KNOWDIVE

KGE - Knowledge Graph Engineering

Knowledge Graphs

Fausto Giunchiglia

Contents

1 KG definiton

2 KGs for applications

3 KG production

4 KG users

Knowledge Graph definition

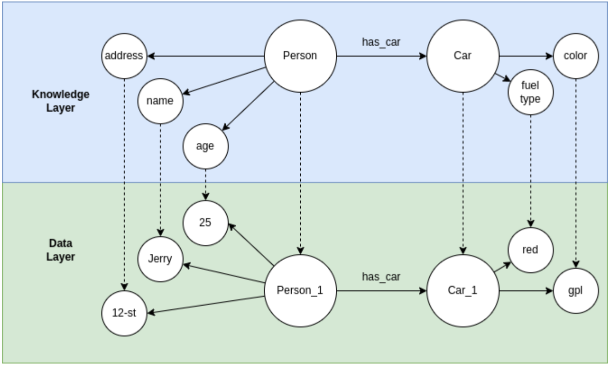
A Knowledge Graph K, can be defined as follows:

$$KG = (E, D, R, A)$$

Where:

- E: is the set of real-world objects types, called *Entity Types* (or ETypes).
- D: is the set of real-world objects representations, called *Entities*. The Entities are ETypes instantiation.
- R is the set of properties used to denote the ETypes. The elements of R, can be properties related to a single EType, called *data properties*, or properties used to define relations among different ETypes, *called object properties*.
- A: is the set of property values denoting the attributes of the Entities. Each attribute, associated to one and only one property, instantiates the relative data/object property.

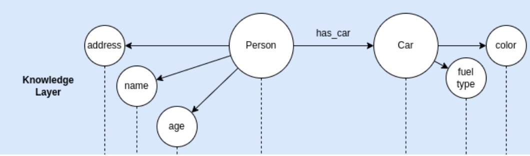
Knowledge Graph example



- E = {Person, Car}
- D = {Person_1, Car_1}
- R = {address, name, age, color, fuel type, has_car}
- A = {12-st, Jerry, 25, red, gpl}

Knowledge Layer

- The KG's Knowledge Layer is composed by the elements of E (ETypes) plus the element of R (properties definition).
- It defines the KG's structure (or schema).
- It is usually defined using an ontology modeled to represent the information to be maintained in the KG.



Ontology

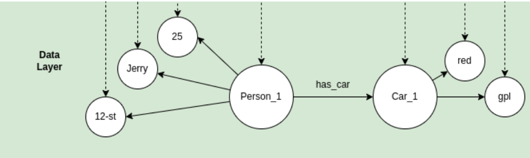
- "An ontology is a formal, explicit specification of a shared conceptualization"
  -by Gruber (1993) and modified by Studer et. al (1998)

- Ontologies are used to capture knowledge about some domain of interest. An ontology describes the concepts in the domain and also the relationships that hold between those concepts

- Ontologies are crucial for attributing semantics to Knowledge Graphs (KGs) which model ground-truth

Data Layer

- The KG's Data Layer is composed by the elements of D (Entities) plus the element of A (attributes definition).
- It contains the data values instantiating the KG's structure.



Contents

1 KG definiton

2 **KGs for applications**

3 KG production

4 KG users

KG-based Apps — examples

- Google Knowledge Panel

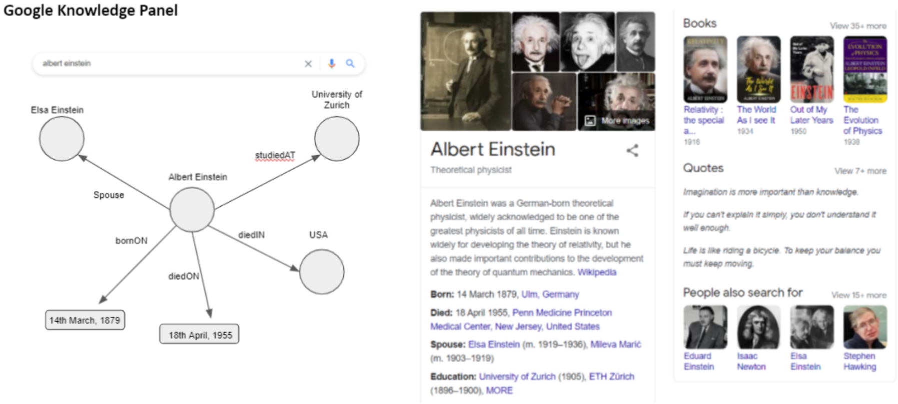


Figure: Mohit M., A guide to Knowledge Graphs, Aug 30, 2021

KG-based Apps — examples

- InteropEHRate EU project

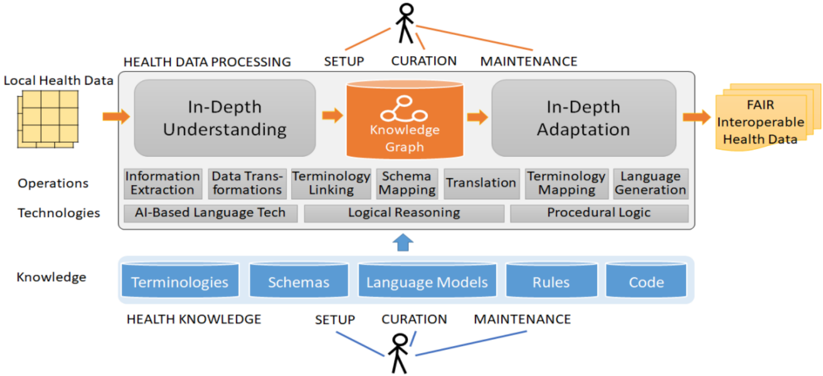


Figure 1: High-level architecture of the InteropEHRate Health Services and the way they are overseen by a human data manager.

KG-based Apps — examples

Many domain specific KGs have been produced supporting tasks like:

- Data Governance
- Automated Fraud Detection
- Knowledge Management
- Insider Trading
- Health Data Interoperability

KG-based Apps — examples

While there are several small-sized and domain-specific KGs, on the other hand, we also have many huge-sized and domain-agnostic KG that contains facts of all types and forms.

- **DBpedia**: is a crowd-sourced community-based effort to extract structured content from the information present in various Wikimedia projects.
- **Freebase**: a massive, collaboratively edited database of cross-linked data. Touted as "an openly shared database of the world's knowledge". It was bought by Google and used to power its own KG. In 2015, it was finally discontinued.
- **OpenCyc**: is a gateway to the full power of Cyc, one of the world's most complete general knowledge base and commonsense reasoning engines.
- **Wikidata**: is a free, collaborative, multilingual database, collecting structured data to provide support for Wikimedia projects.
- **YAGO**: huge semantic knowledge base, derived from Wikipedia, WordNet, and GeoNames.

Contents

1 KG definiton

2 KGs for applications

3 **KG production**

4 KG users

Data Virtualization & Materialization

- In order to support the relative applications/systems/services (see examples above), KGs are produced and exploited in different ways.
- Like their structure, the creation of a KG can be focused only on the knowledge layer, or considering also the data layer.
- Based on the layers considered, the creation of KGs can be divided in two different categories:
  - **Data Virtualization**: only the knowledge layer needs to be modeled when creating the knowledge graph.
  - **Data Materialization**: both knowledge and data layers are involved in the KG's creation.

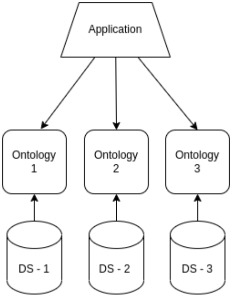
Data Virtualization

- The data virtualization techniques aim to unify the interpretation of multiple data sources, in order to access their information through a single reference schema.

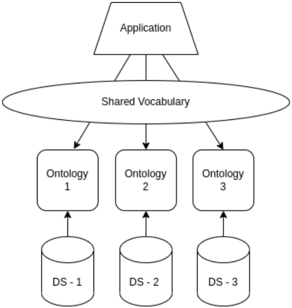
- Also known as Ontology Based Data Access (OBDA), these techniques don't manage (extract, transform) the data required to satisfy a given purpose. Instead, they build on top of multiple data sources a common ontology as single access point for the information of multiple sources.

Data Virtualization - LAV & GAV

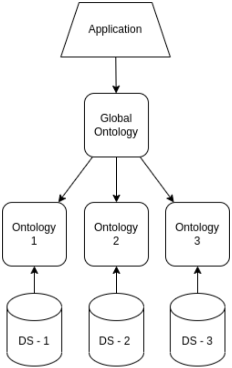
The most common OBDA approaches are:



Local As a View (LAV)



Hybrid LAV



Global As a View (GAV)

Data Materialization

- The data materialization techniques consider both knowledge and data layer during the construction of KGs.
- Known as Knowledge Graph Construction (KGC) these techniques involves the collection (or extraction) of data from the relevant sources, and the subsequent transformation of such data in order to fit the KG final requirements.
- On the other layer, also the knowledge (reference schemas, ontologies) is modeled in order to create the unique KG's schema.

Data Materialization - Activities

KGC includes several activities to be considered in order to build a KG (the order below is not the order of execution):

- Data collection (extraction and scraping)
- Data cleaning and formatting
- Data integration
- Schema modeling
- Schema matching
- Schema integration

Contents

1 KG definiton

2 KGs for applications

3 KG production

4 **KG users**

Data producer & consumer

The KG-based applications, as well as the KG's exploits mentioned, involve two different user categories, defined as follows:

- **Data Producer**: this users aim to produce KGs from scratch, starting with data which needs to be collected, structured, modeled, integrated, ...
- **Data consumer**: this users aim to produce a KG by exploiting already existing KGs. To this end the data consumer needs to identify and compose KGs, and other resources, previously produced for different purposes.

Data producer & consumer - purpose

The needs of data producer and consumer lead them to use different KG production techniques

- The data consumer exploits already existing resources, without the need of (or with a reduced effort in) data management.
  - Therefore, the consumer is the user more interested in data virtualization techniques, supporting her in the definition of a unique knowledge layer for desired output KG.
- The data producer needs to work to produce new (or to improve existing) data and knowledge resources.
  - Therefore, the producer is the user more interested in data materialization techniques, supporting her along the whole KG production process.

KGE - Knowledge Graph Engineering

Fausto Giunchiglia

Knowledge Graphs