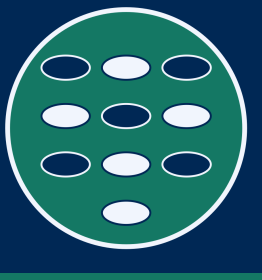
Norges miljø- og biovitenskapelige universitet

Institutt for datavitenskap



Digitalisering på Ås

DAT121
Introduction to data science

2 Data and objects

2.4 Semantic interoperability
2.5 Knowledge graphs
2.6 Querying

Norwegian University of Life Sciences

Discussion

Think of a concrete example where:

- **Multiple systems** (more than two) need to **exchange information**
- They cannot assume that the other parties use exactly the same software

How is this solved in these cases?

- Is the solution based on *translating/converting* or *one agreed standard*?
- To what extent does the methodology succeed at reaching *agreement on*
  - … *the format* or low-level structure of the data?
  - … *the meaning* and information content of what is communicated?
  - … *the protocols*, practices, and patterns of use and access?
- Are the data and the software *co-designed* or *separately designed*?

What do you see?

Use only simple sentences consisting of:

- A **subject**
- A **predicate**
- An **object**

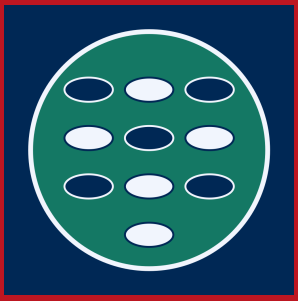
Such as:

"**The elephant is dancing in the-room**."

"**The wheel is part of the-car**."

Noregs miljø- og biovitskaplege universitet

Institutt for datavitskap



Digitalisering på Ås

2 Data and objects

2.4 Semantic interoperability

DAT121

18. august 2023

JSON-LD based data exchange

JSON: JavaScript Object Notation

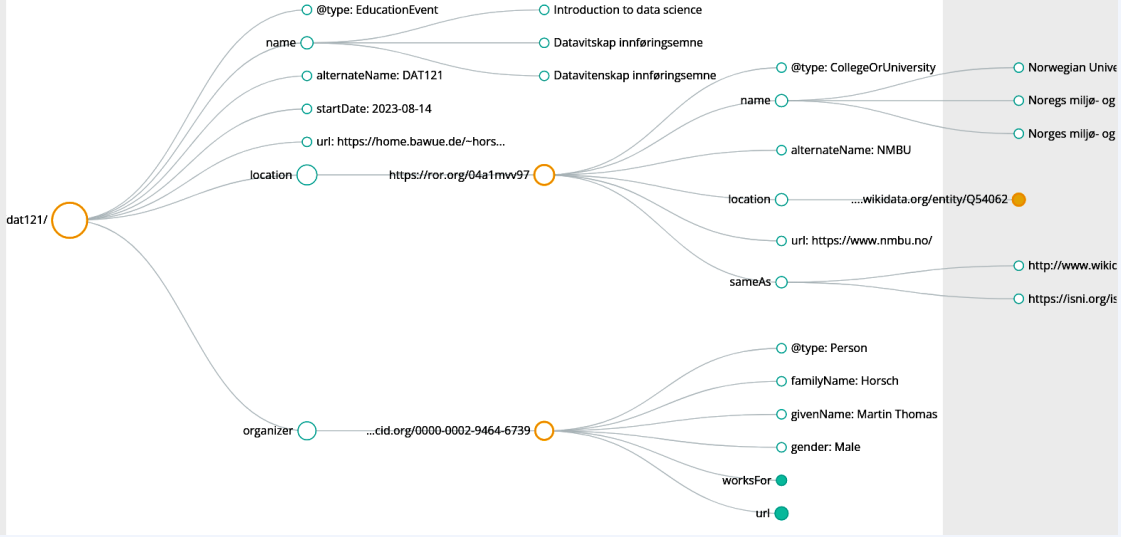
JSON-LD: Extension of JSON to deal with linked data (knowledge graphs)

JSON-LD Playground (https://json-ld.org/playground/)

JSON-LD Input | Options

```
{
   "@context": "https://schema.org/",
   "@id": "https://home.bawue.de/~horsch/teaching/d
   "@type": "EducationEvent",
   "name": ["Introduction to data science", "Datavi
   "alternateName": "DAT121",
   "startDate": "2023-08-14",
   "url": "https://home.bawue.de/~horsch/teaching/d
   "location": {
      "@id": "https://ror.org/04a1mvv97",
      "@type": "CollegeOrUniversity",
      "name": ["Norwegian University of Life Scienc
biovitenskapelige universitet"],
      "alternateName": "NMBU",
      "location": {
         "@id": "http://www.wikidata.org/entity/Q54
         "@type": "City",
         "name": "Ås",
         "alternateName": "Aas"
      },
      "url": "https://www.nmbu.no/",
```

Expanded | Compacted | Flattened | Framed | N-Quads | Canonized | Table | Visualized | Signatures

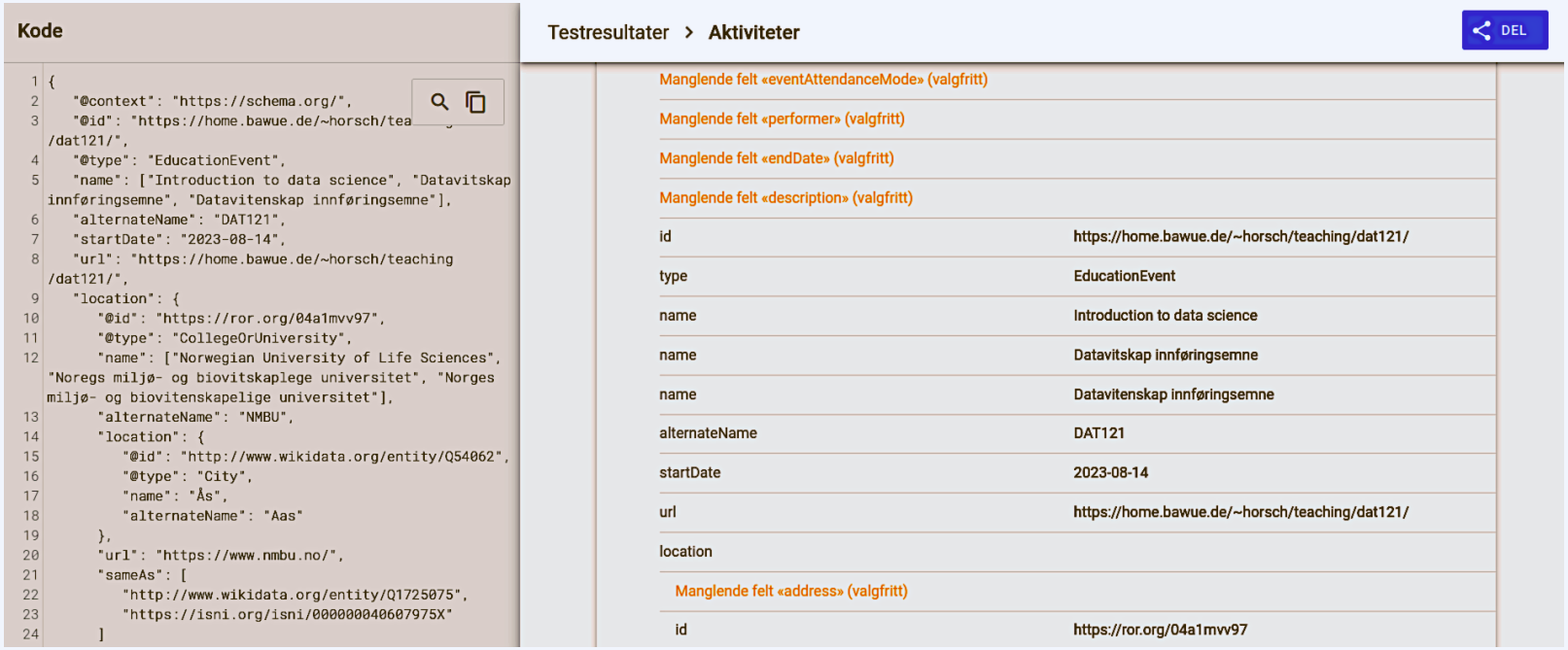


JSON-LD based data exchange

JSON: JavaScript Object Notation
JSON-LD: Extension of JSON to deal with linked data (knowledge graphs)

JSON-LD

JSON-LD Playground (https://json-ld.org/playground/)
Google's Schema Markup Validator (https://validator.schema.org/)
Google's Rich Results Test (https://search.google.com/test/rich-results)



Resource description framework (RDF)

Semantic technology can facilitate the integration of data and software into a coherent framework, permitting multiple components to become interoperable.

On the semantic web, data and metadata are provided as RDF triples:

Triples: Individual Relation Individual. (Subject Predicate Object.)

Example: The fox *f* *eats* the chicken *c*.

(Other kind of triples: Individual "a" Concept. Example: *f* a Fox.)

RDF is the Resource Description Framework, which specifies the semantic web.

triple

Resource description framework (RDF)

resource

Semantic technology can facilitate the integration of data and software into a coherent framework, permitting multiple components to become interoperable.

On the semantic web, data and metadata are provided as RDF triples:

Triples: Individual Relation Individual. (Subject Predicate Object.)

Example: The fox *f* *eats* the chicken *c*.

(Other kind of triples: Individual "a" Concept. Example: *f* a Fox.)

RDF is the Resource Description Framework, which specifies the semantic web. In this context, a **resource** is any of the following:

an **individual** (*i.e.*, object); a **concept** (*i.e.*, class); a property/**relation**.

Resources are referenced by using **Internationalized Resource Identifiers (IRIs)**.

Open world assumption

Triples: Individual Relation Individual. (Subject Predicate Object.)

(1) Frank is_father_of Robert.

(2) Frank is_different_from Nick.

Q: "Is Nick the father of Robert?"
A: "No, he is not."

Human is a concept.
Frank, Robert, etc., are Humans.

Cardinality restriction:
Every Human has exactly 1 father.

(3) Frank is_father_of Anna.

Q: "How many children does Frank have?"
A: "At least two."

Anna is_different_from Robert.

"How many different X are there such that Frank is_father_of X?"

Principle: Open world assumption

Since relevant information may distributed over the semantic web, rather than from the presently considered source only, **available knowledge is assumed to be incomplete**. (Contrast this with a closed, monolithic database architechture.)

Open world assumption

Triples: Individual Relation Individual. (Subject Predicate Object.)

(1) Frank is_father_of Robert.

(2) Frank is_different_from Nick.

Q: "Is Nick the father of Robert?"

A: "No, he is not."

Human is a concept.

Frank, Robert, etc., are Humans.

Cardinality restriction:

Every Human has exactly 1 father.

Related teaching activities: INN351

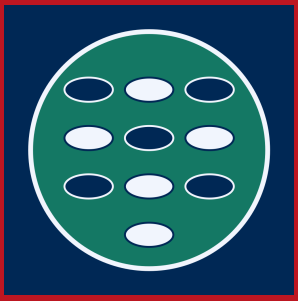
"Enterprise architecture for the digital age"

"digitalization is approached as an offensive move or a defensive action […]. This course takes a strategic and managerial approach to Enterprise Architecture. It considers the problem of Enterprise Architecture as a business, rather than a technical problem. The course clarifies that enterprise architecture is different from IT architecture and therefore it cannot be abdicated to the IT experts." (Yes, this is the actual course description.)

https://www.nmbu.no/en/course/inn351

Noregs miljø- og biovitskaplege universitet

Institutt for datavitskap



Digitalisering på Ås

2 Data and objects

2.4 Semantic interoperability

2.5 Knowledge graphs

DAT121

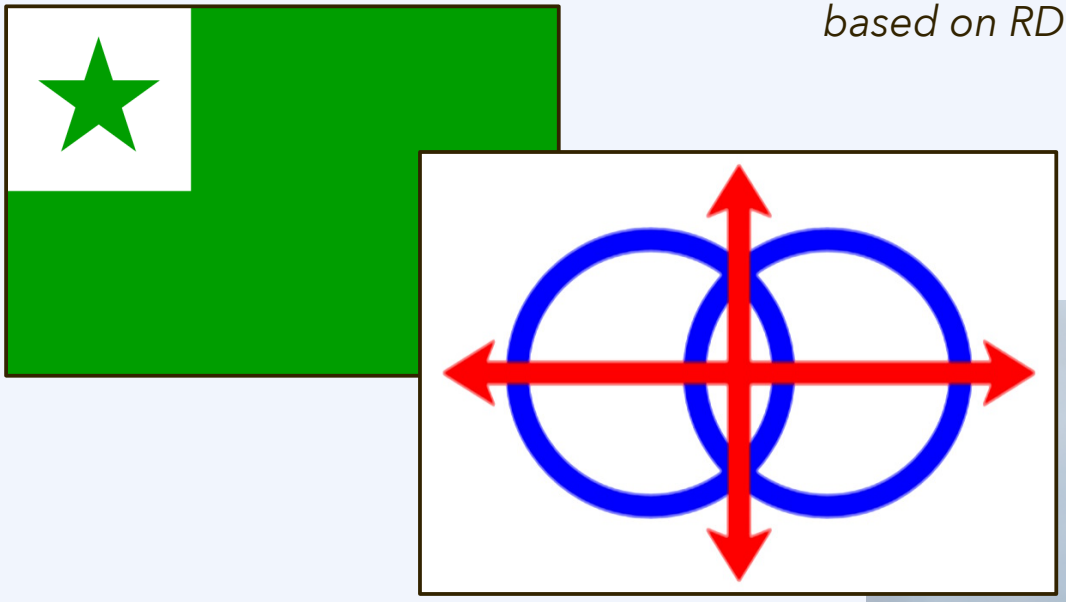
18. august 2023

What did you see?

Ontologies and the web ontology language (OWL)

„One World Language“

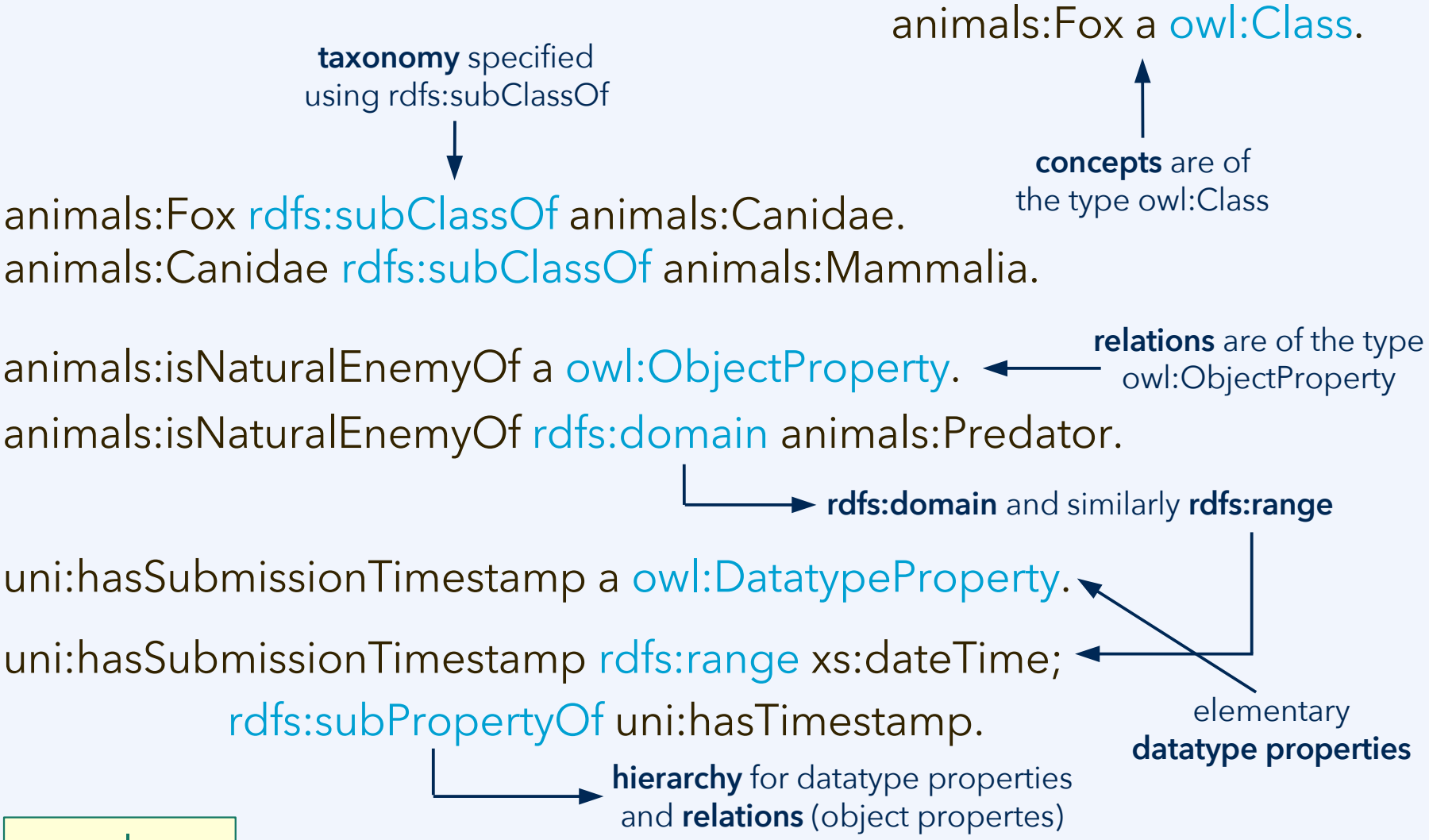
or now, “Web Ontology Lanugage (OWL),” based on RDF Schema (RDFS)



Protégé is an important tool for working with RDFS and OWL:

https://protege.stanford.edu/

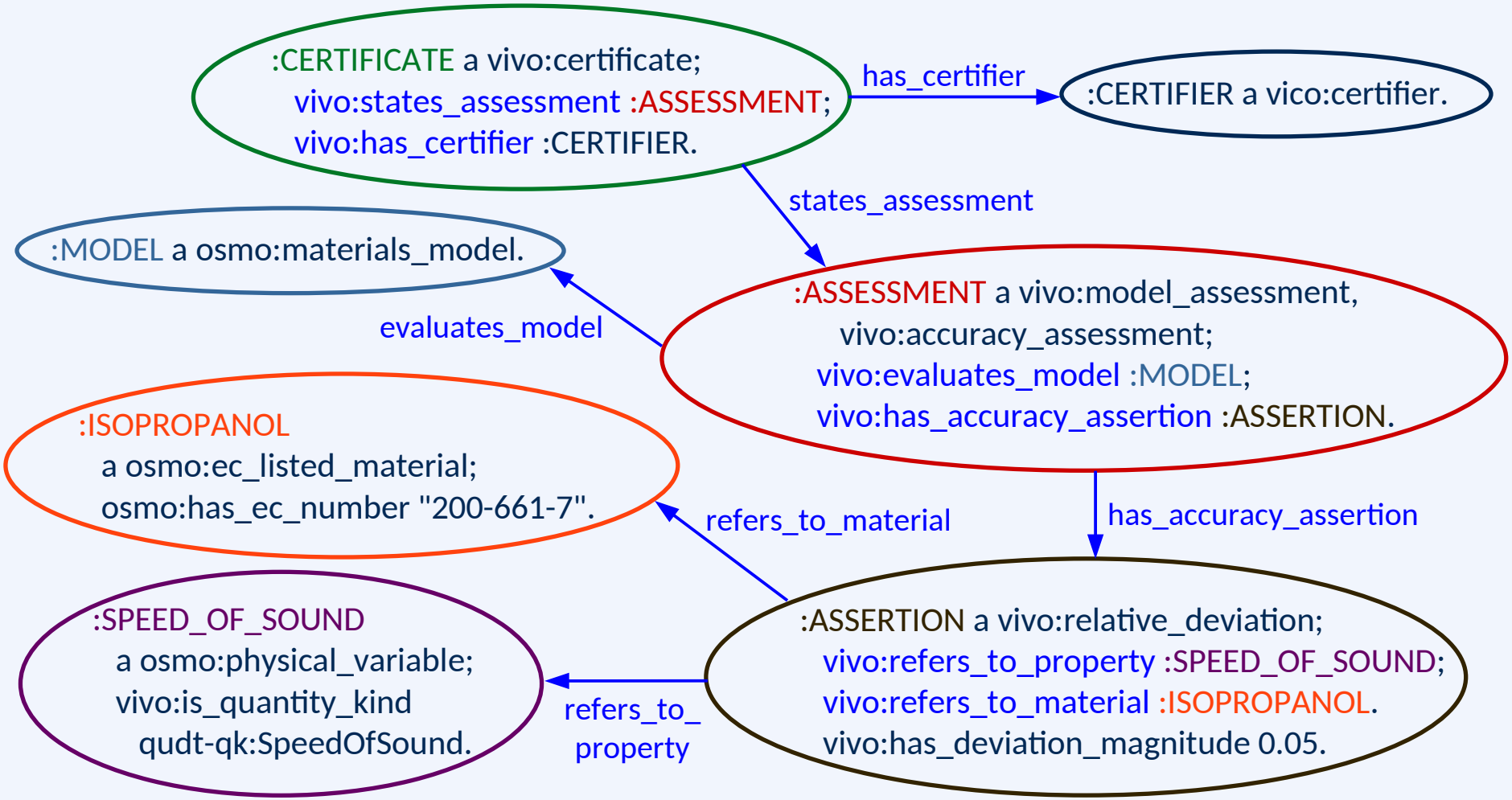
OWL ontologies using terse triple language (TTL)



ontology

Graph representation of linked data

knowledge graph



Knowledge bases: TBox and ABox

A **knowledge base** for linked data consists of two components:

> **Definition:** A **knowledge base**, given by $K = (T, A)$, consists of an ontology T, describing universals, and a set of assertions A describing concrete instances of these universals.

ABox

particular:	**individual** (entity, object)	**relationship**	**property** (sometimes: attribute)

universal:	**concept** (entity type, class)	**relation** (relationship type) (in OWL: ObjectProperty)	**attribute** (sometimes: attribute type) (in OWL: DatatypeProperty)

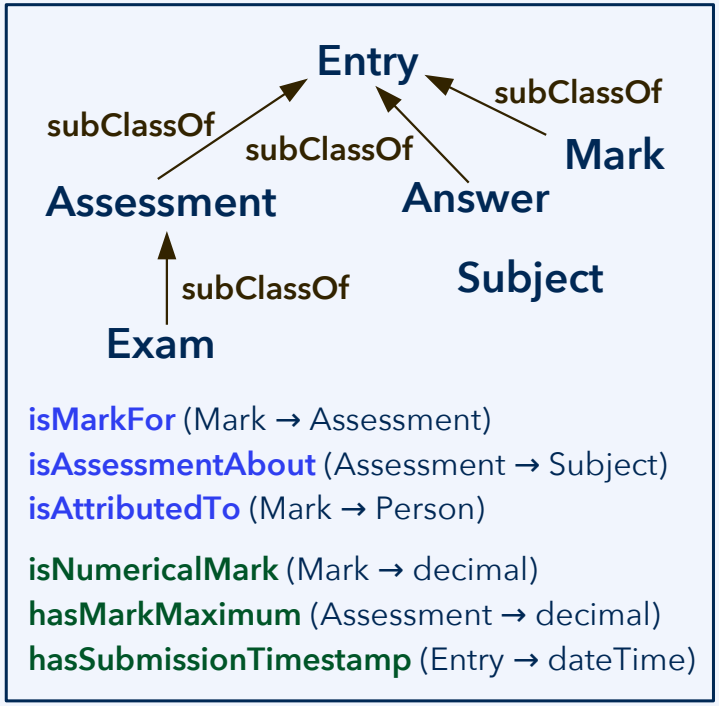
TBox

knowledge base

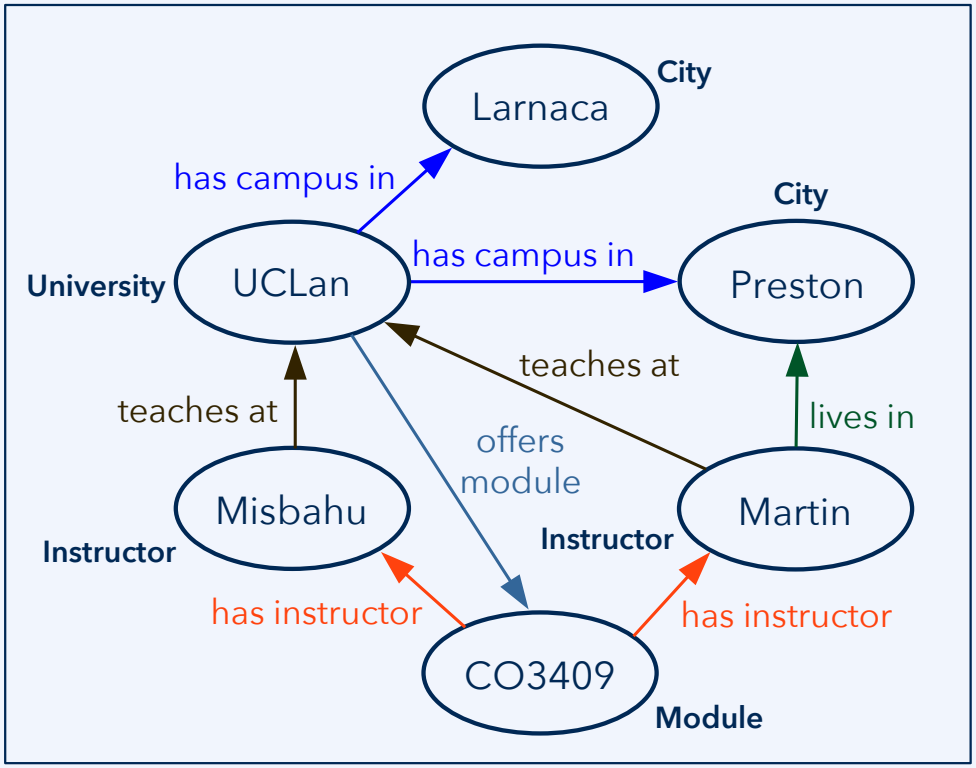
Knowledge bases: TBox and ABox

A **knowledge base** for linked data consists of two components:

terminological box (TBox), **ontology**, or RDF schema

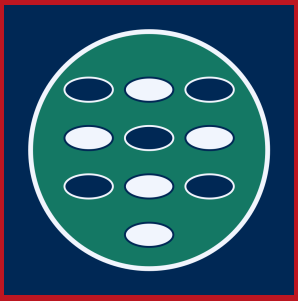


assertional box (ABox) or **knowledge graph**



Noregs miljø- og biovitskaplege universitet

Institutt for datavitskap



Digitalisering på Ås

2 Data and objects

2.4 Semantic interoperability

2.5 Knowledge graphs

2.6 Semantic querying

DAT121

18. august 2023

Querying graph databases using SPARQL

SPARQL is a recursive acronym: "SPARQL Protocol and RDF Query Language." An interface that can handle SPARQL queries is called a *SPARQL end point*.

The **syntax of SPARQL** is reminiscent of SQL, but at its core is given by

- **RDF triples**, using RDF schema and OWL, in **TTL notation**;
- Some elements of the triples are **wildcards**, *i.e.*, free variables.

```
SELECT ?person ?address
WHERE {
  ?person uni:teachesAt ?institution.
  ?institution uni:hasCampusIn ?address.
  ?person uni:livesIn ?address.
}
```

"What persons do you know who are residents of a city in which their institution also has a campus? Output a table of these persons together with the corresponding cities."

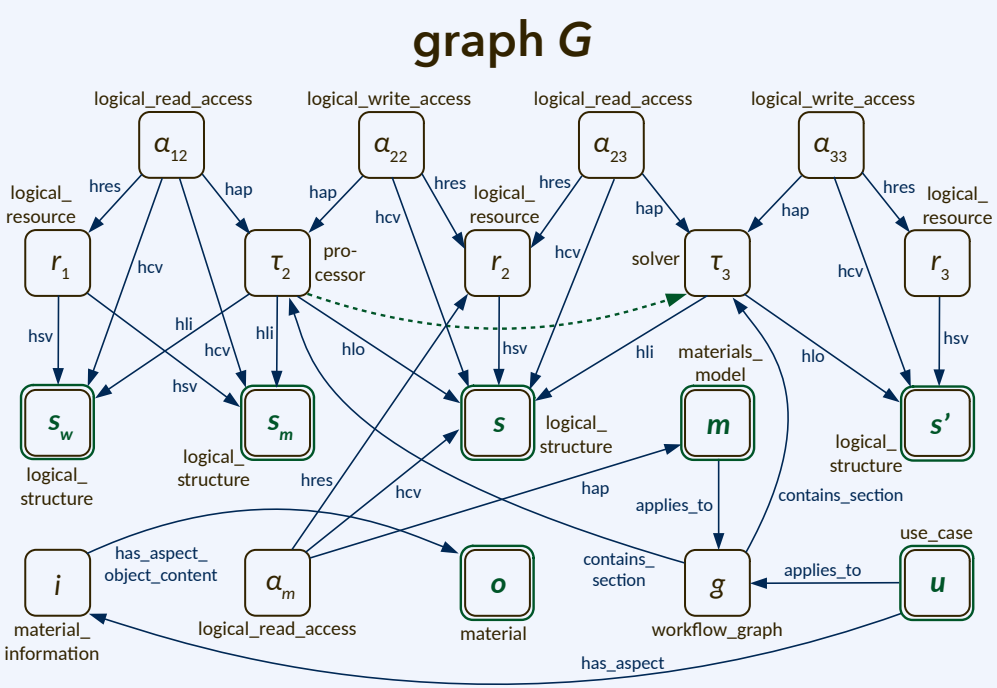
The **semantics of SPARQL** is given by the correct response to a SPARQL query, which consists of a *table with all matching valuations of the selected wildcards*.

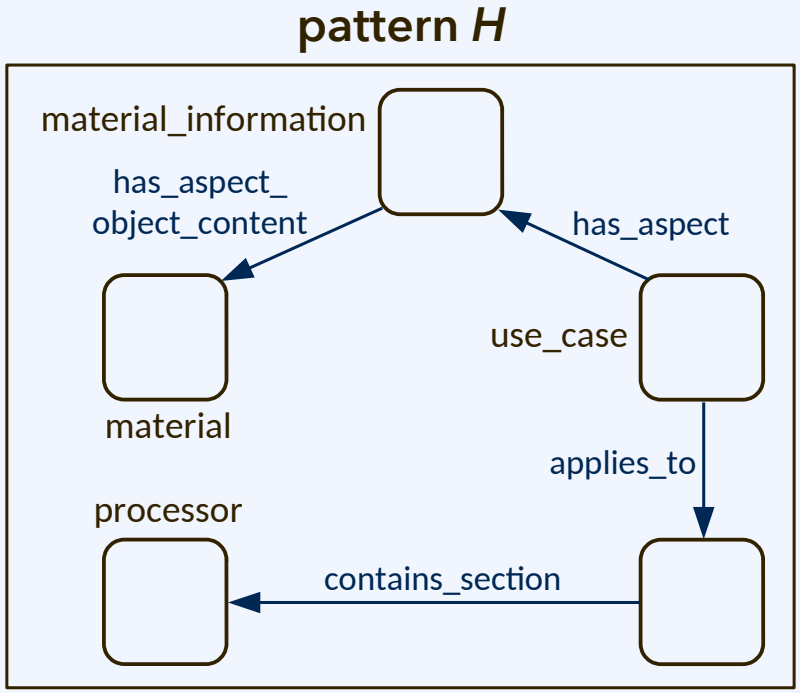
SPARQL querying therefore corresponds to the **subgraph matching problem** from graph theory: It looks for occurrences of a pattern within a larger graph.

Querying graph databases using SPARQL

Subgraph matching problem (NP-complete):

Given a graph *G* and a pattern *H*, does G contain a subgraph isomorphic to *H*?



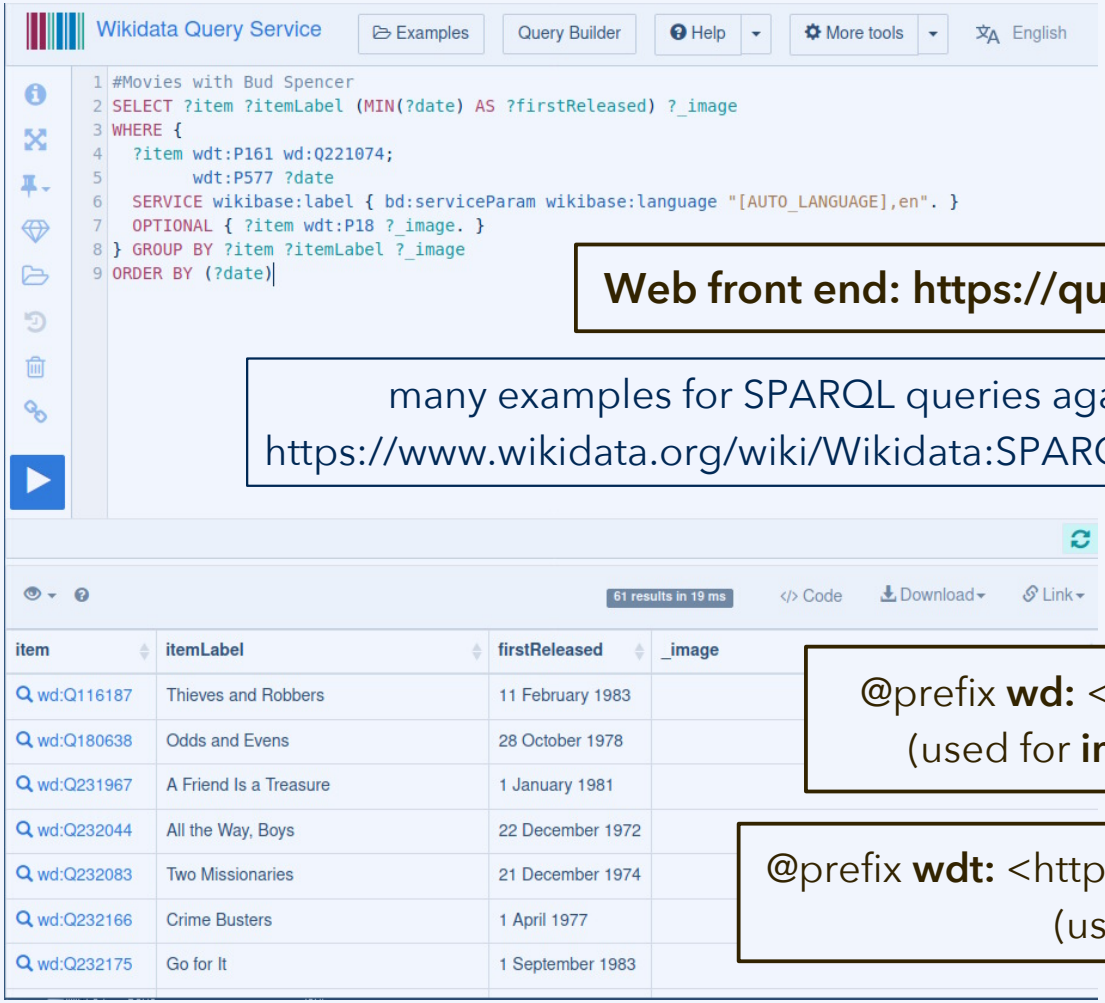


SPARQL querying therefore corresponds to the **subgraph matching problem** from graph theory: It looks for occurrences of a pattern within a larger graph.

(example from P. Klein et al., Proceedings of JOWO 2021)

Querying graph databases using SPARQL

Wikidata SPARQL end point



Web front end: https://query.wikidata.org/

many examples for SPARQL queries against Wikidata are available at
https://www.wikidata.org/wiki/Wikidata:SPARQL_query_service/queries/examples

@prefix **wd:** <https://wikidata.org/wiki/>
(used for **individuals** and **concepts**)

@prefix **wdt:** <https://wikidata.org/wiki/Property:>
(used for **relations**)

Querying graph databases using SPARQL

Linked data that are formalized as RDF triples can be stored:

- in **graph databases**, also called **triple stores** (*e.g.*, Apache Jena Fuseki)
- using JSON based **noSQL databases** such as MongoDB
- architectures containing such components need to facilitate **querying**

Linked data are queried using the SPARQL query language:

- in a "SELECT … WHERE …" query, a **graph pattern** is defined by **triples**
- the graph pattern contains free variables, some of which are selected
- the server needs to *identify all the occurrences* of the graph pattern
- the response **returns tabular data** with one column per selected variable

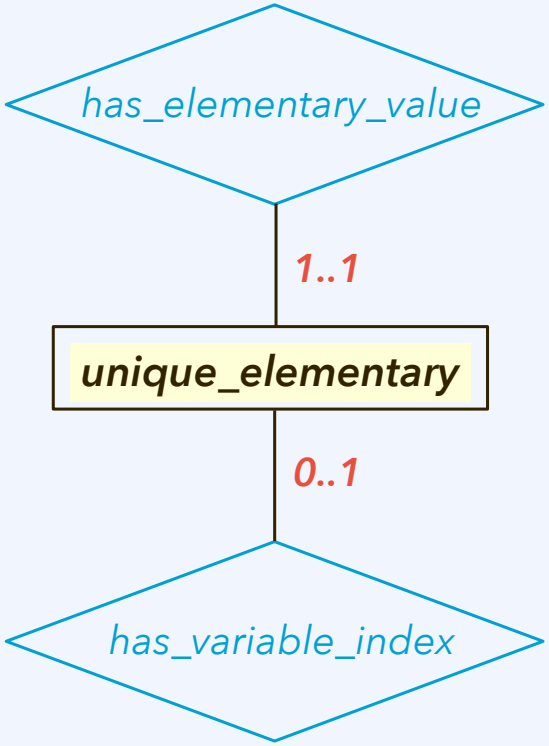
A **SPARQL end point** is a host that provides a SPARQL querying service, most frequently via a RESTful API and accessible through a web frontend.

Shape constraint language (SHACL)

An API usually needs to specify what information content is to be exchanged.

Shapes Constraint Language (SHACL) can be used for such specifications.[1]

```
:unique_elementary_shape a sh:Shape;
  sh:targetClass :unique_elementary;
  sh:property [
       sh:path :has_elementary_value;
       sh:minCount 1;
       sh:maxCount 1
     ], [
       sh:path :has_variable_index;
       sh:maxCount 1
     ].
```

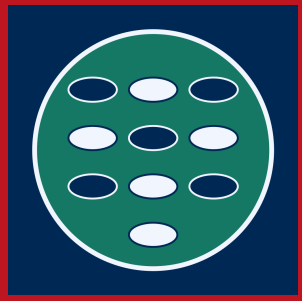


The open world assumption is **not** applied when evaluating SHACL constraints!

[1] W3C recommendation, https://www.w3.org/TR/shacl/, **2017**.

Noregs miljø- og biovitskaplege universitet

Institutt for datavitskap



Digitalisering på Ås

Conclusion

DAT121

18. august 2023

Glossary terms

Proposed glossary[1] terms:

- How do we best define them? Is the definition controversial?
- What is the best translation into Norwegian bokmål/nynorsk?
- Are there more key concepts that would require an agreed definition?

left open for discussion → knowledge graph

(also: ABox)

resource

knowledge base

triple

ontology

(also: TBox)

[1]https://home.bawue.de/~horsch/teaching/dat121/glossary-en.html

Related research activities

SN Computer Science (2022) 3:282
https://doi.org/10.1007/s42979-022-01116-x

SURVEY ARTICLE

Human Emotion: A Survey focusing on Languages, Ontologies, Datasets, and Systems

Mohammed R. Elkobaisi[1] · Fadi Al Machot[2] · Heinrich C. Mayr[1]

Received: 25 May 2021 / Accepted: 28 March 2022 / Published online: 10 May 2022

Abstract
Emotions are an essential part of a person's mental state and influence her/his behavior accordingly. Consequently, emotion recognition and assessment can play an important role in supporting people with ambient assistance systems or clinical treatments. Automation of human emotion recognition and emotion-aware recommender systems are therefore increasingly being researched. In this paper, we first consider the essential aspects of human emotional functioning from the perspective of cognitive psychology and, based on this, we analyze the state of the art in the whole field of work and research to

https://dx.doi.org/10.1007/s42979-022-01116-x

Related research activities

SN Computer Science (2022) 3:282
https://doi.org/10.1007/s42979-022-01116-x

SURVEY ARTICLE

Human Emotion: A Survey focusing on Languages, Ontologies, Datasets, and Systems

https://ceur-ws.org/Vol-1164/PaperDemo03.pdf

Mohammed R. Elkobaisi[1] · Fadi Al

Received: 25 May 2021 / Accepted: 28 March 20

Abstract
Emotions are an essential part of a per
recognition and assessment can play
treatments. Automation of human emo
being researched. In this paper, we fir
tive of cognitive psychology and, base

A Behavior Centered Modeling Tool Based on ADOxx

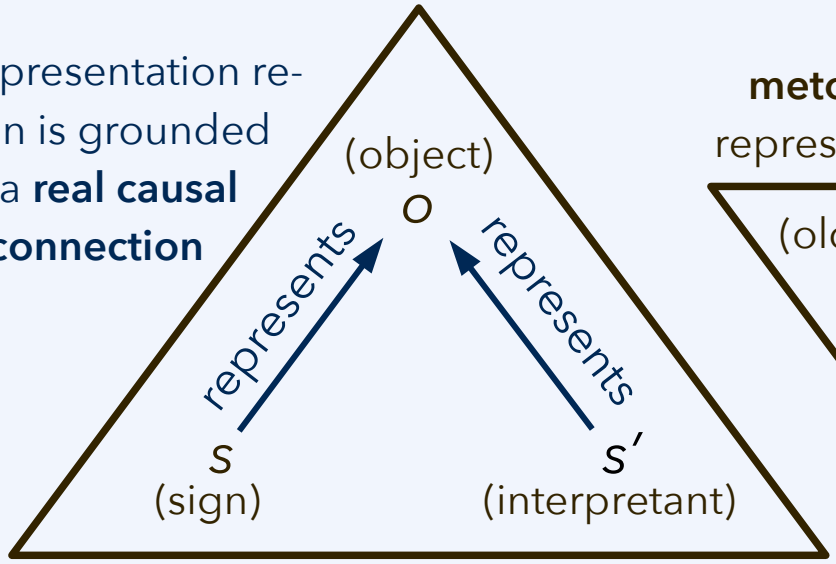
Judith Michael, Fadi Al Machot, Heinrich C. Mayr

Application Engineering Research Group, Alpen-Adria-Universität Klagenfurt, Austria
{judith.michael, fadi.almachot, heinrich.mayr}@aau.at

Abstract. Meta-modeling platforms that support the automatic generation of modeling tools open a new quality in information systems development for engineers: Emphasis can be put on the design and use of a modeling language that is customized to the particular needs and desired features. This may contribute to strengthen the information system design phase as it helps to reduce the developers' aversion against overloaded modeling languages and inflexible or expensive modeling tools. Our demo paper introduces HCM-L Modeler, a model-

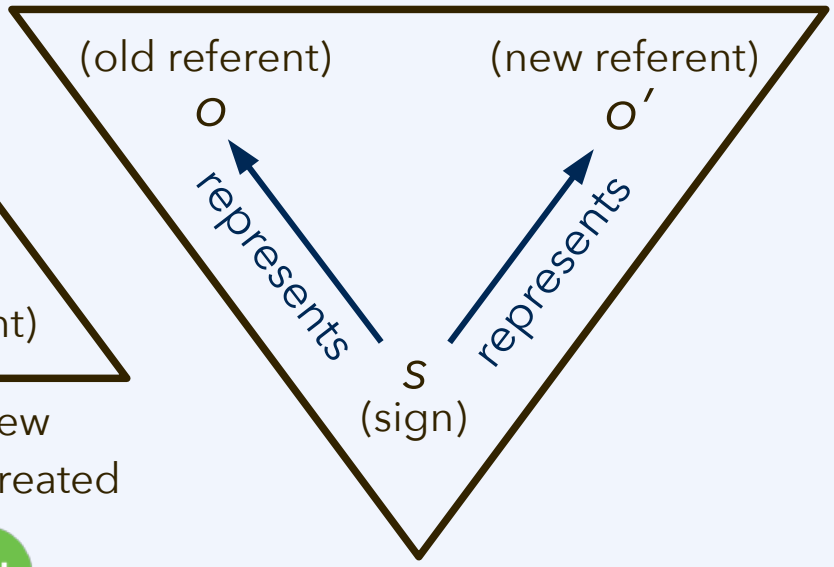
Related research activities

the representation relation is grounded in a **real causal connection**



semiosis, a process by which a new representamen, the interpretant, is created

metonymization, a process by which a representamen is assigned a new referent



C. S. Peirce

Elementary Multi-perspective Material Ontology (EMMO)

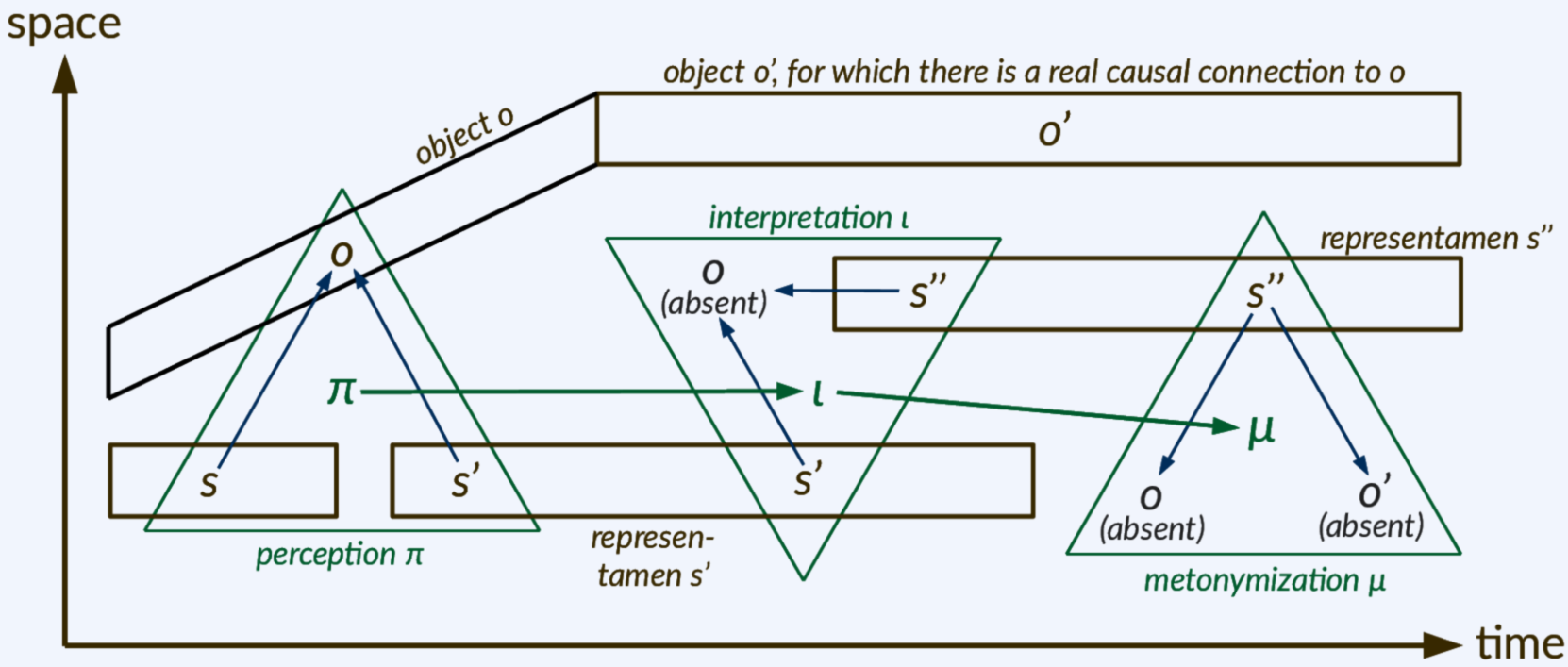
The EMMO[1] combines this with mereocausality – foundational ontology as **mereosemiotics**.

[1]The work on the EMMO (2017 – present) is coordinated by Emanuele Ghedini.

Related research activities

PIMS-II mid-level ontology:[1, 2] http://www.molmod.info/semantics/pims-ii.ttl

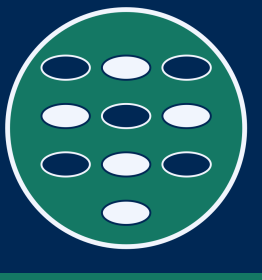
Mereosemiotics:[1-3] Combination of mereotopology and Peircean semiotics



[1]M. T. Horsch, no. 3 in *Proc. JOWO 2021*, **2021**. [2]P. Klein *et al.*, no. 26 in *Proc. JOWO 2021*, **2021**.
[3]M. T. Horsch, S. Chiacchiera, B. Schembera, M. Seaton, I. T. Todorov, in *Proc. ECCOMAS 2020*, **2021**.

Norges miljø- og biovitenskapelige universitet

Institutt for datavitenskap



Digitalisering på Ås

DAT121
Introduction to data science

2 Data and objects

2.4 Semantic interoperability
2.5 Knowledge graphs
2.6 Querying

Norwegian University of Life Sciences

Schedule for 17th and 18th August

Thursday, 17th August 2023

09.15 – 10.00 discussion and Q&A
1.5 Python libraries

10.15 – 11.00 problem solving and start of first lecture on data and objects
2.1 OOP in Python

11.15 – 12.00 problem solving and continued first lecture on data and objects
2.2 Inheritance
2.3 Conceptual modelling

13.15 – 15.00 tutorial session

Friday, 18th August 2023

09.15 – 10.00 discussion and Q&A

10.15 – 11.00 second lecture on data and objects
2.4 Semantic interoperability
2.5 Knowledge graphs
2.6 Querying

11.15 – 12.00 Fadi Al Machot's presentation on research and Master topics

The afternoon of 18th August is reserved for the immatriculation.

Schedule for 21st and 22nd August

Norwegian University of Life Sciences

Monday, 21st August 2023

09.15 – 10.00 discussion and Q&A

10.15 – 11.00 first lecture on regression
- *3.1 Supervised learning*
- *3.2 statsmodels*
- *3.3 Validation and testing*

11.15 – 12.00 problem solving / examples

13.15 – 15.00 tutorial session

Tuesday, 22nd August 2023

09.15 – 10.00 discussion and Q&A

10.15 – 11.00 second lecture on regression
- *3.4 Influence diagrams*
- *3.5 Time series*
- *3.6 Autocorrelation*

11.15 – 12.00 problem solving / examples

13.15 – 15.00 tutorial session