Data Acquisition & Integration

European Commission, first project review

3rd July 2018
Euroforum building, Luxembourg

Tomás Pariente (ATOS)
Claus Stadler (InfAI)
Overview

- Data acquisition and integration overview
- Linked Data Generation & RDF
WP4 works on:
- acquisition of static and real-time data,
- linked data and RDF.

Other work packages provide specific data acquisition and integration tools:
- using crowdsourcing techniques WP3,
- citizen engagement using i-Log app.

The data catalog is based on the use cases data assessment.
Challenges

- List and catalog datasets available for the project
- Establish foundations of data integration in WP4 and beyond
- Deal with different natures of data: static, dynamic and real-time data
- State of the art in different acquisition, data integration and RDF production tools

Main outputs

- Recap on data usage scenarios and their data needs
- Selection of datasets
- Requirements and main functionality for static, dynamic and real-time data acquisition
- Introduction to 1) RDF and Linked Data production tools, 2) data acquisition tools and techniques, 3) data integration tools and techniques for data-at-rest and data-in-motion.
- Introduction to the OASC and FIWARE architectural principles
- Around 60 datasets from Trento, half of them dynamic, plus data from i-Log and TomTom
OASC-Compliance: The Open & Agile Smart Cities initiative is a collaboration between smart cities across the world: http://www.oascities.org/

Replicability: The proposed acquisition framework can be replaced by any other existing framework or technology in other cities as long as it complies with the principles, data models and APIs.

Use of Apache NiFi as preferred data flow to comply with WP8 principles.

FIWARE data models
https://www.fiware.org/developers/data-models/
QROWD-DB FIWARE-based model POIs, Parking...

Outreach
Exploitation

FIWARE
NGSI API

OASC
Community

QROWD
Data Acquisition &
Integration
Principles

Driven-by-experimentation
approach

Trento experiments

Simple and
extensible Data
Model

Publication
CKAN
Target Smart City Platform from FIWARE

- Smart city platform as a Data/Knowledge Hub
- Non-intrusive, open to third parties
Data acquisition processes

- Trento Open Data
- WP3 / WP5 Updates
- Web services
- Trento sensors

**Static Data Acquisition**
- V1
  - Raw
  - GeoJson
  - RDF
  - FIWARE

**Streaming Data Acquisition**
- Nifi Acquisition pipeline
- FIWARE agents

**Sensor Data Acquisition**

D4.2 will explain the internals and the acquisition methodology
- Several origins / Several files in CKAN
  1. Original dataset to CKAN
  2. Intermediate datasets
  3. Fusioned datasets
  4. NiFi processes to recreate the master dataset in different formats
Crowdsourcing data

Existing mobility infrastructure catalogues

Volunteers' geographical information

Crowdsourcing data

**Bike racks data integration flow example**

1. Original datasets transformed to FIWARE data model
2. Upload to CKAN. One dataset may have different formats (json, RDF, dashboard) for the same version
3. Processes of crowdsourcing, data fusion and interlining enrich the data
4. A new version of the dataset is uploaded to CKAN

**Extensible to other POIs**

New version: Event in CB to create a new version of the dataset, and update dashboard, json or RDF formats
Extensible to other POIs

1. Original datasets transformed to FIWARE data model

2. Upload to CKAN. One dataset may have different formats (json, RDF, dashboard...) for the same version

3. Processes of crowdsourcing, data fusion and attribute completion may add new attributes (i.e. capacity) and fire an event to update the dataset to a new version in the desired formats
Dynamic data acquisition and integration

From services or sensors

Usage of FIWARE enablers

- Status of the city: Dynamic data
  1. From sensors or services to the Context Broker
  2. New NiFi NSGI processor created to ingest data to the CB
  3. NSGI consumers to read data from the dashboard or to persist data into QROWD-DB
Overview

- Data acquisition and integration overview
- Linked Data Generation & RDF
Linked Data Generation

Challenges
• Various data formats, prominently XML, JSON, CSV, RDF
• Need for processing spatial and temporal data
• Publishing of the generated Linked Data data sets

Main outputs
• Sparql Integrate (Software) for RDF Generation
  SPARQL Function Extensions + Command Line Interface +
  Apache NiFi Plugin
• DCAT Suite (Software) for executing task from DCAT descriptions
  Deploy datasets to a CKAN
  Load RDF datasets into a triple store
  Retrieve CKAN metadata as DCAT (bidirectional mappings)
  Expand Sparql Integrate output (n-quads) into dataset files
Benefits of Linked Data

• (LD) Knowledge Graphs established means to manage data Disambiguation, Symbolic Learning, Linked Data readiness for Smart Cities
• Simplified data integration due to Uniform graph-based data model Powerful query language, reuse of existing tooling
• As a consequence: Simplified analysis

Goals
• Provide RDF datasets relevant to the use cases Timetables to enrich model split Bikeracks from different sources Background Knowledge
• Bridge RDF and FIWARE schemas E.g. parking models in FIWARE and schema.org
Sparql Integrate

- SPARQL Extensions + Command Line Interface + Apache NiFi Plugin

- Current best practice to design the LD generation process
DCAT Suite

- DCAT-centric tool suite
- Import datasets from CKAN via configurable client-side mappings
- Deploy datasets based on DCAT to CKAN
- Deploy RDF datasets to triple stores
- DCAT-AP support

DCAT Suite components:
- **CKAN Importer**
- **CKAN Deployer**
- **RDF Dataset Deployer**
- **DCAT-AP Mapper**
- **Common Data Model**

Resources:
- [dati.trentino.it](http://dati.trentino.it)
- [github.com/SmarDataAnalytics/dcat-suite](https://github.com/SmarDataAnalytics/dcat-suite)
- [http://ckan.qrowd.aksw.org/](http://ckan.qrowd.aksw.org/)
Overall Process

- Selected data sources from WP2 dataset catalogue
- Workload Definition Generation Script
- Sparql Integrate Scripts
- Process governed by SPARQL
- Deployments
- Dcat Suite Processor
- DCAT-AP
- SPARQL Dataset
- Sparql Integrate Processor
- Datasets (Snapshots)
- CSV
- JSON
- RDF
- XML
Achievements M18

- Achievements data acquisition
  - Data Catalog
  - Data acquisition framework based on NiFi
  - Acquiring data from static and dynamic datasets
  - Ready for the Trento pilot
  - Easily integrate new data sources into FIWARE-compliant data models

- Achievements linked data generation
  - Sparql Integrate
  - Dcat Suite
  - + NiFi Plugins
Future work

- Implement all the NiFi acquisition processes needed for the use cases
- Describe the QROWD methodology for data acquisition
- Dataset versioning methodology
- Deliverable D4.2 (M24) Data acquisition framework

- Cover all Linked Data generation needed for the use cases with SPARQL
- Cover more attributes of (Geo)DCAT-AP
- Deliverables D4.3 / D4.4 (M24) Linked Data generation framework and multilingual data harvesting