

Semantic Technologies: Enabler for Knowledge 4.0

Achim Rettinger, Stefan Zander, Maribel Acosta
and York Sure-Vetter

Abstract Semantic technologies are a key enabler for Knowledge 4.0. Specifically, knowledge graphs have caused significant practical implications for managing knowledge in the digital economy. While most semantic technologies originate from the vision of representing the existing Web in a machine-processable format, it's most notable success so far are large cross-domain knowledge graphs. They are created by collaborative human modelling and linking of structured and semi-structured data. So far, they exhibit only little but still very powerful semantics, which have shown benefits for numerous applications. This chapter introduces the latest innovations in modelling knowledge using knowledge graphs and explains how those knowledge graphs enable value creation by making unstructured content, like text documents accessible by machines and humans. Finally, we show how semantic technologies help to make hard- and software components in cyber physical systems interoperable.

1 Introduction

Semantic technologies intend to bridge the gap between human knowledge and computational knowledge. They try to capture knowledge in an explicit computational knowledge representation that is both, accessible by humans and processable by computers in a meaningful manner. This is a feature that other approaches lack, such as representations created by the nowadays popular deep learning approaches.

Knowledge graphs (KGs) are currently seen as one of the most advanced components to realize the vision of explicit and computational knowledge representations. The term “knowledge graph” was reintroduced by Google in 2012 (Singhal 2012) and is now being used for any graph-based knowledge repository. In

A. Rettinger · M. Acosta · Y. Sure-Vetter (✉)
Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
e-mail: york.sure-vetter@kit.edu

S. Zander
Darmstadt University of Applied Sciences, Darmstadt, Germany

semantic technologies, the recommended data model to publish graph-based data on the Web is defined in the Resource Description Framework (RDF). Therefore, in the remainder we use the term knowledge graph for any RDF graph. An RDF graph consists of a finite set of triples where each triple (s, p, o) is an ordered set of the following terms: a subject s , a predicate p that associates the subject and the object, and an object o . An RDF term is either a URI, a blank node, or a literal. A triple allows to express a statement about a real-world fact. With these basic building blocks, knowledge graphs allow the representation of objects, their abstract relations and classes (groups) of objects, as well as their instantiations as real-world objects, called entities, and their concrete relations and class memberships (Färber et al. 2016).

Based on those basic technological building blocks, in recent years, several noteworthy large, cross-domain, and openly available KGs have been created. These include DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. They have grown to an impressive number of triples, like Freebase which is the largest KG with over 3.1B triples.

In the following sections, we will first outline how human knowledge is being captured in KGs, then explain how they can help to access the content expressed in unstructured sources like text documents and finally demonstrate that semantic technologies can be used to facilitate the interoperability gap between different cyber physical systems' components.

2 Semantic Technologies for Knowledge Engineering

The application of semantic technologies in knowledge engineering allows for the creation and management of knowledge-based systems. Systems that exploit knowledge and data semantics enable advanced capabilities in all the tasks of knowledge engineering processes.

Nowadays, the largest amounts of knowledge collected with the help of semantic technologies are the result of combining data harnessed from a wide range of sources—including humans and the Web—and representing it as semantic interconnected entities in knowledge graphs (KGs). KGs allow a wide range of analytical tasks including query processing through declarative languages, question answering, visualizations, and further data analysis including statistical analysis, data mining, etc. (see Fig. 1). Semantic technologies support the construction of KGs through semantic enrichment of non-semantic sources, data integration of several sources, and data curation. As depicted in Fig. 1, knowledge captured from humans can be stored in Knowledge Bases (KBs) and used across all the processes of knowledge engineering. Semantic technologies exploit human knowledge from KBs and computation knowledge encoded in data semantics to construct KGs.

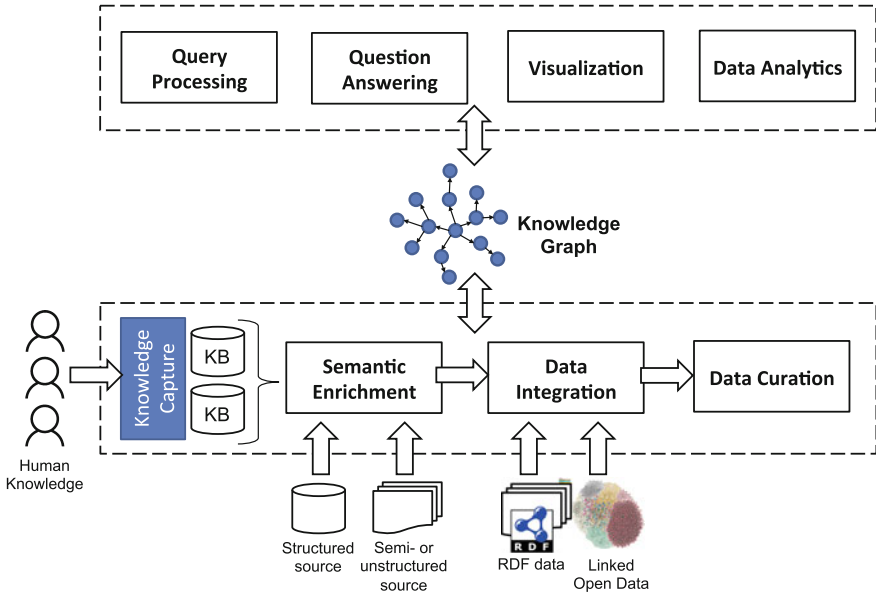


Fig. 1 Overview of the application of semantic technologies for knowledge engineering processes based on knowledge graphs

2.1 Knowledge Capture

Knowledge capture is the process of extracting and representing knowledge from reference sources directly or indirectly provided by humans. In this context, knowledge is modelled as artefacts including rules, ontological terms, and conceptual objects that symbolize physical objects or phenomena of a domain.

Typically, in Enterprise Systems, knowledge is harnessed through knowledge acquisition tasks that involve systematic interviews with domain experts: knowledge is collected through observation of experts including the usage of questionnaires. Nowadays, the rise of the Web and semantic technologies have allowed for capturing knowledge from experts as well as lay contributors to create semantic KBs in a distributed and collaborative way. Exemplary for successful knowledge capture frameworks that exploit Semantic Web technologies are Semantic MediaWiki (Krötzsch et al. 2006), DBpedia (Auer et al. 2007; Lehmann et al. 2015), and Wikidata (Vrandečić and Krötzsch 2014).

Semantic MediaWiki (SMW, <https://www.semantic-mediawiki.org>) is an extension to a wiki implementation that integrates semantic capabilities into the process of collaborative knowledge management through wikis. SMW enables human contributors to capture knowledge by semantically enriching and reusing content in the wiki. KBs created with SMW combine unstructured content and semantic (semi-) structures. KBs acquired with SMW are modelled using RDF,

represented with terms from the Web Ontology Language (OWL) (Bao 2012), and can be queried using declarative languages, for example, the SPARQL query language.¹

DBpedia (<http://dbpedia.org>) is a project that implements an extraction framework to gather data from different language versions of Wikipedia. The result of this framework is the DBpedia KB which is created by extracting data via declarative mappings or wrappers. In this context, wrappers are components that parse the data embedded in Wikipedia articles and translate it into concepts defined in the DBpedia ontology. The DBpedia wrappers are the result of a collaborative effort to the DBpedia project, where human contributors manually specified the wrappers to semantically enrich Wikipedia data. The resulting wrappers are stored in a knowledge base and currently available at the DBpedia Mappings Wiki (<http://mappings.dbpedia.org>). DBpedia data is published following the RDF data model and can be accessed and queried through web services.

Wikidata (<http://www.wikidata.org>) is a Wikimedia project that manages facts mainly from Wikipedia but also from its sister projects like Wikivoyage and Wikisource. Wikidata constitutes a collaborative multi-lingual KB that serves as a centralized source to provide unified and consistent facts across the multiple language versions of Wikipedia. Facts in the Wikidata KB are annotated with a list of reference sources that support the veracity of the facts. Wikidata contributors include humans and machines (bots) that create and maintain data in the KB. The content of the Wikidata KB is exported in different data models including RDF and can be also accessed through web services.

2.2 *Semantic Enrichment*

In semantic enrichment, the data of non-semantic sources is annotated with semantic descriptions from vocabularies or ontologies. The challenges and current solutions for performing semantic enrichment of sources highly depend on the characteristics of the underlying data.

Data accessed in knowledge-based systems may have different data models and structures. In the case of structured (e.g., relational databases) or semi-structured (e.g., XML files, CSV files, etc.) data sources, data can be semantically annotated using rules specified by experts in a knowledge capture process and stored in KBs. These rules translate elements from the sources into ontological concepts, properties, and instances. Particularly, in the case of relational databases, these rules or mappings can be specified using the W3C recommended languages Direct Mappings (DM) (Arenas et al. 2012) or R2RML (Souripriya et al. 2012). DM or R2RML mappings are executed by processors able to generate semantic graph-based data from relational databases following the RDF data model.

¹SPARQL 1.1 Query Language. Technical report, W3C (2013)

Unstructured sources that provide natural language documents or visual information can also be semantically annotated. Depending on the nature of the unstructured data, different semantic enrichment approaches are applied (see Section “Semantic Technologies for Understanding Unstructured Context”).

Depending on the data available at the sources, the process of semantically enriching data might be expensive in terms of time and computational resources. In particular, when dealing with large volumes of data, it is not practical to enrich and materialize the entire content of a source. To tackle this problem, current Big Data architectures (Auer et al. 2017) have focused on solutions that semantically enrich data on-demand, following the paradigm of schema or data on-read.

2.3 Data Integration

Data integration is the process of consolidating data from heterogeneous sources. Heterogeneity may occur at different levels: physical infrastructure (hardware and location), network protocols, data models, and data representation. As explained in the section “Semantic Enrichment”, the integration of data sources can also be performed offline, where all the content of the sources is integrated in a pre-processing step, or on-demand based on users’ queries as performed by traditional data integration systems (Lenzerini 2002).

The integration of semantic data allows for the construction of knowledge graphs from different sources. Data that has been created following the Linked Data principles assumes a common network protocol to access the data (HTTP or SPARQL) and a common graph-based data model (RDF). In cases where a common data representation cannot be assumed, Linked Data integration approaches may exploit rules and other knowledge artefacts (e.g., ontological definitions) specified by users and maintained in KBs and apply reasoning over this knowledge in order to consolidate semantically heterogeneous sources. Nonetheless, when entities, classes, and properties in a Linked Data source are linked or aligned to other sources, it is possible to assume that the sources are providing a common data representation. In line with this assumption, current federated SPARQL engines—e.g., ANAPSID (Acosta et al. 2011), FedX (Schwarte et al. 2011), SPLENDID (Görlitz and Staab 2011), and SemaGrow (Charalambidis et al. 2015)—are able to on-the-fly integrate RDF data from distributed and autonomous sources that are semantically homogeneous during query processing.

2.4 Data Curation

Data curation includes processes to create, maintain, and validate data to ensure the value of the data, in this case, of the KG that represents semantically annotated and

integrated data. To perform data curation, knowledge-based systems may exploit the semantics encoded in the data as well as knowledge captured in KBs.

In the context of knowledge engineering, one of the key tasks in data curation is the creation or completion of data. State-of-the-art solutions have investigated different reference sources or oracles to complete web data and knowledge graphs by, for example, automatically extracting data from web tables (Dong et al. 2014) and NLP graphs (Welty et al. 2012), respectively. Besides automatic approaches, a branch of state-of-the-art solutions resort to crowdsourcing, where humans act as oracles to complete databases or knowledge graphs (Franklin et al. 2011; Marcus et al. 2011; Park and Widom 2013; Acosta et al. 2015, 2017).

One instance of a system that applies crowdsourcing for knowledge graph completion is HARE (Acosta et al. 2015, 2017). HARE is a query engine able to enhance the completeness of knowledge graphs on-demand based on queries posed by users. HARE relies on the topology of the knowledge graph to identify potential missing values. To resolve missing values, HARE exploits the semantics of the data encoded in the knowledge graph to generate human-readable questions to be answered by a crowd composed of experts or lay users (contacted via crowdsourcing platforms). HARE stores the answers collected from the crowd in KBs. In this way, the knowledge captured from the crowd can be reused in subsequent queries. Empirical results evidence that HARE can reliably augment the completeness of knowledge graphs from different domains including Life Sciences. Furthermore, the results show that non-expert crowds can produce high quality answers achieving accuracy values from 0.84 to 0.96. Furthermore, the human-readable questions produced by HARE by exploiting the semantic description of entities in knowledge graphs are able to provide assistance to the crowd to produce high quality answers and to speed up the process of KG completion.

3 Semantic Technologies for Understanding Unstructured Content

The amount of entities in large knowledge graphs (KGs) has been increasing rapidly, enabling new ways of semantic information access, like keyword and semantic queries over entities and concepts mentioned in unstructured content, like text documents and videos. While entity search has become a standard feature (most prominent is the Google Knowledge panels shown next to the search results when searching for named entities), major Web search engines are still limited in their semantic processing capabilities: it is not possible to disambiguate search terms manually, search terms in one query can't be in different languages, the retrieved content items have to be in the same language as the search terms and search results are not gathered across heterogeneous content representations; like natural language and visual information. Most importantly, they don't allow to ask complex queries that spans information across multiple content sources. Recently developed systems

have shown that it is possible to overcome those issues by using semantic technologies.

3.1 Annotation

Semantic technologies enable computers to access the knowledge that is captured in unstructured documents like text or images. The key to semantic processing of unstructured content is annotating it with unique identifiers as provided by KGs. Since KGs have grown considerably over the last years they reached a size that makes high-quality annotations possible for general domains as covered by Wikipedia. The task of identifying mentions of entities in text documents and disambiguating them with their corresponding unique identifier in the KG has been termed entity linking.

One instance of an entity-linking system is X-LiSA (Zhang and Rettinger 2014) which is an infrastructure for cross-lingual semantic annotation. It allows to bridge the ambiguity of unstructured data and its formal semantics as well as to transform such data in different languages into a unified representation. The architecture of X-LiSA is shown in Fig. 2, where cross-lingual groundings extraction is performed offline to generate the indexes used by the online cross-lingual semantic annotation component.

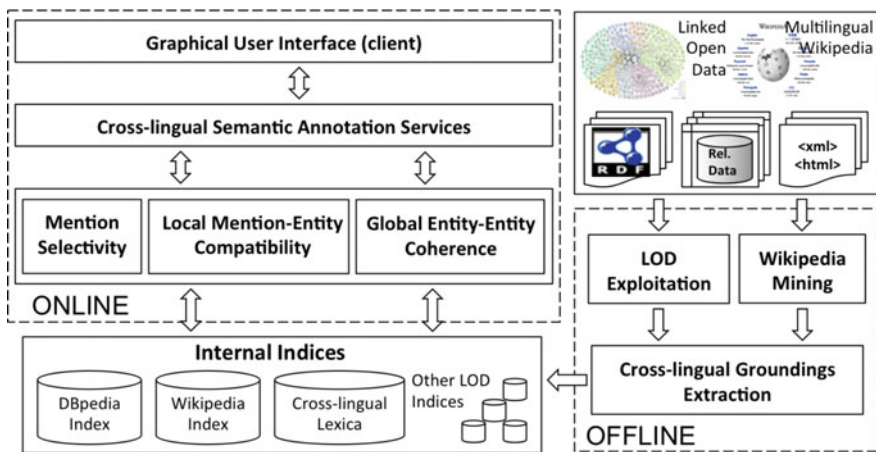


Fig. 2 Technical Components of the X-LiSA cross-lingual semantic annotation system. In an offline pre-processing step, large knowledge sources like Linked Open Data and Wikipedia are exploited and indexed in large repositories. This way, a cross-lingual lexicon of linguistic phrases that refer to entities in the knowledge graph is created. In the online step, text documents obtained for annotation are then split into potential phrases to be annotated, the phrases are scored and linked by the Mention Selectivity, Local Mention-Entity Compatibility and Global Entity-Entity Coherence modules. The output is a text document where all entities mentioned are annotated and linked to the corresponding KG entity

When such a system is used to annotate natural language text that comes from text documents in potentially different languages, text that was extracted from speech in videos and text from social media (again in different languages), a common data model is needed to make the content semantically accessible (see Fig. 3). To allow for semantic interoperability an RDF vocabulary is defined and tailored specifically to the different modalities: text, audio and video. It extends other vocabularies, such as the Dublin Core⁷, SIOC⁸ and KDO⁹. For each entity annotation, the predicates that define the start and end positions of the entity mention are used in a flexible manner and may define character positions, in the case of text, or milliseconds/frame numbers in case of audio/video. Each category annotation captures one topic of the media content. In any case, each entity mentioned in or each topic covered by any content item should relate to a resource in the knowledge base, namely an entity or a category in DBpedia.

Once the content is annotated in this way and if combined with semantic (keyword) query interpretation this allows for a semantic access to cross-modal cross-lingual content.

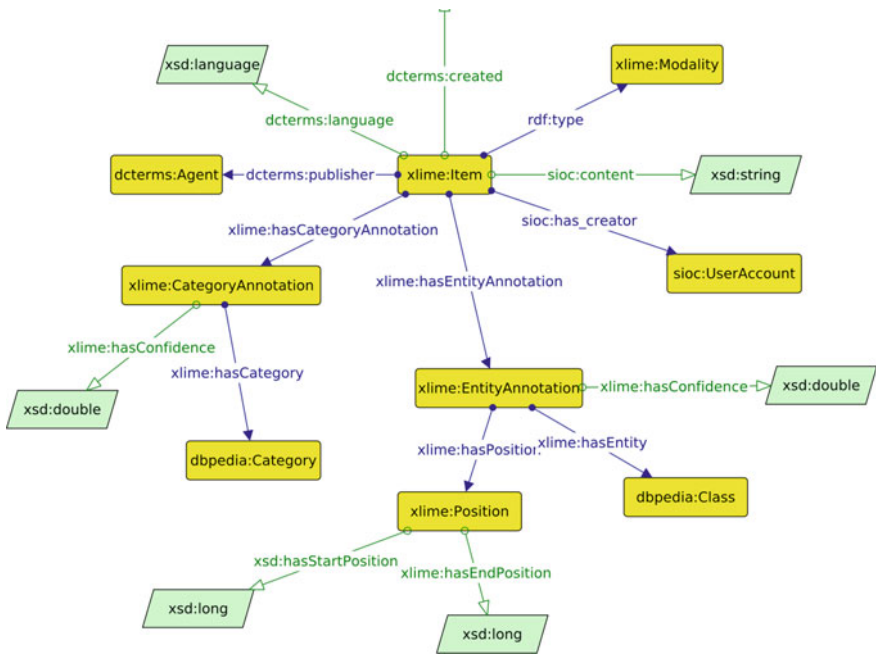


Fig. 3 To capture the meta information about annotations of different media items, a common data model is needed. The figure shows a model used in the xLiME project to annotated text, audio and video content, denoted as “item”. Meta information includes: entity mentions and the position of the mention (phrase in the text, bounding box in the image, time in the video), category of the item is being annotated, as well as the creator, publisher, the modality (text, audio or video) and the language (in case of text)

- **Document retrieval:** The search process starts with a keyword query in any language, which can even contain keywords in multiple languages. Instead of retrieving media items directly by keywords, a semantic search approach first finds the query entity vector (QEV), which represents a subgraph of the semantic graph of the knowledge base with nodes representing entities and edges describing their relations such that for each query keyword there is at least one entity in the subgraphs matching it. For each content item, we construct the data entity vector (DEV), where the entries contain the confidence scores of the annotations (i.e., the linked entities), which are generated by the cross-lingual semantic annotation. The semantic similarity between the QEV and each DEV can be calculated based on standard similarity measures, such as cosine similarity, which is then used for ranking of retrieved media items.
- **Document classification, clustering, recommendation:** The goal of a cross-lingual and cross-media semantic recommendation system is to find the similar content items posted across languages, modalities and channels. Here, we focus on a knowledge-centric approach to semantic recommendation using explicit semantics. This allows the semantic annotations to be further used for finding similar items. Once the entities inside documents are extracted, they can be used to calculate document similarity. Firstly, a subgraph is constructed from the entities identified for each document. As shown (Paul et al. 2016), the subgraphs of both documents are used to find the bipartite graph and graph-based similarity is then applied by computing the pairwise entity similarities based on the hierarchical and traversal scores.
- **Content analytics and complex queries:** Advanced data analytics across unstructured content has become a major necessity, which currently cannot be supported by non-semantic search engines. Using the knowledge extracted by semantic technologies from different media and languages in combination with additional background knowledge in DBpedia, a semantic approach allows to ask complex questions, such as “Which politicians from the Conservative Party of UK were most present in social media in the last two weeks before the Brexit referendum in different languages?”. Such questions are formulated as a SPARQL query and are answered by aggregating, counting and averaging knowledge across several media items. The semantic integration provides the ability to study differences and commonalities among media channels and languages.

The just outlined technologies have shown benefits in several business cases from media monitoring to product recommendation and have been used by social, media and political scientists to analyse content. See e.g. (Zhang et al. 2017) for further details.

4 Semantic Technologies for Cyber-Physical Systems

In the last part of this chapter, we outline some of the benefits semantic technologies in general, and ontology-based description frameworks in particular can provide to cyber-physical systems (CPS) and smart factories. We discuss that along the two aspects: *utilization* and *interoperability* of hard- and software components that constitute such systems. An integral aspect is the usage of ontology languages (Motik et al. 2009) and description logics (Baader 2003; Krötzsch et al. 2006; Rudolph 2011) as knowledge representation frameworks that allow for the axiomatic description of component metadata models (see Zander and Hua 2016). Detailed information together with extensive formal specifications of description logic based formalisms can be found in the previous references. Those axiomatic descriptions can be processed by reasoners, i.e., software programs that are able to understand the formal model-theoretic semantics injected by an ontology-based description framework into the component metadata models. This process is called *reasoning* and allows machines to infer new, implicitly contained knowledge or to check the consistency of a model or knowledge base, particularly when new data are added to it. More details are given in the paragraph about machine processability.

We discuss utilization and interoperability of CPS components simultaneously, as interoperability requires the utilization of a component's metadata model. Throughout this chapter, we understand the term 'metadata model' as a technical description of a component's characteristic features using a specific representation framework and format. In many technical specifications, a component's metadata model is also called *information model*.

The unobtrusive collaboration of a multitude of different and hard- and software components is a central aspect in smart factories that employ intelligent, self-regulatory production lines. Together with the Internet of Things (IoT), cognitive and cloud computing, they are the main pillars of the recently emerging Industry 4.0 paradigm. The multitude of different hardware and software components like services, tools, software agents etc., raises new challenges in addressing the structural, schema and semantic heterogeneity introduced by new technologies, protocols, description frameworks, interfaces, data structures and formats. The importance of addressing these challenges is amplified by the increasing complexity of tools, systems, and other software components embedded in business processes and demand the continuous integration of technical data and expert knowledge throughout the entire value-creation network. The interplay of autonomously operating hardware- and software components is one piece of the puzzle towards the broad realization of smart factories.

In order to reconcile different efforts and emerging formats, leading industrial players such as ABB, Daimler, Huawei, Kuka, Siemens etc. (to name just a few) recently started standardization initiatives for Industry 4.0 specific description frameworks. One such initiative is AutomationML, a well-adopted and fast growing XML-based description standard that already caught the attention of Industry 4.0

communities.² It covers engineering aspects including topology, geometry, kinematics, logic and communication (Drath et al. 2008) that can be used for describing properties and functionalities of a CPS component. Data contained in the AML description of one component can be exposed to the communication network of a CPS system and consumed by other components (Schleipen et al. 2014).

Unfortunately, the expressive power and flexibility of most CPS-related description frameworks are limited (Zander and Hua 2016). From AutomationML alone, for instance, it is not able to determine whether two components are compatible and able to work together based on the interface descriptions they exhibit. As a consequence, many of such description frameworks are not able to provide that form of understanding, in particular not in a machine-processable fashion.

This is the starting point where semantic technologies and ontology-based description frameworks in particular will help complementing such standardized industrial-driven description frameworks in meaningful ways. Thus making a contribution towards the realization of a technical interoperability infrastructure where hard- and software components are able to autonomously collaborate and exchange information together with their intended semantics. Semantic technologies help in doing that through the following aspects:

- (a) **Logical Model:** The description framework of Web ontologies is RDF, the logical model of which is built upon a graph-based data structure rather than a tree-based structure, which is the case for XML and XML-based formats such as AutomationML. Graph-based data structures not only provide a greater flexibility in representing information, they also circumvent the one-to-many mapping problem between graph- and tree-based representation formats and thus mitigating the problem of *structural heterogeneity* and *modelling ambiguity*.
- (b) **Expressivity:** The graph-based representation model of Web ontologies allows to treat relationships (so-called *properties* in terms of the Web Ontology Language (OWL) or *roles* in description logic terms) as *first-class citizens* and explicitly specify their characteristics and semantics. This is one of the main distinguishing features of Web ontologies compared to the object-oriented paradigm in which the creation of a relation is bound to the existence of a class or object. In Web ontologies, relations can be used independently of classes or resources (member of classes) and exist in a self-contained manner. Moreover, their semantics can be defined upon the formal, model-theoretic semantics of the ontology language used to describe and represent an ontology.
- (c) **Machine processability:** The logical theory upon which an ontology language and subsequently the ontology it describes is defined determines the expressive power and accuracy through which elements of a domain of interest (the so-called *universe of discourse* in ontology terms) can be described. Since the formal semantics of an ontology and its language elements (terms and concepts) are defined upon a logical theory, they can be processed automatically by

²Reference architecture model Industrie 4.0. Technical report, ZVEI. (2015).

machines in the form of reasoning engines. Reasoning describes the process of deducing logical entailments from the axioms constituting an ontology, the so-called *knowledge base*. Reasoning allows the determination of the consistency of a knowledge base, i.e., it checks whether an axiom (statement in ontology terms) is satisfiable for the given knowledge base or whether the introduction of a new fact to a knowledge base violates its consistency. Reasoning also allows for the deduction of new facts based on implicitly contained knowledge (as we will demonstrate in the following example).

These are some of the main features that distinguish ontologies and ontology languages from well-established representation frameworks introduced to computer science such as the Unified Modelling Language (UML) (Krötzsch et al. 2014).

In the following paragraph, we demonstrate how ontologies can serve as a semantic shell for enhancing the information models of cyber physical systems' components and allow for the deduction of new knowledge that fosters interoperability and data exchange between collaborating components.

In a first step, an AutomationML description needs to be analysed and transformed, i.e., uplifted into a compliant semantic graph represented as RDF description using transformation rules and domain heuristics (Björkelund et al. 2011; Kovalenko et al. 2015; Zander and Hua 2016). Such an uplifted semantic graph can then be processed by a reasoner in order to automatically classify a component with respect to specific classification systems and complemented its information model with domain knowledge derived from domain ontology axioms (some of them have been developed by several initiatives and projects such as KNOWROB, NIST Robot Ontology, OMRKF, ORA WG, and ReApp) (Schlenoff et al. 2012; Tenorth and Beetz 2013). Figure 4 illustrates an RDF representation of an excerpt of an uplifted AML description describing some technical aspects of a Sick S30B Laser Scanner. For reasons of readability and comprehensibility, ontology namespaces have been omitted throughout this chapter.

The concrete instance of the laser scanner is identified and represented via its UUID (line 1). Several technical parameters are then added in the form of RDF triples such as manufacturer and model information (line 2), starting and end angles (line 3), the maximal measurement range of the laser scanner in meter (line 4) and so on.

```

1 <urn:uuid:f81d4fae-7dec-11d0-765-00a0c91e6bf6>
2 :hasManufacturer "Sick" ; :hasModelName "S30B-2011GA" ;
3 :startAngle "135"^^xsd:integer ; :endAngle "135"^^xsd:integer ;
4 :maxMeasurementRangeInMeter "40"^^xsd:integer ;
5 :maxProtectiveFieldRangeInMM "2000"^^xsd:integer ;
6 :maxWarningFieldRange "8000"^^xsd:integer ;
7 :maxSimultaneousFieldEvaluations "0"^^xsd:integer .

```

Fig. 4 Excerpt of an uplifted AutomationML description of a Sick S30B Laser Scanner. The code represents an excerpt of the corresponding RDF graph and is serialized in the RDF N3 notation (also known as Notation3)

- $$\begin{aligned}
 \text{SafetyLaserScanner} &\sqsubseteq \exists \text{hasCapability.SafeMonitoringOf2DFields} & (1) \\
 \text{SafetyLaserScanner} &\sqsubseteq \text{LaserScanner} & (2) \\
 \text{LaserScanner} &\sqsubseteq \exists \text{hasCapability.MonitoringOf2DFields} & (3) \\
 \text{MonitoringOf2DFields} &\sqsupseteq \text{SafeMonitoringOf2DFields} & (4)
 \end{aligned}$$

Fig. 5 Excerpt of a set of description logic axioms that represent a classification hierarchy for components (Axiom 2 and 3) and capabilities (Axiom 4) and link concepts of a domain ontology to complex concept expression (right part of Axiom 1) in order to represent capability information that a reasoner can infer

Those technical assertions are then processed against the domain knowledge encoded in several hardware, software and capability ontologies that contain axioms as illustrated in Fig. 4. These axioms allow for the classification of a component, i.e., based on the uplifted information, the component represented by its UUID can be classified as `SafetyLaserScanner`. This process is automatically conducted by a reasoner in order to infer additional information and use it to complement the component’s information model.

Axiom 1 states that components classified as safety laser scanners have the default capability of safe monitoring of 2D fields (see Fig. 5). The concept representing ‘`SafeMonitoringOf2DFields`’ is defined in an external capability ontology and linked to the ‘`SafetyLaserScanner`’ concept through the property ‘`hasCapability`’. Such information is encoded as TBox knowledge, i.e., as schema knowledge. Such schema knowledge together with classification information, i.e., asserting that the Sick S30B is a safety laser scanner (usually this can be inferred through the technical information provided by the AutomationML description), allows a reasoner to infer that a Sick S30B laser scanner has the default capability of safe monitoring of 2D fields. When such information is inferred from the constituting knowledge base axioms and additional domain knowledge such as a capability ontology, it can be materialized and added to the uplifted AutomationML information model in the form of RDF assertions, i.e., ABox axioms. Hence, the initial component’s information model is complemented with additional knowledge derived from well-established domain ontologies. The materialization is important as it allows all the inferred knowledge about a component to be indexed by semantic data bases, so-called *triple stores* and retrieved through RDF query languages such as SPARQL.³ Figure 6 illustrates how the uplifted AutomationML description of the Sick S30B component can be complemented with the inferred information.

This complemented description now contains classification information, i.e., it asserts that the concrete instance of a Sick S30B (represented via its UUID; see line 1) is a safety laser scanner (line 2) and so on. The metadata model also contains information about a component’s capabilities (‘`{SafeMonitoringOf2DFields}`’, ‘`{MonitoringOf2DFields}`’ and ‘`{Monitoring}`’ (line 3), the purposes for which it

³SPARQL 1.1 Query Language. Technical report, W3C (2013).

```

1 <urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>
2   rdf:type :2DSaftyLaserSensor , :LaserSensor , :HardwareComponent ;
3   :hasCapability :{SafeMonitoringOf2DFields} , :{MonitoringOf2DFields} , :{Monitoring} ;
4   :hasPurpose :{HazardousAreaProtection} , :{AccessProtection} , :{PersonnelSafety} ;
5   :hasOperationEnvironment :iIndoor, iOutdoor .

```

Fig. 6 An uplifted RDF description of a Sick S30B Laser Scanner component (concrete instance) with additional information derived by inferring additional and implicit information contained in domain ontologies

can be used by default together with information about its operation environments (the derivation of this entailments is not depicted in the example). With that complemented information models, software agents are then able to infer whether two components are compatible and able to collaborate in more sophisticated ways. More information together with detailed examples can be found in (Zander and Awad 2015), (Zander and Hua 2016) as well as in (Zander et al. 2016).

5 Summary

In this chapter we introduced the latest innovations in modelling knowledge using knowledge graphs and explained how those knowledge graphs enable value creation by making unstructured content, like text documents accessible by machines and humans. We covered different steps of the knowledge creation lifecycle including (manual) knowledge engineering and (automatic) understanding of unstructured content. Last, but not least, we have shown how semantic technologies help to make hard- and software components in cyber-physical systems interoperable.

References

- Acosta, M., Simperl, E., Flöck, F., & Vidal, M.-E. (2015). HARE: A hybrid SPARQL engine to enhance query answers via crowdsourcing. In *Proceedings of the 8th International Conference on Knowledge Capture*. New York: ACM.
- Acosta, M., Simperl, E., Flöck, F., & Vidal, M.-E. (2017). Enhancing answer completeness of SPARQL queries via crowdsourcing. *Web Semantics: Science, Services and Agents on the World Wide Web*.
- Acosta, M., Vidal, M.-E., Lampo, T., Castillo, J., & Ruckhaus, E. (2011). ANAPSID: An adaptive query processing engine for SPARQL endpoints. *The Semantic Web–ISWC 2011*, 18–34.
- Arenas, M., Bertails, A., Prud, E., & Sequeda, J. (2012). A direct mapping of relational data to RDF. W3C Recommendation. See <https://www.w3.org/TR/rdb-direct-mapping/>
- Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., & Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. *The Semantic Web*, 722–735.

- Auer, S., Scerri, S., Versteden, A., Pauwels, E., Charalambidis, A., & Konstantopoulos, S., et al. (2017). The BigDataEurope platform—Supporting the variety dimension of big data. In *International Conference on Web Engineering*. Berlin: Springer.
- Baader, F. (2003). *The description logic handbook: Theory, implementation and applications*. Cambridge: Cambridge University Press.
- Bao, J. (2012, December). OWL 2 Web Ontology Language document overview. W3C Recommendation. *World Wide Web Consortium*, 201(2).
- Björkelund, A., Malec, J., Nilsson, K., & Nugues, P. (2011). Knowledge and skill representations for robotized production. *IFAC Proceedings Volumes*, 44(1), 8999–9004.
- Charalambidis, A., Troumpoukis, A., & Konstantopoulos, S. (2015). SemaGrow: Optimizing federated SPARQL queries. In *Proceedings of the 11th International Conference on Semantic Systems*. New York: ACM.
- Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., & Murphy, K., et al. (2014). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York: ACM.
- Drath, R., Luder, A., Peschke, J., & Hundt, L. (2008). AutomationML—the glue for seamless automation engineering. In *IEEE International Conference on Emerging Technologies and Factory Automation*, 2008. *ETFA 2008*. New York: IEEE.
- Färber, M., Bartscherer, F., Menne, C., & Rettinger, A. (2016). Linked data quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. *Semantic Web(Preprint)*, 1–53.
- Franklin, M. J., Kossmann, D., Kraska, T., Ramesh, S., & Xin, R. (2011). CrowdDB: Answering queries with crowdsourcing. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*. New York: ACM.
- Görlitz, O., & Staab, S. (2011). Splendid: Sparql endpoint federation exploiting void descriptions. In *Proceedings of the Second International Conference on Consuming Linked Data-Volume 782*, CEUR-WS.org.
- Kovalenko, O., Wimmer, M., Sabou, M., Lüder, A., Ekaputra, F. J., & Biffi, S. (2015). Modeling automationml: Semantic web technologies vs. model-driven engineering. In *2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA)*. New York: IEEE.
- Krötzsch, M., Simancik, F., & Horrocks, I. (2014). Description logics. *IEEE Intelligent Systems*, 29, 12–19.
- Krötzsch, M., Vrandečić, D., & Völkel, M. (2006). Semantic mediawiki. In *International Semantic Web Conference*. Berlin: Springer.
- Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., & Mendes, P. N. (2015). DBpedia—A large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, 6(2), 167–195.
- Lenzerini, M. (2002). Data integration: A theoretical perspective. In *Proceedings of the Twenty-first ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. New York: ACM.
- Marcus, A., Wu, E., Karger, D. R., Madden, S., & Miller, R. C. (2011). Crowdsourced databases: Query processing with people, Cidr.
- Motik, B., Grau, B. C., Horrocks, I., Wu, Z., Fokoue, A., & Lutz, C. (2009). OWL 2 web ontology language profiles. *W3C Recommendation*, 27, 61.
- Park, H., & Widom, J. (2013). Query optimization over crowdsourced data. *Proceedings of the VLDB Endowment*, 6(10), 781–792.
- Paul, C., Rettinger, A., Mogadala, A., Knoblock, C. A., & Szekely, P. (2016). Efficient graph-based document similarity. In *International Semantic Web Conference*. Berlin: Springer.
- Rudolph, S. (2011). Foundations of description logics. In *Reasoning Web. Semantic Technologies for the Web of Data* (pp. 76–136). Berlin: Springer.
- Schleipen, M., Pfrommer, J., Aleksandrov, K., Stogl, D., Escaida, S., Beyerer, J., & Hein, B. (2014). Automationml to describe skills of production plants based on the ppr concept. In *3rd AutomationML User Conference*.

- Schlenoff, C., Prestes, E., Madhavan, R., Goncalves, P., Li, H., & Balakirsky, S., et al. (2012). An IEEE standard ontology for robotics and automation. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. New York: IEEE.
- Schwarte, A., Haase, P., Hose, K., Schenkel, R., & Schmidt, M. (2011). Fedx: Optimization techniques for federated query processing on linked data. In *International Semantic Web Conference*. Berlin: Springer.
- Singhal, A. (2012). Introducing the knowledge graph: Things, not strings. <https://googleblog.blogspot.co.at/2012/05/introducing-knowledge-graph-things-not.html> 2016.
- Souripriya, D., Seema, S., & Richard, C. (2012). R2RML: RDB to RDF Mapping Language, W3C Recommendation.
- Tenorth, M., & Beetz, M. (2013). KnowRob: A knowledge processing infrastructure for cognition-enabled robots. *The International Journal of Robotics Research*, 32(5), 566–590.
- Vrandečić, D., & Krötzsch, M. (2014). Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57(10), 78–85.
- Welty, C., Barker, K., Aroyo, L., & Arora, S. (2012). Query driven hypothesis generation for answering queries over nlp graphs. In *International Semantic Web Conference*. Berlin: Springer.
- Zander, S., & Awad, R. (2015). Expressing and reasoning on features of robot-centric workplaces using ontological semantics. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. New York: IEEE.
- Zander, S., & Hua, Y. (2016, February). Utilizing ontological classification systems and reasoning for cyber-physical systems. In *Karlsruhe Service Summit Research Workshop*.
- Zander, S., Merkle, N., & Frank, M. (2016). Enhancing the utilization of IoT devices using ontological semantics and reasoning. *Procedia Computer Science*, 98, 87–90.
- Zhang, L., & Rettinger, A. (2014). X-LiSA: cross-lingual semantic annotation. *Proceedings of the VLDB Endowment*, 7(13), 1693–1696.
- Zhang, L., Thalhammer, A., Rettinger, A., Färber, M., Mogadala, A., & Denaux, R. (2017). The xLiMe system: Cross-lingual and cross-modal semantic annotation, search and recommendation over live-TV, news and social media streams. *Web Semantics: Science, Services and Agents on the World Wide Web*.

Author Biographies

Achim Rettinger, PD Dr. is an Research Group Leader at AIFB. Before joining KIT, he did his Ph.D. studies in Computer Science at the Technical University of Munich and at the Siemens AG in Munich, Germany. His research interests and publications are in combining machine learning with semantic technologies. He published extensively on cross-lingual and cross-modal representation learning, text and structured embeddings, cross-lingual semantic annotation, cross-lingual and multimodal retrieval, knowledge representation and statistical relational learning.

Stefan Zander, Prof. Dr. is a Full Professor for Web technologies and User-centered Design at Darmstadt University of Applied Sciences. He is also Guest Researcher at the FZI Research Center for Information Technology at KIT, where he was a research group leader before. His research interests cover topics related to knowledge representation, semantic technologies and its application in different domains such as the Internet of Things or cyber-physical systems in general.

Maribel Acosta, Dr.-Ing is a Postdoctoral Researcher at AIFB. She conducted her Ph.D. studies at the Karlsruhe Institute of Technology (KIT), Germany. Her research interests include data management over semantic data, with a special focus on data integration, data curation, and query processing. Her publications include works on enhancing Semantic Web technologies with human-based computation via crowdsourcing, adaptive techniques for Linked and Big Data management, and querying federated data sources on the web.

York Sure-Vetter, Prof. Dr. is Full Professor for Web Science at the Karlsruhe Institute of Technology (KIT), Germany. He is the spokesperson of the Institute AIFB, Director at the FZI Research Center for Information Technology at KIT and Director at the interdisciplinary Karlsruhe Service Research Institute (KSRI). His research interests include web science, knowledge representation, machine learning, Industry4.0 and the Internet of Things.