INFO- 2089 UDK : 303:681.3:001
Primljeno / Received: 2013-13-03 Izvorni znanstveni rad / Original Scientific Paper

NODES OF KNOWLEDGE METHOD FOR KNOWLEDGE REPRESENTATION

METODA ČVOROVA ZNANJA (NOK) ZA PREDSTAVLJANJE ZNANJA

Mile Pavlić, Alen Jakupović[1], Ana Meštrović
Department of Informatics, University of Rijeka, Rijeka, Croatia; Bussines Department, Polytechnic of Rijeka, Croatia[1]
Odjel za informatiku, Sveučilište u Rijeci, Rijeka, Hrvatska; Poslovni odjel, Veleučilište u Rijeci, Rijeka, Hrvatska[1]

Abstract
This paper introduces a graphical based method for knowledge representation named Nodes of Knowledge (NOK). Basic concepts for graphical representation (nodes and links) as well as their variations are described. Furthermore, the idea of application the NOK method in different domain of knowledge is presented. The conceptual model of a system based on the NOK method designed for generating new knowledge out of existing database is described. The result of the NOK method implementation into the existing information system (IS) is the whole new set of possibilities that leads to the creation of new knowledge that cannot be derived directly from the database. The NOK method can capture different kinds of knowledge: knowledge from existing databases, knowledge embedded in business processes, knowledge stored in business documents, knowledge from dictionaries and encyclopedias.

Sažetak
Ovaj rad uvodi grafičku metoda za predstavljanje znanja nazvanu Čvorovi znanja (skraćeno NOK). Opisani su osnovni koncepti za grafičko predstavljanje i to čvor i veza. Prikazana je mogućnost korištenja NOK metode u raznim područjima predstavljanja znanja. Opisan je model sustava zasnovan na metodi NOK koji bi trebao podatke iz postojećih baza podataka transformirati u mrežu znanja. Uvođenje NOK metode u postojeći informacijski sustav (IS) omogućava stvaranje novih znanja koja se ne mogu generirati iz baze podataka. Metoda NOK može oblikovati i prihvatiti različite vrste znanja: znanje iz postojećih baza podataka, znanje o poslovnim procesima, znanje pohranjeno u poslovnoj dokumentaciji, znanje iz pisanih tekstova.

1. **INTRODUCTION**

The term "knowledge" is often used nowadays. Perhaps the most obvious aspect of its use is in the concept of "knowledge society", which stands for knowledge-based society development. The main goal of knowledge management is to get the most out of knowledge resources /**1**/. Results in this research area are different knowledge management applications, knowledge management systems, knowledge-based systems, and different business intelligence tools /**2**/, /**3**/, but there is still room for improvement and new approaches.

An important aspect of knowledge is certainly knowledge representation. One of the greatest challenges in the domain of knowledge representation is to develop a general representation formalism that can capture the complex systems' requirements. Our motivation was to define knowledge representation formalism general enough to deal with natural language semantics. Furthermore, the main goal was to overcome the drawbacks of the existing information systems that have limited capabilities of generating new knowledge from the existing knowledge. In the classic IS the data is stored in a relational database, which is restrictive. There are many additional business intelligence (BI) systems and tools, such as OLAP or data warehousing tools, which may improve reasoning with data in the information system. However, there is still a problem with natural language semantic capturing. The idea of

our approach is to develop a system that can improve relational database capabilities and traditional IS in organization. The starting point of the presented approach is the new knowledge representation method named Nodes of Knowledge (NOK), which is described in this paper. The NOK method can capture different kinds of knowledge: knowledge from existing databases, knowledge embedded in business processes, knowledge stored in business documents, knowledge from dictionaries and encyclopedias, etc. The NOK method /**4**/, /**5**/, /**6**/ has a graphical representation form, named Diagram of Nodes of Knowledge (DNOK). DNOK uses two elements for graphical representation - nodes and links. Different kinds of nodes are used for term representation. Link between nodes enable grouping terms into more complex expressions. Specifically, a process node is introduced as an aggregation point for the representation of knowledge described in sentences. An array of interconnected process nodes can represent knowledge expressed in a sequence of sentences. The NOK method can be applied to parts of an information system such as documents, databases, reports, screen forms, Web contents or business applications. In this paper, a system that provides relational database extension using the NOK method is described.

2. KNOWLEDGE REPRESENTATION FORMALISMS

The evolution of information technology results in diversity of knowledge representation formalisms and methods. Different formalisms defined in the domain of artificial intelligence have been implemented in a variety of projects for knowledge-based systems. Traditionally, two main approaches of knowledge representation are defined in the domain of artificial intelligence: declarative and procedural. The declarative approach includes logic schemas, network schemas and frames; while the procedural approach is related to production systems, also known as rule systems. Logic schemas concerns first order predicate calculus (FOPC) and a number of other formalisms based on logic: fuzzy logic, modal logic, temporal logic, higher order logic, frame logic, etc. In FOPC knowledge is represented as objects and relations between objects. Network schemas include different formalisms based on graphical notations. The most influential are semantic networks introduced by Quillian /**7**/ and conceptual graphs (CG) introduced by Sowa /**8**/ and later developed and elaborated in /**9**/, /**10**/. Frame based knowledge representation formalisms were introduced by Minsky /**11**/ in order to overcome drawbacks of semantic networks. Apart from these traditional approaches, there is connectionist approach in the field of knowledge representation that introduces neural networks /**12**/. There is a comprehensive classification of knowledge representation methods. It includes data models of traditional database management systems (relational data model, hierarchical data model, network data model and entity relationship model), object-oriented methods (semantic networks, cognitive semantic networks, frame representations and structured inheritance networks), lexical methods (thesauri, computer lexicons), rule-oriented methods (logic-oriented methods, grammatical methods and production-rule systems) and procedural methods (agents, methods and programs). Another category of knowledge representation formalisms are ontologies that are currently considered to be important in the domain of knowledge representation. Ontology is a formal explicit specification of a shared conceptualization /**13**/. There is a great number of formalisms and languages defined and used for ontologies representation (CycL, KL-ONE, OIL, DAML, DAML+OIL, RDF(S), OWL, etc.). Ontology-based languages are also based on traditional formalisms such as logic schemas, semantic networks or frame approach. Furthermore, there is a number of specific formalisms, methods and languages for knowledge representation that aim to represent natural language syntax and semantics. Natural language processing is another important field in the AI domain that has a great impact on knowledge representation. Natural language representation possibilities are studied in numerous researches that may include linguistic approaches /**14**/, /**15**/, /**16**/.

In /**17**/ a method named Multilayered Extended Semantic Networks (MultiNets) is defined as a method for the semantic representation of natural language expressions which can be used as a universal knowledge representation paradigm in human sciences. In /**18**/ some specific graphical methods based on conceptual graphs are defined and described, such as the basic conceptual graphs (BGs) and the simple conceptual graphs (SGs)

methods. Knowledge representation technique named Hierarchical Semantic Form (HSF) is introduced as a method for representing patterns in natural language sequences.

3. CONCEPTS OF THE NOK METHOD

The NOK method consists of the following concepts: node, link (arc), process node, context node, context link and link role. Let us describe the meaning of individual concepts.

3.1. NODE

A Node is a drop of knowledge (term, entity) different from any other knowledge in the model. All concepts that have their own meaning are in nodes. Thus nodes can represent particular named people, things, events, actions, ideas, but also concepts on a higher level of abstraction, such as a person, a table, a sporting event, learning, a feeling, activities etc. A node is the smallest unit of knowledge that cannot be further divided. Larger units of knowledge are represented by a group of connected nodes. In addition, nodes are not groups (relations, tables, classes) of similar entities. A node is not a classification of entities, but rather one individual entity. Every new term is a new node. A node may only contain one term. The meaning (name) of a node is entered in the node symbol (rectangle), see Fig. 1. Even the smallest semantic difference between two terms leads to the creation of a new node. A new node is also created when there is a difference in "essence" between two concepts (e.g. two different persons named Marko). The name of a node is its attribute and gives the semantic identification of the concept in DNOK. The name does not have to be one word only; it can be a group of words, a compound or a derivative with a specific meaning. E.g. "I don't know", "army branch", "come to mind", company "good vibrations". A node can have several attributes. While implementing the method, it is necessary for a node to have a name, but also an identification attribute that identifies it unambiguously. Names of DNOK nodes may not be polysemantic, that is, all homonyms have to be broken down and their precise word versions have to be used in the knowledge record. In human language, it is presumed that, because of their natural intelligence, readers will understand the knowledge that one tries to transfer to them. Our goal is to define such a DNOK that relies on the reader's prior knowledge, but leaves no possibility of alternative interpretations and ambiguities. We have shown the essential requirement of the NOK method: there is a drop of knowledge (essence, concept) which has its own identity (name or identifier) and semantics (essence of the concept, the corresponding fact it replaces) different from anything else, both in reality and in the mind, and such drops are represented by nodes. There are following types of nodes: node (ordinary node, static node, entity, concept, and term), context node (abstract, group, sort, class, framework, type), data node (place where data are kept) and different kinds of process nodes (binding, relative, dynamic, functional, action nodes, conditioned nodes).

3.2. LINK

The second basic concept is link, which has the role of connecting a maximum of two nodes in the network. Links do not have link names, but can have role names. Role name (as described below) is the information belonging to the node, and this name questions the role of the connection between that node and another node. A link is represented by a line, with or without an arrow. A link cannot connect three or more nodes. Only binary links are allowed. If there is a need to connect three or more nodes then process nodes, which connect several nodes by binary links, can be introduced. The cardinality of a link is always (1, 1): (1, 1) /**19**/. This means that the first node in the connection must always exist and that it is connected with a maximum of one other node and vice versa. An example of two nodes and their links, which correspond with the sentence: "City of Zagreb", but also with the sentence "Zagreb City", is shown in Fig. 1. If we take another city, e.g. Sarajevo, it will be connected to City by a new link.

3.3. CONTEXT LINK AND CONTEXT NODE

A Context link is a special link between a context node (node on the higher level of abstraction, general, superior, class, generic, superterm) and a specialised node (described, specific, node on the lower level of abstraction, of phenomenon, of the pertinent).

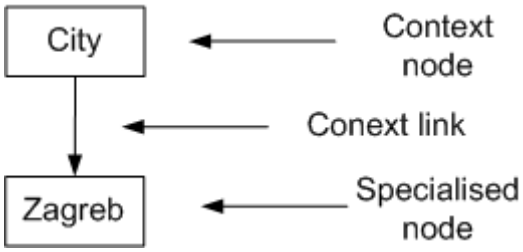

Figure 1. Relationship between a context node and a specialized node

The assumption on which this knowledge modelling concept lies upon is that knowledge can be presented and organised in nodes among which there is a certain relationship: abstract term – specific term. The complete knowledge network consists of several levels of knowledge as it is presented in Figure 3. The first level is the level of the first (phenomenal) nodes, that is, the schemes of knowledge about relationships between particular phenomena. The second and higher levels are the levels of classes and their relationships. The higher level node is a context node with respect to the node described on the lower level. The knowledge network imposes no limitations regarding the connection of any node with any other node on any level.

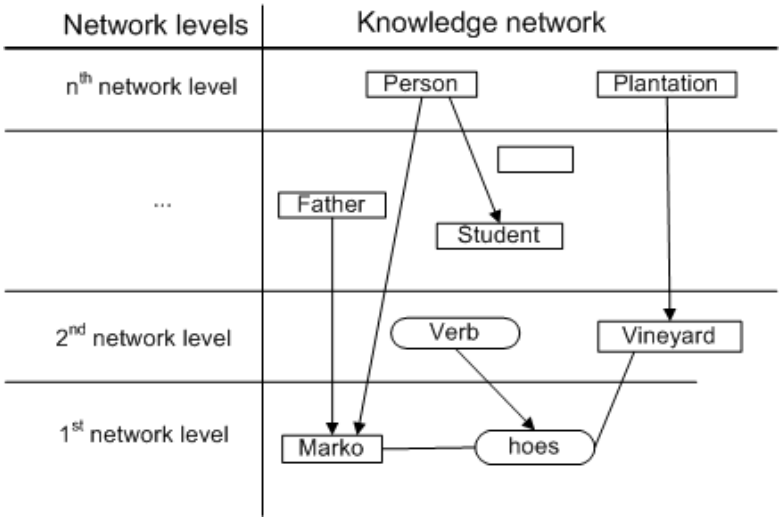

Figure 2. Knowledge network nodes at different levels

It is possible to interconnect nodes from all levels; then the higher level node is called the context node with respect to the node it is connected to. This link is called the context link. This is not a generalisation link in which supertype attributes belong to the subtype and both have several occurrences. A context link for a particular node answers to the following questions: what is the node, which sort is the node, of which type it is, which class is it, to which group it belongs, etc. If two nodes from different levels are connected, then the context link is represented by a line with an arrow. The arrow points to the lower level node. Figure 3 shows certain context links. One node can be the context node for an unlimited number of specialised nodes. One node can have an unlimited number of superior context nodes. A context link can be established between different sorts of nodes (ordinary, process nodes).

3.4. PROCESS NODE

A process node is a node whose links connect the nodes and together with them create a more complex presentation of knowledge in form of aggregated knowledge. The process node is intended for representing knowledge that cannot be represented by ordinary nodes because it stands for: relationships between nodes, activities, links between several nodes, actions, occurrences, feelings etc. The process node is graphically represented by an oval. See Fig. 2 and Fig. 3 for an example of a process node. Names of ordinary and context nodes are usually nouns. Names of process nodes can belong to different parts of speech or word groups but are usually verbs or gerunds. Process nodes are the glue that links and connects words in a sentence into superterms. A process node can represent an action happening between

the nodes. If an action is composed of several parts, then it is represented by several connected process nodes.

3.5. LINK ROLE

Every link has two ends which connect it to two nodes. The link and node touching point is called the link role. There are two roles in every link (see Fig. 3a). On a DNOK we can be interested in naming neither, one or both roles. The role name is entered on the link line beside the node whose role we are defining (see Fig. 3b).

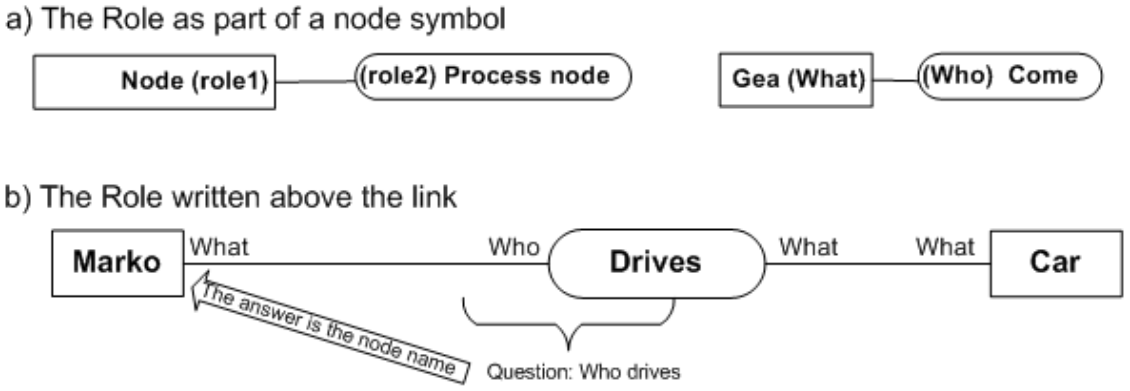

Figure 3. Role is an integral part of the node

The Link role name is usually a simple question such as: who, what, how, when, where, whereby, whose, why. It can also be a complex question consisting of several words, such as: from when, how long, which is not, where it cannot, etc. The same process node can have several identical questions. These names provide a clearer description of knowledge which has an event logic (time, place, actors, manners and similar; all these states can be questioned). Role names can also be other terms which tell us why some nodes have come together. An example for role names is given in Fig. 3 for the sentence "Marko drives a car". Beside the process node, there is the question Who on the link line towards the Marko node. The way to read the DNOK is by selecting a process node, e.g. Drives, and then start asking questions from the roles such as: Who Drives? The answer is the node at the other end of the link, therefore, the answer is: Marko. The same applies in the opposite direction. We ask Marko What? The answer is Drives. Starting from the node towards the process node the question is always simply What and the answer is the name of the process node. This question can be left out of a DNOK, but the possibility of adding it is implied. Therefore, we have the following record on the model: Marko drives a car, but also four questions: Who drives, What does he drive, What he does with the car and What does Marko do and four answers: Marko, Car, Drives and Drives.

3.6. CONNECTING PROCESS NODES

Let us analyze the complex sentence "While running quickly, Ivica saw a BMW driver when the driver stopped at traffic lights". This sentence can read as follows in Croatian: "Ivica brzo trčeći ugleda vozača kada je BMW-om stao na semaforu". This sentence has several actions connected to each other in both languages, and corresponds to several connected process nodes shown in Fig. 4. The knowledge network models are not identical.

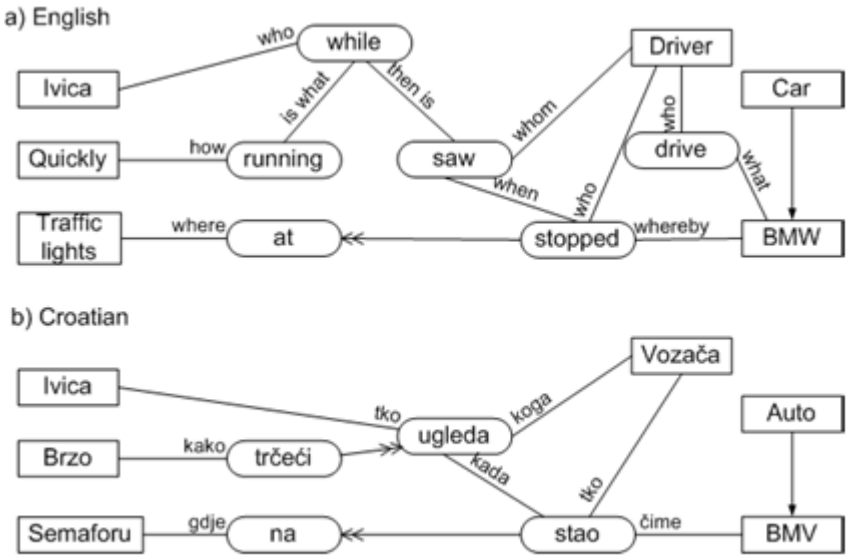

Figure 4. An example of process node linking

The order of movement across the DNOK is represented by a double arrow on the link line. The conjunction While was not used in Croatian (although this is possible since there are several good translations), so the order of actions is denoted by double arrows. Based on the analysis of other parts of speech we suggests that, apart from verbs, process nodes should be also used for the following parts of speech: conjunctions, prepositions, adverbs and whenever it is necessary to describe an action or a state between two or more terms, such as: translation into another language, linking of synonyms, linking of all meanings of homonyms, linking of pronouns and nouns etc. Nodes are used for: nouns, pronouns, numbers, symbols, adjectives, words in the dictionary regardless of the part of speech it belong to, derivatives, compounds and, in general, every term which has its own essence.

4. APPLICATION OF THE NOK METHOD IN BUSINESS ORGANIZATIONS

In making management and strategic decisions, business organizations use reports that show aggregated data that is structured within a business database that serves as a backend to a business application. Business applications do not have the ability to enhance databases with new data types and to create new laws, which exist within existing and new data, without changing applications and database schemas. The NOK method can be applied in the development of a software system that would be able to connect to existing business applications and databases and would be able to register, search and discover new business knowledge without changing existing applications. The basic idea is to connect and transform business data from an existing business application into a special form of written human language sentences presented by the NOK method. This transformation would create a knowledge base over that could be used by a computer program to discover and propose new knowledge. A special user interface would allow execution of queries and updates of the knowledge base. These queries and updates would be composed of elements in the form of written human language. The proposed software system based on the NOK method should consist of 5 subsystems (Figure 5).

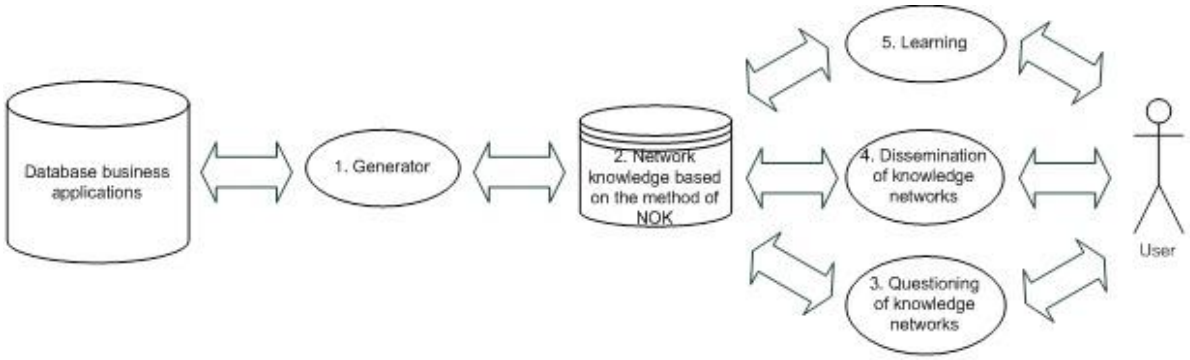

Figure 5. Subsystems of the software system based on the NOK method

1. The Generator is the basic programmatic solution that transforms the structure of a business database into a knowledge network. This solution should convert the structure and key data from a business database into a knowledge base. Key data is transferred in case that we wish to enter new data related to a specific database row that does not exist in the database. The Generator module is also an interface over which the business database and the knowledge network communicate.

2. The Knowledge Network Subsystem is a relational database that contains knowledge about the NOK method in the form of a meta-model. Therefore, this additional database contains knowledge that is not contained within the business database. Apart from data, this database also contains relationships among data as well as questions and answers, as it is designed in the NOK method.

3. The Knowledge Network Query Subsystem is a programmatic solution that should let the end user to enter a query on the knowledge network using elements of written language. The result of that query should be displayed by the same program. Therefore, this user interface can be used to create queries that are similar to human language.

4. The Knowledge Network Expansion Subsystem is a programmatic solution that can add, change and delete knowledge from the knowledge network. Updating of the knowledge network would be performed in the following two ways: Free User Updating and Guided Updating. Free user updating would permit users to update the knowledge network freely by adding elements of written human language. On the other hand, guided updating is based on deduction, i.e. on knowledge that already exists in the knowledge network and in the predefined node classes. This means that while entering new knowledge, the user can specify a node and its belonging class. Based on the selected class and its definition (i.e. the definition of possible roles in a relationship), the system would ask questions to the user. The user can, but does not have to answer those questions. Questions that do not have an answer can be answered at another time. Moreover, the system can, based on existing knowledge, conclude which type of questions can usually be answered and therefore sort the questions by importance before presenting them to the user.

5. The Knowledge Network Learning Subsystem is a programmatic solution that is capable of finding and suggesting new knowledge based on existing knowledge in the knowledge network. This algorithm should be also based on the principles of mathematical induction. For example, if multiple nodes (that belong to a contextual node) are connected to the same type of node or the same node, the system will be able to suggest new knowledge on a higher level of abstraction (more general knowledge) that will connect the contextual node with appropriate nodes that were used to bring such a conclusion. New knowledge created in this manner represents the foundation for Guided Updating of the Knowledge Network. Through this subsystem, the user will be given suggestions of new knowledge that the user can but does not have to accept.

The basic purpose of the described software system is achieved through its ability to connect to an existing company database, which results in a

"general purpose" expert system based on existing data. The established expert system will provide a more "intelligent" view on data that already exists in a company and therefore make it possible to bring better business decisions. The reason why a software system based on the NOK method would be unique lays in its ability to expand of the knowledge network automatically by transforming parts of an existing business database. Furthermore, the NOK method itself, which is the foundation of such a knowledge base, enables the creation of simple queries by using the name of a node and the name of its role (i.e. question). The formed query consists of elements of written human language. A very important part of the system is its ability to guide the user through updates of the knowledge network by asking questions that can be sorted by importance. Also, the system can suggest new knowledge from existing knowledge, and this new knowledge can be used during Guided Updating of a knowledge network. The authors of this article created the first version of the software that implements a NOK-based knowledge base. The software possesses basic functionalities such as knowledge base expansion and query creation. The results are promising and serve as a motivator for further software upgrades and for creating other Subsystems, as presented on Figure 6 (Generator and Learning Module first of all).

5. CONCLUSION AND FUTURE WORK

NOK provides an alternative way of storing knowledge, different from the way the human mind stores spoken or written words, and different from other existing formalisms. What this method introduces is a special kind of process nodes that represent a link between nodes, but at the same time a new term related to other terms. The method emphasizes contextuality of all knowledge. A special feature is the ability to observe any knowledge through basic questions and answers, which will enable easier development of future software for communication between users and knowledge bases. This idea of incorporating questions in the knowledge network enables us to learn, to ask ourselves what else can be added in order to expand knowledge and retrieve it from the knowledge base. The result shown in this paper is a conceptual model of system that can be an extension of the existing IS and relational database. We introduce a knowledge base that is in relation to the relational base of the traditional IS. The NOK method is fundamental part of knowledge base and it enables the storage of the semantic component of the whole system. Furthermore, three different modules that communicate with the knowledge base are defined (Query Module, Learning Module and Network Expansion Module). It is a complex system that enables (more intelligent) reasoning about knowledge stored in the relational database. Using semantic component stored as knowledge network in the knowledge base, easier questioning, using natural language is provided. As well, the proposed system can extend the existing database with new knowledge. The knowledge network enables easier development of expert systems, particularly modules for communication between experts and knowledge bases, both for asking questions and for expanding the knowledge base. NOK is a new method and, as such, it will continue developing and upgrading. Further lines of research can go in the direction of creating concepts that are more complex. As well, we plan to implement the described system for a concrete business application and evaluate the results. Based on the NOK method it is possible to create adequate software that would enable entry of a large number of nodes and their links and therefore the creation of more complex knowledge bases. These knowledge bases would contain entire specialized dictionaries and based on such templates enable automatic expansion of the knowledge base with specialized texts dealing with a specific field of knowledge. In this way, knowledge would be stored regardless of other knowledge, that is, the storage of knowledge would be completely flexible in terms of amount and sequence of origin.

Notes

/1/ Becerra-Fernandez, I., Gonzalez, A., & Sabherwal, R. (2004). Knowledge Management Challenges, Solutions, and Technologies, Prentice Hall.

/2/ Garner, B. J. (1987). Expert systems: from database to knowledgebase, Information and Software Technology, 29(2), 60–65.

/3/ Soutou, C. (2001). Modeling relationships in object-relational databases, Data & Knowledge Engineering, 36(1), 79-107.

/4/ Jakupovic, A., Pavlic, M., Mestrovic, A., & Jovanovic, V. (2013). Comparison of the Nodes of Knowledge method with other graphical methods

for knowledge representation, Proceedings of MIPRO 2013 - 36th International Convention, Mipro, Rijeka, 1276 – 1280.

/5/ Pavlic, M. (2013). Development of a method for knowledge modeling, Department of Informatics, University of Rijeka, Rijeka.

/6/ Pavlić, M., Meštrović, A., Jakupović, A. (2013). Graph-Based Formalisms for Knowledge Representation, Knowledge Generation, Communication and Management: KGCM 2013, the 17th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI 2013, Orlando, Florida, USA

/7/ Quillian, R. (1968). Semantic memory, Semantic Information Processing, MIT Press. 227 – 270.

/8/ Sowa, J. F. (1976). Conceptual graphs for a data base interface, IBM Journal of Research and Development, 20(4), 336 – 357.

/9/ Sowa, J. F. (1979). Semantics of conceptual graphs, Proceedings of the 17th annual meeting on Association for Computational Linguistics, Association for Computational Linguistics, Stroudsburg, PA, USA, 39 – 44.

/10/ Sowa, J. F. (2000). Knowledge representation: logical, philosophical and computational foundations, Brooks/Cole Publishing Co., Pacific Grove, CA, USA.

/11/ Minsky, M. (1974). A Framework for Representing Knowledge, Technical Report, Cambridge, MA, USA.

/12/ Russell, S. J., & Norvig, P. (2010). Artificial Intelligence: A Modern Approach, Pearson Education, New Jersey.

/13/ Gruber, T. R. (1993). A translation approach to portable ontology specifications, Knowledge Acquisition, 5(2), 199 – 220.

/14/ S. C. Shapiro, J. W. Rapaport, Knowledge representation for natural language processing, in: 1987 Natural Language Planning Workshop, Northeast Artificial Intelligence Consortium,Minnowbrook, NY, pp. 56 – 77.

/15/ D. Jurafsky, J. H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2000.

/16/ N. Chomsky, Studies on Semantics in Generative Grammar, Janua Linguarum. Series Minor, Mouton

/17/ Helbig, H. (2006). Knowledge Representation and the Semantics of Natural Language, Springer, Berlin.

/18/ Sowa, J. F. (1976). Conceptual graphs for a data base interface, IBM Journal of Research and Development, 20(4), 336 – 357.

/19/ Pavlic, M., (2011). *Oblikovanje baza podataka*, Department of Informatics University of Rijeka