UNIVERSITÉ CÔTE D'AZUR

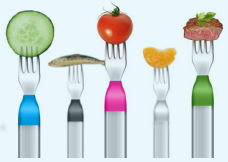
i3S sophia antipolis

Semantic Web of Things (SWoT)

An introduction

Gérald Rocher and Thibaut Gonnin, 08/07/2021

Looking for a display...



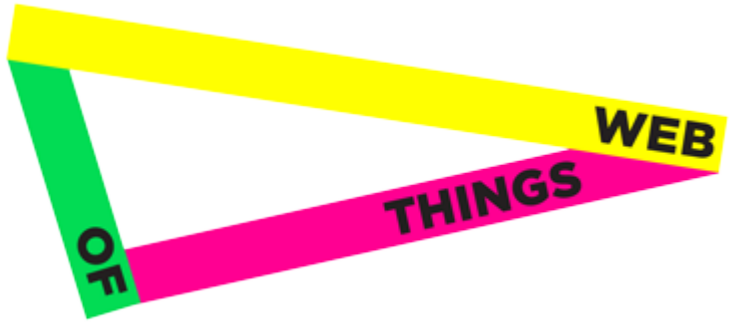
How machine can **reason** about networked things functionalities ?

Web of Things (WoT)

Introduction

- Approaches, software architectural styles and programming patterns that **allow networked things to be part of the World Wide Web**...

- **Dominique Guinard --** PhD thesis (2011)
  - Four layers IoT **common application architecture**
    1. **Accessibility,**
    2. **Findability,**
    3. **Sharing,**
    4. **Composition.**



A Web of Things Application Architecture – Integrating the Real-World into the Web.

PhD thesis No. 19891, ETH Zurich, Zurich, Switzerland, August 2011

Accessibility Layer

- **Resource Oriented Architecture (ROA)**
  RESTful services & description (RSDL)
  data centric: over HTTP (GET, POST, PUT, DELETE, etc.)

- **Service Oriented Architecture (SOA)**
  SOAP services, WS-* (functional control),
  API descriptions (WSDL).

Source : Wikipedia

- **A common & generic way to access things data and API through gateways**
  - Technological abstraction,
  - Fixing the IoT Technological Heterogeneity issue

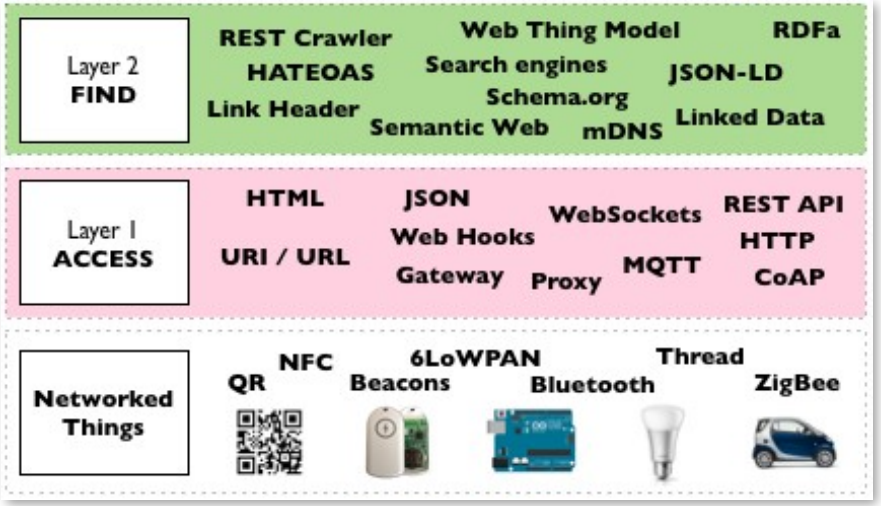
Once things are accessible, web applications and tools can be used...

Findability Layer

- Providing a way to **find** and **locate** relevant services (devices) on the Web
  - Search engines,
  - Crawlers,
  - Etc...

Metadata model for describing things and their services...

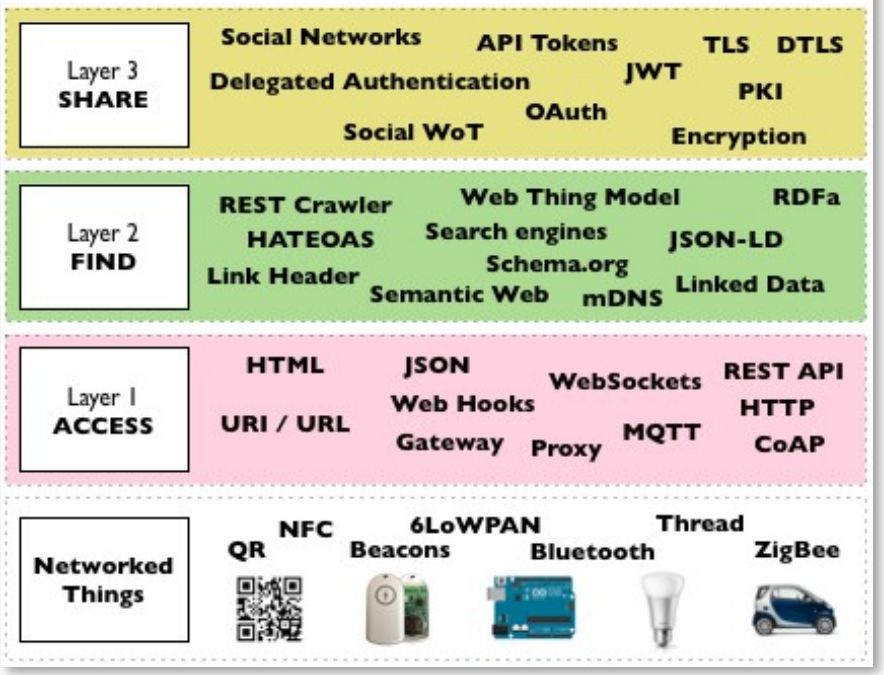
- **Integration/indexation of smart things to existing search engines**,
- **Semantic annotations** based on microformat/ RDFa to describe static properties (product, service) and dynamic properties (Location, Quality of Service, etc.).



Source : Wikipedia

Sharing Layer

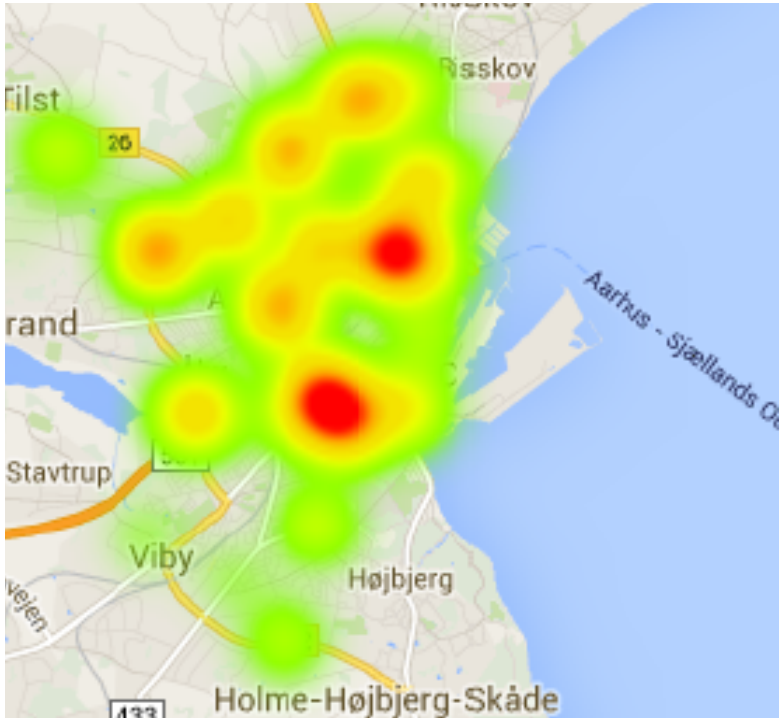
- Ensuring that data generated by devices are shared in an efficient and secure manner.
  - **Security (encryption, authentication)**
  - JSON Web Token (JWT), API tokens, etc...

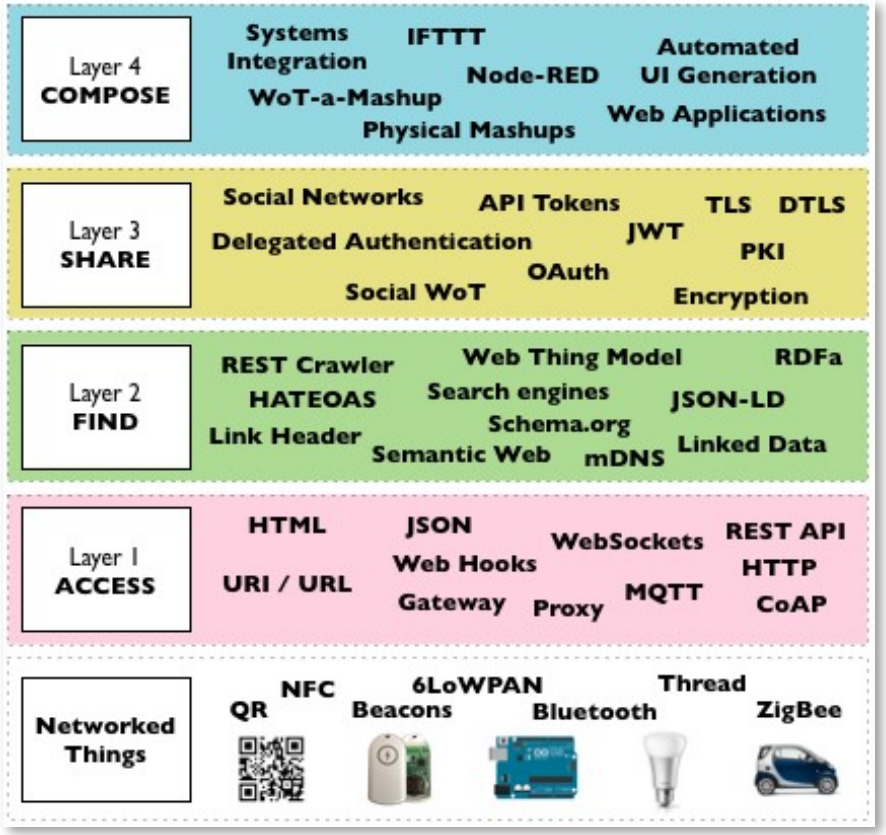


Source : Wikipedia

Composition Layer

- Integrating services and data offered by the devices into a set of higher level web tools
  - Analytics software,
  - **Physical mashups, composite applications**,
  - Etc...





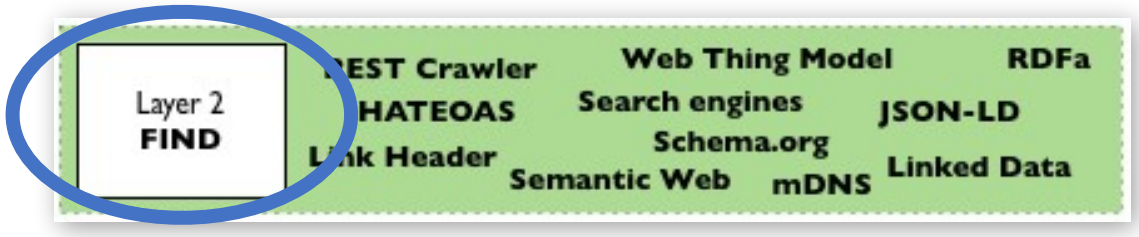
Source : Wikipedia

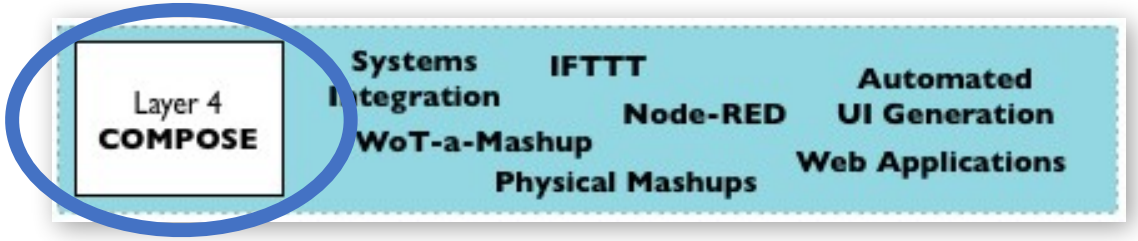
What's the problem?

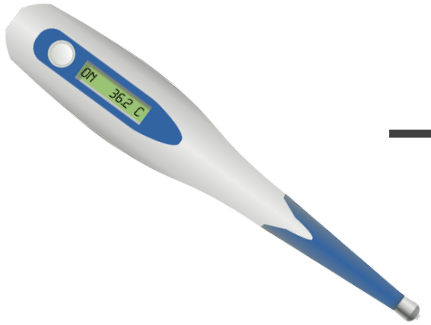
Semantic heterogeneity issue

- As many ways to describe things, services and data as manufacturers…
- Numerous data sources and types…

- How to describe data to get it **machine understandable** and establish **collaboration** among devices (full interoperability) far beyond microformat, RDFa, …?







37.8°C

Sensor data

160.0°C

Sensor data

73.0°F

Sensor data

How can machines "understand" the semantics of data?

Semantic Web of Things (SWoT)

Rhayem, A., Mhiri, M. B. A., & Gargouri, F. (2020). **Semantic web technologies for the internet of things: Systematic literature review**. *Internet of Things*, 100206

Ruta, M., & Scioscia, F. (2020). **Information-Centric Semantic Web of Things**. *Open Journal of Internet Of Things (OJIOT)*, *6*(1), 35-52.

Introduction

- Reuse **Semantic Web standards** to formally describe things, their services and the data they publish… **with semantically enriched annotations**

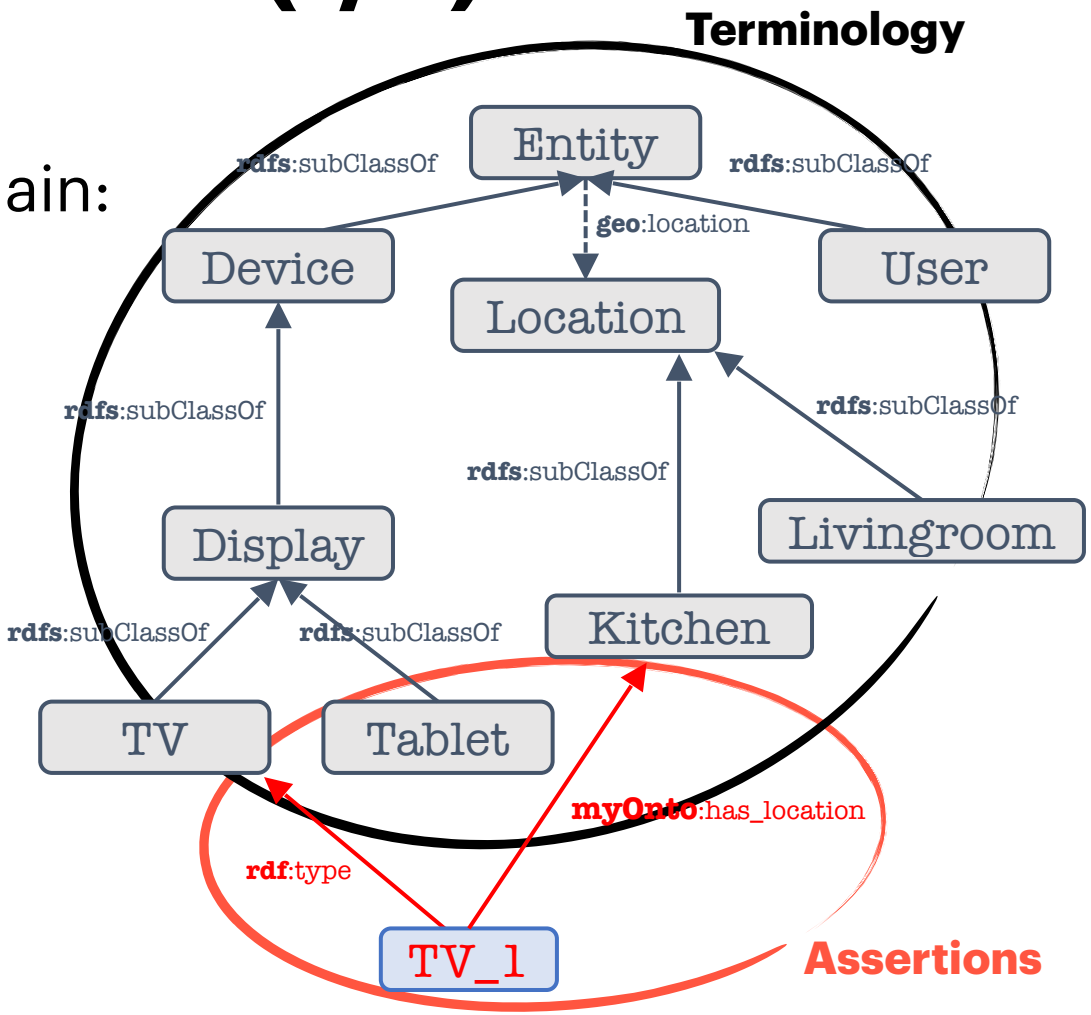
In a nutshell:

- Knowledge **formal description languages** (RDF, RDFS, OWL, etc.),
- **Reasoning** (built-in/custom inference rules),
- **Querying** (SPARQL, DL Query).

https://www.w3.org/standards/semanticweb/

Semantic Web standards basics (1/3)...

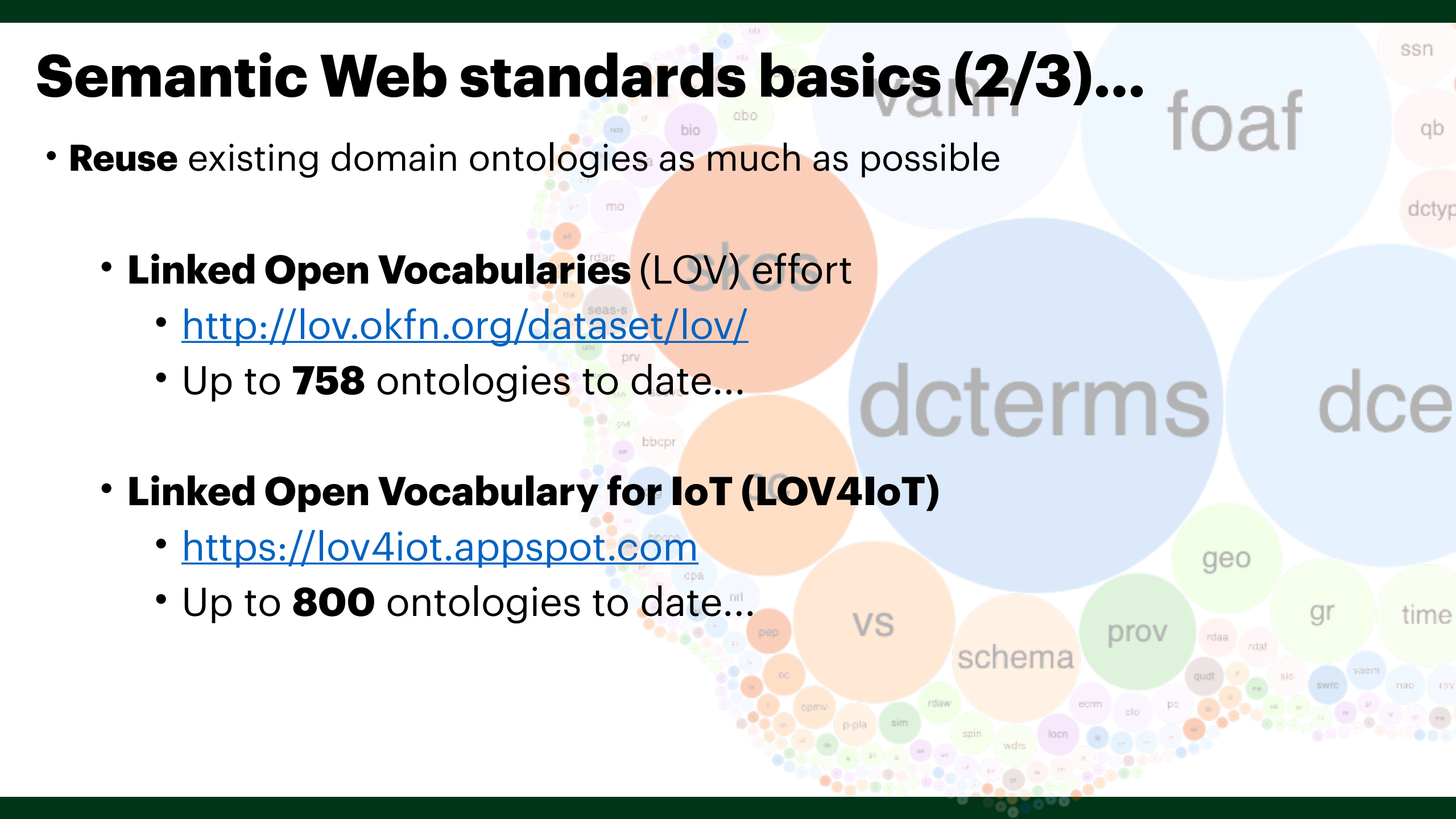
- **Ontology**
  - **Formal knowledge description** of a domain:
    - Class (concept), Class hierarchy,
    - Properties, instances,
    - Restrictions, etc...
  - A **consensual domain knowledge** to be **reused** and **shared** across applications and people.
  - **Everything is defined through URI**s (Uniform Resource Identifiers)



Knowledge expressivity (and inferences) depends on the description language used (RDFS, OWL-LITE, OWL-DL, OWL2, etc...).

Semantic Web standards basics (2/3)...

- **Reuse** existing domain ontologies as much as possible
  - **Linked Open Vocabularies** (LOV) effort
    - http://lov.okfn.org/dataset/lov/
    - Up to **758** ontologies to date...
  - **Linked Open Vocabulary for IoT (LOV4IoT)**
    - https://lov4iot.appspot.com
    - Up to **800** ontologies to date...

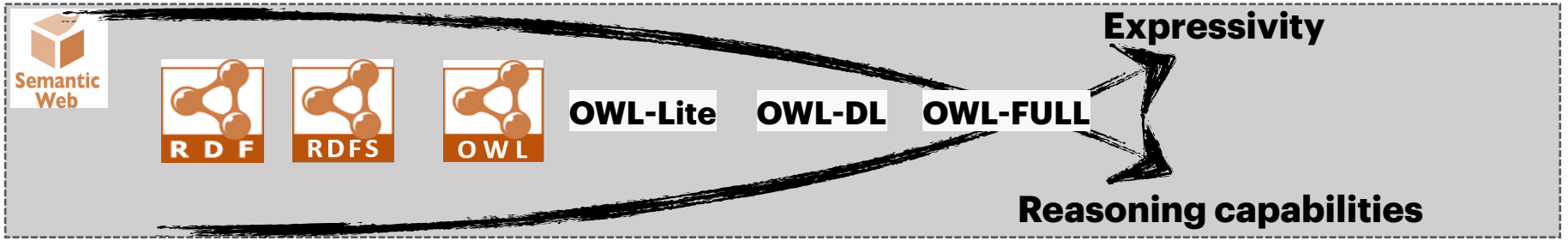


Semantic Web standards basics (3/3)...

- **Knowledge Base (KB)**
  - **Stores** intentional (**Tbox → Terminology**) and assertional knowledge (**Abox → Assertions**),
  - From which...
    - An **Inference** engine can reason about the knowledge and use rules and logic to **infer new knowledge** or **detect inconsistencies**.
    - A **query engine** (SPARQL) can retrieve, add or remove RDF data.

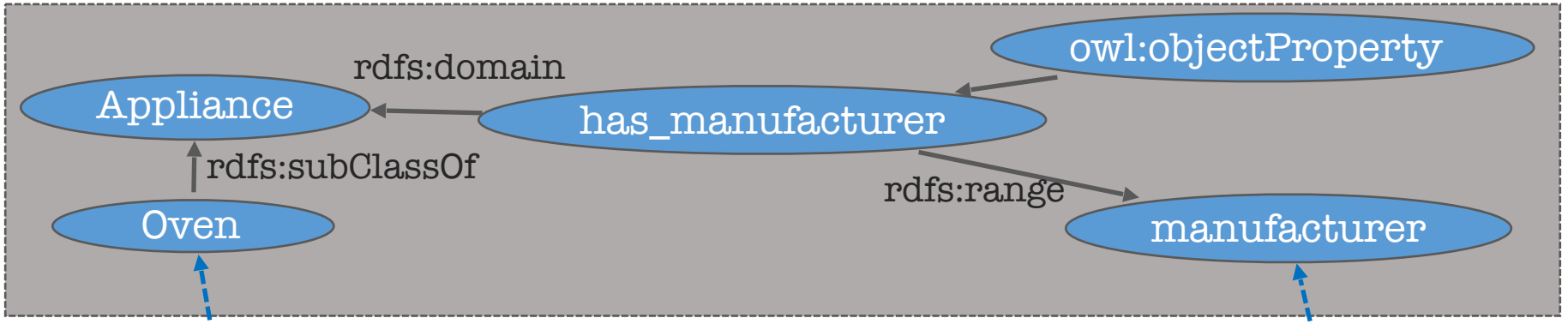
Ontology modeling layers

Meta meta model
(Ontology modeling languages)



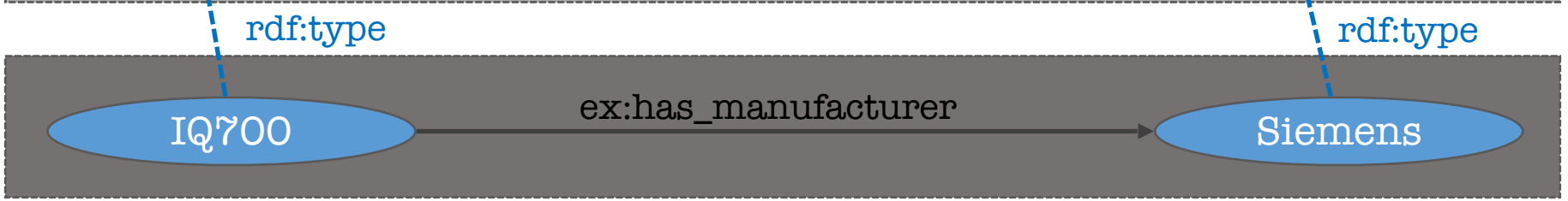
Relies on

Meta model
(Specific domain ontology,
aka **upper ontology**)



Relies on

Model
(Assertions about the world)



Serialization formats

RDF/XML **Notation3 (.n3)** **JSON-LD (.jsonld)** **...**

Turtle (.ttl) **N-Triples (.nt) N-Quads (.nq)**

Example

source: https://ontola.io/blog/rdf-serialization-formats/

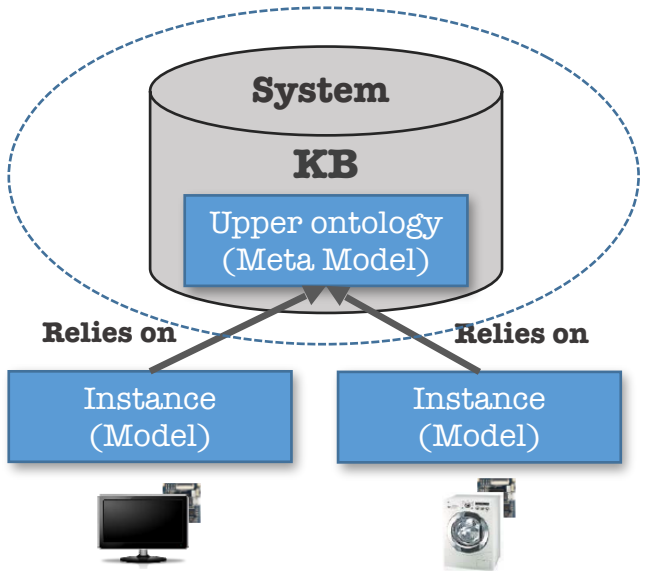
RDF/XML

```
<?xml version="1.0"?>

<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:schema="http://schema.org/">

<rdf:Description rdf:about="https://www.w3.org/People/Berners-Lee/">
  <schema:birthDate>1966-06-08</schema:birthDate>
  <schema:birthPlace rdf:resource="http://dbpedia.org/resource/London"/>
</rdf:Description>

</rdf:RDF>
```

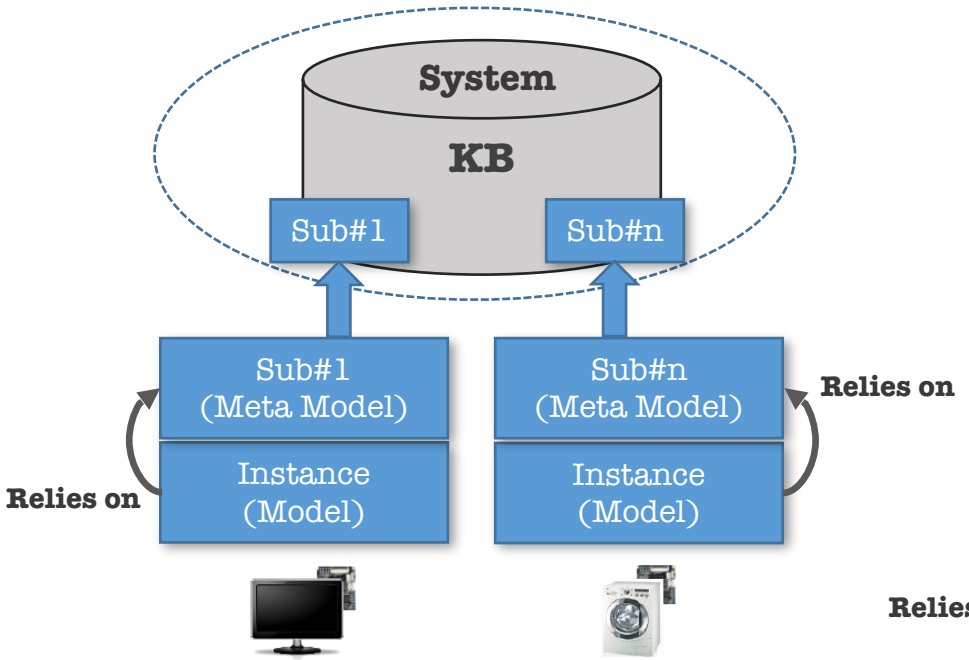
JSON-LD

```
{
  "@context": {
    "dbpedia": "http://dbpedia.org/resource/",
    "schema": "http://schema.org/"
  },
  "@id": "https://www.w3.org/People/Berners-Lee/",
  "schema:birthDate": "1955-06-08",
  "schema:birthPlace": {
    "@id": "dbpedia:London"
  }
}
```

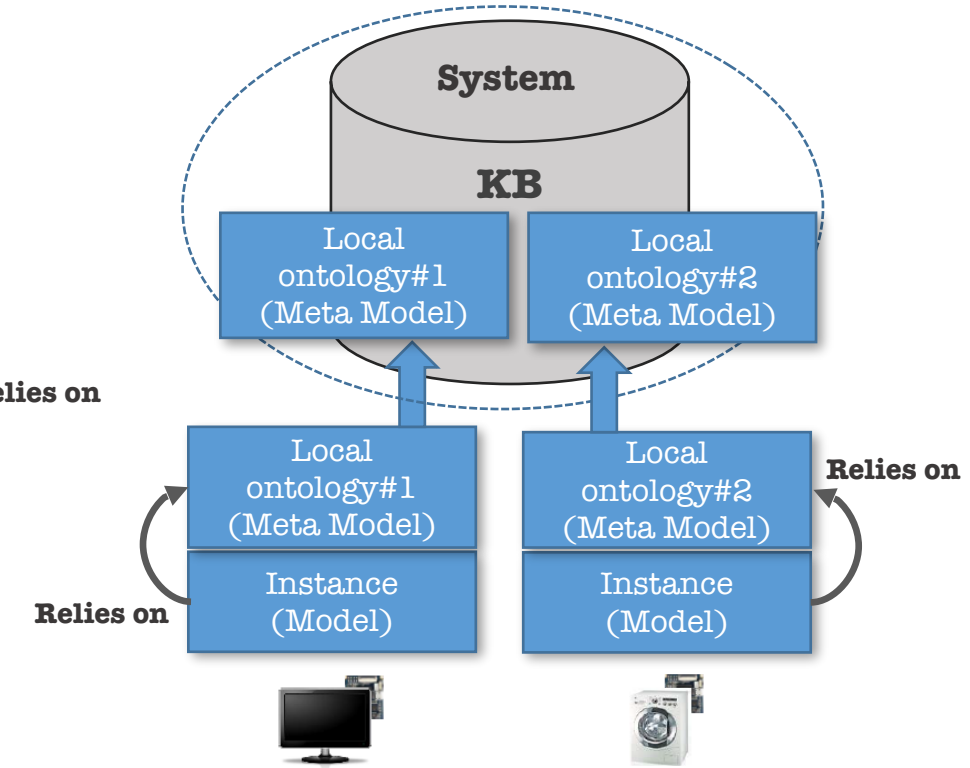
Architectural approaches in SWoT



Global approach : instance descriptions are based on a common upper ontology



Fragmented upper ontology approach : instance descriptions are based on fragments of a common upper ontology. The KB is enriched over time and contains only the needed knowledge.

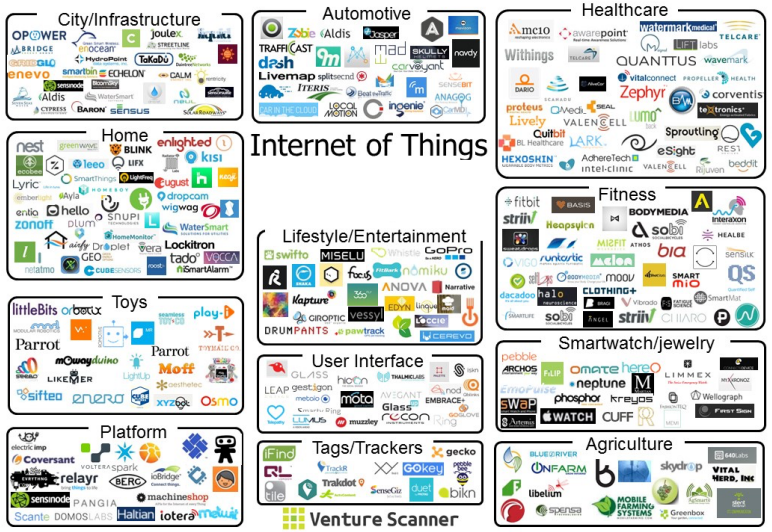


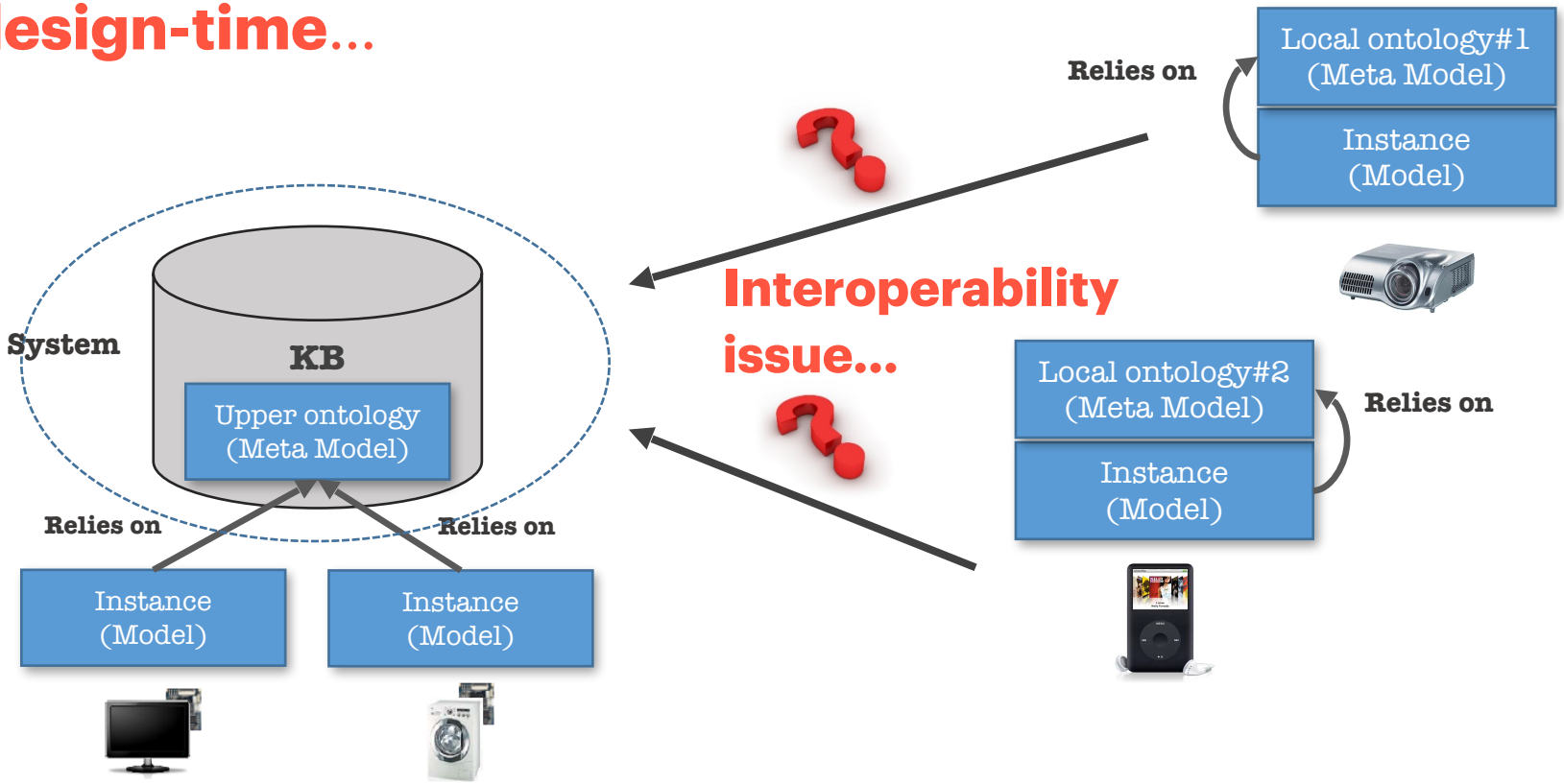
Heterogeneous approach : instance descriptions are based on their own ontology. The KB is enriched over time with new meta models.

Research challenges

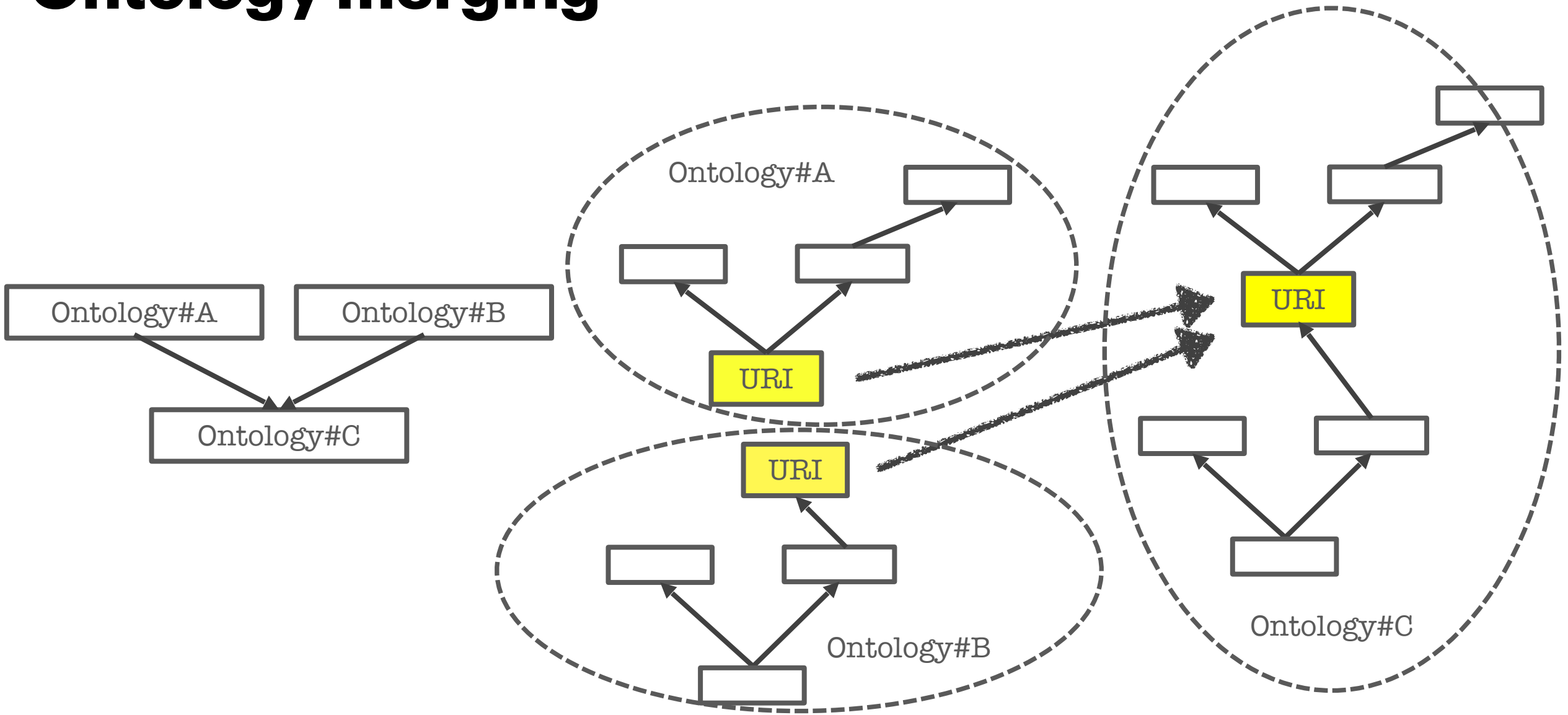
Full interoperability not yet a reality…

- Numerous actors in the IoT world, developping (or not) their own ontologies
- Most of the current SWoT applications rely on **application/domain specific ad-hoc ontologies**
- **Problem :** SWoT applications have to **integrate heterogeneous models & meta models unknown @design-time**…



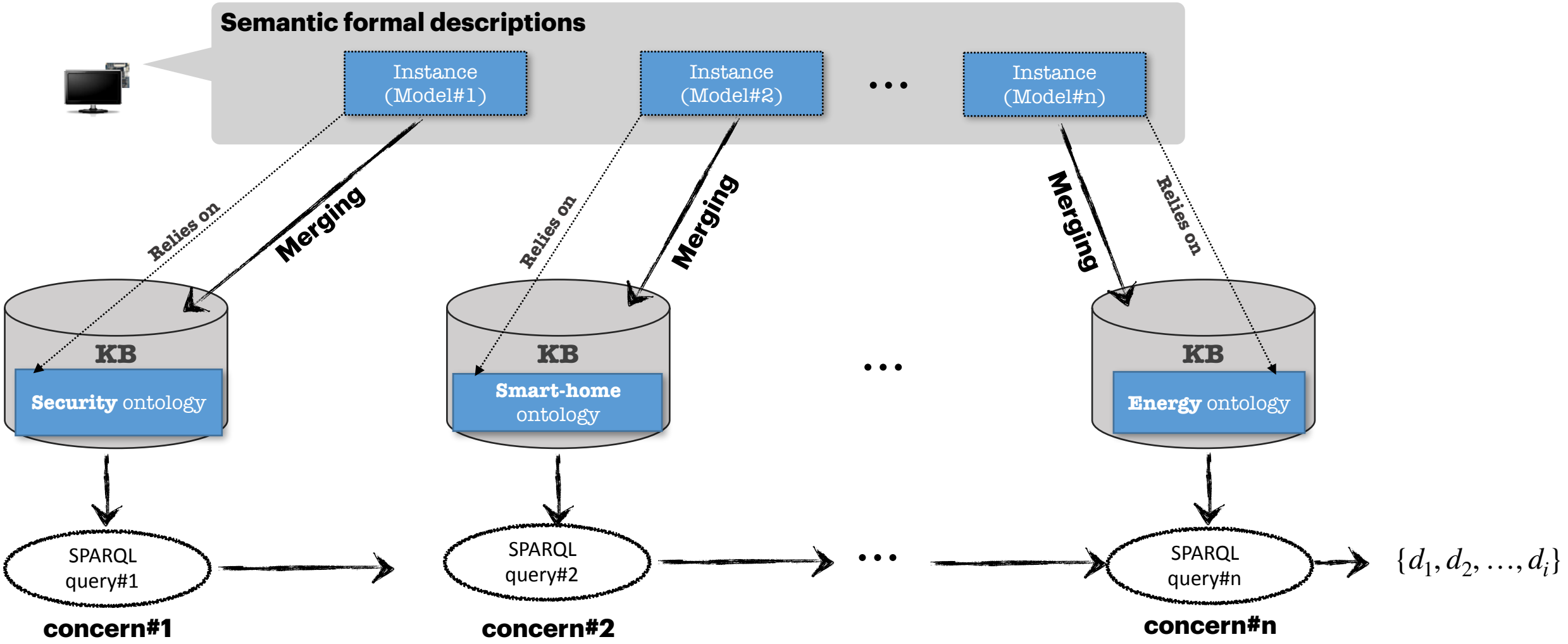


Ontology merging

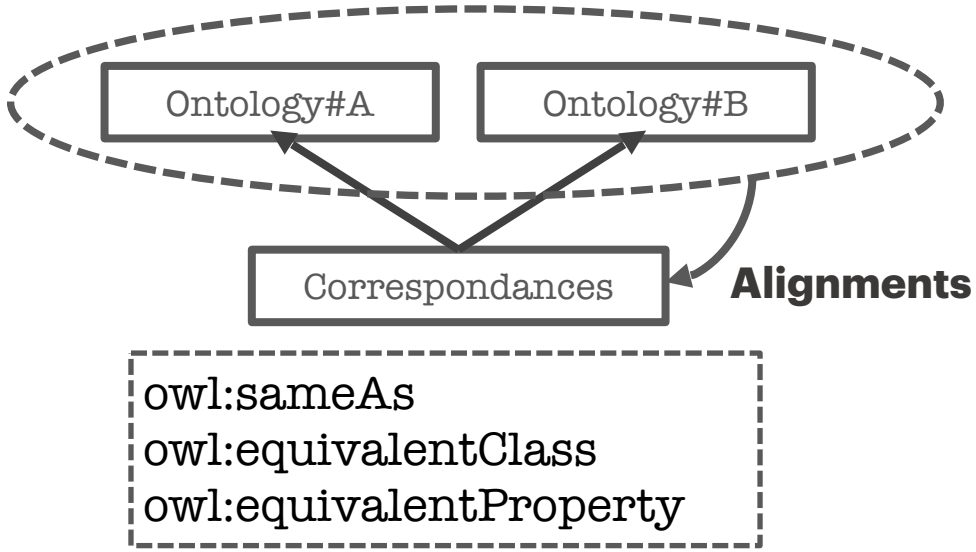


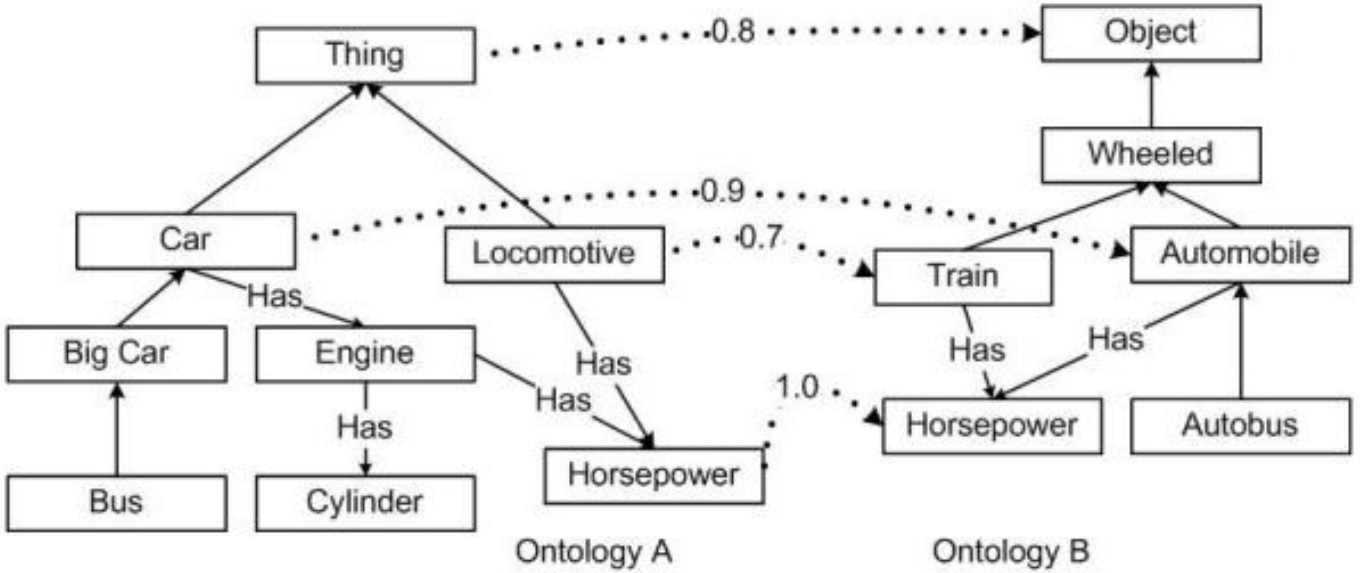
Semantic-based service selection

Concern-centric approach [DEMO]



Ontology alignment & Mapping 🔥





Example from http://www.webology.org/2006/v3n3/a28.html

- Several algorithms available...
  - **Syntactic** alignment,
  - **Structural** alignment,
  - etc.

- Alignments are not perfect and depend on a pre-defined threshold value...

http://lov4iot.appspot.com/?p=ontoMappingTool_sota

Some tools...

Some tools (1/2)...

- **Ontology engineering**
  - Protégé (http://protege.stanford.edu/)
- **Java framework**
  - **Knowledge bases**
    - Apache Jena (https://jena.apache.org/)
    - The OWL API (http://owlapi.sourceforge.net/)
  - **Reasoners**
    - Pellet (https://github.com/Complexible/pellet)
    - HermiT (http://www.hermit-reasoner.com/)
  - **Alignment engines**
    - Alignment API (http://alignapi.gforge.inria.fr/)

Some tools (2/2)…

- **Ontology search engines**
  - Watson (http://watson.kmi.open.ac.uk/WatsonWUI/)
  - Swoogle (http://swoogle.umbc.edu/)
- **Ontology online validators**
  - W3C (https://www.w3.org/2001/sw/wiki/SWValidators)

- **Additional tools can be found here :**
  - http://www.sensormeasurement.appspot.com/?p=semanticTool