Socrata
Los Angeles	1635	Uses API	3	Socrata
Washington DC	1333	Uses API	3	ArcGIS Hub
Toronto	425	CSV (175)	3	WordPress + CKAN
Montreal	320	CSV (227)	3	CKAN
Melbourne	221	Uses API	3	Socrata
Sydney	176	Uses API	3	ArcGIS Hub

Table 3.2: Analysis of European and Global cities' Open Data Portals. The data reported in this table was collected during April 2022.

As for Italian cities, all cities belonging to European Union provides their catalog as Linked Open Data using the `DCAT_AP` ontology for metadata, in order to make the resources accessible through the European portal.

A similar approach applies to U.S., Canadian and Australian cities, whose catalog of data is collected using the local government API and is made available also in the central government data portal.

To conclude the analysis on Italian, European and global cities' approach to Open Data, we can definitely see that no cities publish Linked Open Data, but they prefer publish resources in using non-proprietary (and in some cases proprietary) format, reaching a score of three or fewer stars. This is probably due to the fact that convert and publishing data as Linked Open Data has a greater cost in terms of time and economic resources [BK11]. The examples of Milano and Bologna, which have stopped investing in Linked Open Data, are proof of this. However, these costs can be covered by states, ministries, government institutions, or regions, which instead publish a portion of their data as Linked Open Data. Some examples are the Europeana project,[10], Ministero della Cultura,[11] ISPRA,[12] Regione Veneto,[13] or Regione Sicilia.[14]

Of interest is the approach of Italian and EU cities in publishing the catalog in RDF format using the DCAT metadata profile, which allows them to provide some semantic information to the datasets, such as the owner of the data, the frequency of update, or the topic to which the data refer.

Finally, it is also interesting to analyze the choices of different cities regarding the software chosen to publish Open Data. The most popular tool is CKAN (14 out of 28 cities), especially for Italian and European cities (12 out of 21 cities) use CKAN for their data portal. The reasons for this choice certainly lie in the potential offered by CKAN, which, as explained in Section 2.7, is highly customizable and expandable with plugins, is an open source program and easily installed, and offers the possibility of sharing the catalog in RDF with the DCAT metadata profile, facilitating the interoperability with the central governments.

[10] https://www.europeana.eu
[11] https://dati.cultura.gov.it/
[12] http://dati.isprambiente.it/
[13] https://www.culturaveneto.it/it/
[14] https://dati.regione.sicilia.it/i-linked-open-data-nel-catalogo-regionale/

4

Requirements analysis

As introduced in Chapter 1, this thesis aims to facilitate the publication and dissemination of Open Data, and in particular Linked Open Data, by Italian municipalities by designing and developing an ontology to describe the data, and to develop a web application to make it usable for local government, citizens and businesses to consult the data. In particular, the ontology and the web applications are designed taking in consideration the needs and the data of the Comune di Sona[1], as part of the *Innovation Lab*[2] project, a project financed by Regione Veneto that aims to spread digital and Open Data culture.

One of the best practices in designing an ontology is to reuse, where possible, existing ontologies [NM+01]. Following this principle, the OntoIM ontology imports the ontologies of the OntoPiA network. As described in Section 2.8, OntoPiA is maintained by AgID and the Italian Digital Transformation Team, and aims to describe different domains of the Italian public administrations, and in particular: people, public and private organizations, addresses and locations, point of interests, accommodation facilities, paths, cultural heritage, cultural events, public services, parking, public contracts, transparency obligations, projects, routes, IoT events, indicators, and higher education and research. Where possible, these ontologies are also aligned with existing ontologies on the web.

The OntoIM ontology was therefore designed and developed as an extension

[1] `https://comune.sona.vr.it/`
[2] `https://innovationlab.regione.veneto.it/`

of the existing OntoPiA ontology, and with the aim that it would become an integral part of the network.

The first part of the design phase involved not only analyzing the data provided by the Comune di Sona, but also analyzing which data the major Italian cities share on their Open Data portals. This choice is due to the fact that we want to create an ontology that can also be reused by other administrations, and takes into account possible future extensions. The work described in Chapter 3, therefore, served not only to analyze how data are made public by various cities, but also what data is available. In addition, some data have been collected from Italian government portals or public agencies, such as Camera di Commercio, ISTAT or Agenzia delle Entrate, to have uniformly structured data across cities.

The data collected comprehends: private organizations, associations, municipal offices, events, cultural heritage, point of interests, accommodation facilities, street directory, traffic and road accidents, municipal heritage, concession acts, waste production, schools and courses organized by private organizations, and demographic statistics (which also includes statistics on tourism, association members, students, and event attendance). In addition to these requests, to make an ontology that is adaptable to other municipalities as well, the census of plants, green areas and street signs, and hospitals were added. Of course, since this is Government Open Data, the privacy of organizations and citizens must also be guaranteed.

Once the data were collected, it was necessary to understand how well OntoPiA ontologies could describe the areas involved. After that, we proceeded to design the ontology by adding the missing classes and properties, and going on to modify the existing ones where necessary.

5

Description of the OntoIM Ontology

The next two sections will describe more in details the OntoIM ontology, designed to describe the semantic areas presented in Chapter 4. In particular, Section 5.1 presents the choices made in developing the ontology, while Section 5.2 will describe the main semantic areas and will present the principal classes in each of them.

5.1 Overall design principles

The design of the ontology started analyzing the data collected from Comune di Sona, and public agencies. Table 5.1 shows what data were collected and where they were collected from.

As said in Chapter 4, the best practice of using existing ontologies where possible was followed. The next step then was to figure out which areas were already described by OntoPiA ontologies and which, instead, needed to be created or imported. The new classes created, moreover, following the design principles of OntoPiA, are subclasses of others existing in OntoPiA ontologies and, in particular, the top-level ontology `LO`. Indeed, as said in Section 2.8, this ontology allows all the ontologies to be linked, enabling the network of ontologies.

The first version of OntoIM ontology is composed of 526 classes, 570 object properties, 405 data properties. The URI of the ontology, the controlled vocabularies and the resources are secure and permanent by using the W3 Permanent Identifier Community Group, which let create permanent Uniform Resource Lo-

Data	Source
Demographic statistics (citizens by location and year, citizenship of foreigners, statistics on names and surnames)	Comune di Sona
Associations	Comune di Sona
Civil status events (births, deaths, emigrations, immigrations, marriages, civil unions, divorces)	Comune di Sona
Concession acts	Comune di Sona
Cultural events	Comune di Sona
List of majors	Comune di Sona
Municipal heritage	Comune di Sona
Museums and cultural heritage	Comune di Sona
Point of Interests	Comune di Sona
Popular University (courses and subscribers)	Comune di Sona
Traffic observations	Local police
Accommodation facilities	Regione Veneto and Comune di Sona
Tourism (arrivals and presences by nationality/region)	Regione Veneto
Private organizations	Camera di Commercio
Addresses and civic numbers	Agenzia delle Entrate
Municipal offices	IPA (AgID)
Waste production	ISPRA
Road accidents	ISTAT
Schools	Ministero dell'Istruzione

Table 5.1: The data collected as reference for designing the OntoIM ontology, and their source.

cators (URLs) that redirects to defined locations on the Web. Moreover, thanks to this service it was possible to implement a content negotiation mechanisms, to return serialized resources and ontologies in different formats (such as RDF/XML or Turtle), or its visualization/documentation, depending on the request.

The persistent URI, and the full documentation of OntoIM ontology is available at `https://w3id.org/ontoim`.

Finally, all the files, and the documentation of the ontologies and the controlled

vocabularies are Open Source and available on a GitHub repository[1], which also allows for a permanent location to place the serialization and documentation of the ontology and the other resources.

5.1.1 Semantic areas

5.1.2 Controlled vocabularies

5.2 Area-by-Area

[1] https://github.com/luca-martinelli-09/ontoim

6

Ontology Development and Data Mapper

7 Web Applications

8

Conclusions and Future Works

References

[Gru95] Thomas R Gruber. "Toward principles for the design of ontologies used for knowledge sharing?" In: *International journal of human-computer studies* 43.5-6 (1995), pp. 907–928.

[BHL01] Tim Berners-Lee, James Hendler, and Ora Lassila. "The semantic web". In: *Scientific american* 284.5 (2001), pp. 34–43.

[NM+01] Natalya F Noy, Deborah L McGuinness, et al. *Ontology development 101: A guide to creating your first ontology*. 2001.

[Gen+03] John H Gennari et al. "The evolution of Protégé: an environment for knowledge-based systems development". In: *International Journal of Human-computer studies* 58.1 (2003), pp. 89–123.

[Bec+04] Sean Bechhofer et al. "OWL web ontology language reference". In: *W3C recommendation* 10.2 (2004), pp. 1–53.

[Ber06] Tim Berners-Lee. "Linked Data - Design Issues". In: (July 2006). URL: `http://www.w3.org/DesignIssues/LinkedData.html`.

[Fei+07] Lee Feigenbaum et al. "The semantic web in action". In: *Scientific American* 297.6 (2007), pp. 90–97.

[EMS08] Jérôme Euzenat, Adrian Mocan, and François Scharffe. "Ontology alignments". In: *Ontology Management*. Springer, 2008, pp. 177–206.

[Hit+09] Pascal Hitzler et al. "OWL 2 web ontology language primer". In: *W3C recommendation* 27.1 (2009), p. 123.

[Tay10] Mohammad Mustafa Taye. "Understanding semantic web and ontologies: Theory and applications". In: *arXiv preprint arXiv:1006.4567* (2010).

[BK11] Florian Bauer and Martin Kaltenböck. "Linked open data: The essentials". In: *Edition mono/monochrom, Vienna* 710 (2011).

[BHB11] Christian Bizer, Tom Heath, and Tim Berners-Lee. "Linked data: The story so far". In: *Semantic services, interoperability and web applications: emerging concepts*. IGI global, 2011, pp. 205–227.

[Con+13] World Wide Web Consortium et al. "SPARQL 1.1 Overview". In: (Mar. 2013). URL: https://www.w3.org/TR/sparql11-overview/.

[Con+14a] World Wide Web Consortium et al. *RDF 1.1 Concepts and Abstract Syntax*. Feb. 2014. URL: https://www.w3.org/TR/rdf11-concepts/.

[Con+14b] World Wide Web Consortium et al. "RDF 1.1 Primer". In: (June 2014). URL: https://www.w3.org/TR/rdf11-primer/.

[IB14] Mirjana Ivanović and Zoran Budimac. "An overview of ontologies and data resources in medical domains". In: *Expert Systems with Applications* 41.11 (2014), pp. 5158–5166.

[Dig17a] Agenzia per l'Italia Digitale. "Linee guida nazionali per la valorizzazione del patrimonio informativo pubblico". In: (2017). URL: https://docs.italia.it/italia/daf/lg-patrimonio-pubblico/.

[Dig17b] Agenzia per l'Italia Digitale. "Piano triennale per l'informatica nella Pubblica amministrazione 2017-2019". In: (2017). URL: https://docs.italia.it/italia/piano-triennale-ict/pianotriennale-ict-doc/it/2017-2019/.

Acknowledgments

No thanks