Teachers for the Knowledge Society

Transforming the Knowledge Incorporated in e-Learning Software into an Automatable Explicit Knowledge for the Teacher and the Learner

Vasile Avram\textsuperscript{a}, Diana Avram\textsuperscript{b}\textsuperscript{*}

\textsuperscript{a}Academy of Economic Studies, 6 Romana Plaza, District 1, Bucharest 010374, ROMANIA
\textsuperscript{b}SIGMA Publishing Co., 38 General Berthelot Str., District 1, Bucharest 010169, ROMANIA

Abstract

The aim of this paper is to introduce a solution for the disclosure and recovery of the educational knowledge incorporated, and hidden to the user, in the support software for e-Learning. The solution, called SIK (software inherited knowledge), is to add an architectural loose connected component to new or existing software. This component is represented by knowledge repositories that are used in order to systematically acquire, structure, store and maintain knowledge, formalized as rules sets. It investigates also the main benefits that different categories of users – teachers, students, administrators – may have from the recovery of the incorporated knowledge.

Keywords: Teaching; Learning; e-Learning; Knowledge; Knowledge management; Business rules; SIK

1. Introduction

In what follows, we designate by the term teaching and learning (TL) organization any organization, school, university, etc. that operates in education or have some educational activities. We designate also by the term teaching and learning support software the educational components of e-Learning applications (we try to exclude in that way a lot of technical components required for access, backup/restore, maintenance etc).

The today’s world is characterized by an increasing complexity of relationships between business actors and an increasing complexity of the global competition. This induces an increasing complexity and dependence of TL organizations on Information Technology & Communications (IT&C). In turn, this complexity makes more and more important for TL organizations to use all educational, business, and IT&C best practices, as high valuable knowledge. Brand software, sold and delivered by leader IT&C companies, for teaching and learning support, incorporates good practices (that are knowledge) inspired from those of leader organizations in various TL domains. Thus the educational brand software incorporates the knowledge allowing the organization, application, and the operation of those good practices. This knowledge is expressed and documented in a formal way by what is named in knowledge management by the term business rule. The business rules, when expressed in a natural language, as for example the ‘working procedures’ you use at your job, allows uncertainty and misunderstandings generated by the nature of the language itself. To avoid that, the effort and trend in business is to represent these rules in a formal

\textsuperscript{*} Tel.: +4-072-338-2371; fax: +4-021-323-7319; Tel.: +4-072-338-2372; fax: +4-021-323-7319

\textit{E-mail address}: vasileavram@ie.ase.ro; diana@avrams.ro

Open access under CC BY-NC-ND license.

Selection and peer-review under responsibility of Masterprof team.

1877-0428 © 2011 Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
doi:10.1016/j.sbspro.2011.01.057
way, as rules sets or formalized business rules, that can be interpreted and executed (acted) by software applications called business rules engines running on computers (it means by machines). The artificial intelligence applications can learn from these formalized business rules, can ‘understand’ and recognize patterns based on these business rules, and even can use their automated reasoning component to infer new rules. When included in a program, expressed as commands of a programming language or as an executable code, this knowledge (or business rule, in that case) becomes hidden to normal users, as teachers and students/learners are, and even to computers that executes these programs. A complex e-Learning system includes in the software applications components, taken as a whole, different categories of knowledge. These categories refer to the knowledge specific to IT&C activities required by IT&C teams to design and realize the support software, to the knowledge about the human-computer interface required by the user to operate the application, and to the knowledge specific to the modelled domain. We are interested here mainly in the knowledge of the modelled domain, in our case the educational activities, and secondarily in the knowledge required to operate and interact with the software application. A TL organization efficiently adopting and using such software, in its current activities, will start, maybe, these activities and their associated competitiveness, at least from that level of good practices and associated knowledge. We consider, for the support software of the educational domain, three broad categories of users: teachers, students, and administrators (software administrators, librarians, accountants etc). To be competitive, an educational organization using e-Learning support software must provide better knowledge support to every category of its users.

The main obstacle in providing better knowledge support is that, at present, the meaning of business rules content (the educational rules) incorporated in these software products is not accessible to humans and is hidden to both users and computers (it is represented as a string of bits, a large finite number of 0 and 1’s). In other words, a lot of valuable knowledge remains hidden to all the actors that manipulate and interact with teaching and learning support software, in the same way the implicit knowledge belonging to the minds of teachers remains hidden to others if that is not made explicit knowledge. Generally the software applications are always accompanied and delivered with many user guide manuals specialized to user’s roles, such as ‘Teacher’s guide, Learner’s guide’ etc. These manuals allow those actors to use and interact with these applications. The manuals can contain details about the modelled business rules and their implementation or computational algorithms. In this context one can argue that the knowledge is explicit, in that way, to the involved actors. But here there is a small problem. Remember how embarrassing is that you work on your computer and the operating system (a Microsoft Windows version for more than 87% of the Internet users) of this one interrupt you and announce that it is necessary to stop your work and restart the computer to apply critical updates. In all the cases the updates to software do not makes changes in the user guide manuals they provided you when the software was sold. In that way a lot of knowledge corresponding to those updates remains completely hidden to the user or can be hardly accessible. The user is faced with the same problem when using the teaching and learning support software and applies updates to this one. Another problem faced by software applications is that there is a gap between how information is presented in the textbook and how the software delivers it effectively. When people use software only by providing the required inputs and without any knowledge about how information is processed in order to obtain the output, they use that software as a black-box. A black-box induces in time the so called ‘black-box syndrome’: each mistake or bad past experience is amplified, and fear appears every time one operates the software.

Considering these assumptions, the purpose of this study was to find a feasible solution to transform the knowledge incorporated into a teaching and learning support software from hidden and implicit-like knowledge into explicit knowledge for all human users and software actors wanting to know and use it. The study was also directed to make explicit knowledge organized and described in a formal manner to favour automation of its future processing. In the following paragraphs we consider a generalized architecture for e-Learning software systems and we put that together with SIK architecture component that allows software having a vector for communicating the incorporated knowledge and the inheritance of knowledge. In that way we obtain a solution to make explicit the implicit educational knowledge incorporated in e-Learning software, making it accessible to both humans and machines.

2. e-Learning Architecture

An organization responsible for training students and/or employees has as main objective to transfer knowledge to as many learners as possible while using its resources most efficiently. E-learning provides the means to deliver
education and training (and other kinds of) content, and is even able to change the knowledge or skill level of the learner, as training is progressing. By changing the way content is organized, delivered, received, and possibly even assimilated, education and training methodologies might also be changed.

E-learning involves communication, collaboration, training, and assessment efforts within any organization that deals with professional development of individuals (Kelly & Nanjiani, 2004). It is an integrated system of e-communications, e-training, and e-assessment components whose deserving tools work in a networked environment. E-learning is a process that meets learner needs with relevant content and with technology-enabled tools that are convenient to the learner’s context. It changes how and what ‘we say’, to whom, and when.

Effective e-Learning requires several different skills, such as instructional design, media design, software engineering, and economics (Horton, 2006). Figure 1 outlines the e-Learning systems global architecture, built by corroborating many sources (Kelly & Nanjiani, 2004; Horton, 2006; McVay Lynch & Roecker, 2007) and expressed in terms of components category together with their required functionalities and services, to be offered to the user, from the perspective of the required level.

Each functionality/service is deserved by software components that incorporate the domain knowledge, expressed as software commands, of the domain they serve. We are not interested here about the IT knowledge required by the IT teams when designing and realizing the software itself. For the actors involved in the educational process, such as instructors and learners, the instructional knowledge incorporated in different tools expressed as assessment rules, personalization rules, evaluation rules, organizing content rules etc. or in other words the “business rules” of the educational process, is extremely valuable.

For this architecture, Web 2.0 technologies can improve collaboration and communication within educational organizations by its capacities to support traditional training, to modify/develop the training content, to support both asynchronous and synchronous training, and to codify and distribute training content (Andriole, 2010).

3. SIK Architecture for e-Learning

SIK concept was introduced first in Avram (2009) and is built on the basis of a loose connected software architectural component, materialized in the form of one or multiple knowledge repositories (Avram, 2010). It is used for systematically acquire, structure, store and maintain knowledge, formalized as business rules for all domain business rules that are incorporated in the software product itself. The knowledge refers mainly to the domain knowledge of the business domain modelled by the application software together with the knowledge about the operation of the current application. The formalization of the business rules is realized through RDF/XML (Resource Description Framework/Extended Mark-up Language) serialized triples. A triple is defined by <subject, predicate, object> as specified by RDF standard which was defined with the scope to represent and manipulate knowledge in web space. RDF usage is based on the specifications of the standard defined by the RDF Working Group of World Wide Web Consortium (W3C), the non-governmental organization, that defines and owners the Internet standards. The standard includes also the guidelines for Web architecture as defined by the creator of Internet Sir Tim Berners-Lee (Berners-Lee et al., 1999). RDF is based on XML but differs on this one at least by that an RDF document includes in a single unit both metadata and data while an XML don’t. The SIK does not restrict the stored knowledge at the domain business rules. Being an RDF document, which in turn is based on XML, the SIK component has all the characteristics of both RDF and XML. SIK allows defining and keeping in the

![Figure 1: E-learning system – components, functionalities, and services](image)
same document (file) a variety of rules grouped in separate spaces by intermediate of namespaces. The spaces also allow storing business rules regarding the application programming interface (used to connect applications over cyberspace) or the rules applied to operate the application in current use. In that way each e-Learning component, as investigated in Figure 1, can have his own spaces in the same repository and/or can have his own repository. SIK can be physically defined across several repositories each one having his own specialization and that situation is described by the SIK configuration component. For every application software having a compliant to SIK architecture, the companion repository act as vector for transport and communication of the knowledge inherited from software. Figure 2 shows a simple example taken from the open source e-Learning product Moodle version 1.9.10. In the left side is shown an excerpt from both source code and executable code as required interpreting and acting by the microprocessor. Moodle offers to the teacher to choose between nine aggregation types to determine the ‘Grade’ (a final mark or a mean) that can be listed as number of points, as a percent, or as a letter (such as ‘A’, ‘A-’, ‘B+’, ‘B’, etc., for example). The options are given as a name (associated to an aggregation type such as Weighted mean, Simple Weighted Mean, etc.). The names have short explanations associated that are displayed on setup screens and also provided in the ‘Teacher’s guide’ but, unfortunately, this one does not include the computation formula, which is knowledge. The teacher, even having mathematical skills, must guess or imagine how effectively this works. The business rules, representing educational domain knowledge, as the procedure (rule) for marks aggregation and computation formulas in our above example, are hidden in traditional software application. Figure 2 highlights the difference between traditional software applications and the compliant SIK architecture. The part labelled A is an excerpt from the Moodle PHP source code, and the part labelled B is an excerpt from an executable as required to interpret and act commands by the computer microprocessor. Parts C and D are excerpts from the SIK content, respectively, their interpretation by the parser.

The same rules can be revealed to the user, teacher in our case, by adopting a compliant to SIK architecture for the software application as exemplified in Figure 2 by the parts labelled C, for the SIK repository content (excerpt), and labelled D, for what is revealed to the teacher by a very simple RDF parser, the component used to connect SIK to a Semantic Web Application. A Semantic Web Application, by his components, is able to query in a more human
meaningful way the SIK repositories and knowledge databases and to infer new knowledge from the existing one (artificial intelligence). The existing software can be easily adapted to the new architecture by adding its knowledge repository in a similar way with this example for Moodle. The output of the parser is not so pleasant but this is happening because the RDF parser is a tool used more to check the validity of the defined triples, and it certifies that the triples are machine and user meaningful. With a semantic search engine or a query engine this aspect changes in favour of the end user, regarding the searching method and formatted result.

4. Conclusions

The possibility to access the assessment rules, evaluation rules, personalization rules etc., revealed by SIK repository, allows better collaboration and communication to both instructors and learners. This in turn will increase the ability to coordinate discussions, reach more people faster, synchronize projects and tasks, and audit the communications streams (Andriole, 2010). SIK act as a source of enriching, with the contained knowledge, every knowledge base deserving/assisting knowledge workers (teachers, instructors) and favour the integrative architecture for knowledge management system and associated software. In that way, it answers to the requirement for ability of organizations to increase their stock of automatable knowledge which is a premise for sustainability of the organization and for success when facing competitions.

The IT&C companies realize their e-Learning products by collecting valuable practices from high skilled teachers and from the successful educational experiments. Every educational organization using e-Learning components having a SIK architecture will help its teachers/instructors to better interact with those applications and even to learn about all best educational (training, assessment, evaluation, performance management etc) practices incorporated into the product and thus to develop their competence. The SIK can be visible to any authorized person, preventing the 'black-box syndrome' common to all monolithic applications. The user – teacher/instructor, learner, and administrator – can learn and understand from the system behaviour, making easier his/her interaction with the application. Teacher can learn from other teachers’ experiences or successful experiments incorporated in the software, at any moment, the educational knowledge repository SIK representing a strong support and a challenge for his/her career. This characteristic of e-Learning applications is emphasised by the ‘disclosures’ of the domain specific knowledge as business rules in SIK. We can argue that SIK allows learners and teachers to have access to a source of the latest research and findings in the domain. Being parts of a repository that is updatable, in an easier manner than the software is, any upgrade or patch applied to software can be accompanied by the corresponding upgrade/patch of the corresponding business rules.

Because the knowledge stored in SIK is processable, the teacher can use inference engines to infer new knowledge from it, that is, to find new rules. These new rules can be, for example, new assessment procedures and criteria, new metrics for measurements and evaluation, new personalization ways for specific contexts etc. These new rules allow the teacher to develop his/her professional competence and adapt towards a continuous professional challenge. By offering the possibility to analyse and compare the stored business rules, SIK can be used as a real basis for comparisons and evaluations of e-Learning products.

References