Database Strategy in Development of Digital Ecosystem for Quality Control in Education

Ivan Evdokimov[1,*], *Halina* Danilava[2], *Tatiana* Yamskikh[1], and *Roman* Tsarev[1]

[1]Siberian Federal University, 79 Svobodny Prospect, Krasnoyarsk, Russia
[2]Belarusian State University of Informatics and Radioelectronics, 6 P. Brovki Street, Minsk, Belarus

Abstract. The purpose of this research in educational environment is to improve the quality of the educational process with the help of new information technologies. The paper examines practical issues of digital ecosystems implementation in education on the basis of relational databases, as, from the authors' point of view, in the organization data are less subject to change in time than business processes. Therefore the strategy of information systems development where the database functions as a cornerstone provides the result, that is more stable than, for example, a functional approach in software development. Within the frames of this research information technologies of quality control in education are being created. They are intended for the exact and full representation of teaching materials as a system used as a source of information for the assessment of learning and teaching materials and creation of electronic resources on this platform. The structure of the created database, operation practices and topical issues of its introduction in more large-scale digital ecosystems are considered. As a result the database ready for use as a management solution for higher education institutions is created. This is important as information technologies provide additional opportunities for the development in the sphere of higher education, quality improvement of the educational process. The authors conclude that introduction of innovative information technologies in management system of higher education institution is relevant and perspective.

1 Introduction

Development of an effective education system is an invariable priority of State policy.

At the present moment, the problem of development and deployment of new technologies based on the methods which increase the efficiency and quality of educational process [1-3] is very important as information technologies penetrated into all fields of human activity. This will create opportunities for supplying all branches of the national economy with the qualified employees.

The auxiliary software is widely used for the educational purposes. Moreover, there have appeared learning management systems, such as Moodle, SharePointLMS [4, 5]. However some spheres of management in education still wait for professional developers, in particular the provision of learning and teaching materials [6, 7].

Besides, the importance of higher education in modern society requires using valid and reliable instruments to measure the quality of education [8, 9].

Reforms in the field of higher education, implementation of innovative technologies to educational process will cause renewal of the traditional strategy of management, its transformation into a new, more flexible, capable to adapt to changing external and internal conditions.

Now the information system of the enterprise is developed as a project. Project is a temporary endeavor undertaken to change a particular system. It has specific goals and their achievement means the end of the project which should be executed over a fixed period and within certain cost, resources, risks and organizational structure.

The process of creating an effective information system includes several stages:
- data collection and analysis;
- database design and implementation (creation of domain model, logical and physical design);
- generating user interface to work with the existing database and visual representation of information.

The development of such a system is considered in this paper. One of the main stages in the design process was study of problem domain and creation of hierarchical domain model. The domain model has declarative character; in fact, it is the knowledge base about the object of research that is used to formalize the representation of an object (task) and analyze requirements for the modeling system (solution). The declarative character of domain model does not impose restrictions on the subsequent choice of the model type for the creation of separate components of the system

---

* Corresponding author: evd-ivan@yandex.ru

under study and interactions between them. The purpose of this model is to create exact and full representation of teaching materials as a system used as a source of information for the assessment of learning and teaching materials and further design of the database.

The work of the faculty responsible for methodical support of educational process has been defined as a problem domain.

"The Curriculum", "The Federal State Educational Standard of Higher Education" and "The provision of learning and teaching materials" were used as source documents to generate the subject domain model.

2 Logical database model

For the development of desired software product it is necessary to consider such concepts as "model" and "domain model".

A model is a representation or copy of an object (process, phenomenon) reproducing the features and properties of a genuine object, that are the most essential in relation to the outcomes the user is interested in.

A domain model represents the key concepts of the problem domain and identifies the relationships among all of the entities within the scope of the domain.

The alternative description of the considered problem domain can be represented as follows:

- the faculty comprises the departments;
- the department is composed of the teaching staff;
- the department offers Bachelor's degree programs designed to prepare students for a variety of careers in the field;
- training is provided in accordance with the curriculum;
- the curriculum is based on the Federal State Educational Standard of Higher Education;
- the Course Syllabus is developed in accordance with the curriculum;
- the Course Syllabus has to be registered by its developer – the teacher who provides the course;
- each course should be provided with the Instructional Practice Guidelines for the completion of this or that type of tasks.

On the basis of data obtained it is possible to create a visual model of problem domain as presented in figure 1.

On the stage of database, logical design the problem consists in representing the domain model of problem domain in the data model supported by DBMS chosen for the system creation. For this research, Microsoft Access has been used to create the database. In order to understand the system it is necessary to identify domain entities and their relationships.
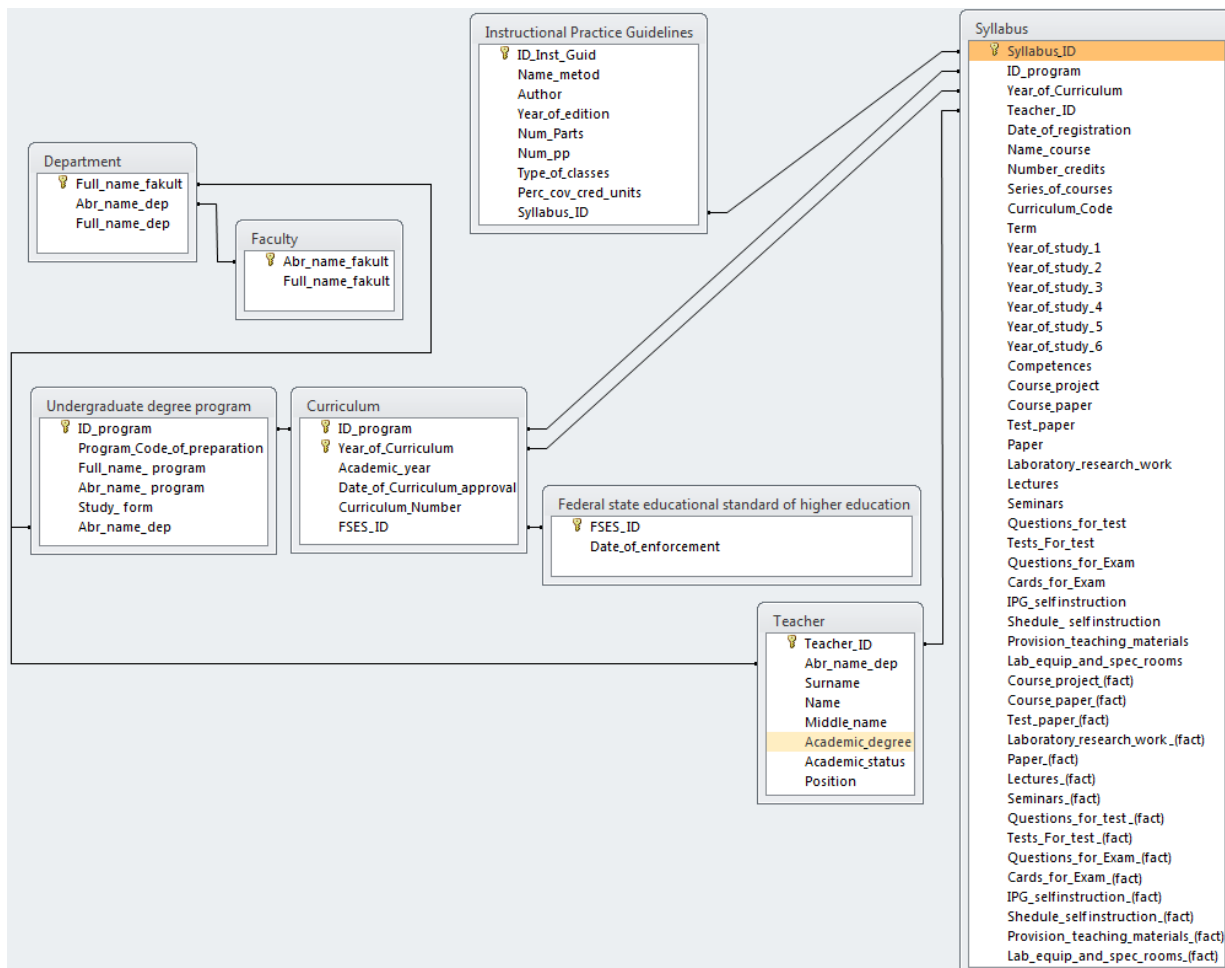

Fig. 1. Visual model of problem domain.

An entity is a system component specifically intended for modeling an object, process or phenomenon of the problem domain to be stored. A point of distinction should be made between entity types and entity instances.

An entity type defines a set of entities that have the same attributes. An entity instance is a single item in this set.

An attribute is a characteristic of an entity object.

The unique identifier of an entity is an attribute or a set of attributes, distinguishing any entity instance from other instances of the same type. Attributes describe each entity and identify entity instances.

Relationship is the graphic representation of associations established between two entities. Degree of relationship refers to the number of entity types that participate in the relationship, i.e. relationship between two entities is called binary. If a relationship connects three entities, it is called ternary or "3-ary." If a relationship connects three or more entities (n entities), it is called an "n-ary" relationship. Most relationship sets in a database system are binary.

The rules of drawing an entity-relationship diagram for the information system created within this research can be described as follows:

a) each faculty can comprise several departments. Each department can belong to one faculty only. There are faculties which do not have any department, but there are no departments which do not belong to any faculty;

b) each department can offer several undergraduate degree programs, each program belongs to one department only. There are departments which do not offer undergraduate degree programs, but there are no undergraduate degree programs which do not belong to any department;

c) each department employs several teachers, but each teacher can be employed by one department only. There are no departments which do not employ any

teaching staff; there are no teachers who are not employed by any department;

d) each undergraduate degree program can have several curricula, each curriculum belongs to one undergraduate degree program. There are no undergraduate degree programs which do not have any curriculum and there are no curricula which do not belong to any undergraduate degree program;

e) each curriculum is based on at least one federal state educational standard of higher education, each federal state educational standard can regulate several curricula. There are no curricula which are not regulated by federal state educational standard of higher education and there are no federal state educational standards of higher education which do not regulate any curriculum;

f) each curriculum makes the bases for the development of several course syllabuses, each syllabus is based on a particular curriculum. There are no curricula which do not have syllabuses and there are no syllabuses which are not specified by a curriculum;

g) each teacher can develop several syllabuses, each syllabus is developed by one teacher. There are syllabuses which are not registered by a teacher; however there are no teachers who haven't developed any syllabus;

h) each syllabus should contain several Instructional Practice Guidelines to fulfill this or that type of tasks for different kinds of high school classes, each Instructional Practice Guidelines has to belong to one course only. There are no syllabuses which do not have any Instructional Practice Guidelines, and there shouldn't be any Instructional Practice Guidelines which do not belong to any syllabus.

In order to study and understand the problem domain the rules discussed above were implemented. This has allowed us to identify relationships that exist between the entities. Entities, relationships, type of participation and degree of relationship are specified in Table 1.

Table 1. Power of entities – relationship.

Entity 1		Relationship		Entity 2	
Name	Type of participation	Name	Degree of relationship	Name	Type of participation
Faculty	Mandatory	comprises	1:n	Department	Mandatory
Department	Optional	offers …	1:n	Undergraduate degree program	Mandatory
Department	Mandatory	employs	1:n	Teacher	Mandatory
Teacher	Mandatory	develops	1:n	Syllabus	Mandatory
Undergraduate degree program	Mandatory	is based on	1:n	Curriculum	Mandatory
Federal state educational standard of higher education	Mandatory	regulates	1:n	Curriculum	Mandatory
Curriculum	Mandatory	regulates	1:n	Syllabus	Mandatory
Syllabus	Mandatory	contains	1:n	Instructional Practice Guidelines	Mandatory

The relational data model of problem domain represents a set of relationships which allows storing data and modeling the relations between them.

With specially formulated set of rules, it is possible to generate the relation schema of relational data model on the basis of domain model. There is a rule that if the degree of binary relationship is equal to 1:n and the type of participation is mandatory, then it is enough to use two relationships: one for each entity, provided that the entity key serves as the primary unique identifier for each entity instance, the key of contiguous entity has to be added as an attribute to the relationship for n-tuply connected entity.

Preliminary relation schemes are generated on the basis of this rule:

1) The faculty comprises Departments. Relationship 1: [n] – then we use:

- Faculty (Abr_name_fakult, …);

- Department 1 (Abr_name__dep, Full_name_fakult…).

2) The department offers Undergraduate degree program. Relationship 1: [n]:

- Department 2 (Abr_name__dep, …).

- Undergraduate degree program 1 (ID_program, Abr_name__dep, …).

3) The department is composed of the teaching staff. Relationship 1: [n]:

- Department 3 (Abr_name__dep, …);

- Teacher 1 (ID teacher, Abr_name__dep, …).

4) Training is provided in accordance with the curriculum. Relationship 1: [n]:

- Undergraduate degree program 1 (ID_program, …);

- Curriculum 1 (Year_of_Curriculum, ID_program, …).

5) The curriculum makes the bases for the development of the course syllabus Relationship 1: [n]:

- Curriculum 2 (Year_of_Curriculum, …);

- Course syllabus 1 (Syllabus_ID, Name_course, …).

6) Federal state educational standard regulates the curriculum. Relationship 1: [n]:

- Federal state educational standard of higher education 1 (FSES_ID, …);

- Curriculum 3 (Year_of_Curriculum, FSES_ID, …).

7) The teacher develops the syllabus. Relationship 1: [n]:

- Teacher 2 (Teacher_ID, …);

- Course syllabus 2 (Syllabus_ID, Teacher_ID , …).

8) Course syllabus contains Instructional Practice Guidelines. Relationship 1: [n]:

- Course syllabus 3 (Syllabus_ID, …);

- Instructional Practice Guidelines 1 (ID_Inst_Guid, Syllabus_ID, …).

The other attribute relationships are created with the nonkey attributes taking into account that they are single valued. Then data model is normalized to Boyce Codd Normal Form (BCNF).

The final relation schema looks as follows:

- Faculty (Abr_name_fakult, Full_name_fakult);

- Department (Abr_name__dep, Full_name_dep);

- Undergraduate degree program 1 (ID_program, Program_Code_of_preparation, Full_name_ program, Abr_name_ program, Study_ form);

- Teacher (Teacher_ID, Surname, Name, Middle_name, Academic_degree, Academic_status, Position);

- Curriculum (Year_of_Curriculum, Academic_year, Date_of_Curriculum_approval, Curriculum_Number);

- Federal state educational standard of higher education 1 (FSES_ID, Date_of_enforcement);

- Course syllabus (Syllabus_ID, Date_of_registration, Name_course, Number_credits, Series_of_courses, Curriculum_Code, Term, Year_of_study_1, Year_of_study_2, Year_of_study_3, Year_of_study_4, Year_of_study_5, Competences, Course_paper, Test_paper, Paper, Laboratory_research_work, Lectures, Seminars, Questions_for_test, Tests_For_test, Questions_for_Exam, Cards_for_Exam, IPG_selfinstruction, Shedule_selfinstruction, Provision_teaching_materials, Lab_equip_and_spec_rooms, Course_paper_(fact), Test_paper_(fact), Paper_(fact), Laboratory_research_work_(fact), Lectures_(fact), Seminars_(fact), Questions_for_test_(fact), Tests_For_test_(fact), Questions_for_Exam_(fact), Cards_for_Exam_(fact), IPG_selfinstruction_(fact), Shedule_selfinstruction_(fact), Provision_teaching_materials_(fact), Lab_equip_and_spec_rooms (fact));

- Instructional Practice Guidelines (ID_Inst_Guid, Name_metod, Author, Year_of_edition, Num_Parts, Num_pp, Type_of_classes, Perc_cov_cred_units).

At a phase of logical design the logical structure of database corresponding to logical model of problem domain is developed. The logical level of data modeling using Erwin provides several levels of representation: Entity, Relationship, Unique Identifier and other Attributes.

Logical structure of relational database is generated at the end of the process of database logical modeling. For this purpose the elements of type, data size, uniqueness, type of participation are set for each attribute.

During the physical design process, data gathered during the logical design phase are converted into a description of the physical database. The physical model depends on a particular DBMS as there no standards for database objects.

The Erwin Data Modeler used to create the information system "Teaching materials" provides the processes of forward and inverse modeling that considerably reduces the time for detection and correction of errors and obtaining approval for a prototype from the customer.

When we use Erwin Data Modeler on the physical level views are created. These views are the database objects in which data are not constantly stored as in a table and the result sets are produced dynamically from queries. Application of views allows the developer to provide each user with the ability to treat data differently, that solves the problems of usability and data security.

Among other functions, Erwin provides with the ability to estimate the approximate size of the database,

---

tables, index data and other objects in a certain period of time after the beginning of operation.

Thus, entities and attributes of the problem domain were identified on the bases of functional specification required for the project development "Teaching materials" for Bachelor`s degree program "Software engineering" and University Curricula documents. The data model and the database were created.

3 Digital ecosystem

Implementation of information technologies for quality control in education has shown that in conditions of digital economy when all stakeholders of educational process (students, teachers, parents, potential employers, academic authorities, etc.) have the right to control and, as a rule, exercise it, a digital ecosystem has to become a foundation for the development of IT infrastructure at the conceptual level.

At the moment there is no unambiguous and standard understanding of this concept, therefore it is possible to define this biometaphor for a software system through the description of its functions and abilities:

- decentralization – the ability of a system to continue at the same level of performance even though one or more components have failed, thus it is important for a system to be fault-tolerant, i.e., have functions and resources distributed between several elements;

- adaptability – the ability of a software system created to improve its parameters under the influence of external environment, first of all – stakeholders in the sphere of professional education;

- openness - the ability of a system to interact continuously with the external environment, this interaction can take, for example, the form of information or material transformations on the border with the system, in other words – transferability and compatibility with the other computer systems;

- self-organization – the ability of a system to spontaneously arrange its elements? in a purposeful manner, using internal factors, without external specific influence (change of external conditions can have stimulating or overwhelming affect);

- scalability – the ability of a system to increase its total output under an increased load when resources(typically hardware) are added;

- stability – the ability of a system to return to its normal or stable conditions after being externally disturbed.

It should be mentioned these abilities are inherent and perennial in many social and technical systems. A digital software ecosystem should also possess specific abilities:

- naturalness and discreteness – the evolving software agent exists (it is not distinguished for the research purposes only). It is a system of interacting social, technical and information objects (e.g., students interacting with a software system on the hardware platform) which is functionally separated from other similar objects. Digital software ecosystems are distinct

systems, so, one can speak about independent evolution of the neighboring agents;

- the key role of ecosystem interactions in identifying speed and direction of evolution when evolution of software agent is considered as a process of developing software upgrade.

So, it is possible to define the concepts which are essential for the generalized digital ecosystem: environment; interactions; agent; multi-agent community; evolution, as it is shown in [10] "The concept of an object is naturally (one is tempted to say, inevitably) arises in the process of evolution". As in educational institution data are less subject to change in time than business processes, the strategy of information systems development where the database functions as a cornerstone, was taken as a foundation in this research. This will provide the result, which is more stable than, for example, functional approach in software development.

4 Conclusion

There are serious problems with the modern educational system around the world driven by a number of factors, including those connected with changing technical environment and the amount of daily processed information. Moreover, there are other factors influencing the process of education. Both sides are interested in its computerization; those who are going to learn and gain knowledge, and those who teach.

The importance of higher education in modern society requires using valid and reliable instruments to measure the quality of education.

Transformation of management system, use of innovations in education result in gradual modernization of traditional management strategy which was formed in the past taking into account new and more flexible methods and tools capable to adapt to changing external and internal conditions. The database developed can serve as an example.

In the countries of Post-Soviet Space, there is an increased demand for job candidates with at least a bachelor's degree. To satisfy this demand it is necessary to use traditional methods and new technologies to increase the quality of education.

The development of management system for higher education institution is a perspective task in ensuring quality of educational process. Ongoing efforts have to contribute notably to the development in this sphere. Encouraging development and deployment of innovative technologies in management of higher education institutions is rather important.

This task represents only the part of the great number of problems which need to be solved for computerization of education in transition to mass higher education.

References

1. B. Barna, S. Fodor, *Advances in Intelligent Systems and Computing* **715**, 684-692 (2018)

2. A. Hermanto, *MATEC Web of Conferences* **154** art. no. 03008 (2018)

3. M. Khouja, I.B. Rodriguez, Y.B. Halima, S. Moalla, *International Journal of Human Capital and Information Technology Professionals* **9**, 52-67 (2018)

4. M.N. Bulaeva, O.I. Vaganova, M.I. Koldina, A.V. Lapshova, A.V. Khizhnyi, *Advances in Intelligent Systems and Computing* **622**, 406-411 (2018)

5. S. Karpenko, G. Kuzenkova, N. Shestakova, N. Borisov, A. Kuznetsov, *CEUR Workshop Proc.* **1761**, 151-157 (2016)

6. O. Barabash, *Proc. of the 12th Int. Scientific and Technical Conf. on Computer Sciences and Information Technologies CSIT 2017* **1**, 313-317 (2017)

7. J.P. Pereira, I. Zubar, E. Natalya, *Advances in Intelligent Systems and Computing* **745**, 422-432 (2018)

8. A. Gunn, *Educational Review* **70**, 129-148 (2018)

9. A.S. Mikhaylov, A.A. Mikhaylova, *Quality - Access to Success* **19**, 111-117 (2018)

10. V.F. Turchin, *Kybernetes* **22**,10-30 (1993)