Towards Information Management System for Licensing in Higher Education: An Ontology-Based Approach

Olga Cherednichenko[a], Dmytro Kuklenko[a], Sergiy Zlatkin[b]

[a)] Computed-Aided Management Systems Department,
National Technical University "Kharkiv Polytechnic Institute",
Kharkiv, Ukraine
marxx@kpi.kharkov.ua, dmytro.kuklenko@gmail.com

[b)] Department of Information Systems,
Massey University, New Zealand
s.zlatkin@massey.ac.nz

Abstract. Higher education licensing, i.e. the process of granting permissions to provide certain educational services, is an important process of public administration. Governmental bodies handle licensing requests submitted by education providers regularly. Such requests are supplemented with large amounts of information that demonstrates the actual situation in an education provider. At present this process is paper based and involves a lot of manual labor. This situation is similar for other types of education, e.g. primary and secondary, as well as for many countries. We, therefore, aim to computerize this process by creating a universal licensing information system (LIS). We base our approach on a domain ontology that defines the main concepts in the licensing process. In this paper we show how licensing process works now and propose the ways to make it more efficient. Then, we envisage the usage models of LIS, identify its main business actors and use cases, and provide a high-level architecture of the system. Finally, we present a fragment of the domain ontology in higher education domain.

1 Introduction

Education is one of the most important aspects in the life of any society. Educational level of citizens is the crucial success factor of country's development. It defines the possibility of using ICT, innovations success, and high living standards. The main task of the national education system is to train highly qualified specialists with up-to-date knowledge and skills in different branches and professions.

The national education system requires and produces enormous amount of information. In general, such information is related to the interaction of higher education service consumers (students and entrants), universities, and governmental bodies. In [Ch06] we presented our approach to design the common architecture of information system that solves the task of higher education coordination based on e-government concepts.

That information system would allow to increase the efficiency of information management in the national higher education system as well as to support the solution of various specific tasks. These tasks are associated with coordinating the higher education and include the following [GC03]:

- Licensing universities, departments, programs, majors, and courses.
- Forming strategies and development plans for the higher education system.
- Distributing grants among universities.

According to Wikipedia [Wi07] ”to license is to give permission”. A party may grant license to another party as an element of an agreement between those parties. Using licensing in education area the government manages the education activities of universities and other education providers. Having carried out a respective audit, the government guaranties that citizens can obtain the education of recognized quality standards. Most of the countries provide the licensing in higher education and the respective processes have many things in common (see e.g. [Nz06]).

The licensing process is carried out on a regular basis (the license validity is limited), it requires processing of large data volumes, and supposes that the information related to education providers is available for public access. This information in particular may include curriculums, course programs, lectures, university infrastructures, etc. All these data characterize the actual state of education provider. To estimate this state we need to have a set of indices. Normally these indices are defined when the initial data are received from an education provider (EP).

Collecting and processing initial data is not only EP’s concern but also governmental one. Large number of EPs working with similar data as well as the needs of data analysis and common presentation bring us to another important problem – to define the common terminology used in licensing process. We believe using ontology may help solving this problem.

In this paper we provide architecture LIS – licensing information system - that supports the licensing process. The information system is aimed to be used by a respective Qualification Authority. However, it can be used by EPs as self-assessment tool. The system is also capable of providing information to public and serves in that way as an e-Government system.

The rest of the paper is structured as follows. In the next section we describe licensing process as it is now and discuss how the process can be improved. In section 3 we discuss LIS building principles. We derive main business actors and use cases from the LIS usage models as we envisage those. And we provide LIS high level architecture showing how different components realize use cases. In section 4 we provide a general ontology for licensing process. We conclude the paper and provide our references in section 5 and 6 respectively.

2 Licensing process

Below we consider the licensing process as it exists now. Although we refer to the licensing in Ukraine, as we mentioned, the process is also similar to other countries with only minor differences. For example, the role of Qualification Authority is played by the Ministry of Education in Ukraine, while in New Zealand a respective governmental body, namely New Zealand Qualification Authority, delegated necessary permissions to the NZVCC, New Zealand Vice-Chancellors Council [Nz06]. We propose to keep the existing process but to improve it by introducing LIS to make the information processing more effective and less resource consuming. The proposed licensing process is shown in Figure 1.

The process starts when EP submits a licensing application into LIS. After this the off-site audit process is initiated. During the off-site audit LIS verifies the formal aspects of the submitted application according to the requirements specified by existing legislation. If some of formal conditions are violated the licensing application is refused automatically.

Then, the Qualification Authority assigns an auditor to process the application. This assignment can be also done automatically by LIS. In this case an auditor is selected from the list of eligible auditors, or the Qualification Authority can assign the auditor directly. The goal of an auditor is to verify both the formal and informal aspects of licensing application. This procedure includes also the off-side audit stage again, in order to verify the requirements that have not been formalized through LIS.

Finally, the on-site audit should be performed. The auditor verifies the application details when physically visiting EP's campus(es). This stage results in the audit report. Based on that, the Qualification Authority either accepts or denies the license application. The license application acceptance means issuing the license with appropriate eligibility period specified.

Also, EP can submit a self-assessment application. In this situation no formal acceptance/refusal decision should be made. LIS verifies formal aspects of the application only and generates the recommendations on how to improve the application in order to process with official audit successfully.

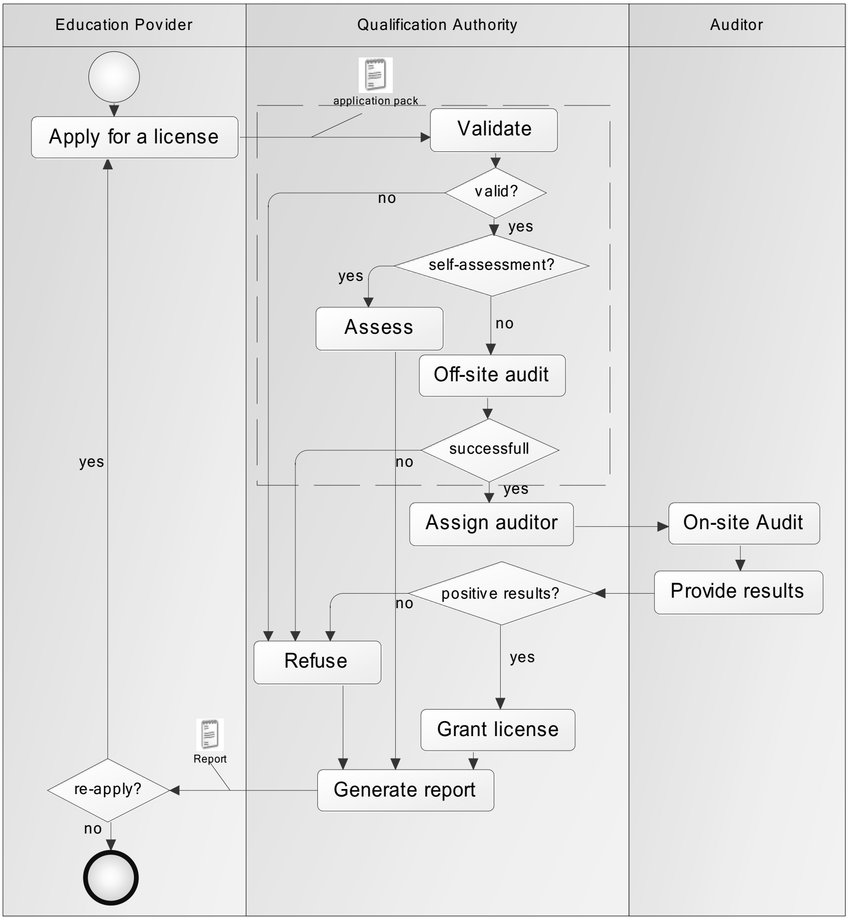

Figure 1. Licensing process in higher education

3 LIS Usage Models and Architecture

3.1 LIS Usage Models

According to the licensing process model introduced in the previous section, we anticipate the following usage models of LIS.

1. *Initialization*
We assume that EP possesses an information system (EPIS) of any kind that can prepare required data for LIS. To enable an efficient communication between LIS and EPIS it is required both systems being capable '*to speak the same language*'. To ensure that EPIS and LIS should share the common vocabulary, i.e. ontology (because there may exist a

number of ontologies for different domains, e.g. higher education, secondary education, etc.; or for different countries, and these ontologies may be amended regularly - every time a new program or major is introduced). LIS should provide a respective service to EPIS allowing ontology exchange and synchronization. The communication process in this case is the following. EPIS requests from LIS an up-to-date version of the ontology. LIS in response provides the ontology in a transferable and parseable form, which is XML [GP04]. Finally, EPIS updates its vocabulary.

2. *Self-assessment*
After vocabularies of LIS and EPIS are synchronized, EP may request a self-assessment service of LIS. EPIS generates an application pack and submits to LIS. First, LIS parses an application documents and validates if all required data are provided, and if an up-to-date ontology is used. In case of a failure, a respective refusal is returned. If all details are correct, LIS scans the case store looking for cases with initial parameters similar to the case submitted. Provided similar cases found, LIS then analyzes what changes had been made to these cases to make them successful, and generates recommendations. We use ripple down rules for knowledge acquisition [Pr96], and case-based reasoning for solving new problems based on the solutions of similar past problems [AP94, Wa97]. We, though, leave the details of these methods out of the scope of this paper. If requested, LIS may assign an external auditor who will then analyze the application and provide his/her assessment together with the one generated by LIS. After returning the response to EPIS LIS updates the case store with the details from that application.

3. *Application*
In case of an official application the process is quite similar. If documents are valid LIS fulfils off-site audit. It compares application data with the normatives and computes respective indices. If off-site audit fails LIS refuses the application. Otherwise, LIS selects an auditor, independent from EP, and assigns him/her the task of on-site audit. Because auditors normally are not full-time employees of the Qualification Authority, LIS should be capable of sending auditor notification e-mails or SMS. When on-site audit is completed and respective auditor's recommendations are provided, LIS generates the respective report and sends it to EP. Then EP is either granted a license or refused. In both cases LIS updates the case store.

4. *On-site audit*
We distinguish two different methods how an auditor can log into LIS and provide necessary data. It should be possible for auditor to log into LIS using either browser, when sitting at his office, or using PDA or other mobile device, when physically presenting at the EP site. LIS then builds GUI according to auditor's device and personal preferences prior stored in the LIS. Auditor enters and updates respective data and suggests his recommendations and log offs.

5. *Administration*
Administration usage model is standard for many systems. Administrator manages database, access permissions, etc. Apart from that administrator is responsible for importing and exporting ontologies from and to external systems such as Protégé, for example.

Based on these models we can identify the main business actors and main use cases of LIS. We provide the list of actors and use cases below in tables 1 and 2 respectively.

Term	Description
EPIS	Information system of the University, Polytechnic, etc. that communicates with LIS on B2B principles according to SOA (service-oriented architecture)
Auditor	An employee of QA (or sometimes external) who access LIS remotely using mobile device or web browser. Access granted individually by administrator.
Expert	An employee of QA who has power rights for LIS and who is responsible for keeping ontologies consistent and up-to-date.
Administrator	An employee of QA who manages all required data.
Guest	An anonymous user interesting in the licensing data. In particular, current and future students, lecturers, journalists, etc.

Table 1. LIS Business Actors

Use Case Name	Participating Actors
Apply for self-assessment	EPIS Auditor Administrator (optional)
Apply for licensing	EPIS Auditor Administrator (optional)
Carry out off-site audit	No actor involved
Load on-site audit results	Auditor
Generate report	No actor involved
Import/export ontology	Administrator, Expert
Modify ontology	Expert
Administration	Administrator
Provide public access to licensing information	Guest

Table 2. LIS Use Cases

3.2 LIS Architecture

LIS is a web-based information system and is supposed to aid in executing the functionalities identified in the previous section. The figure 2 shows its high-level architecture.

The key components in that diagram are briefly explained below.

- *Screen engine*, for generating respective GUI according to user device and personal settings and preferences.
- *Web Service Interface*, for communicating with LIS partners, such as EPIS.
- *Workflow engine*, for invoking respective components according to prescribed routines. For example, in case of self-assessment the routine is to validate application data, then run the respective self-assessment analysis, and finally generate a report and send it back to the EPIS that initiated the process.
- *Validator,* for checking the application data against a respective ontology.
- *Analyzer*, for carrying out off-site audit.
- *Advisor*, for providing respective recommendations in case of self-assessment using case-based reasoning and ripple-down rules.
- *Case Manager,* for manipulating case store as well as managing current cases in a session context.
- *Case store*, for storing cases both successful and failed as well as submitted for self-assessment.
- *Importer/Exporter,* for importing ontologies from and exporting to Protégé.
- *Administration*, for managing permissions, updating workflow routines, manipulating data, and other administrative tasks.

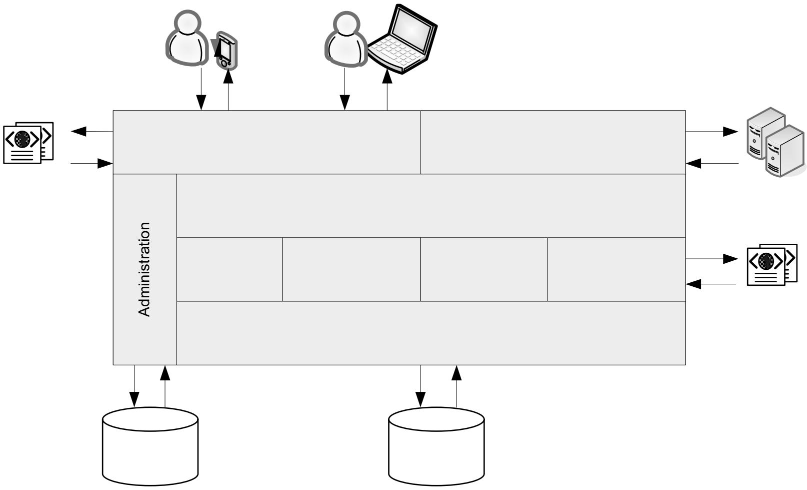

Figure 2 – LIS Architecture

LIS is an e-Government system of G2C type (Government-to-Zitizen). We use the term G2C instead of B2C (Business-to-Consumer) as the owner of LIS is a governmental body. It provides its users access to its services via web browser. At the same time LIS is a G2B (Government-to-Business) system. It provides a web service to external systems of LIS's owner partners, i.e. EPs.

We find our architecture complying with a model-view-controller design pattern (see [Fo03, Ga95] for details). Screen engine plays the role of the view; workflow engine is a controller, while all the rest components are the model.

4 Licensing ontology: Proposal

A lot of definitions of the term 'ontology' exist; some of them are given in [Gu98, GG95]. However, most of researchers come to the point that ontology can be helpful when there is a need to define a common vocabulary of terms for some application domain. From this point of view ontologies are classified as top-level ontologies, domain ontologies, task ontologies, and application ontologies.

There exist ontologies in the domain of higher education, for example the one in [Bu04]. But we believe that these ontologies are mostly oriented to the problems of teaching process formalization, sharing courses, etc. By now the problem of licensing has not been the point of consideration for ontology developers.

For the problem of higher education licensing it makes sense to specify domain ontology. This ontology can formalize the notions used in licensing process.

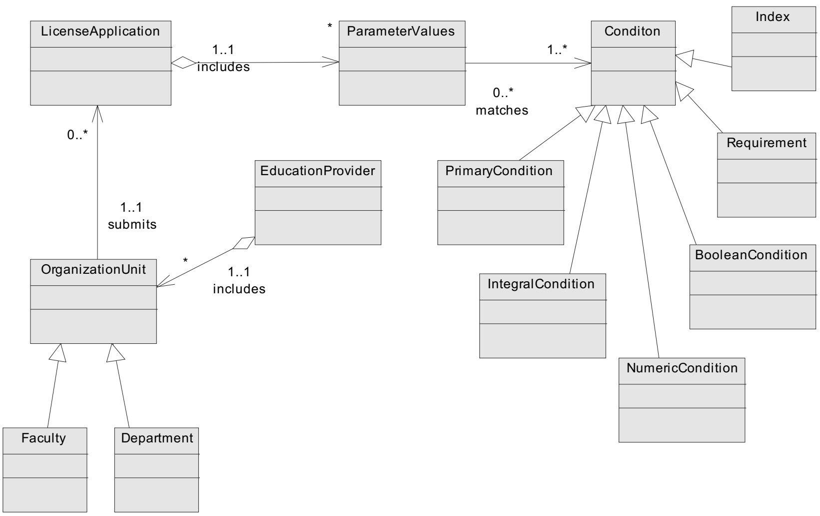

Figure 3 – Fragment of licensing domain ontology

Here we present a fragment of licensing domain ontology that is related to the licensing audit. During the licensing audit EP is being estimated according to the license **conditions.** By the nature of conditions they can be **formal** and **informal**. Formal conditions can be verified through LIS. To verify informal conditions the Qualification Authority requires explicit auditor participation. Only the formal conditions are considered in the ontology fragment that we present in the paper. These conditions can be further classified in following manner (see Figure 3 for graphical view):

- by the scope of the condition we can distinguish **indices** and **requirements**. Indices are parameters EP is being estimated with, as well as minimally possible values (thresholds) for the indices that allow EP to be audited positively. The different threshold values can be specified for various application modes (e.g. license for bachelor or master studying). The requirements relate to all application types and describe EP in general (e.g. presence of first-aid post or high-speed internet connection);

- by the type of values conditions operate. The conditions can have either **numerical** or **Boolean** type. The values of numerical indices can be supplied as absolute values or in percentage terms;
- by the actor that defines the condition values. In a license application EP supplies the values of indices and requirements that can be easily received at the scene – **primary** parameters (e.g. the list of staff, their workload, parameters of program and major curriculum, etc.). Using this data the Qualification Authority calculates the values of **integral** parameters and then compares them with thresholds. These calculations can be made off-site. Based on their results an application could be refused if some of the requirements are violated.

On the left hand side of the diagram there are classes of **License Application** and **Organization Units** of **Education Provider**. Here we pay much attention to a **Department** class because departments are usually the main stakeholders in the process of licensing application submissions. The Department class includes indirectly the information on curriculums, teaching staff, equipment, etc.

5 Conclusions and Future Work

We have presented an approach to effective and useful licensing process in higher education. This approach supposes that the ontology-driven information system is used to process licensing applications submitted by education providers. We have provided architecture of this information system and specified its main use-cases. The fragment of domain ontology built for licensing area has been also shown. The proposed Licensing Information System should serve also as an e-Government tool, i.e. it can provide the necessary information to various users interested in learning about education services.

References

[AP94] Aamodt, A.; Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. Artificial Intelligence Communications 7, no. 1 (1994): pp. 39-52, 1994.

[Bu04] Bulat, A.; Gray, E.; Ermolayev, V.; et.al.: The Infrastructure for Electronic Data Interchange. Reference Architecture Specification. URL: http://www.unit-net.org.ua/Default.aspx?page=9&lng=2 &doc=2. 2004. Last accessed 22/02/2007.

[Ch06] Cherednichenko, O.; Kuklenko, D.; Zlatkin, S.: Coordinating Higher Education as an e-Government Initiative. In Dimitris Karagiannis and Heinrich C. Mayr (eds.) Information Systems Technology and its Applications. Proceedings of ISTA'2006, Klagenfurt, Austria, 2006.

[Fo03] Fowler, M.: Patterns of Enterprise Application Architecture, Addison-Wesley, 2003.

[Ga95] Gamma, E.; Helm, R.; Johnson, R.; Vlissides. J.: Design Patterns: Elements of Reusable Object-Oriented Software, 1995.

[GC03] Godlevsky, M., Cherednichenko, O.: Model of funds allocation among higher education institutions based on higher education system development plan, Printed scientific works

of National Technical University “KhPI”, Issue 1, vol. 7, pp. 15-20, Kharkiv, Ukraine, in Russian, 2003.
[GG95] Guarino, N.; Giaretta, P.: Ontologies and Knowledge Bases: Towards a Terminological Clarification. In N. Mars (ed.) Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing. IOS Press, Amsterdam: pp. 25-32, 1995.
[GP04] Goldfarb, C.F.; Prescod, P.: XML Handbook, 5th Edition, Prentice Hall, 2004.
[Gu98] Guarino, N.: Formal Ontology in Information Systems. In N. Guarino (ed.) Formal Ontology in Information Systems. Proceedings of FOIS'98, Trento, Italy. IOS Press, Amsterdam, pp. 3-15, 1998.
[Mi07] Ministry of Education and Science of Ukraine, URL: http://www.mon.gov.ua (last accessed 15.02.2007), in Ukrainian, last updated 2007.
[Nz06] New Zealand Vice-Chancellors’ Committee: Committee on University Academic Programmes. Functions and Procedures. URL: http://www.nzvcc.ac.nz/files/cuap/FANDP05.pdf, 2006. Last accessed 22/02/2007.
[Pr96] Preston, P.; Compton, P.; Edwards, G.; Kang, B.: An Implementation of Multiple Classification Ripple Down Rules. URL: http://ksi.cpsc.ucalgary.ca/KAW/KAW96/preston/MCRDR_Tut.html, 1996. Last accessed 22/02/2007.
[Wa97] Watson, I.: Applying Case-Based Reasoning: Techniques for Enterprise Systems. Morgan Kaufmann, 1997.
[Wi07] Wikipedia: License. URL: http://en.wikipedia.org/wiki/License, last accessed 22/02/2007.