

REPRESENTATION AND APPLICATION OF DIGITAL STANDARDS USING KNOWLEDGE GRAPHS

Luttmer, Janosch;
Ehring, Dominik;
Pluhnau, Robin;
Nagarajah, Arun

University Duisburg-Essen

ABSTRACT

Standards are an important source of knowledge in product development. Due to the increasing digitization of the product development process, standard development organizations aim to develop machine-actionable standards that automatically enforce operations in output devices. However, the current representation format in PDF or XML does not meet the requirements of machine-actionable standards. This paper examines existing approaches towards the representation of XML data in knowledge graphs and their transferability towards the domain of digital standards. Based on these approaches, the paper aims to develop and validate a concept for transferring standard content from XML format to a graph-based representation, using the example of formulas. For this purpose, a concept for the automatic identification, extraction and modeling of formulas will be presented. Afterwards, the concept is validated using the example of DIN ISO 281 whereas a chatbot application serves as conversational user interface. It is proven, that knowledge graphs are suitable for the representation of machine-actionable standard content. Future work will investigate the abstraction towards a general approach as well as further information objects of standards.

Keywords: Standards, Conceptual design, Knowledge graphs, Knowledge management, Semantic data processing

Contact:

Luttmer, Janosch
University Duisburg-Essen
Institute of Product Engineering
Germany
janosch.luttmer@uni-due.de

Cite this article: Luttmer, J., Ehring, D., Pluhnau, R., Nagarajah, A. (2021) 'Representation and Application of Digital Standards Using Knowledge Graphs', in *Proceedings of the International Conference on Engineering Design (ICED21)*, Gothenburg, Sweden, 16-20 August 2021. DOI:10.1017/pds.2021.516

1 INTRODUCTION

The product development process is subject of change. Not only the increasing complexity of products, but also the growing individuality of customer requirements pose great challenges to companies (Rauch et al., 2016). In order to overcome those challenges the digitization of development processes as well as the interoperability of application systems is increasingly important. Since product development is also characterized as knowledge-intensive activity, the integration of digital knowledge management is of particular interest. (Zhenyong et al., 2020)

Standards are an essential source of knowledge and enable companies to, among others, meet regulatory requirements and decrease development costs (Ziegenfuss, 2020). Furthermore, they support the decision making process of many engineering activities along the product development process (Manoharan and Koehler, 2018). One important stakeholder of standards is the designer who ensures conformity to standards in the early phases of development. Because standards are mostly offered in print or PDF, a non-negligible part of his work consists of searching, reading, and understanding standards. Afterwards, the designer extracts standard content and transfers it into company specific tools such as requirements engineering tools. Thus, the current usage of standards is inefficient and not compatible with digital knowledge management and its integration into product development processes. (Manoharan et al., 2019; Ziegenfuss, 2020; Loibl and Nagarajah, 2019)

1.1 Current trends in standardization

As digitization continues to increase in importance, standard development organizations have growing interest in distributing their content outside of typical print or PDF formats (Ziegenfuss, 2020). The German Institute for Standardization (DIN), for example, started to build a central XML database in 2015 and completed the transfer of more than 30,000 standards in 2018 (Wischoefer and Koch, 2016; Schuch and Wischoefer, 2018). Additionally, the National Information Standards Organization (NISO) created a standardized schema for XML-structured standards which is called NISO STS (Standard Tag Suite) and applies to the entire set of DIN standards (Schuch and Wischoefer, 2018; National Information Standards Organization, 2017).

However, while standards are already being offered in XML format, a study by Blind and Hess (2018) has shown that 83 % of 1035 companies surveyed still use standards in Word or PDF format. At the same time, a survey by the German Federal Ministry of Economics and Energy revealed that 47 % (n = 620) demand machine-readable and machine-actionable standards (Bundesministerium für Wirtschaft und Energie, 2017). Thus, there is deficit between content offered and content actually consumed.

Compared to legacy documents in print or PDF, XML has the advantage that content is represented in a structured manner which can be characterized as machine-readable (Ziegenfuss, 2020). Further development stages of digital standards include the semantics of information in order to automatically enforce operations in output devices. They are defined by Loibl et al. (2020) as machine-actionable and machine-interpretable whereas the latter represents the highest level of automation. Currently, standardization organizations are in transition between machine-readable and machine-actionable content. In order to guide this process, Loibl and Nagarajah (2019) propose the Duisburg 3M model which consists of three pillars, namely modularization, modeling, and management. Besides scientific institutes, standard development organization such as DIN or ISO as well as companies themselves have initiated to work on digitized, machine-actionable standards (Ziegenfuss, 2020; Esser and Mueller, 2019; DIN and DKE, 2020).

1.2 State of the art

While XML is powerful in representing the syntactic, structural properties of data and is, thus, particularly suitable for their exchange, the representation of semantic relations is limited which negatively effects the semantic interoperability of XML data (Tekli et al., 2016). However, since semantics is a prerequisite for machine-actionable standards, it can be assumed that the current representation of standards in XML format is not sufficient (Loibl et al., 2020).

In the context of standardization, the absence of semantic information in XML documents and potential other types of representation have not been investigated in detail. Loibl et al. (2020) describe a procedure to classify standards according to their degree of formalization and to model and store standard content in a graph database. Additionally, Manoharan and Koehler (2018) conclude that the

highly interconnected information of standards should not be stored in tabular form, but in graph databases such as Neo4j whereas the so called knowledge graph could then be provided to the user via micro services.

A more comprehensive investigation of the present problem is done in the context of automatic text processing and analysis. The transformation of XML data into a graph-based representation is focus of several research projects. [Hacherouf et al. \(2019\)](#) propose a transformation of XML data into the Web Ontology Language (OWL), which is based on the Resource Description Framework (RDF) and extends the XML format by additional methods for expressing the semantics of data. In order to automatically transfer XML data into an OWL ontology, graph-based rules can be defined. Thus, a linkage between the semantic (OWL) and syntactic layer (XML) is established. ([Cruz and Nicolle, 2010](#))

[Sana and Suganthi \(2017\)](#) propose a general procedure for transforming XML data into a graph-based representation which consists of four steps. Firstly, XML content is converted into a triple structure as subject, predicate, and object (RDF). Afterwards, these triples are modeled in graph patterns and stored in the graph database Neo4j. The user is then able to access the database through the Neo4j specific query language Cypher. The main benefits are the representation of semantic relations between data elements as well as the high flexibility and performance of graph databases. ([Sana and Suganthi, 2017](#))

The development of graph patterns from XML documents and their aggregation to knowledge graphs are also considered in the context of mechanical engineering and product development. One example is the work of [Kwon et al. \(2020\)](#), who model product information in STEP format and inspection data in XML format and merge both models to a single knowledge graph. In this context, XML data is automatically converted into an OWL/RDF schema and, thus, a semantically enriched layer is created which enables reasoning mechanisms and logic-based methods ([Kwon et al., 2020](#)).

Another research approach of [Pinquié et al. \(2019\)](#) refers to the digitization of design rules that are typically present in unstructured documents such as Word or PDF as well as structured documents like XML. In order to develop a cognitive assistant for design engineers, the authors propose following steps:

1. Extraction of design rules from input documents (e.g. Word, PDF, XML)
2. Design of context-specific graph patterns
3. Modeling of design rules in graph patterns
4. Consolidation of graph patterns in order to provide a single graph data model
5. Storage in graph database
6. Provision of interfaces to different channels

A possible field of application is the integration of the cognitive design assistant into computer-aided engineering tools. Moreover, the authors propose to provide informal design rules in textual form through a chatbot application which enables the assistant to answer personalized questions of the designer. ([Pinquié et al., 2019](#))

1.3 Goal of research paper

In order to automatically integrate digital standards into development processes and tools, standard content needs to be offered in machine-actionable format ([Loibl et al., 2020](#)). As this requires the representation of semantic relationships, which cannot be achieved by the currently used XML format, content needs to be transferred to an appropriate representation type.

As shown in section 1.2 knowledge graphs are used to provide semantic enriched content. This research paper aims to transfer the approach towards the domain of digital standards. Hereby, the paper focuses on formulas that are an essential information object of standards and are characterized by a high degree of clarity. Due to the correlation of clarity and suitability for machine-actionability, formulas serve as an appropriate starting point for the digitization of standards ([Loibl et al., 2020](#)). Hence, the goal is to develop and validate a concept for the automatic transformation of formulas from XML standards into a knowledge graph as well as their usage through an appropriate user interface.

For this purpose, section 2 contains the development of a concept which is derived from existing approaches. Afterwards, the concept is further elaborated in section 3 and validated in section 4. The paper concludes with a brief discussion in section 5 as well as a summary and an outlook in section 6.

2 CONCEPTUAL DESIGN

Based on the previously identified literature and especially the approaches by [Sana and Suganthi \(2017\)](#) as well as [Pinquié et al. \(2019\)](#), the transformation of XML content into knowledge graphs can be summarized by the following steps:

1. Modeling of XML content in predefined graph patterns (sub models)
2. Aggregation of sub models to provide an entire graph data model
3. Storage in graph database
4. Content usage via different channels

These rather general steps can serve as basis of the concept which will be developed in this section. It is conceivable that individual process steps can be reused or tailored to satisfy the use case of formulas. To guide the concept development, several requirements are specified which are analyzed with respect to the existing solution approaches (see Figure 1).

Requirement	Characteristics of Sanal and Suganthi	Characteristics of Pinquié et al.
The concept shall consider XML documents as input.	Input: XML	Inputs: Word, PDF, XML
The concept shall ensure the automatic identification of formulas in XML documents.	-	-
The concept shall provide a graph-based modeling pattern for formulas.	Modeling of RDF triples	Modeling of context-specific graph patterns (submodels)
The modeling pattern shall consider semantic relations between formulas.	RDF triples ensure semantic relations	Submodels are merged in order to represent semantic relations
The concept shall include a single point of storage for all relevant information.	Graph data model stored in Neo4j graph database	Graph data model stored in graph database
The concept shall ensure that the graph data model can be accessed by different user interfaces.	Query language Cypher	Querying of graph data model, potential interfaces: chatbot, CAE environment
The concept shall aim for a high degree of automation.	-	-

Figure 1. Analysis of solution approaches ([Sana and Suganthi, 2017](#); [Pinquié et al., 2019](#))

It is evident that the existing approaches already cover several requirements. This includes the input document type as well as the principle to represent XML data with the help of graph patterns to create sub models. These sub models are then aggregated and stored as a single knowledge graph in a graph database. Ultimately, the graph database can be accessed via specific query languages which enables the integration into various application programs. In order to abstract the already existing solution approaches to the domain of formulas, concept adaptations are required. While the approaches are based on the conversion of entire XML documents, in this context only formulas and information associated with formulas need to be identified and extracted (Requirement no. 2, see Figure 1). Furthermore, there is no specific graph pattern for modeling formulas. While [Sana and Suganthi \(2017\)](#) point out that modeling in RDF triples is a general solution, [Pinquié et al. \(2019\)](#) introduce context-specific graph patterns. Ultimately, no statement is possible about the degree of automation of existing solution approaches (Requirement no. 7). However, it should be noted that in the concept presented below, each individual step aims at a high degree of automation.

Based on the requirements and the analysis of existing solutions, the overall concept is developed. The first part of the concept consists of four steps and deals with the identification and extraction as well as the modeling, aggregation, and storage of formulas in a graph data model (see Figure 2). For this purpose, especially the first two steps have to be developed in more detail, since no suitable solution already exists. The methods for aggregating and storing the knowledge graph in a graph database can be taken from the existing solutions. Lastly, to ensure access to the graph database, specific query

languages are used. This concept is reused from existing solutions and enables the integration of the knowledge graph into various application programs.

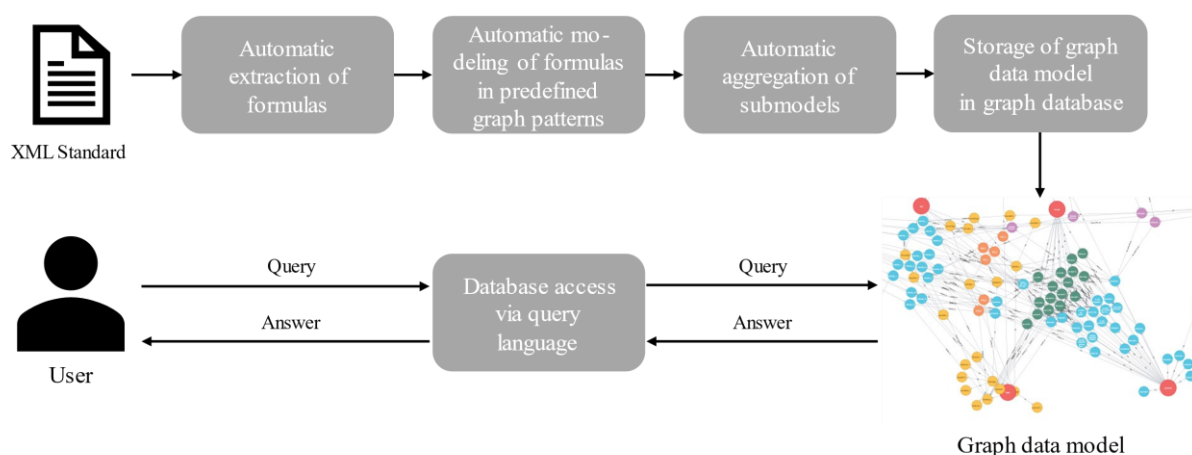


Figure 2. Concept for transforming formulas from XML towards a graph data model and their usage

3 ELABORATION OF CONCEPT

In order to realize the concept described above, the process steps *Automatic extraction of formulas* and *Automatic modeling of formulas in predefined graph patterns* are elaborated in more detail. In addition, the steps taken from existing solutions are briefly presented.

3.1 Automatic extraction of formulas

Standards in XML format are based on the NISO STS standard. Hereby, formulas are identified by the enclosing tags `$` and `$`. Additionally, the mathematical information that lies between the tags is based on a standard named MathML. MathML can be understood as a mathematical markup language whereas the mathematical description of the formula follows a predefined syntax. (National Information Standards Organization, 2017)

As an example, the representation of parameter L_{10} in MathML will be considered:

```
<math>
  <msub>
    <mi> L </mi>
    <mn> 10 </mn>
  </msub>
</math>
```

In this context, the element *msub* indicates that the subscript 10 is attached to the base L. Moreover, L is declared as an identifier through the tag *mi* and 10 is tagged as number using *mn*. Since formulas do not exclusively consist of a mathematical description, additional content such as parameter names or definitions needs to be extracted. Again, these information objects are tagged according to the NISO STS standard.

To ensure a high degree of automation, an XML-parser has been developed which identifies formulas via the characterizing tags and extracts the information in between. For this purpose, the MathML logic is used in order to transfer the syntax of formulas into a semantic enriched representation in graph patterns. Additionally, the parser is able to extract parameter names and definitions as well as further information such as formula ID or standard ID.

3.2 Graph pattern for modeling formulas

After identifying and extracting the relevant content, a fixed graph pattern is used to model every formula as separate sub model. In general, different modeling variants are conceivable that will be investigated in following research papers.

In this paper, the basic schema maps the mathematical description of the formula by *left* and *right* relations to either operators or parameters or values. Hereby, the graph pattern consists of five node

types and five relationship types which are shown in Table 1. In order to provide further semantics, the additional extracted information such as parameter names and definitions can be used as properties which are attached to nodes.

Table 1. Graph pattern - nodes and relationships

Node type	Explanation
Formula node	Contains the characteristic formula ID
Symbol node	Contains variables and parameters as well as their definitions
Operator node	Contains operators
Number node	Contains numeric values
Standard node	Contains the indication of the respective standard
Relationship type	Explanation
Left	Target node is left of predecessor node
Right	Target node is right of predecessor node
Is	Used between formula node and operator node with value =
Has	Used between standard node and formula node
if	Indicates a condition which restricts the usage of the formula

Every formula is assigned to one or more standards. Therefore, the starting point of the graph pattern is the corresponding designation of the standard. Firstly, this maps formulas to a superordinate document layer and, secondly, enables the graph database to be extended by further standards. The standard nodes are linked to the respective formula nodes via the *has* relationship, which contains information about the formula's designation. Furthermore, formula nodes are connected via an *is* relationship with the operator = (equal to). Starting from this, a network of operators, symbols and numerical values develops, which are connected by *left* and *right* relations (see Figure 3).

The main benefit of using this graph pattern is its flexibility and adaptability. Moreover, the order of calculation steps is explicitly stored in the graph pattern which simplifies the usage in application software. Additionally, due to *left* and *right* relationships, the calculation is performed step by step. Therefore, no additional expression parser or evaluator is needed.

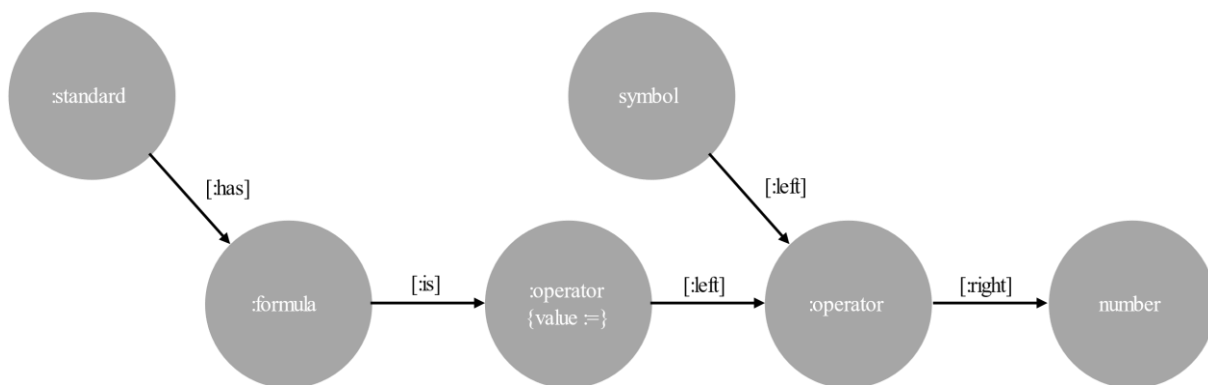


Figure 3. Graph pattern for modeling formulas

3.3 Aggregation of sub models and storage in graph database

According to the existing solutions, the sub models are aggregated to one entire graph data model which contains all necessary content (Sana and Suganthi, 2017; Pingué et al., 2019). For this purpose, redundant nodes are identified and connected so that the number of nodes is reduced and the performance of the graph is increased. The entire graph data model is then stored in a graph database such as Neo4j.

3.4 Database access via query language

Generally, the application of the graph data model can be realized using specific query languages, e.g. Cypher. Based on queries, relevant information is extracted from the graph database and provided to application programs. As this serves as a highly flexible interface, different fields of application are

conceivable, for example the integration into computer-aided engineering tools and direct human to machine communication via chatbots.

4 VALIDATION

In order to validate the previously described concept, an expert system which is also known as knowledge-based agent or cognitive assistant (Pinquié et al., 2019) was developed. It is based on a graph data model that is automatically created from an XML document and stored in the graph database Neo4j. The expert system's goal is to support engineers in searching formulas and performing calculations based on the standard DIN ISO 281 - Rolling bearings (Dynamic load ratings and rating life) (DIN, 2010). To provide a conversational user interface, a chatbot application will be used. Figure 4 shows the context diagram of the expert system.

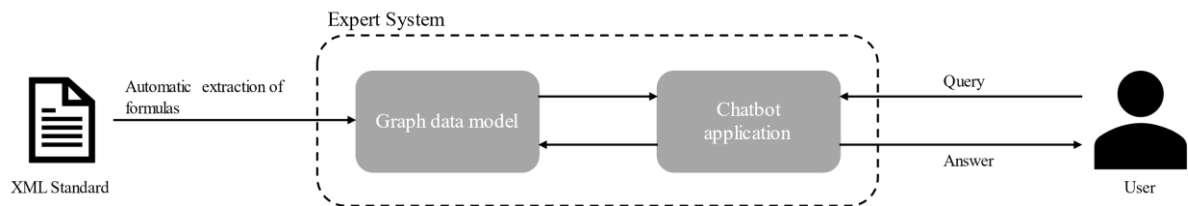


Figure 4. Context diagram of expert system

4.1 Transformation of formulas to graph data model

DIN ISO 281 consists of 59 formulas. Since the formulas are based on the standards NISO STS and MathML, the identification and extraction can be automated according to section 3.1. Besides, the previously stated XML-parser extracts additional information such as formula ID, parameter names and definitions.

Afterwards, every formula is modeled according to the defined graph pattern. Figure 5 shows the exemplary representation of formula 4 of DIN ISO 281 which specifies the nominal life of a radial rolling bearing (DIN, 2010):

$$L_{10} = \left(\frac{C_r}{P_r}\right)^3 \quad (1)$$

Whereas the formula's mathematical description is expressed by the network of symbol, operator and number nodes, additional information about the formula ID as well as the assigned standard is provided (see Figure 5). After modeling every formula in the graph pattern, the sub models are merged and stored in the Neo4j graph database.

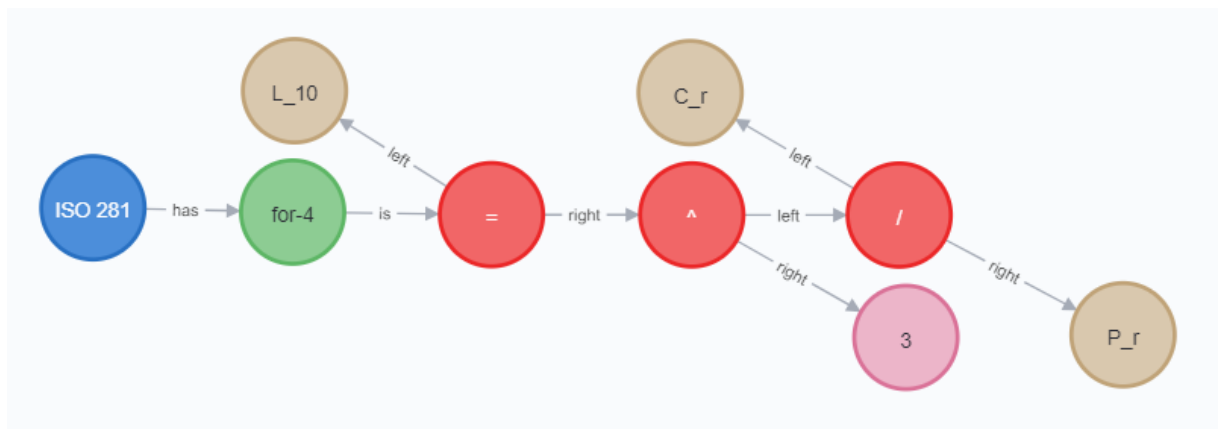


Figure 5. Example of formula graph pattern (sub model)

4.2 Chatbot application using graph data model

The chatbot application was implemented using the Microsoft Azure Bot Framework and in particular the visual authoring canvas *Bot Framework Composer* which manages conversation flows of the chatbot. In addition, it contains the language understanding service LUIS, which analyzes user

requests in natural language and provides appropriate answers. For this, LUIS creates a language model and uses machine learning techniques to improve it through exemplary inputs. (Microsoft Corporation, 2020a)

After development and testing of the chatbot, the application can be published via the Azure Portal on various channels, such as Microsoft Teams or Cortana (Microsoft Corporation, 2020b).

In order to develop the user interface, following use cases shall be considered:

- Search and show formulas
- Perform calculations
- Support the design of rolling bearings
- Search and show definitions and parameter names

The functions are realized by implementing conversation flows that guide the user appropriately. During a conversation flow, necessary user input is requested and stored in the working memory. Afterwards, a query is triggered to search the graph data model for required information. This query is written in Cypher language whereas an exemplary database request is shown in the following:

```
MATCH p = (r:formula)-[*]->(x) where r.value = "for-4"  
UNWIND nodes(p) AS Vertex RETURN DISTINCT Vertex;  
MATCH p = (r:formula)-[*]->(x) where r.value = "for-4"  
UNWIND relationships(p) AS Vertex RETURN DISTINCT Vertex;
```

These queries retrieve the nodes and relationships which are connected to formula 4. Afterwards, the nodes are interconnected with the help of relationships and the formula is rebuilt step by step. Finally, the formula is returned to the user. If a calculation is desired, the user specifies the corresponding values of the parameters. These are inserted into the formula, the result is calculated and returned to the user.

The main function of the chatbot is to support the user in designing rolling bearings. The conversation flow developed for this purpose is based on the procedure in the standard itself and is optimally tailored to the application of DIN ISO 281.

Additionally, the user has the possibility to query definitions and parameter names which is primarily for the sake of user-friendliness. Moreover, the integration of LUIS offers the advantage that the user can ask the chatbot questions in natural language. LUIS analyzes these questions and initiates appropriate actions. For example, in the query "What is the formula of nominal life of radial rolling bearing?" the language understanding software identifies the entities *formula*, *nominal life*, and *radial rolling bearing* and triggers the search of a suitable formula.

5 DISCUSSION

With respect to the requirements defined in section 2 and the validation of the derived concept, the transformation of formulas from XML documents towards a graph-based representation proves to be sufficient in order to provide semantic enriched content. Due to the knowledge graph, semantic relationships are represented and can be used to automatically perform actions in various output devices like a chatbot application (see section 4). Hence, the concept realizes the transfer of machine-readable XML content into machine-actionable content.

The graph data model consists of individual graph patterns. Although the modeling approach was chosen intuitively, it offers great flexibility and adaptability. However, it also leads to a strong increase in the number of nodes when extracting standards with many formulas. This in turn can lead to a reduction in query performance, especially when several standards are mapped to an entire graph. Furthermore, the XML-parser extracts information from different parts of the XML document such as the definition section or the formulas itself. It is conceivable that future XML standards contain building blocks whereas all information that is relevant for a specific formula is stored in one building block. This enhances the quality of extraction and the representation of semantic relations.

The concept presented here uses formulas as an example. Nevertheless, the transfer to other information objects of standards such as tables or text elements should be investigated. For this purpose, techniques need to be developed to identify and extract relevant content in the XML document. Furthermore, suitable modeling patterns have to be created, which can be integrated into

the existing structure of the knowledge graph. On higher level of abstraction, following steps are suggested in order to transfer XML standards into machine-actionable content:

1. Identify and extract relevant information (automatically/manually)
2. Model the information in defined graph patterns
3. Merge graph patterns to one entire graph data model
4. Store graph data model in graph database

6 CONCLUSION AND OUTLOOK

Standards are available in various forms of representation like XML. Although XML offers machine-readable, structured format, the representation of semantic relations is limited. Since this is a prerequisite of machine-actionable standards, a different form of representation is needed. (Loibl et al., 2020)

This research paper has shown that existing approaches for transferring XML data into a semantic, graph-based representation can be adapted to the domain of digital standards and especially formulas as an information object. Therefore, a concept based on existing approaches was developed and elaborated. This contains an XML-parser as well as a specific graph pattern for modeling formulas. In order to validate the concept, an expert system for the design of rolling bearings based on DIN ISO 281 was presented. For this purpose, formulas were extracted from the XML document and modeled using the predefined graph pattern. Lastly, the sub models were merged in a single graph database and accessed through a chatbot application.

Future work will focus on the generalization of the developed approach. Therefore, the structure of formulas in standard documents will be further analyzed in order to develop a building block for formulas. This, in turn, allows conclusions to be drawn about the actual design of future digital standards. In addition, further information objects such as texts, tables and graphics can be examined regarding their suitability for machine-actionability. In this respect, the procedure for creating the knowledge graph presented in this research paper can be extended and generalized. Finally, future work will focus on other fields of application for machine-actionable standards. It is conceivable to develop micro services which can be accessed via web and integrated in existing design tools such as CAD. Moreover, machine-actionable standards could serve as source for product requirements or test specifications.

REFERENCES

- Blind, K. and Hess, P. (2018), *Deutsches Normungspanel - Normungsforschung, -politik und -förderung*. [Online] Deutscher Förderverein zur Stärkung der Forschung zur Normung und Standardisierung e.V.. Available at: <https://www.normungspanel.de/> (11/15/2020) (in German).
- Bundesministerium für Wirtschaft und Energie (2017), *Die Rolle der Normung 2030 und Gestaltungsoptionen unter Berücksichtigung der technologiespezifischen Besonderheiten der IKT in der Normung und Standardisierung*. [Online] Bundesministerium für Wirtschaft und Energie. Available at: <https://www.bmwi.de/Redaktion/DE/Publikationen/Studien/rolle-der-normung-2030.html> (11/15/2020) (in German).
- Cruz, C. and Nicolle, C. (2010), "Graph-based rules for XML data conversion to OWL ontology", *WEBIST 2010 - Proceedings of the 6th International Conference on Web Information Systems and Technology*, Vol. 1, pp. 175-178. <https://doi.org/10.5220/0002791501750178>.
- DIN and DKE (2020), *DIN and DKE Roadmap - German Standardization Roadmap Industrie 4.0*. [Online] German Institute for Standardization. Available at: <https://www.din.de/blob/65354/57218767bd6da1927b181b9f2a0d5b39/roadmap-i4-0-e-data.pdf> (11/15/2020).
- DIN (2010), *DIN ISO 281: Rolling bearings - Dynamic load ratings and rating life (ISO 281:2007)*. [Online] German Institute for Standardization. Available at: <https://www.beuth.de/de/norm/din-iso-281/134168364> (11/15/2020). <https://dx.doi.org/10.31030/1718715>.
- Esser, M. and Mueller, S. (2019), "Semantische Extraktion aus der zentralen XML-Datenbank", *DIN Mitteilungen* (in German).
- Hacherouf, M., Nait-Bahloul, S. and Cruz, C (2019), "Transforming XML schemas into OWL ontologies using formal concept analysis", *Software and Systems Modeling*, Vol. 18. <https://doi.org/10.1007/s10270-017-0651-4>.

- Kwon, S., Monnier, L.V., Barbau, R. and Bernstein, W.Z. (2020), "Enriching standards-based digital thread by fusing as-designed and as-inspected data using knowledge graphs", *Advanced Engineering Informatics*, Vol. 46. <https://doi.org/10.1016/j.aei.2020.101102>.
- Loibl, A. and Nagarajah, A. (2019), "Classification of standards to determine their suitability for a machine-actionability", *17. Gemeinsames Kolloquium Konstruktionstechnik*, Aachen, 10/01/2019 - 10/02/2019, RWTH Aachen University, Aachen, pp. 174-185. <http://doi.org/10.18154/RWTH-2019-08786> (in German).
- Loibl, A., Manoharan, T. and Nagarajah, A. (2020), "Procedure for the transfer of standards into machine-actionability", *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, Vol. 14 No. 2. <https://doi.org/10.1299/jamdsm.2020jamdsm0022>.
- Manoharan, T. and Koehler, P. (2018), "Knowledge Provision as a service", *16. Gemeinsames Kolloquium Konstruktionstechnik*, Bayreuth, 10/11/2018 - 10/12/2018, pp. 224-235. (in German).
- Manoharan, T., Loibl, A., Nagarajah, A. and Koehler, P. (2019), "Approach for a Machine-Interpretable Provision of Standard Contents Using Welded Constructions as an Example", *Proceedings of the Design Society: International Conference on Engineering Design*, Vol. 1 No. 1, pp. 2477-2486. <https://doi.org/10.1017/dsi.2019.254>.
- Microsoft Corporation (2020), *Introduction to Bot Framework Composer*. [Online] Microsoft Corporation. Available at: <https://docs.microsoft.com/en-us/composer/introduction> (11/15/2020).
- Microsoft Corporation (2020), *Microsoft Bot Framework*. [Online] Microsoft Corporation. Available at: <https://dev.botframework.com/> (11/15/2020).
- National Information Standards Organization (2017), *STS: Standards Tag Suite*. [Online] National Standards Information Organization. Available at: <https://www.niso.org/publications/z39102-2017-sts> (11/15/2020).
- Pinquier R., Véron P., Segonds F. and Zynda T. (2019), "A Property Graph Data Model for a Context-Aware Design Assistant", *IFIP International Conference on Product Lifecycle Management*, Moscow, 07/08/2019 - 07/12/2019, Springer, Cham. https://doi.org/10.1007/978-3-030-42250-9_17.
- Rauch, E., Dallasega, P. and Matt, D.T. (2016), "The way from Lean Product Development (LPD) to Smart Product Development (SPD)", *Procedia CIRP*, Vol. 50, pp. 26-31. <https://doi.org/10.1016/j.procir.2016.05.081>.
- Sana, A. and Suganthi, G. (2017), "Modeling and Storage of XML Data as a Graph and Processing with Graph Processor", *2017 World Congress on Computing and Communication Technologies (WCCCT)*, Tiruchirappalli, Tamil Nadu, India, pp. 16-19. <https://doi.org/10.1109/WCCCT.2016.14>.
- Schuch, W. and Wischhoefer, C. (2018), "Abschluss des Projekts XML100", *DIN Mitteilungen* (in German).
- Tekli, J., Charbel, N. and Chbeir, R. (2016), "Building Semantic Trees from XML Documents," *Journal of Web Semantics*, Vol.37-38, pp. 1-24. <https://doi.org/10.1016/j.websem.2016.03.002>.
- Wischhoefer, C. and Koch, H. (2016), "Die XML-Datenbank der DIN-Gruppe", *DIN Mitteilungen* (in German).
- Zhenyong, W., Xinguo, M., Lina, H. and Goh, A.M. (2020), "Product Development-Oriented Knowledge Service: Review, Framework, and Solutions", *IEEE Access*, Vol. 8, pp. 64442-64460. <http://dx.doi.org/10.1109/ACCESS.2020.2984631>.
- Ziegenfuss, A. (2020), *Digital Standards Systems — An Integrated Approach to Engineering Standards Usage*, [Online] SAE Technical Paper. Available at: <https://www.sae.org/highlights/digital-standards-system-whitepaper> (11/15/2020).