An Extensible Semantic Data Fusion Framework for Autonomous Vehicles

E. Kontopoulos, P. Mitzias, K. Avgerinakis, P. Kosmides
Catalink Ltd, Cyprus

N. Piperigkos, C. Anagnostopoulos, A. S. Lalos
Industrial Systems Institute, "ATHENA" Research Centre, Greece

N. Stagakis, G. Arvanitis, E. I. Zacharaki, K. Moustakas
Electrical & Computer Engineering Dept., Univ. of Patras, Greece

Presenter Info

https://www.linkedin.com/in/skontopo/

@skontopo

catalink

https://catalink.eu/

https://www.linkedin.com/company/catalink-ltd

- PhD in Semantic Web Technologies
- >15 years of experience on Semantic Web theory & practice in academia & industry

---

- Catalink's areas of expertise
  - Knowledge-based Applications
  - Computer Vision
  - Machine Learning
- Strong presence in research projects
- Wide network of contacts with Universities, Research Institutes, Industry, Municipalities & Public Orgs

CPSoSaware

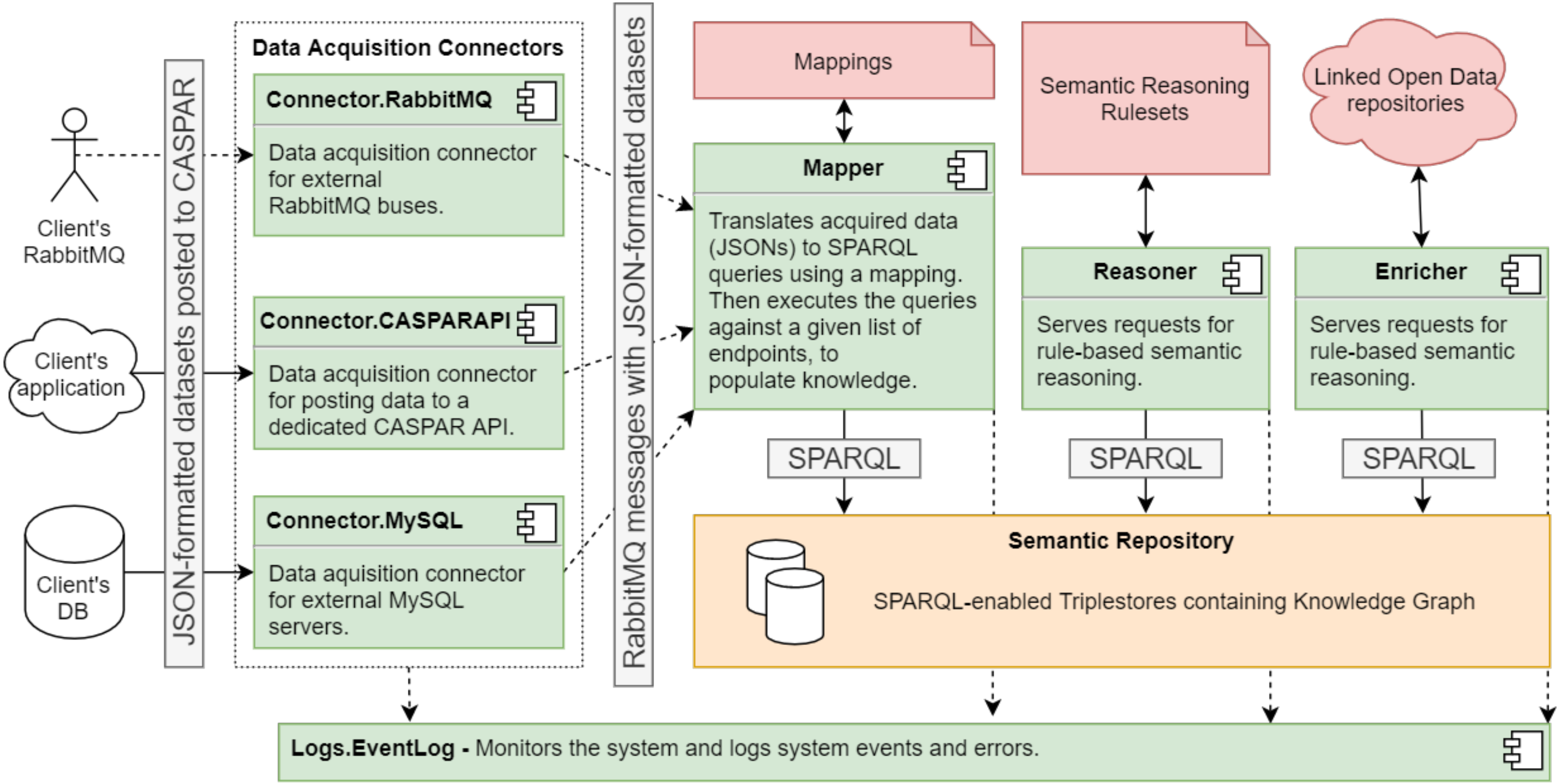
Motivation

- Rapidly increasing interest in (semi)autonomous connected vehicles
- Automotive AI market estimated at **$10.5 billion by 2025**
- Major automotive manufacturers increasingly investing in the field
- Key challenge: AI systems must deal with **large volumes of heterogeneous data** generated by sensors on the vehicle (e.g., camera, radar, LiDAR, GPS, etc.) & by the vehicle's environment (e.g., pervasive inputs, weather data, other vehicles, etc.)
- Semantic technologies are a natural fit for **semantically integrating** heterogeneous pieces of information into a uniform knowledge representation model, i.e., **a semantic KG**

Background – CPSoSaware H2020 Project

- CPSoS as a living organism:
  - Behaves autonomously
  - Is aware of its physical & cyber environment
  - Reacts to environment to match intended purpose
- Aims:
  - Develop models & tools for describing a CPSoS in a holistic & abstract way
  - Allocate computational power/resources to the CPS devices towards the overarching goal
  - AI for reliability, fault tolerance & security at system level
  - Increase human situational awareness for human-in-the-loop & consider human behavior in the CPSoS design & operation phase
- Two use cases:
  - **Connected semiautonomous vehicles**: Assess driver's state (DMS) + analysis of scene outside the vehicle
  - **Human – robot interaction in a manufacturing environment**: Operator's safety + design operation continuum support

CASPAR – Architecture



CASPAR – Data Acquisition Mechanisms

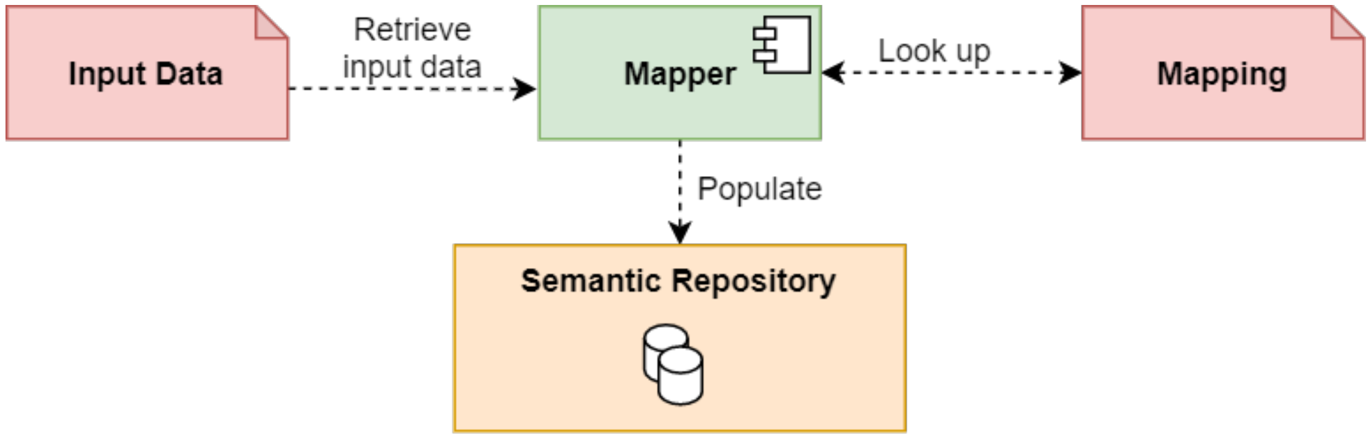
Aims:

- Heterogeneous input from multiple (diverse) sources
- Periodically retrieved & normalized into a common format
- Non-intrusive & seamlessly integrated to existing data-intensive systems

Modular data connectors:

- **API Connector**: RESTful API expecting HTTP requests with payload
- **MySQL Connector**: Definition of MySQL queries for extracting desired datasets from end user's DB
- **RabbitMQ Connector**: Directly consume messages from end user's RabbitMQ deployment
- **Future plans**: More query languages & DBs (SQL, SQLite, NoSQL, SPARQL); More message brokers (e.g. Apache Kafka); Periodically querying REST APIs on end-user side; Directly handling datasets in XML, CSV, Excel, JSON, etc.

Mapper – Mapping Definition & Semantic Data Integration



- Heterogeneous input from multiple (diverse) sources
- Transforms data into SPARQL queries & does the semantic data integration
- **Mappings** specify how input data is associated with concepts and relationships in the KG
  - **Template**: Mechanism for focusing on specific parts of the input (multiple templates per source input)
  - **Individuals**: Specify nodes to be created or updated in the KG
  - **Properties**: Specify object or data properties in the KG

So, where is the Novelty?!?

Other tools:

- Directly generate RDF triples from input source
- Insertion of duplicate nodes & branches

CASPAR:

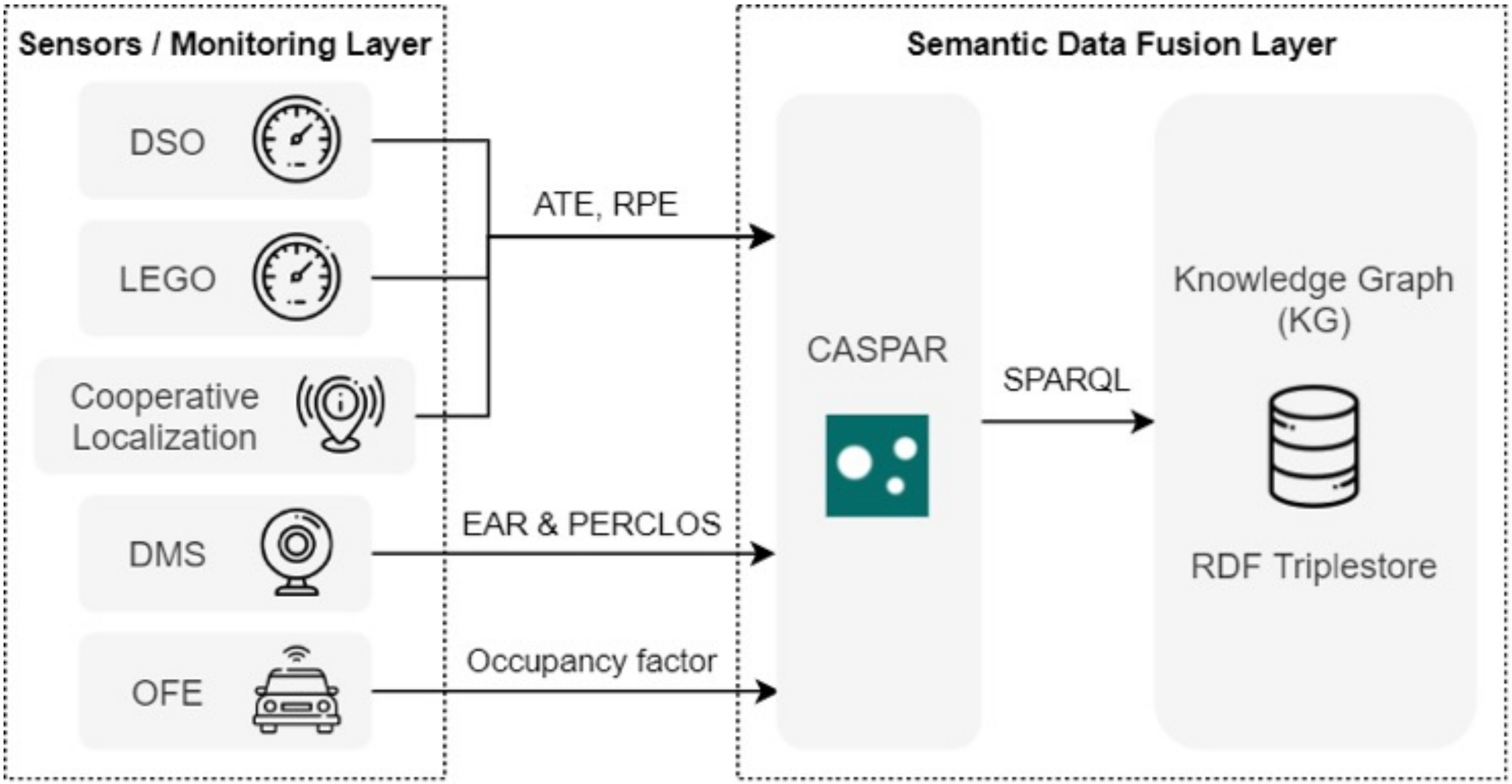
- Optional **update_on** field for each individual
- Takes advantage of SPARQL UPDATE

```
"individuals": [
  {
    "path": "customers.[*]",
    "namespace": "http://www.example.com/caspar#",
    "classes": [
      "ex:Customer"
    ],
    "properties": [
      {
        "predicates": [
          "ex:hasName",
          "rdfs:label"
        ],
        "object": {
          "path": "customers.[*].Name"
        }
      },
      {
        "predicates": [
          "ex:hasID"
        ],
        "object": {
          "path": "customers.[*].ID"
        }
      },
      {
        "predicates": [
          "ex:hasDietPreference"
        ],
        "object": {
          "path": "customers.[*].Dietary preferences.[*]"
        }
      }
    ],
    "update_on": [
      {
        "predicates": ["ex:hasID"],
        "object": {
          "path": "customers.[*].ID"
        }
      }
    ]
  },
```

Two Application Scenarios

- Semantic data integration from multiple diverse sources in the CPSoSaware autonomous driving use case
  - Populate a semantic model with data from other system components
- Monitor non time-critical parameters via offline reports
  - **Scenario #1**: Evaluate robustness of odometry algorithms
  - **Scenario #2**: Calculate risk levels for the driver

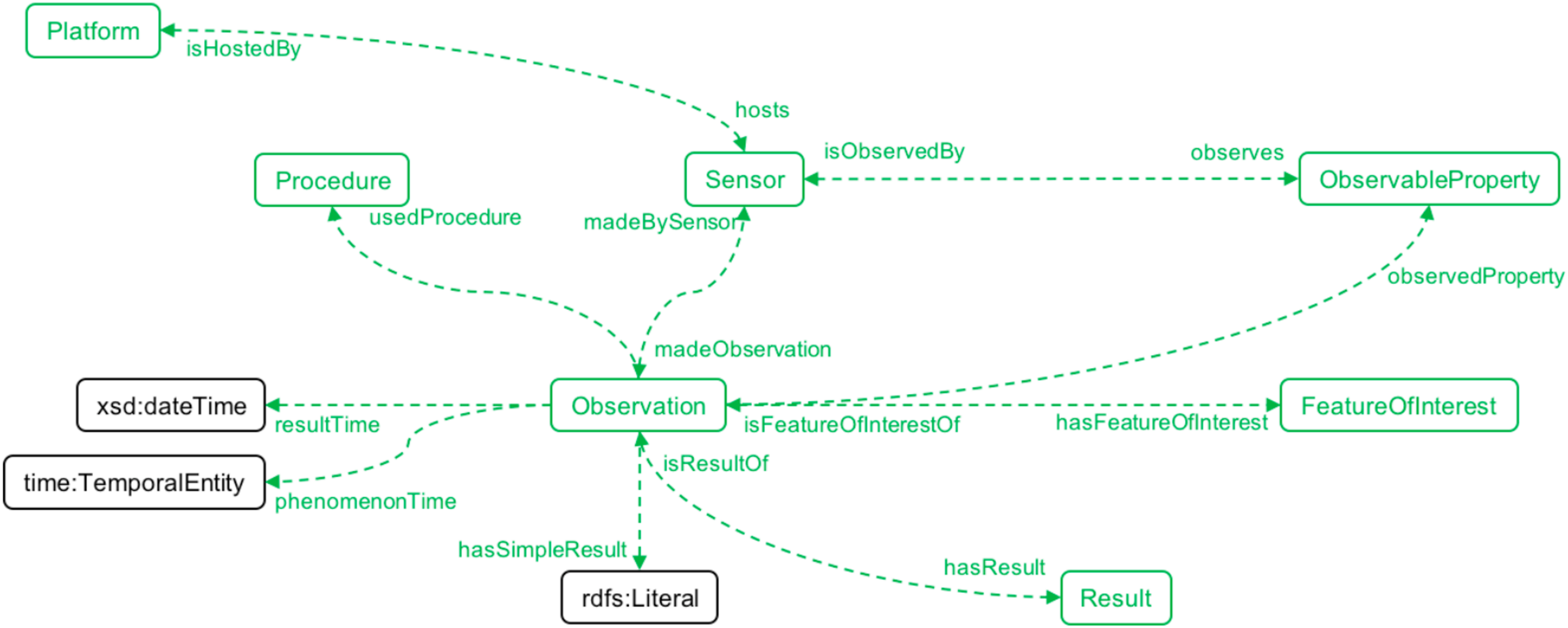
Overview



Sensors / Monitoring Layer
DSO
LEGO
Cooperative Localization
DMS
OFE
ATE, RPE
EAR & PERCLOS
Occupancy factor
Semantic Data Fusion Layer
CASPAR
SPARQL
Knowledge Graph (KG)
RDF Triplestore

Core Semantic Model

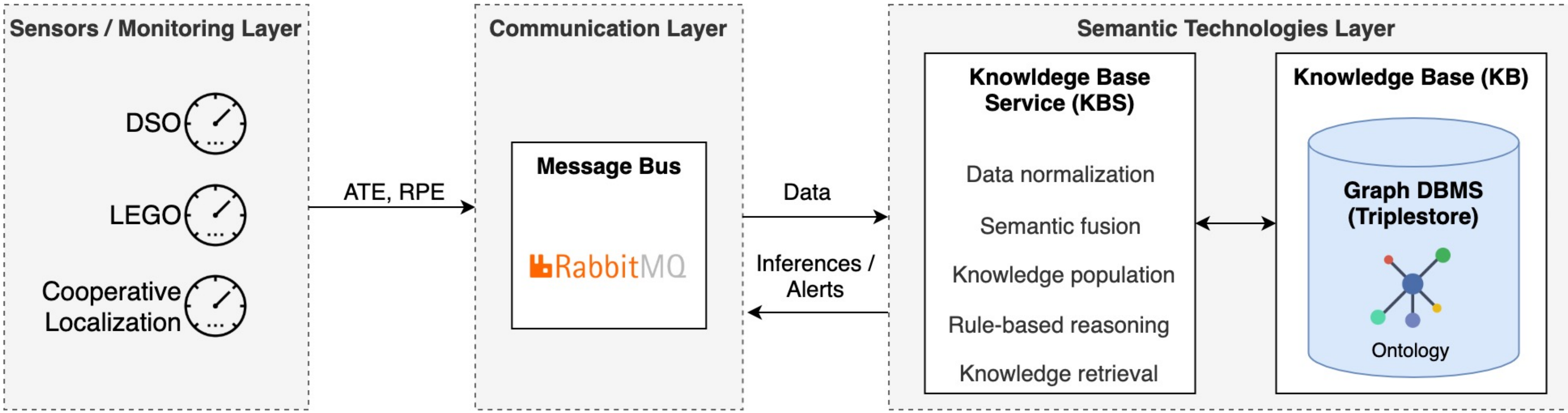
SOSA ontology (Sensor, Observation, Sample, & Actuator)*



*W3C Recommendation 19 October 2017: http://www.w3.org/ns/sosa/

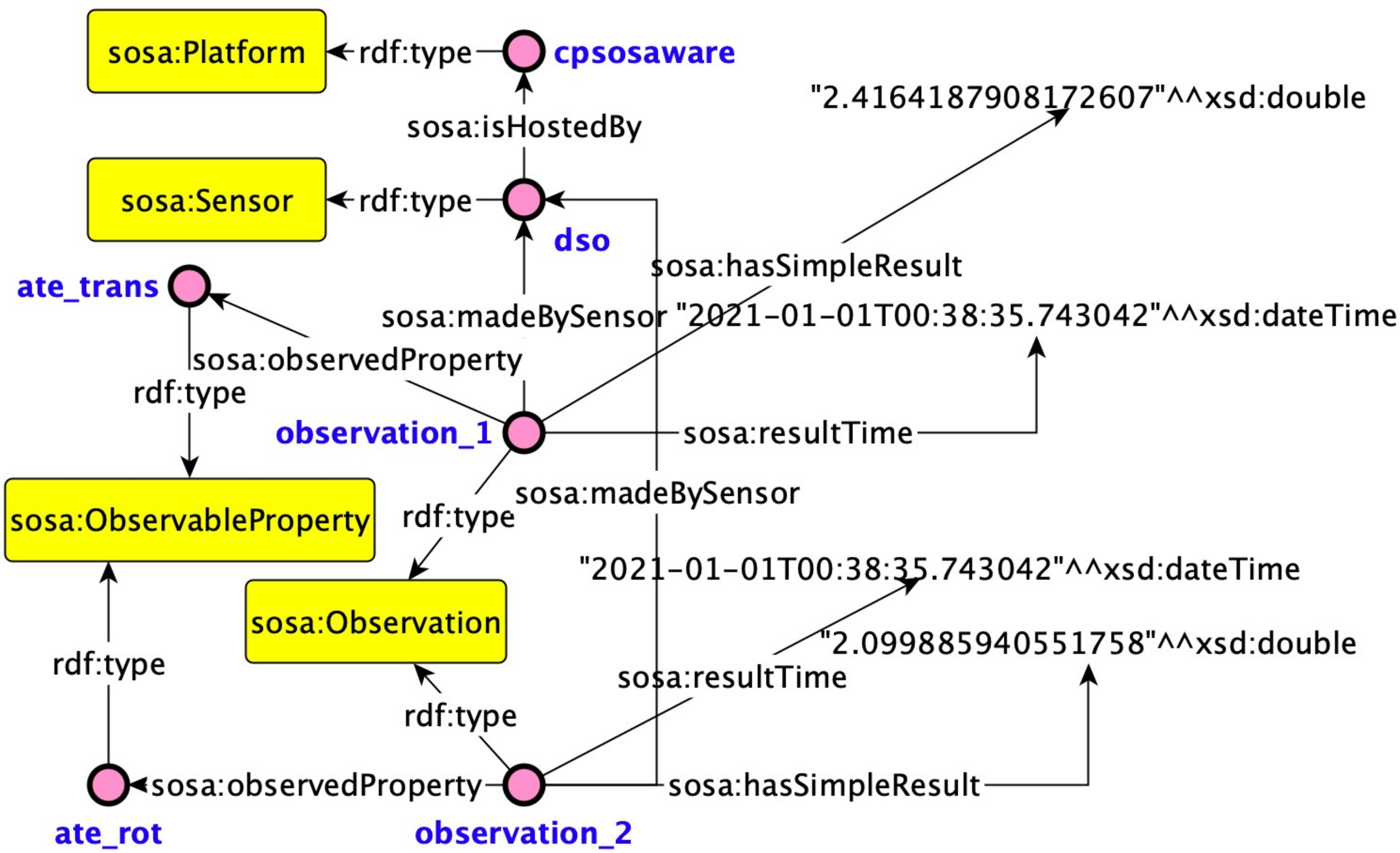
Scenario #1: Evaluate Robustness of Odometry Algorithms — Setup

ATE & **RPE** calculated for 3 algorithms (LegoLoam, DSO, cooperative localization) utilizing 3 different modalities (lidar, rgb, gnss)




```json
{
  "observations": [
    {
      "timestamp": "1970-01-01T00:38:35.743042",
      "property": "ate_trans",
      "result": 2.4164187908172607,
      "source": "dso"
    },
    {
      "timestamp": "1970-01-01T00:38:35.743042",
      "property": "rpe_trans",
      "result": 0.8676187992095947,
      "source": "dso"
    },
    ...
  ]
}
```

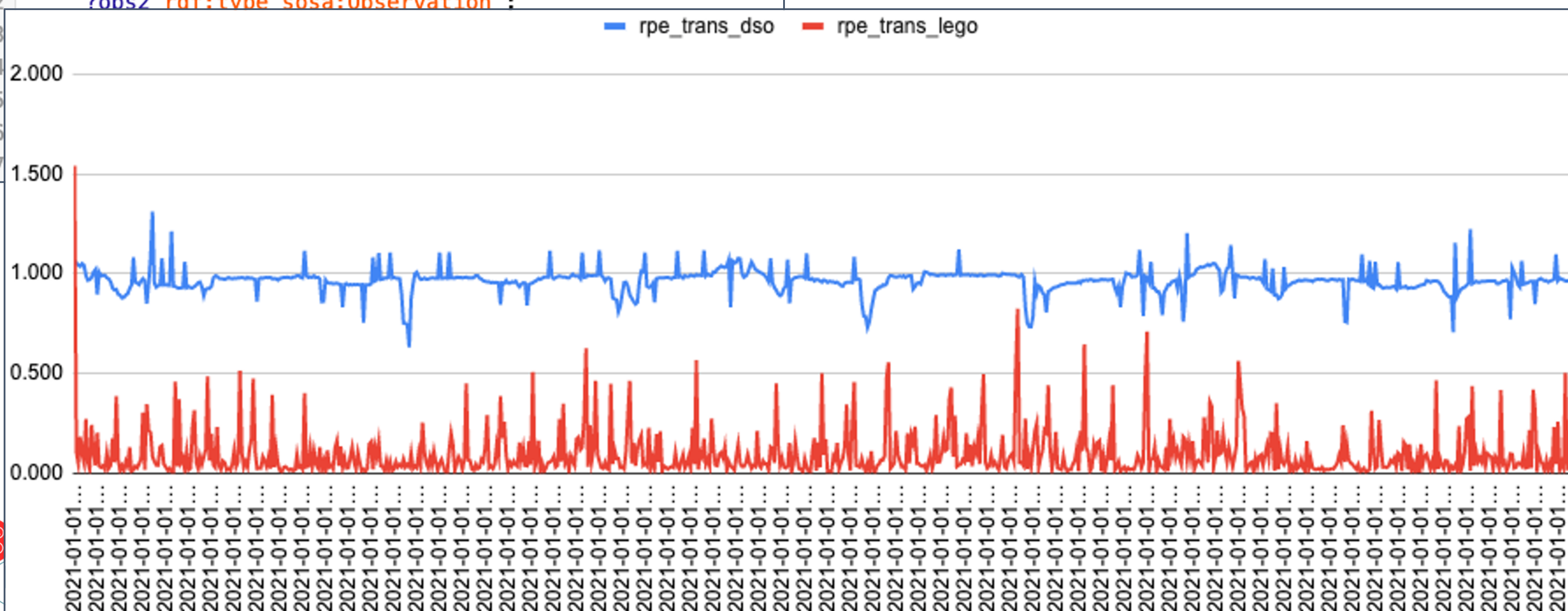
Scenario #1: Evaluate Robustness of Odometry Algorithms — Semantic KG



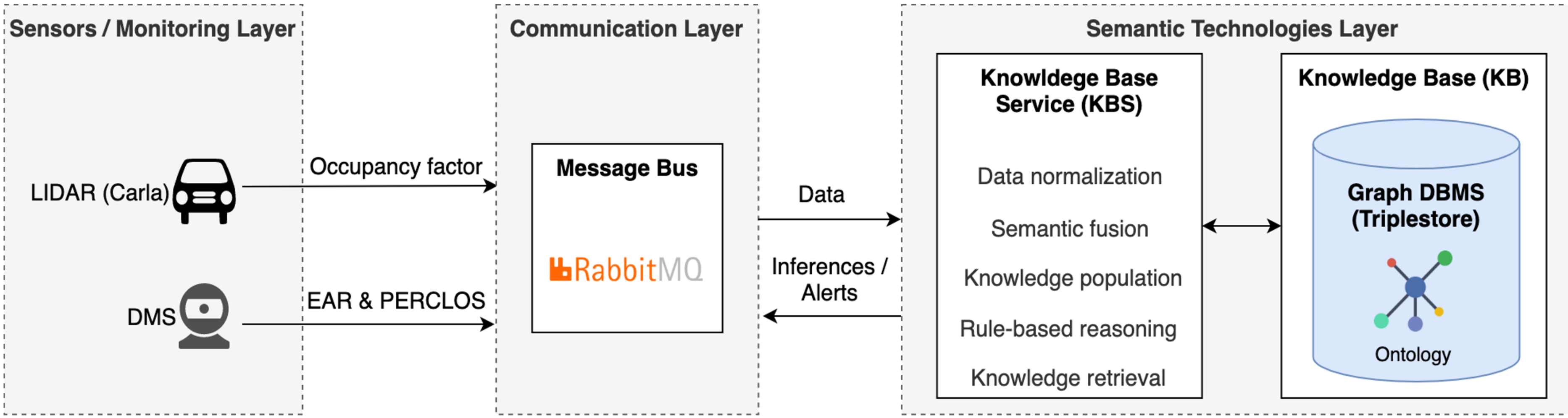
Scenario #1: Evaluate Robustness of Odometry Algorithms — Results

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX sosa: <http://www.w3.org/ns/sosa/>
PREFIX : <http://www.cpsosaware.eu/ontologies/kg#>

SELECT DISTINCT ?timestamp ?rpe_trans_dso ?rpe_trans_lego
WHERE {
    ?obs1 rdf:type sosa:Observation ;
        sosa:resultTime ?timestamp ;
        sosa:madeBySensor :dso ;
        sosa:hasSimpleResult ?rpe_trans_dso ;
        sosa:observedProperty :rpe_trans .
    ?obs2 rdf:type sosa:Observation ;
```



Scenario #2: Calculate Risk Levels — Setup



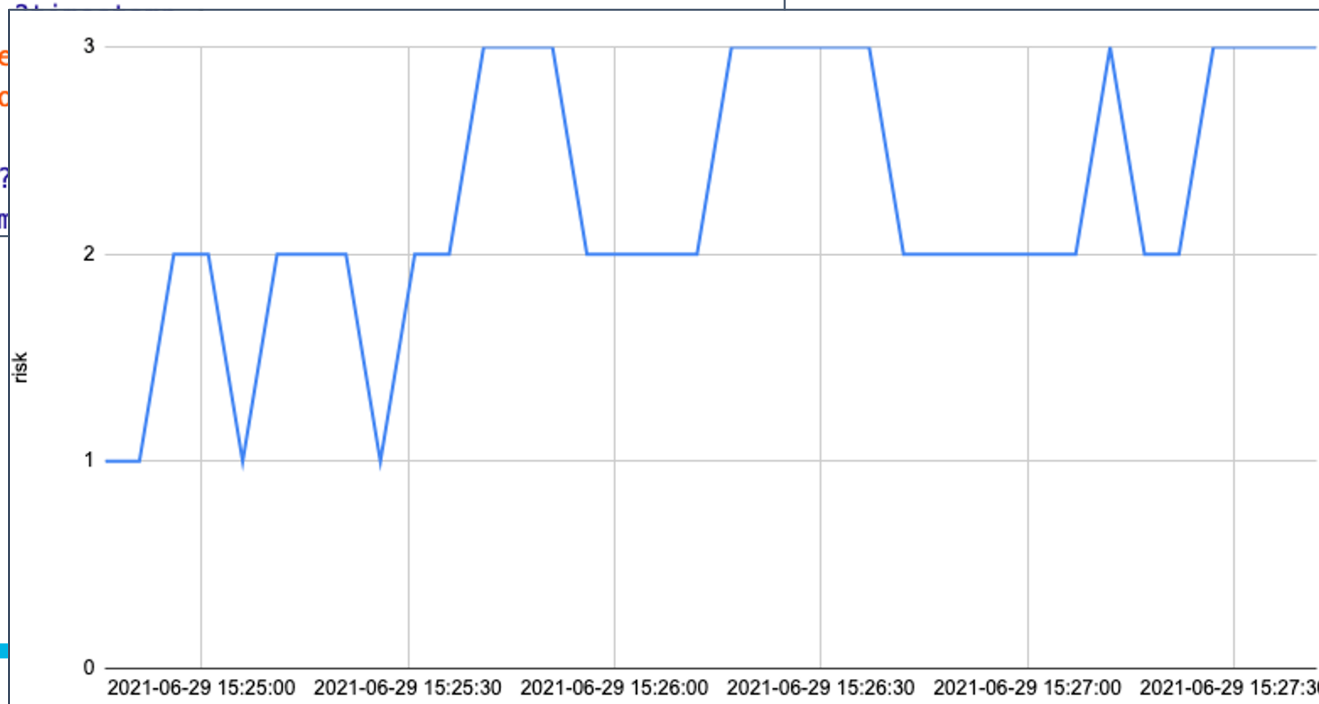

```json
{
  "observations": [
    {
      "timestamp": "2021-03-29T15:21:33.358+03:00",
      "result": 0,
      "property": "perclos",
      "source": "dsm_camera"
    },
    {
      "timestamp": "2021-03-29T15:21:33.358+03:00",
      "result": 319.435173411598,
      "property": "occupancy_factor",
      "source": "carla_lidar"
    },
    {
      "timestamp": "2021-03-29T15:21:33.358+03:00",
      "result": 0.32744966651627755,
      "property": "ear",
      "source": "dsm_camera"
    }
  ]
}
```


	PERCLOS < 0.25	PERCLOS >= 0.25 & < 0.7	PERCLOS >= 0.7
Occupancy factor > 280	Low risk	Be aware	High risk
Occupancy factor <= 280 & > 200	Low risk	Be aware	High risk
Occupancy factor <= 200	Be aware	High risk	High risk

Scenario #2: Calculate Risk Levels — Results

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX sosa: <http://www.w3.org/ns/sosa/>
PREFIX : <http://www.cpsosaware.eu/ontologies/kg#>

SELECT DISTINCT ?timestamp ?risk ?perclos ?of
WHERE {
    ?obs1 rdf:type sosa:Observation ;
        sosa:resultTime ?timestamp ;
        sosa:hasSimpleResult ?perclos ;
        sosa:observedProperty :perclos .
    ?obs2 rdf:type sosa:Observation ;
        sosa:resultTime
        sosa:hasSimpleRe
        sosa:observedPro
    BIND (IF(?perclos <
&& ?perclos < 0.7), IF(?
} ORDER BY ASC(?timestam
```



Next Steps

- Unified KG
- Run more complex simulation scenarios
  - Add more sources of information: steering frequency & biometrics (DMS), weather info
  - Dynamic objects, reduced visibility, sudden braking, etc.
  - Augment the rule-base accordingly
- Handle real-time semantic reasoning
- Deploy semantic data integration in the 2nd use case: manufacturing

Thank you for your attention!

CASPAR – Performance