Ontology Construction and Knowledge Base Feeding and Cleaning for Smart-city Services

Pierfrancesco Bellini, Paolo Nesi, Nadia Rauch

Dipartimento di Ingegneria dell'Informazione, DINFO
Università degli Studi di Firenze
Via S. Marta 3, 50139, Firenze, Italy
Tel: +39-055-4796567, fax: +39-055-4796363
DISIT Lab
http://www.disit.dinfo.unifi.it *alias* http://www.disit.org
{pierfrancesco.bellini, paolo.nesi, nadia.rauch}@unifi.it

Smart-cities

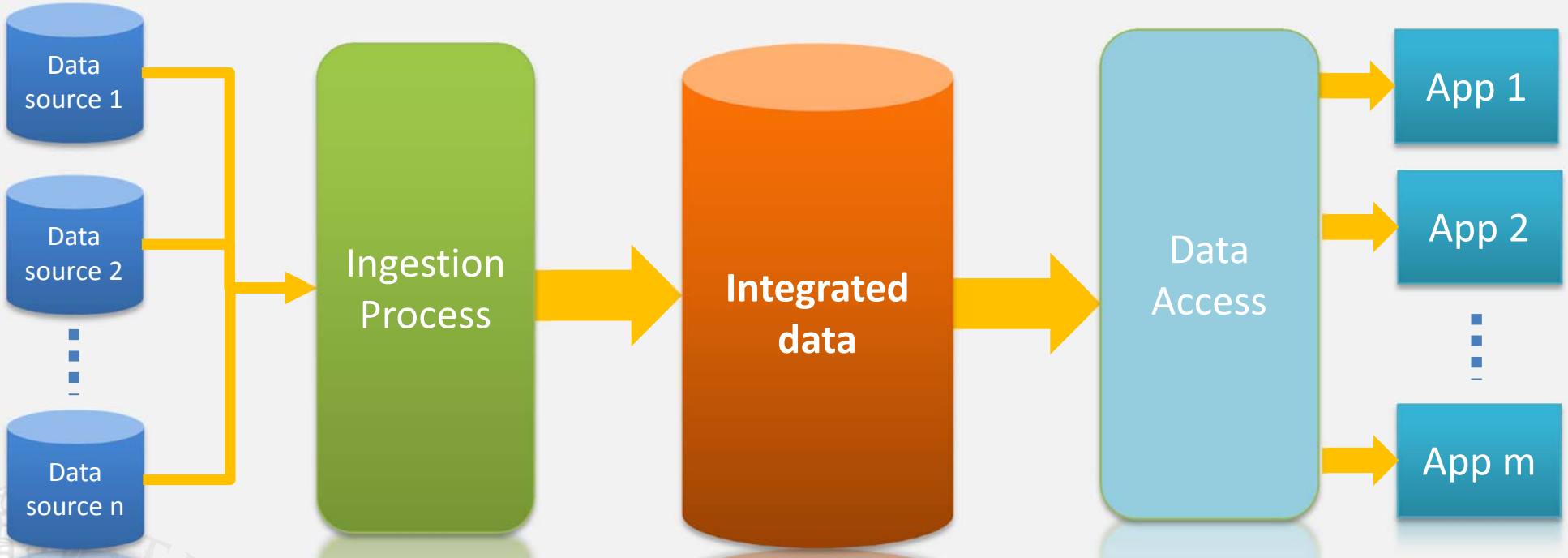
- Cities produce a HUGE amount of data every day
  - **'Static' data**
    - Road graph
    - Bus/train graph
    - Services
    - …
  - **Dynamic data**
    - Weather conditions
    - Traffic conditions
    - Pollution status
    - Bus/train positions
    - Parking status
    - …
  - **Open/Private Data**

Smart-cities

- Aim
  - Provide a platform able to ingest and manage all these data and provide services to applications that exploit these information to provide services to citizens
    - Is there any parking near here?
  - Leverage on the ongoing Semantic Web effort
- Problems & Challanges
  - Data provided in many different formats and protocols and from different institutions!
  - Data not aligned (e.g. street names)
  - BIG DATA!

Smart-city services

- Get all the data and make it available for applications to build services for the citizens

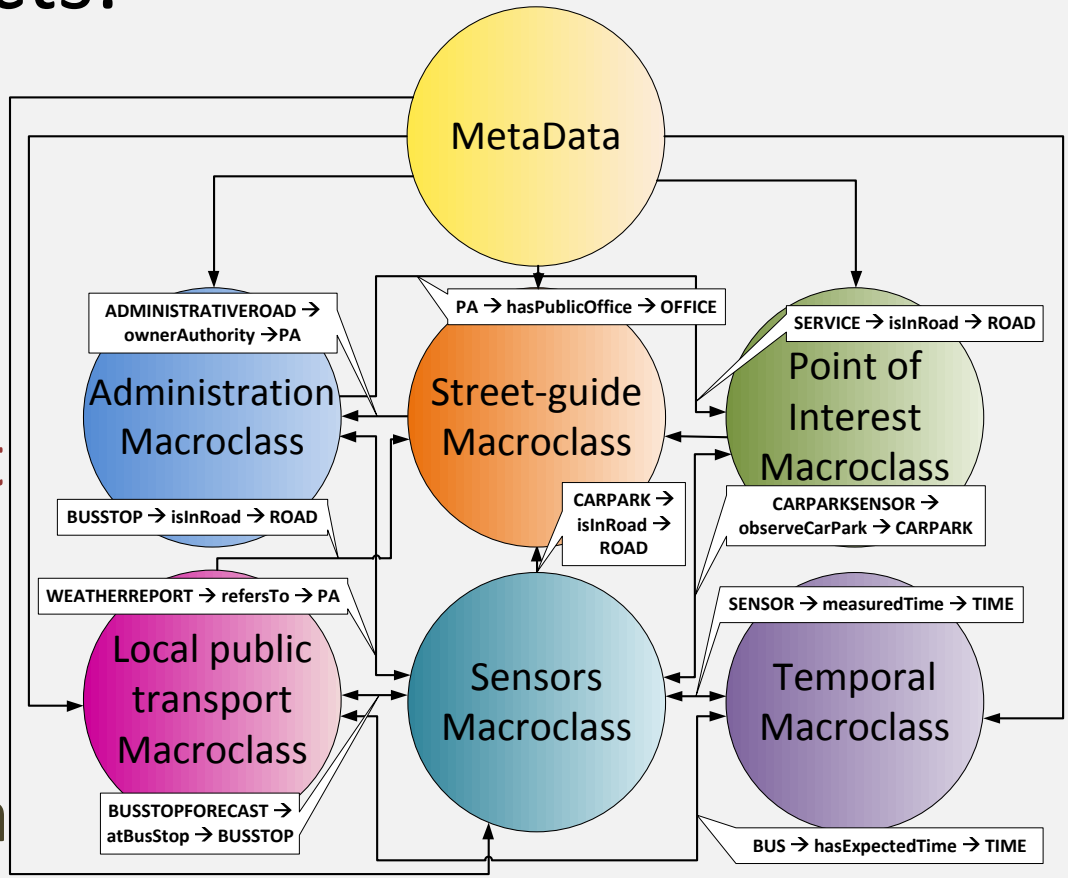


Using SemanticWeb standards

- **W3C Resource Description Framework (RDF)**
  - for the representation of data
  - All represented as <subject> <predicate> <object> graph
  - Everything identified by URIs
- **W3C OWL Ontology**
  - for the description of information
  - Similar to UML + inference (based on Description Logic)
- **W3C SPARQL query language**
  - to access to the information available
  - Graph matching based language

Smart-city Ontology

- The data provided mapped to the ontology, it covers different aspects:
  - Administration
  - Street-guide
  - Points of interest
  - Local public transport
  - Sensors
  - Temporal aspects
  - Metadata on the data



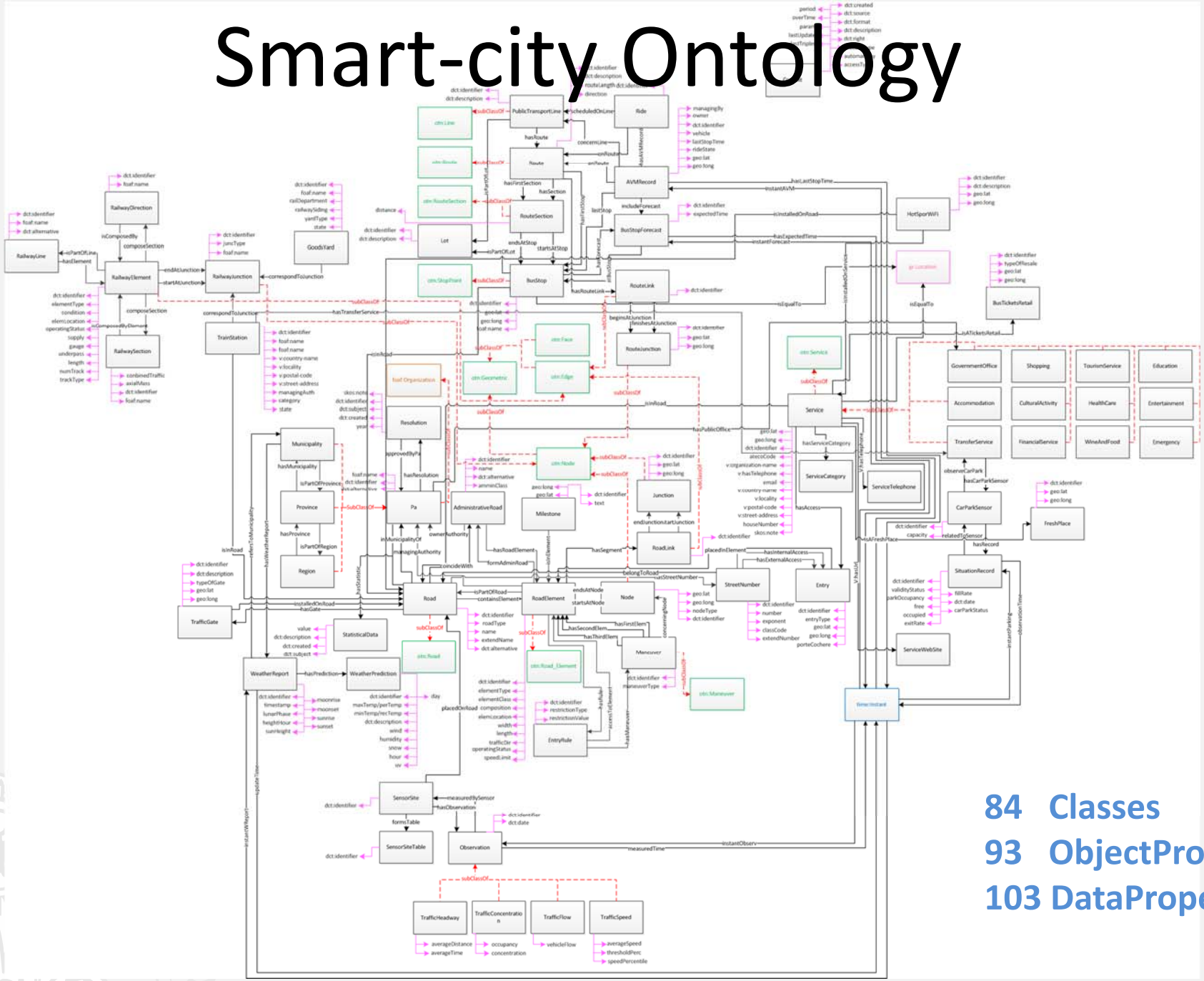
Smart-city Ontology

- ***Administration:*** structure of the general public administrations (*Municipality*, *Province* and *Region*) also includes *Resolutions* (represents the ordinance issued by administrations that may change the viability)
- ***Street-guide:*** formed by entities as *Road*, *Node*, *RoadElement*, *AdministrativeRoad*, *Milestone*, *StreetNumber*, *RoadLink*, *Junction*, *Entry*, *EntryRule*, *Maneuver*,… represents the entire road system of the region, including the permitted maneuvers and the rules of access to the limited traffic zones. Based on OTN (Ontology of Transportation Networks) vocabulary
- ***Points of Interest*****:** includes all *Services*, activities, which may be useful to the citizen and who may have the need to search for and to arrive at.

Smart-city Ontology

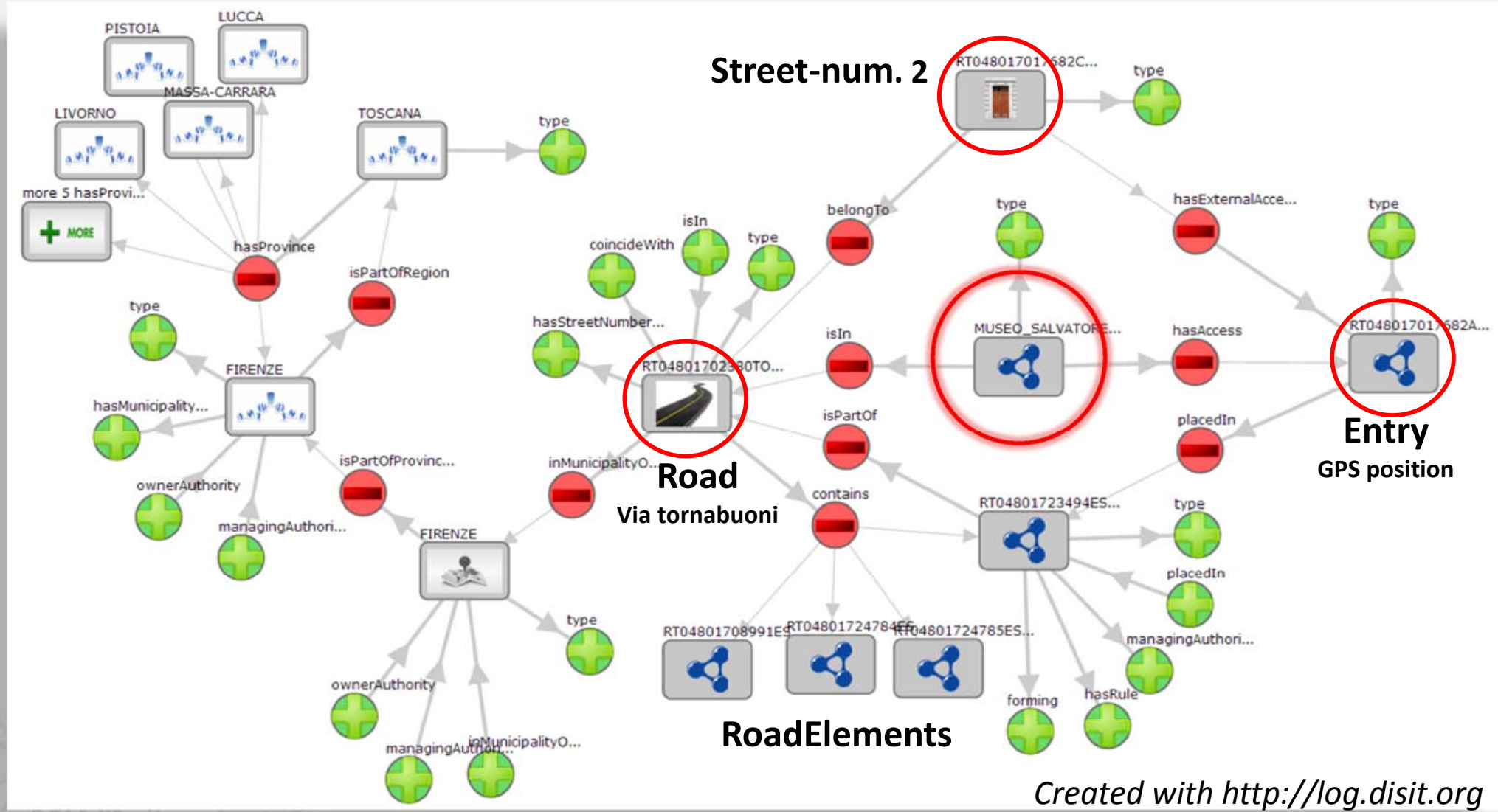
- ***Local public transport*:** includes the data related to major local public transport companies as scheduled times, the rail graph, and data relating to real time passage at bus stops.
- ***Sensors***: data provided by sensors: currently, data are collected from various sensors (parking status, meteo, pollution) installed along some streets of Florence and surrounding areas, and from sensors installed into the main car parks of the region.
- ***Temporal***: that puts concepts related with time (time intervals and instants) into the ontology, so that associate a timeline to the events recorded and is possible to make forecasts. It uses time ontologies such as OWL-Time.
- Defined reusing other basic ontologies
  - Dublin core; *FOAF* for the description of the relations among people or groups; *vCard* for a description of people and organizations; *wgs84_pos* for latitude and longitude.

Smart-city Ontology



84 Classes
93 ObjectProperties
103 DataProperties

Example: the Ferragamo Museum

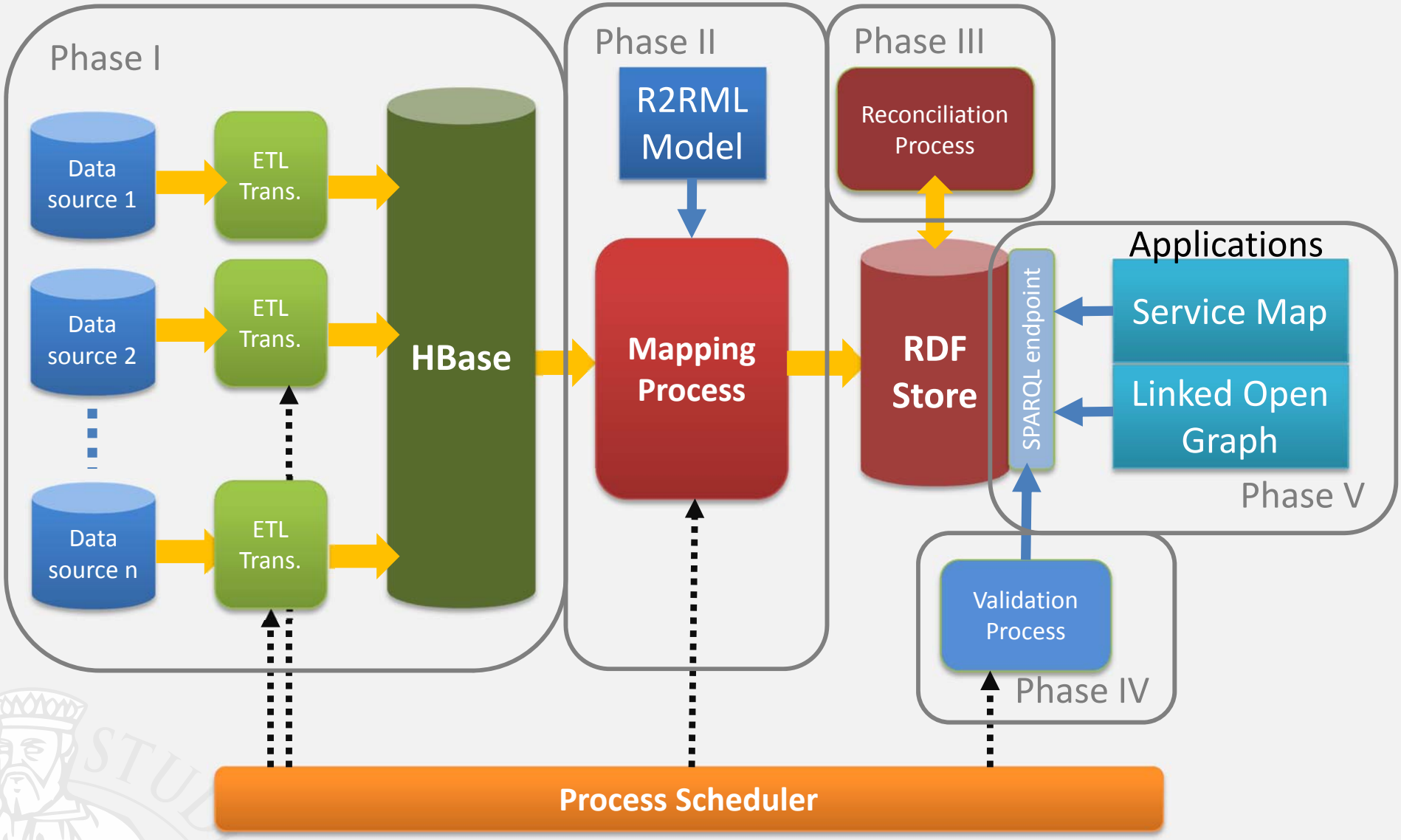


Created with http://log.disit.org

SPARQL query example

```
SELECT ?s ?c ?longitude ?latitude WHERE {
    ?s rdf:type sm:Service.
    ?s sm:serviceCategory ?c.
    ?s sm:hasAccess ?a.
    ?a geo:long ?longitude.
    ?a geo:lat ?latitude.
    ?a omgeo:nearby(43.77 11.24 "0.1km")
}
```

Data Engineering Architecture



Phase I - Data Ingestion

- Ingesting a wide range of public and private data, static or dynamic data.
- For the case of Florence/Tuscany area, we are addressing about **150 different data sources** of the 564 available.
- Using ***Pentaho - Kettle*** for data integration (Open source tool)
  - using specific ETL Kettle transformation processes (one for each data source)
  - data is stored in HBase (Bigdata NoSQL database)
- **Static and semi-static data** include: points of interests, geo-referenced services, maps, accidents statistics, etc.
  - Public files in several formats (SHP, KML, CVS, ZIP, XML, etc.)
- **Dynamic data** mainly data coming from sensors
  - parking, weather conditions, pollution measures, bus position, etc. using Web Services.

Phase II - Data mapping

- Transforms the data in Hbase into RDF
- Using **Karma Data Integration tool**, a mapping model from SQL to RDF on the basis of the ontology was created
- Data to be mapped first temporarly passed from Hbase to MySQL and then mapped using Karma (in batch mode)
- The mapped data is uploaded to the **RDF Store** (OpenRDF – sesame with OWLIM-SE)

Phase III - Data Reconciliation/alignment

- After the mapping each dataset ingested is not connected with the others, strongly limiting the usage of the knowledge base e.g. the services are not connected with the road graph.
- We want to associate each **Service** with a **Road** and an **Entry** on the basis of the street name, number and locality
- **It is not easy!** data coming from different sources

Phase III - Data Reconciliation/alignment

- Examples:
  - Typos;
  - Missing street number, or replaced with "0" or "SNC";
  - Municipalities with no official name (e.g. Vicchio/Vicchio del Mugello);
  - Street names and street numbers with strange characters (-, /, ° ? , Ang., ,);
  - Road name with words in a different order (e.g. Via Petrarca Francesco, exchange of name and surname);
  - Red street numbers (for shops);
  - Presence/absence of proper names in road name (e.g. via Camillo Benso di Cavour / via Cavour);
  - Number wrongly written (e.g. 34/AB, 403D, 36INT.1);
  - Roman numerals in the road name (e.g., via XXVII Aprile).

Phase III - Data Reconciliation/alignment

- Steps:
  1. *SPARQL Exact match* – match the strings as they are
  2. *SPARQL Enhanced Exact Match* – make some substitutions (Via S. Marta → Via Santa Marta, ...)
  3. *Last Word Search* – use only the last word of street name
  4. Use Google GeoCoding API
  5. Remove 'strange chars' (-, /, °, ? , Ang., ,) from Street number
  6. Remove 'strange chars' from Street name
  7. Rewrite wrong municipality names

Phase IV - Verification and Validation

- total 30,182 services/POI.
  - 13,185 reconciled at street number-level (43%),
  - 21,207 reconciled at street-level (70%)

No. Step	Method	No. hasAccess Triples created	No. isIn Triples created
1st Step	Exact search	5,627	8,329
2nd Step	Enhanced exact search	1,698	6,971
3rd Step	Last Word Search	5,160	5,415
4th Step	Google GeoCoding API	552	492
5th Step	Street number with strange chars	43	0
6th Step	Street name with strange chars	47	47
7th Step	Wrong municipality name	58	58
Total Reconciliated Services		13,185	21,207

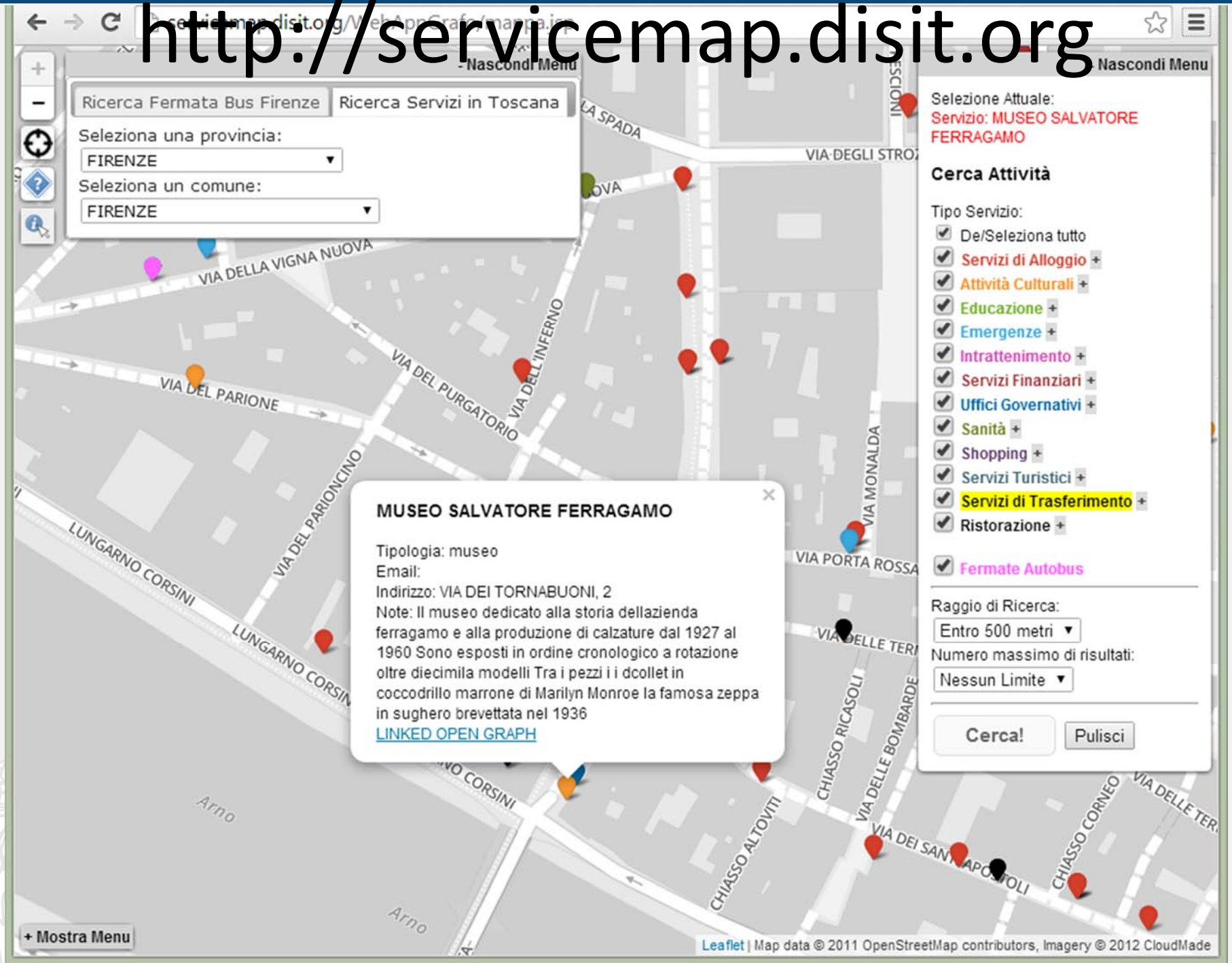
Phase IV - Verification and Validation

- The validation process is performed by defining a set of SPARQL queries that verify the knowledge base conditions with the aim of detecting **inconsistencies and incompleteness**, and verifying the correct status of the model.
- The validation process may lead to identify changes in the ingested datasets that may imply changes into the ontological model or in the ingestion processes.

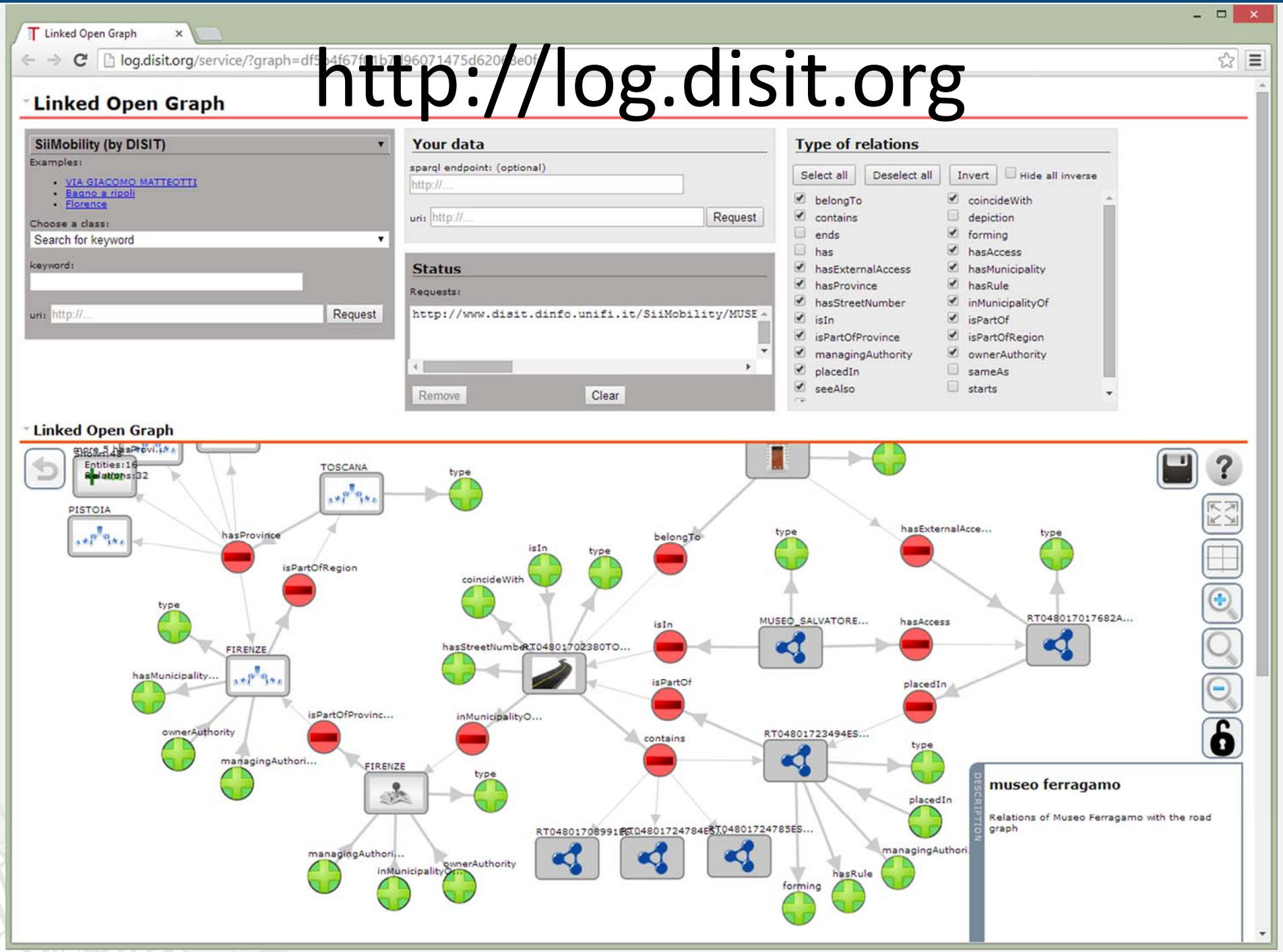
Phase V - Data access

- Applications can access the data using the SPARQL endpoint, currently we have two applications:
  - ServiceMap (http://servicemap.disit.org) for a map based application
  - Linked Open Graph (http://log.disit.org) for browsing the data from SPARQL/Linked Data sources

http://servicemap.disit.org



http://log.disit.org



Conclusions

- Developing a platform for smart-city data ingestion and data alignment based on Semantic Web Tools
- Developing a new Smart-city Ontology
- Future/Ongoing activities
  - Improvement of data alignment
  - Data cleaning
  - Comparison with other data alignment tools (Silk)
    - High precision (97%), suff. recall (72%) but better than Silk (67%)
- The platform will be used in **Sii-mobility project**
  - Adding prediction algorithms
  - Adding user-generated information
  - Adding more applications using the data

Thank you!