INTEGRATION OF TECHNOLOGICAL DATA USING ONTOLOGIES IN THE INNOCAST DATABASE MAPPING TO CASTNODE KNOWLEDGE BASE

REGULSKI Krzysztof[1], WILK-KOŁODZIEJCZYK Dorota[1,2], KLUSKA-NAWARECKA Stanisława[2]

[1]AGH University of Science and Technology, Krakow, Poland, EU
[2] Foundry Research Institute, Krakow, Poland, EU

Abstract

CastNode platform is a knowledge base dedicated to casting technology and foundries. Applications database (the InnoCast) integrates four components of knowledge: (1) foundries and supplier base, (2) the base of innovations in the field of casting, forging and rolling, (3) database of publications and (4) patents for casting techniques and products. The schema of the relational database is a diagram of the data structure and is divided into several parts although physically it forms one component. Each module is treated separately. This is due to the complexity of the application and requirement to process it into a knowledge base where different parts are merged by keywords, which are common to all components. Additionally, the system is enhanced by a module of knowledge in the form of ontologies, which shows what data are stored and what relationships take place between them. This method of archiving has many advantages. The main advantage is ease of searching the contents according to users' criteria. In a very extensive database containing several thousands of records, we can search for information in a short period of time. The paper presents the architecture of the CastNode platform and the methodology of integration of technological data using mapping of database schema to ontology.

Keywords: Application of Information Technology to the Foundry Industry, Innovative Foundry Technologies and Materials, Ontology, Knowledge Sharing, Knowledge Base

1. INTRODUCTION

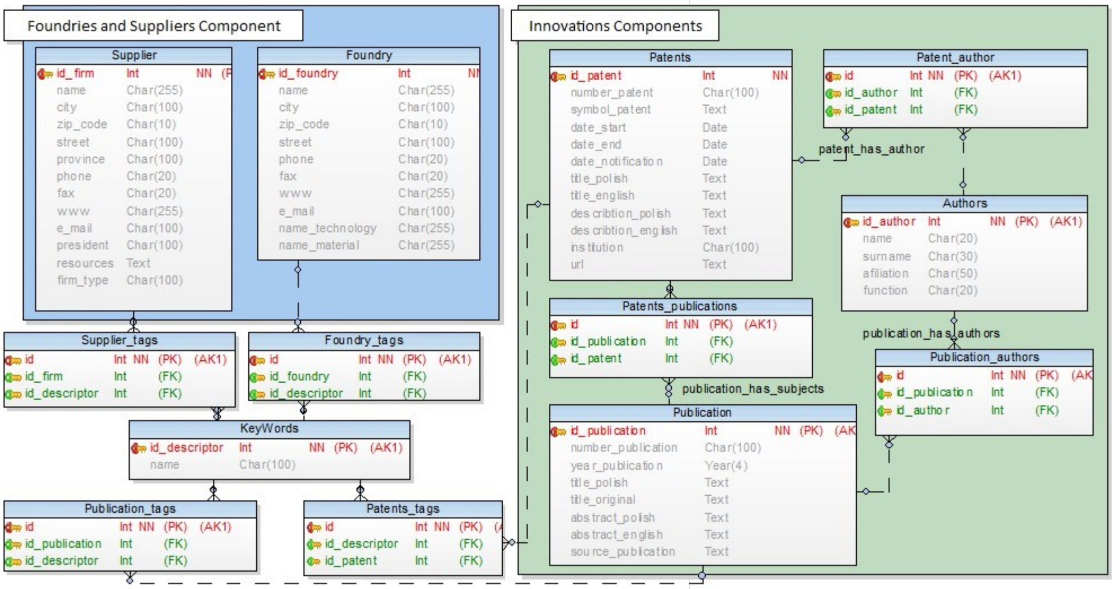
For several years, Foundry Research Institute in Krakow has been running projects, in which data sets are being developed making components of knowledge about casting and metallurgy. These components are intended to provide the user with a constant supply of current information and knowledge in the field of casting, not only derived from literature, but also from other sources, including various sources available on the Internet.

InnoCast module discussed in this report refers to the knowledge about innovations in foundry and metallurgy. These innovations are understood as published patents, research projects, and publications in journals. In addition, data were collected on modern materials for the foundry and on suppliers of these materials. All these data have been organized for the purpose of import into a database, and then a common, relational data model was created for them and the data loading operation into a MySQL database was performed. Up to date information on innovations in foundry and metallurgy was collected. The data collected included various types of information: (1) concerning the companies involved in the metallurgical industry, and specifically foundries and suppliers for foundries, (2) on articles and publications [1-5] and (3) on patents and research projects for foundry products and techniques.

The collection included knowledge about: (1) process improvements, new processes of rolling, forging and continuous casting and (2) theoretical (metallurgical) foundations to improve the properties of products manufactured by rolling, forging, and continuous casting. It was divided into different files with different structures for projects from various sources. It has been adapted to the needs of a relational database, and a common, unified data structure was developed.

1.1. Relational Data Model

The database application integrates four components of knowledge provided by the Foundry Research Institute in Cracow: supplier base, base for innovations, base of publications and base of patents in the field of casting, forging and rolling. The base is a diagram of data structures needed to record information received from the Foundry Research Institute. It is divided into several parts and, although physically it forms a single component, each module is treated separately. This is due to the complexity of the application and attempt to process it into a knowledge base; its individual parts are merged by the table of keywords, which are common to all components.



. **Fig. 1** Entity Relationship Diagram for the InnoCast

The database can be edited (supplemented and updated) via a standard mysql input interface or via ready-made CRUD service tools.

2. CASTNODE KNOWLEDGE BASE

CastNode platform is a system integrating knowledge about casting and metallurgy archived and made available through the components of the knowledge accumulated in InnoCast databases. The platform uses an ontological model to navigate the system and support the management of the knowledge components.

This concept consists in mapping relational database tables to the base of knowledge, i.e. to the base that has well-defined contextual relationships between individual tables. The tables are treated as "data" and the defined links as compounds. Combining the data in an appropriate context creates information, while combining multiple information with the relationships taking place between this information and the context creates knowledge. All this serves the possibility of creating rules of inference based on knowledge and ontology for a given topic. The use of the described procedure allows processing the ordinary relational database into a knowledge base.

2.1. Ontological knowledge base

Currently, computers are becoming "smarter". They are able to process more and more information and can also be faster in drawing conclusions and making decisions. However, the way in which the computer operates is based on mathematics, and more specifically on logic. Ontology in terms of information technology is also a formalism of logic, because knowledge is made up of rules or arguments, i.e. of interrelated information. In contrast, information consists of data connected by relations.

Ontology is a kind of mediator between man and computer. The requirement imposed onto a computer is that it should understand knowledge in the way it is understood by human. The problem is not understanding the knowledge in the grammatical sense, because this goal has already been largely implemented, but understanding it in the semantic sense, or understanding the links between the concepts expressed in natural language and abstract concepts. Here the main role is played by the "semantics" of inference [6,7].

Possessing some data, we can give them context. Combining this data with appropriate relationships we create, so-called, "triples" that are simple phrases in natural language containing subject, predicate and object. In RDF, each of these parts of ontology has its own description in the form of URI. Data is written as a class of ontology. One can also store additional components i.e. the properties defined as DatatypeProperty or relationships between individual ObjectProperty resources. In this way, the basic objects are stored that make up the entire system, which also includes the rules of inference. In addition to the built-in rules of hierarchy and subsumption, the user is able to define his own rules, on the basis of which the requesting machines automatically process the knowledge base. One of the most popular tools for creating and editing of ontology is Protege program. It allows defining all the aforementioned components. In addition, it may create graph dependencies, which also form an important part of the data processing system from the point of view of the user. It is visualization, which another user is able to quickly understand and thus learn from the knowledge offered to him.

2.2. Ontology-Based Data Integration

For data integration it is necessary to provide in a transparent and uniform manner the user or machine with access to heterogeneous data sources. Information technologies long moved towards dispersion of data due to the dynamics of the processes of creating repositories, the use of local information across applications, and hence the need to diversify data structures and syntactic formalisms. Currently, we have a situation in which access to, generated by various sources, information is heavily congested, the growth of data is not suppressed, and therefore it is not possible to dispense with the idea of having scattered and varied archiving systems. The developing data integration techniques [8-9] pursue a common solution where there is a need for application of an additional layer in systems architecture, which means that it is suggested to place between the data layer and application layer a conceptual layer allowing for semantic description of data sources. As a conceptual layer we use ontologies to ensure uniform access to sources in terms of the naming and data redundancy [10-11].

When designing the systems we often use the ER or UML models acting as interpreters of data structures. Usually, however, these models are not utilized during use of the system. In the proposed solution, it is ontology that performs this role.

2.3. Process of the ontology construction

Creating ontology may be done in small fragments, depending on the application needs. In this case, knowledge model is determined by the data. The ability to embed it in the context of domain knowledge is provided by the ontological model discussed in the earlier part of the work. Ontology written in the language of OWL will enable in the future the integration of model data structures with decision rules resulting from the processes of data mining [12]. Such rules may be extracted e.g. on the basis of the studies of regression and correlation. Systemic use of thus created knowledge ensures the correct semantic description of the data structures. To this end, a database schema transformation to the form of an ontology stored in OWL was performed.

Currently, methods are already known for semi-automatic transformation of database schema into ontologies written in OWL. Most of them, however, require interaction with the ontology designer at one of the transformation stages. It turns out that for simple database, the transformation can be successfully performed manually, using tools for the edition of ontologies, as well as basic transformation rules.

The first step is to save all the tables and individual fields in graphical form. Most tables will reflect the future classes. Linking tables, which are the embodiment of a many-to-many (n:m) relation, form the basis of future relations in the ontology (object properties). Also, fields being foreign keys establish relationships in the ontology. In this way, one can limit the number of attributes in the diagram to obtain a basis for future ontology.

The tables, in which the foreign key is also the primary key, will form subclasses in the ontology combined by is_a relationship. After transformations performed in this way, what remains for the designer is to decide how to process the remaining fields. Some of them will automatically create in OWL the datatype properties, but the designer can decide that individual attributes become subclasses, if this is relevant to the description field. The correct description of each class involves the description of an object by object properties and datatype properties. The difference between these properties is that the object properties connect a particular class with another class using the relationship. Datatype properties allow us to assign to a class properties which are expressed by given values. The next step is mapping of tuples from a database to instances of classes in the ontology. For this purpose, one can use ready-made tools accelerating this process or manually enter individual instances. Supplementing of instances is done to adapt a model to future processing using rules stored in e.g. SWRL.

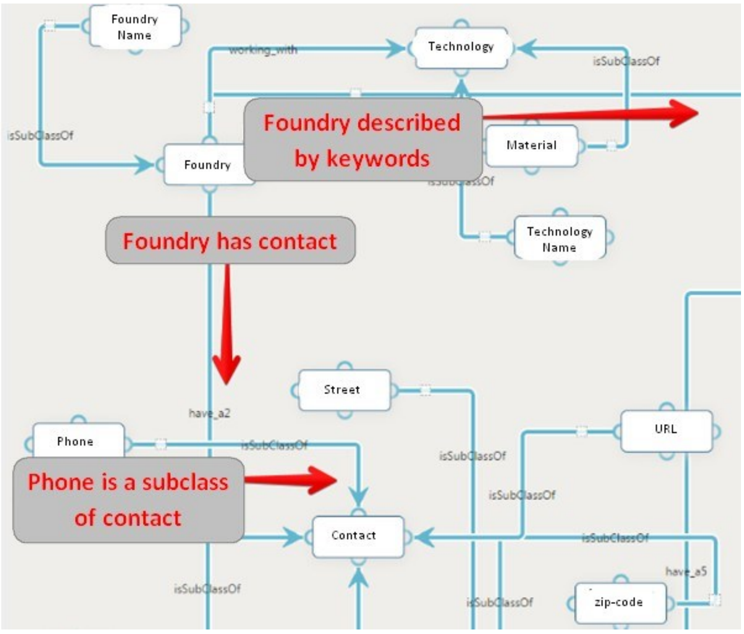
With the dependence created, we are able to generate a directed graph showing the whole ontology (Figure 2.).

2.4. Implementation of the CastNode platform

The project assumption was to create a Web application that supports relational database, along with its mapping to ontology, which is a component of knowledge. The application performs all CRUD operations on the database for application components. Additional functionality is the ability to import large batches of data to repository. This is accomplished by allowing the import of .xls files directly from the view level. This allows the user to quickly back up data that is stored on computer disks or enter data from the documents. The limitation imposed on the import mode is defined structure of .xls files themselves. To make import correctly, it is recommended to use the templates provided in import section, which is available for each specific application module.

An important feature of the application is providing a database in the context of knowledge. To this end, the ontology module reproduces relationships that occur in the database, giving them a context and thereby building knowledge. Each table is mapped to the class of ontology and appropriately connected to the base by semantics. Browsing the graph, the user is able to determine what data are stored in the application, of which parts a particular module consists, and what messages can be obtained from this module. Knowledge about the application is presented in a way natural for the user, making it easier to understand the information held in the database. Additionally, connecting the application components to the appropriate classes of ontology allows direct navigation across the application to perform data processing.

The system architecture is based on a division into several smaller modules. The modules use a variety of tools realizing the objectives that are assigned to them. Dividing the project and avoiding bottlenecks we make the application prone to expansion by further components. Just by adding a reference to the relevant project, which requires a new component, it is immediately possible to work with its contents. This is consistent with one of the principles of good S.O.L.I.D design namely "Open-Closed Principle," which says that a class or project will be open for extension but closed for own modification.



. **Fig. 2** Fragment of the ontology of the Foundry module

CONCLUSIONS

CastNode platform is intended for employees of Foundry Research Institute in Cracow, as a "reservoir" for the two types of information: (1) concerning the companies involved in the metallurgical industry and specifically foundries and suppliers for foundries, (2) on publications and patents for products and techniques of casting. The system has a module of knowledge in the form of ontology, which shows which data are stored and which dependencies take place between them. The purpose is to enable transparent sharing of knowledge and relationships between the data store creating knowledge in a broader context. Such a method of recording has a number of advantages. The main advantage here is ease of searching the content according to one’s own criteria. In a very extensive database it provides the ability of semantic search based on the relationships and not just on keywords. Expanding the application to include the system of inference can further increase the efficiency and clarity of the data collected and allow easy extension of the structure by new components, e.g. data from the Web [13]. The presented work discusses the assumptions and CastNode platform architecture as well as methodology of data model mapping to ontology, which is one of the most important aspects of semantic data integration.

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 technologies).

REFERENCES

[1] OLEJARCZYK-WOŻEŃSKA I., ADRIAN A., ADRIAN H., MRZYGŁÓD B., Parametric representation of TTT diagrams of ADI cast iron, Archives of Metallurgy and Materials, vol. 57 (2012), pp. 981-986.

[2] DAVID J., JANCIKOVA Z., FRISCHER R., VROZINA M., Crystallizer's Desks Surface Diagnostics with Usage of Robotic System, Archives Of Metallurgy And Materials, Vol. 58, Is. 3, 2013, pp. 907-910.

[3] GLOWACZ, A., GLOWACZ W., GLOWACZ Z., Recognition Of Armature Current Of DC Generator Depending On Rotor Speed Using FFT, MSAF-1 and LDA, Eksploatacja i Niezawodnosc - Maintenance And Reliability, vol. 17 is. 1, 2015, pp. 64-69.

[4] OLEJARCZYK-WOŻEŃSKA I., ADRIAN H., MRZYGŁÓD B., Mathematical Model of The Process of Pearlite Austenitization, Archives of Metallurgy and Materials, vol. 59 (2014), pp. 613-617.

[5] DAVID J., SVEC P., FRISCHER R., GARZINOVA R., The Computer Support of Diagnostics of Circle Crystallizers, METALURGIJA, Vol. 53, Is. 2, 2014, Pp. 193-196.

[6] KLUSKA-NAWARECKA S., WILK-KOLODZIEJCZYK D., REGULSKI K., DOBROWOLSKI G., Rough Sets Applied to the RoughCast System for Steel Castings, Intelligent Information and Database Systems, Lecture Notes in Computer Science, eds. Nguyen, N., Kim, C.G., Janiak, A., ACIIDS 2011, Pt II Volume: 6592, 2011, Pages: 52-61.

[7] KLUSKA-NAWARECKA S., ŚNIEŻYŃSKI B., PARADA W., LUSTOFIN M., WILK-KOŁODZIEJCZYK D., The use of LPR (logic of plausible reasoning) to obtain information on innovative casting technologies, Archives of Civil and Mechanical Engineering, vol. 14 iss. 1, 2014, pp. 25–31.

[8] KLUSKA-NAWARECKA S., REGULSKI K., KRZYŻAK M., LEŚNIAK G., GURDA M., System of semantic integration of non-structuralized documents in natural language in the domain of metallurgy, Archives of Metallurgy and Materials, Polish Academy of Sciences. vol. 58 iss. 3, 2013, pp. 927–930

[9] KLUSKA-NAWARECKA S., WILK-KOŁODZIEJCZYK D., DAJDA J., MACURA M., REGULSKI K., Computer-assisted integration of knowledge in the context of identification of the causes of defects in castings, Archives of Metallurgy and Materials, vol. 59 iss. 2, 2014, pp. 743-746.

[10] ANGELE J., GESMANN M., Data integration using semantic technology: a use case, Rules and Rule Markup Languages for the Semantic Web, 2006, pp.58-66, DOI: 10.1109/RULEML.2006.9

[11] UZDANAVICIUTE V., BUTLERIS R., Ontology-based Foundations for Data Integration, BUSTECH 2011, The First International Conference on Business Intelligence and Technology, 2011, pp.34-39

[12] REGULSKI K., SZELIGA D., KUSIAK J., Data Exploration Approach Versus Sensitivity Analysis for Optimization of Metal Forming Processes, Key Engineering Materials, 2014, vols. 611–612, pp. 1390–1395.

[13] OPALINSKI A., DOBROWOLSKI G., Framework for Opinion Spammers Detection, Multimedia Communications, Services and Security. Springer International Publishing, 2014. 202-213.