ARCHIVES
of
FOUNDRY ENGINEERING

ISSN (1897-3310)
Volume 15
Issue 2/2015

39 – 42

8/2

Published quarterly as the organ of the Foundry Commission of the Polish Academy of Sciences

# The Use of Formal Knowledge Representation in Operating on Resources Concerning Cast Iron Processing

**S. Kluska-Nawarecka [b], D. Wilk-Kołodziejczyk [a, b,] *, G. Rojek [a], K. Regulski [a], G. Polek [a]**
[a] AGH University of Science and Technology, Mickiewicza 30, Cracow, Poland,
[b] Foundry Research Institute, Zakopianska 73, Cracow, Poland;
*Corresponding author. E-mail address: wilk.kolodziejczyk@gmail.com

Received 08.12.2014; accepted in revised form 31.03.2015

## Abstract

The problem of materials selection in terms of their mechanical properties during the design of new products is a key issue of design. The complexity of this process is mainly due to a multitude of variants in the previously produced materials and the possibility of their further processing improving the properties. In everyday practice, the problem is solved basing on expert or designer knowledge. The paper is the proposition of a solution using computer-aided analysis of material experimental data, which may be acquired from external data sources. In both cases, taking into account the rapid growth of data, additional tools become increasingly important, mainly those which offer support for adding, viewing, and simple comparison of different experiments. In this paper, the use of formal knowledge representation in the form of an ontology is proposed as a bridge between physical repositories of data in the form of files and user queries, which are usually formulated in natural language. The number and the sophisticated internal structure of attributes or parameters that could be the criteria of the search for the user are an important issue in the traditional data search tools. Ontology, as a formal representation of knowledge, enables taking into account the known relationships between concepts in the field of cast iron, materials used and processing techniques. This allows the user to receive support by searching the results of experiments that relate to a specific material or processing treatment. Automatic presentation of the results which relate to similar materials or similar processing treatments is also possible, which should make the conducted analysis of the selection of materials or processing treatments more comprehensive by including a wider range of possible solutions.

**Keywords:** Application of information technology to the foundry industry, Automation and robotics in foundry, Semantic analysis, Semantic search

## 1. Introduction

Selection of materials in the design of new products as well as the selection of appropriate treatment providing the required properties is topical issue all the time. Decisions in this scope are taken by a designer or a technologist based on technical conditions and his own knowledge or by reference to literature in the field of materials research. In both cases, taking into account the rapid growth of data, the importance are gaining computer tools to manage information (data from experiments or literature) on the properties and change in the properties of materials as a result of treatment. Well-known is the fact that for a given product one can typically use an alloyed material providing certain properties in as-cast state or cast iron that meets specified requirements when subjected to properly conducted heat treatment. To find the material that offers the specified properties it is enough to construct an appropriate query and address it to a database, but to get the result in the form of a list of similar materials that can meet the required criteria after treatment (including information about the treatment itself), a more advanced search is necessary.

## 2. Formal Knowledge Representation

The specificity of the Internet is the best example of problems faced by the continuously expanding knowledge bases possessed by research institutes and large companies with research departments. Unfortunately, most of resources are intended to be read only by humans (in the form of text or hypertext documents) and are not in any way friendly to crawlers that search the repository. Robots are not able to intelligently detect the meaning of the resources they search. Therefore, it is currently so important to search for keywords that are in the meta-description of the document. Thanks to them, the search engines are able to fit in better or worse way the resources to the phrase entered by the user. Unfortunately, this is not an effective solution. The search for relevant information, or going over entire documents becomes the challenge more and more severe.

This paper proposes the use of formal knowledge representation as a bridge between physical repositories of data in files and user queries, which are usually formulated in natural language. Ontology as a formal form of knowledge will take into account the known relationships between concepts in the field of cast iron, materials used and treatment processes. This allows the user to receive support, which consists in searching for the results of experiments that relate to a specific material or processing treatment. It is also possible to automatically indicate the results concerning related materials or similar processing methods, which should make the carried out analysis for the choice of material or treatment process more complete by taking into account a broader range of possible solutions (Fig. 1).

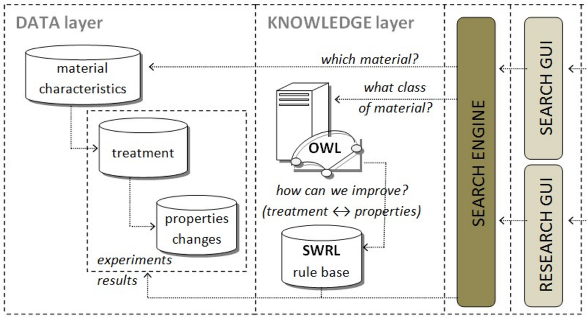


Fig. 1. Schema of multiple layers of the system architecture for semantic search in material characteristics and treatment databases

The two technologies that support the creation of knowledge bases in a way allowing the formalisation of knowledge are eXtensible Markup Language (XML) and Resource Description Framework (RDF). XML allows building one's own tags and using them to describe resources, e.g. <component> </ component>. With XML, we can build the structure of the resource in the way we want. Unfortunately, in XML we are not able to encode the meaning of each tag, which makes the tags created by us useless for the processing computer programme. To save the meaning, RDF language is used, which stores individual data in groups of, so called, triples, consisting of <subject, predicate and object> e.g. <Cementite is a structural component>.

In short, RDF is a general-purpose language for writing/presenting semantic information in the knowledge base. With RDF we can create such dependencies as "belongs to", "is a type of" or "has", which is not possible in a standard XML.

Subjects and objects are presented in RDF in the form of a unique Universal Resource Identifier (URI), in the same way as individual pages in the Web, which allows preserving backward compatibility with existing solutions, and also creation of unique identifiers for the resources of knowledge base. Using URI, we are able to describe things that should be easily found in the repository. Due to the use of RDF triples, the individual pieces of information start to be linked together and this enables computers learn the meaning of the sentences from the knowledge base. All these technologies are used to create the logic of semantic applications.

The problem that immediately shows up in the practical application of URI is the use of two different identifiers to represent the same designated object, e.g. material. The computer system should be aware that the definition, though represented by different URI addresses in different knowledge bases (e.g. in different languages or different standards), means the same and refers to the same being. It should be able to apply this information to any knowledge base that will be used, without the need to modify the code or the database itself [1]. The problem has been solved by the use of ontologies.

RDF and XML are the pillars supporting domain-knowledge models which are formal ontologies. OWL ontology (a file in Ontology Web Language) is treated as a document that contains a representation of knowledge, which consists of a set of concepts and relationships between them. Ontologies allow for the implementation of simple grammar used in the knowledge base, due to which the search for data in two databases becomes easier for a computer programme. For further discussion of issues related to ontologies, the Reader is referred to a rich world literature [2-5] or to previous publications of the authors [1, 6-12].

### 2.1. Semantic Analysis

One of the solutions to the problem posed earlier, concerning searching of data resources by robots, is to create a semantic layer that would give meaning to individual resources. The task of the semantic layer is to create a friendly environment to the, so called, agents which, installed in search engine, are searching the resources and can serve many roles. Agents would be able to search through the successive resources (e.g. sample after sample), selecting only the results relevant to the user. For example, an agent for metallurgical applications could search the semantic layer, taking into account different classes of materials according to the criteria of carbon content in the alloy and property such as corrosion resistance, finding the material or group of materials looked for. The problem of semantic analysis in the context of searching the resources can be divided into two tasks, fundamentally different in the conception of action. This is

(1) an analysis of the meaning of indexed resources involving natural language processing and (2) the use of search engines processing the resources through semantic layers. Both types of search engines are trying to discover the meaning of the indexed content, but they do it in two different ways. The system discussed here uses search engine of the second type. It is a search engine searching the resources of semantic layer, which means searching the knowledge base stored in the form of ontology. Its operation mainly consists in searching the indexed ontologies and reasoning using SWRL (SemanticWeb Rule Language) rules and dependencies contained therein. With the possession of knowledge base, the user is able to ask the search engine complicated questions using SPARQL (SPARQL Protocol And RDF Query Language) or submitting them in the form of keywords.

# 3. Architecture of the System

The utilitarian aim of this work was to propose a model and create a simple semantic search engine based on domain ontology in the field of material properties to show the advantages and opportunities offered by semantic search. Creating a semantic search engine based on the domain ontology in the field of cast iron, accessible from a web browser, enabled using effectively the advantages of semantic analysis in the work of researcher and technologist. Thus processed tool can be interesting for several reasons. The first is the possibility to create an innovative search engine showing the advantages and disadvantages of semantic search for data on materials research. The basic concept of the system was that the developed semantic search engine should return correct results, i.e. the most relevant to the user query, including an element from the searched ontology (class of material asked by the user), along with its properties inherited from parent elements, and write a list of items related to the base element, that is, indicate other materials having the required properties, or likely to obtain the required properties upon the application of a particular treatment.

## 3.1. Semantic Search Engine

As with any web application, also the described semantic search engine must be implemented in a specialised environment that will support all aspects of the system. The client (user) is required to have only a web browser that supports HTML5 and JavaScript. The application server instead should be on a machine running a Linux distribution or any Windows XP or later. The required software is Java 1.6+, Scala 2.3 + and sbt in version 0.13. These are the minimum requirements that are necessary for the proper operation of the system in a production environment.

As an application framework and basis for further development of functionality, the gaining popularity Play Framework version 2.2 was used. Its main advantage is the use of proprietary server, which allows skipping the use of the Java plug-in in the user's browser. This allows speeding up user interaction with the system and ensures the availability of the system on more platforms. Play Framework allows creating scalable web applications in Java and Scala. Both these languages can be used simultaneously in the project, which provides a variety of solutions to specific problems. Play Framework uses the MVC (Model-View-Controller) model, which is nowadays the basis for the creation of web applications. User interaction with the system should be as simple and intuitive as possible. Example of user interaction with the system (in Polish language) is shown in the Figure 2. The semantic search engine on the home page contains only one field to interact with the system, i.e. the text box "Search". The flow of activities is as follows:

1. User enters a search string in natural language;
2. The system processes the user's request in terms of:
   a. language of the query;
   b. meaning of the query.
3. The system searches the ontology with SPARQL.
4. The system sorts the obtained results based on the set search priorities.
5. The system generates a result based on the data found in the ontology.
6. The system sends the result to the user.
7. User is viewing the result.


Fig. 2. Screen shoot from Polish interface of the described search engine

The created model of domain ontology used in the application is based on the popular framework of Semantic Jena Web. Jena provides an API for extracting and storing the data in RDF graphs. Graphs are represented in the form of an abstract model which can be created based on a file, on URI or on a relational database. The framework has many built-in reasoners and the ability to configure external libraries as extensions to the system. One of the main strengths of Jena framework is full support for OWL Full. In the project, Jena was used to create a model of the ontology which, by processing a metal.owl file containing semantic data such as XML/RDF, produces a graph containing the relationships between various nodes. Additionally, the model assumes the possibility of using SWRL reasoner that allows expanding the graph to include nodes obtained by inference from the rules contained in the ontology. Owing to the graph produced, it is easy to search the ontology and receive elements of interest to the user. Ontology is a file that contains the data saved in OWL (Web Ontology Language), which is a derivative of RDF/XML format. Compared with the RDF format, OWL contains more definitions, types, and elements. The ontology used in the project is divided into the following classes: owl:Class representing the basic entities that describe a given field of knowledge, owl:ObjectProperty including the properties of objects which allow adding features to each class, and owl: NamedIndividual instances that represent individual objects in the knowledge base.

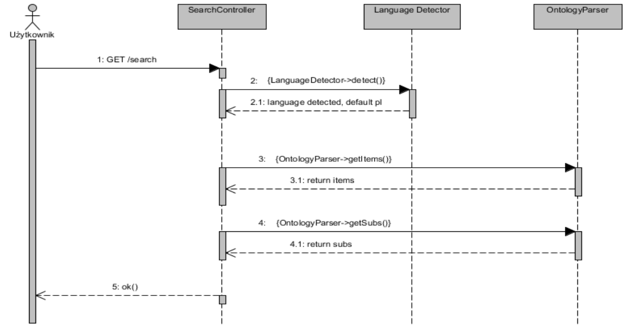


Fig. 3. Graph showing the sequence of ontology parsing

The above graph (Fig. 3) shows a sequence diagram of the ontology parsing, which includes a class of #LanguageDetector, #SearchController and #OntologyParser. Parsing of an ontology conducts as in following example: Ontology includes i.e. a class of #white_cast_iron and relationships between this class and other objects of ontology The property named rdfs:subClassOf allows creating a hierarchy of classes and inheritance of properties by the descendants. In contrast, rdfs:domain and rdfs:range allow assigning properties to an object/class. The class of #white_cast_iron has got the assigned property "carbon is present in the form of", which refers to the #Cementite object. Due to this design of RDF/XML and OWL languages, complex knowledge bases can be created, which allow a given domain to be described in a manner "understood" by the computer programmes. The search engine also allows performing complex queries using for this purpose the SPARQL language, which was created specifically for searching RDF graphs. To get all the items matching the query written by the user, all classes that have a label in Polish and correspond to the query received from the user are selected.

# 4. Conclusions

Under this project the Authors have created a semantic search engine using the domain ontology in the field of cast iron in the form of interactive applications accessible from a web browser. The main recipient of this application are engineers working in industry. The semantic search engine enables them to find data on the properties of materials with links between them.

The search engine based on a formal knowledge representation allows for a more flexible approach to the search for resources - in addition to specific materials found in the database, one can search for similar materials belonging to the same group, or having similar properties, or even - having similar characteristics after processing.

Ultimately, the search engine can also support the management of the experiments allowing easy-to-use search of the interesting results of research or descriptions of the experiments to edit or add new records.

# Acknowledgements

The work was financed within the framework of the international project No. 820/N-Czechy/2010/0 of 30 November (part: knowledge and databases). Financial support of the National Centre for Research and Development (LIDER/028/593/L-4/12/NCBR/2013) is also gratefully acknowledged (in the part of semantic search engine).

# References

[1] Regulski, K., Rojek, G. & Kusiak, J., 2014, Process of Ontology Construction of Rolling Metal Sheets Industrial Process. *Key Engineering Materials*. 622-623, 978-985.

[2] Goczyła, K. (2011). *Ontologie w systemach informatycznych*, Oficyna Wydawnicza EXIT, Warszawa. (in Polish)

[3] Staab, S., Studer, R. (2004). *Handbook on Ontologies*, Berlin: Springer-Verlag.

[4] Noy, N., McGuinness, D. (2001). *Ontology Development 101: A Guide to Creating Your First Ontology*. Technical report, Stanford Knowledge Systems Laboratory.

[5] Cristiani, M. & Cuel, R. (2005). A Survay on Ontology Creation Methodologies. *International Journal on Semantic Web & Information Systems*. 1(2), 49-69.

[6] Kluska-Nawarecka, S., Mrzygłód, B., Durak, J. & Regulski, K. (2010), Analysis of the applicability of domain ontology in copper alloys processing. *Archives of Foundry Engineering*. 10(2). 69-74.

[7] Wilk-Kolodziejczyk, D., Rojek, G., Regulski, K. & Kluska-Nawarecka, S. (2014). The Implementation of Computer Platform for Foundries Cooperating in a Supply Chain. *Archives of Foundry Engineering*. 14(3), 111-116.

[8] Wilk-Kolodziejczyk, D., Regulski, K., & Rojek, G. (2014). The Decision Support System in the Domain of Casting Defects Diagnosis. Archives of Metallurgy and Materials. *Archives of Foundry Engineering*. 14(3).

[9] Regulski, K., Kluska-Nawarecka, S. (2012). *Knowledge integration computer tools and algorithms in the improvement of the production processes of cast-steel castings*. Artificial Intelligence in the Knowledge and Information Systems, Instytut Odlewnictwa, Kraków.

[10] Kluska-Nawarecka, S., Durak, J. & Regulski, K. (2012). Aspects of knowledge integration using ontological model in issues of decision support in the process of die forging, *Computer Methods in Materials Science*. 12(2), 63-69.

[11] Wilk-Kołodziejczyk, D., Regulski, K., Dziaduś-Rudnicka, J. & Kluska-Nawarecka, S. (2012). Overview of activities on the Internet devoted to casting technology. *Archives of Foundry Engineering*. 12(2), 245-250.

[12] Mrzygłód, B. & Regulski, K. (2011). Model of knowledge representation about materials in the form of a relational database for CAPCAST system. *Archives of Foundry Engineering*. 11(3), 81-86.