European Journal of Educational Research
Volume 6, Issue 3, 261 - 267.
ISSN: 2165-8714
http://www.eu-jer.com/

Digital Learning Characteristics and Principles of Information Resources Knowledge Structuring

Margarita Belichenko *
Ariel University, ISRAEL

Nitza Davidovitch
Ariel University, ISRAEL

Yuri Kravchenko
National University of Kyiv, UKRAINE

Received: June 7, 2017 ▪ Revised: July 5, 2017 ▪ Accepted: July 14, 2017

Abstract: Analysis of principles knowledge representation in information systems led to the necessity of improving the structuring knowledge. It is caused by the development of software component and new possibilities of information technologies. The article combines methodological aspects of structuring knowledge and effective usage of information resources which are designed for the scientific and methodological study of practical recommendations in Digital learning implementation. It is shown at the paper that one of the most important problems, which face the processing of knowledge or construction systems, is knowledge representation. The main idea of the paper is increasing the efficiency at the university learning process on the basis of a possible applications and rational structuring knowledge. The proposed concept of using information resources of Digital Learning is based on the idea of using the principles of abstraction, encapsulation, modularity, hierarchy, typing, concurrency preservation and implementation in such stages of this process as algorithmic support of knowledge structuring and structured transfer of knowledge to students. The topic of the article is unique as we try to see Digital learning in multidisciplinary aspects: education, mathematics, and statistics. The article combines a theoretical approach to structuring knowledge that is based on the integrated usage of fuzzy semantic network, theory of predicates, Boolean functions, theory of complexity of network structures and practical aspects and ways of construction Digital Learning Systems at the University.

Keywords: *Information resources, knowledge structuring, digital learning, fuzzy semantic network, knowledge representation, higher education.*

To cite this article: Belichenko, M., Davidovitch, N. & Kravchenko Y. (2017). Digital learning characteristics and principles of information resources knowledge structuring. *European Journal of Educational Research*, *6*(3), 261-267. doi: 10.12973/eu-jer.6.3.261

Introduction

Methodological aspects of structuring knowledge and effective usage of information resources are designed for the scientific and methodological study of practical recommendations in Digital learning implementation. Researches have shown that one of the most important problems which face the processing of knowledge or construction of systems is knowledge representation.

This is due to the fact that knowledge representation ultimately determines the characteristics of the system. The selection appropriate method of knowledge representation can avoid unnecessary complexity of system and solve many issues. The analysis of existing principles knowledge representation at information systems led to the conclusion regarding to the necessity of improving existing approaches to structuring knowledge. It is caused by the development of hardware component and software as well as new possibilities of information technologies in general.

The proposed concept of using information resources in Digital Learning is based on the idea of using the principles of abstraction, encapsulation, modularity, hierarchy, typing, concurrency preservation and mplementation such stages of this process as algorithmic support of structuring knowledge and structured transfer of knowledge to students.

Problem Statement

Digital learning systems in the world are developed and successfully used offering services to study the various training programs and courses. Network systems of distance education consist of the following basic elements (Bykov & Kucherenko, 2005):

* **Corresponding author:**
Margarita Belichenko, Ariel University, Ramat HaGolan Str., 28, Ariel, Israel.
Email: naukakyiv@gmail.com

- the institution, as the organizational structure of digital learning;
- information resources - databases training and reference materials;
- hardware and software technologies providing digital learning;
- digital learning teachers (tutors) and students, trainees.

There is the most visual and informative, on our opinion, structured diagram of Digital Learning Network System. Figure 1.

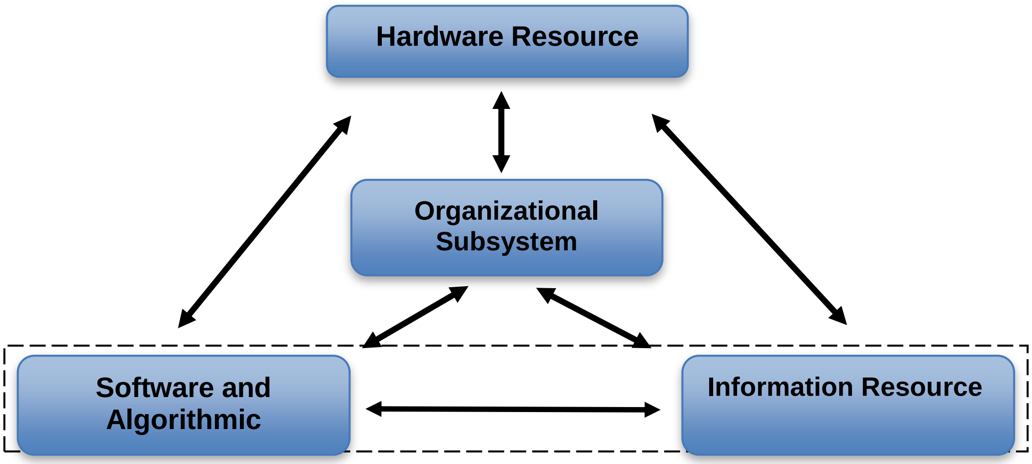

Fig. 1. Digital Learning System

Information resource - a set of documents in information systems (libraries, archives, data banks, etc.). [Law of Ukraine "On the National Informatization Program", p. 1].
Software and algorithmic resource appropriately consider a subsystem of information resource.

The basic characteristics of an information resource as a complex system

1. Information resource consists of interconnected resources which can be divided from under-resources down to the lower levels of the abstraction.
2. Choosing a lower level of the abstraction is arbitrary and more determined by the observer. Internally element usually is stronger than connections between the elements connections.
4. Hierarchical systems consist of several different types of subsystems which are implemented in different orders and various combinations.
5. Advanced information resource is an inevitably result of the simple resource development.

We describe the resource Ω, as a set of some abstract elements

$$\{\omega_1, \omega_2, ..., \omega_N\} \in \Omega,$$

then it is logical, based on the basic principles of the theory of artificial intelligence, complex systems theory and systemic approach which adds the following, $\{\omega_1, \omega_2, ..., \omega_N\} = v_j \cup d_k$, $j,k \in N$ abstract elements that present information resource as a union set of an information resource elements $\{v_j\} \in V$, and a plurality of links between elements of the information resource $\{d_k\} \in D$ where $j,k \in N$.
The researches have shown that structuring of knowledge or knowledge of structural analysis today becomes significant because of the necessity of implementation of new information technologies. In our view, the concept of "structured knowledge" can be defined in different ways. Some experts the term "structuring" replace to the term

"modeling". However, the first term is broader towards to the second and covers a greater range of concepts and processes. On the other hand all the techniques used in modeling are true for the term "structuring."
Modeling or knowledge representation is the creation of a model, which can be reflected by the existing structure of knowledge and methods of knowledge objects.

Theoretical Background

The subject of modeling is called as the researching of the real object-reproducing model, a sign modeling is the study of the object character based on logical mathematical structures. Modeling is one of gnoseological categories that allow to learn object-based study of its model [Lavrov A.].
The term "structuring" is fairly new and can be interpreted in two ways - on the one hand, the Concept of structuring may be one of the characteristics established gnoseological model object and in this sense the Concept of structuring is partial towards modeling, on the other hand, the definition of structuring, as a certain process, is more general in relation to the knowledge.
As it is known, knowledge of structural analysis is the process of creating semi-formalized description of subject knowledge. The purpose of structuring knowledge is creating adequate virtual or logical reflection of the real object. Virtual reality is understood as created by artificial means an environment that is issued or adopted by the subject in it's actions to a real or close to real.
The basic principles of information resources usage are: abstraction, encapsulation, modularity, hierarchy, typing, parallelism and preservation (Aliev, 1990).

Proposed sequences of these principles are applied for the first time.

The main principles:
- Abstraction;
- Encapsulation;
- Modularity;
- Hierarchy.

Additional principles:
- Standardization;
- Parallelism;
- Preservation.

Abstraction is one of the main methods used to solve complex problems. Abstraction emphasizes parts essential for consideration and usage that puts over those which currently are insignificant. To sum up different points of view, we get the following definition of abstraction: "Abstraction provides essential characteristics of an object that distinguish it from all other kinds of objects and thus clearly defining its conceptual boundaries in terms of the observer" (Schleer. 1993).
Abstraction focuses on the external features of the object and separation the essential features of the treatment from non-essential. Grady Booch considers useful one additional principle which is called the Principle of least surprise, according to which abstraction should cover the entire behavior of the object, but no more and no less, and do not bring surprises or side effects that are beyond its scope of applicability. Choosing the right set of abstractions for a given subject area is the main task of its designing (Buch, 1992).
There are variety of abstractions, beginning with objects that almost exactly match the realities of the domain, and ending objects that have no right to exist.

Abstractions (the most useful to the least useful):

- Abstraction of entity - the object is the essence of a utility model in the subject area;
- Treatment Abstraction - object consists of generalized set of operations;
- Abstraction of virtual machine - an object groups operations or uses with higher levels of management a set of lower-level operations;
- Arbitrary Abstraction - the object contains a set of operations that do not have common to each other.

Abstraction focuses on constitutive from the point of view of the observer, characteristics of the object. The central idea is the concept of abstraction invariant. Invariant is a logical condition which value (true or false) should be maintained.

For each operation object it is possible to set the pre-conditions (invariants predictable surgery) and post-conditions (invariants that satisfies operation). Changing the invariant violates the contract is associated with abstraction. In case of violation of any condition excited exceptional situation.
All abstractions have both static and dynamic properties. For example, a file object requires a certain amount of memory on a particular devices that has a name and content. These attributes are static properties.

The specific values of each these properties and the dynamic changes in the use of the facility: the file can be increased or decreased, changed its name and content. In procedural style of programming steps that change the dynamic characteristics of the objects are the essence of the program. Style of programming is focused on rules, characterized by the fact that exposure to certain conditions, certain rules are activated which in turn cause other rules.

Abstraction and **encapsulation** are complementary: abstraction aims to conduct observation of the object and encapsulation deals internal device. Most encapsulation performed using concealment of information that disguise all internal parts that do not affect the external behavior. And of course, it hides the internal structure of the object and its implementation methods. Encapsulation defines clear boundaries between different abstractions. Abstraction makes sense only with encapsulation. It means that there are two parts in the class, interface and implementation. The interface reflects the external behavior of the object, describing the abstraction behavior of all objects of this class. The internal presentation describes the implementation of abstractions and mechanisms to achieve the desired behavior of the object.

The principle of interface separation and implementation corresponds to the essence of things: in the interface part contains everything related to the interaction of the object with any other object; implementation hides from other objects all the details that have no relationship to the process of interaction between objects.

Encapsulation hides implementation details of the object. Thus, the encapsulation can be defined as follows: encapsulation is a process of separation between object elements that define its device and behavior; encapsulation serves to isolate contractual obligations abstractions of their implementation.

Modularity is associated with the division of program modules, and can reduce its complexity. There are set well-defined and documented interfaces inside the modular programs. These interfaces are invaluable for a comprehensive understanding of the program as a whole.

This feature is particularly useful when the system consists of many hundreds of grades. In the most languages that support a principle of modularity as an independent concept, an interface module is separated from its implementation. Thus, modularity and encapsulation virtually are inseparable. Different programming languages are supported by modularity differently.

Proper distribution of program modules is almost as challenging as choosing the right set of abstractions. The modules act as physical containers with the definitions of classes and objects in a logical system. For small tasks it is allowable description of all classes and objects in a single module. However, for most applications (except the trivial) the best solution would be grouped in a separate module logically related classes and objects, leaving open those items which are absolutely necessary to see the other modules.

Modularity allows to keep abstractions separately. The traditional structural designing of modularity is the art to decompose in a part of routines which use each other or they are changing together. The Digital learning system should be physically divided into classes and objects that make up the logical structure of the project. So modularity is a property system that was expanded to internally coherent, but poorly interconnected modules.

Thus, the principles of abstraction, encapsulation and modularity are complementary. Logical object determines certain boundaries of abstraction and encapsulation, and modularity makes them physically immutable.

Significant simplification in understanding complex problems is achieved through the formation of abstractions hierarchical structure. Hierarchy is the ordering of abstractions, the location of the levels. The example of hierarchy is single inheritance.

Inheritance means a relationship between classes, where one class borrow structural or functional part of one or more other classes (respectively, single and multiple inheritance). In other words, inheritance creates a hierarchy of abstractions in which subclasses inherit the structure of one or more superclass. Often subclass completes or rewrites components of higher class.

The principle of typing. The concept of the type is taken from the theory of abstract data types. Grady Booch defines the type as "accurate description of the properties, including the structure and behavior related to a certain set of objects" (Schleer, 1993). For our purposes it is necessary assume that the type and class terms are interchangeably. Casting is a way to protect against the using one class instead of another, or at least control such using. The idea of reconciliation types takes the notion in typification of a central place.

The principle of parallelism. There are tasks where automated systems have to handle many events simultaneously. In other cases, the necessity for computing power exceeds the resources of one processor. In each of these situations it is naturally to use multiple computers for solving the problem or engaging in multitasking multiprocessor computer. The process of flowing control is the fundamental unit of action in the system. Each program has at least one flow control system which has many parallel streams: some of them are short, others live throughout the session system. Real parallelism is achieved only on multiprocessor systems and systems with one processor simulating parallelism by time allocation algorithms. Many modern operating systems provide direct support of parallelism, and this is very beneficial to the possibility of the parallelism using information resources in Digital learning.

Using the information resources in Digital learning system is based on abstraction, encapsulation and inheritance, concurrency main emphasis on abstraction and synchronization processes. The object is the Concept on which these two points of view come together: each object (derived from the abstraction of the real world) can be a separate control flow (abstraction process). This object is called active. On this basis, we give the following definition of parallelism.

Parallelism allows different objects to operate simultaneously. Parallelism is a property that distinguishes active from passive objects.

Three approaches of parallelism in modern conditions

Firstly, parallelism is an intrinsic property of some programming languages. Thus, the Ada language mechanism is implemented as a parallel process task. Smalltalk class is a process, which inherit all active objects. There are many other languages with built-in mechanisms for parallel execution and synchronization of processes - Actors, Orient 84 / K, ABCL / 1, which provide similar mechanisms of parallelism and synchronization. In all these languages active objects can be created, the code is constantly performed in parallel with other active objects (Buch, 1992).

Secondly, we can use the class library that implements some kind of "light" parallelism. For example, AT & T library for C ++ contains classes Shed, Timer, Task etc. Its implementation directly dependents on the platform, but the interface is quite effective. This approach parallel execution mechanisms are not embedded in the language (and, hence, do not affect the system without parallelism), but at the same time almost perceived as a embedded.

Finally, creating the illusion of multitasking usage can be interrupted. For example, in our implementation class Active Temperature Sensor we could have a hardware timer that periodically interrupts the application, after which all sensors measure temperature and would apply if necessary to their calling features.

Once we administer into the system of parallelism, it immediately raises the question of how to synchronize active relationships of objects to each other and to other objects that are in series. For example, if two objects send messages to the third, there must be some mechanism that ensures that facility, which directs the action without the simultaneous collapse of two active sites trying to change its position.

Therefore, the importance of information resources in digital learning is obvious, because it incorporates the knowledge and skills that are needed by the trainee. In this regard, particular relevance gets the process of transferring information resource to students. The task of distribution of any kind of resource is always relevant and requires more attention, because of its efficiency depends on the decision process for which a resource is designed (Kravchenko, 2009).

It is logical to highlight the following stages of the process:

- Converting data into knowledge;
- Structuring knowledge (including modeling or representation);
- Algorithmic providing of structured knowledge;
- Structured knowledge transmission to the audience (Fig. 2).

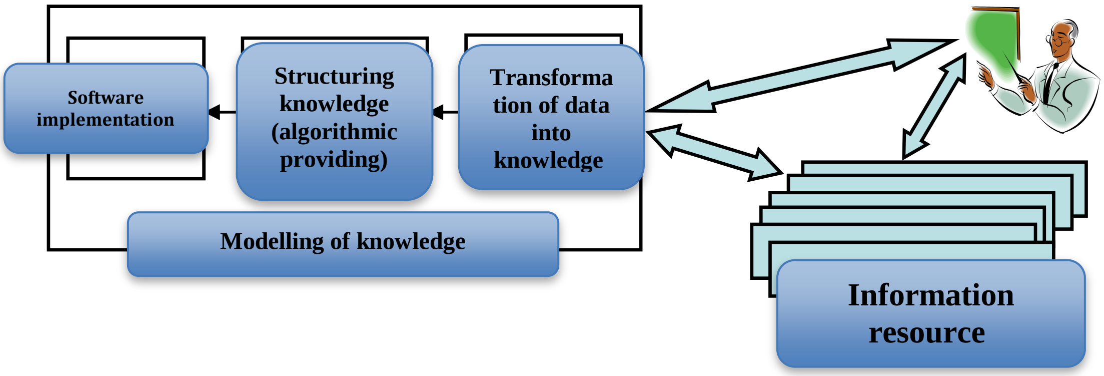

Fig. 2. The Concept of Using Information Resources

Conclusion

Any program's object exists in memory and lives in time. It is considered that there is an infinite set of duration objects of existence: there are objects that are presented only during the calculation formula, but there are also such databases that exist independently of the program. This spectrum includes the preservation of objects:

- Intermediate results of calculation expressions;
- Local variables in a procedure call;
- Own variables, global variables and dynamically generated data;
- Data stored between sessions of the program;
- Data stored in the transition to the new version;
- Data which are experiencing all the program.

Typically, the first three levels are involved in the programming language, and the last into database. This conflict of cultures leads to unexpected solutions, programmers develop special schemes for preservation between startup and programs designers and database alter the technology under the short-lived objects. Unification of principles of parallelism for objects allowed to make a parallel programming language. The introduction of preservation as part of the usual object-oriented approach leads us to the object-oriented databases.

In practice, these databases are based on proven models - sequential, indexed, hierarchical, network or relational, but the programmer can implement object-oriented abstraction interface through which database queries and other operations are performed in terms of facilities, whose lifetime exceeds the lifetime of a particular program.

This standardization simplifies the development of certain types of applications, allowing, in particular, to apply a common approach to different segments of the program, some of which are related to databases, while others have no such connection. Typically, survival is achieved by using an object-oriented wrapper for the relational database management systems; it better, particularly for those which have already invested facilities in a relational system. Preservation is not only the problem of storage. It makes sense to save classes, so that the program can correctly interpret the data. This creates great difficulties with increasing amounts of data, especially if the class object had suddenly changed. Preservation is the ability of an object to exist in time and in space, moving from its original address space. Therefore, analysis of principles knowledge representation in information systems led to the Concept of using Information resources and principles of information resources knowledge and structuring of Digital learning processes. It is implemented in stages as algorithmic support of knowledge structuring and structured transfer of knowledge to students in practical aspects of designing Digital Learning Systems.

References

Altbach, P. G. (2007). Introduction: The underlying realities of higher education in the 21st century. In P. G. Altbach, & P. M. Peterson (Eds.), *Higher education in the new century: Global challenges and innovative ideas*. Rotterdam/Taipei: Sense Publishers.

Aliev R.A. (1990) Production systems with artificial intelligence / Aliev R.A, Abdikeev N.M, Shakhnazarov M.M - M.: *Radio and Communication,* - 264 p.

Altbach, P. G., Reisberg, L., & Rumbley, L. E. (2009). Trends in global higher education: Tracking an academic revolution. *Report prepared for the UNESCO2009 World Conference on Higher Education, Paris: United National Educational, Scientific, and Cultural Organization.*

Bridges, D., Juceviciene, P., Jucevicius, R., Mclaughlin, T. H., & Stankeviciute, J. (Eds.). (2014). *Higher education and national development: Universities and societies in transition.* London: Routledge.

Buch G. (1992). *Object-oriented programming* / Buch G. - M.: Concord, - 519 p.

Bushuyev, S., & Bushueva, N. (2010). Project Management. Fundamentals of professional knowledge and competence assessment system of project managers *(National Competence Baseline, NCB UA* Version 3.1).

Bykov V., & Kucherenko V. (2005) *Distance learning process: a tutorial [ed.]. - K: Millennium, - 292 p.*

Iegorchenkov, O. V., Lisitsyn, O. B., & Kataev, D. S. (2012). Optimization of information management in product design management system. *Management of Complex Systems*, *13*, 28-31.

Kukharenko V. V. (2001). Practice of distance learning: a manual. - *Kharkiv: NTU "KhPI"*, - 124 p.

Kravchenko Yu. V. (2009). Concept of the structure of the informational resource of the remote systems / Kravchenko Yu. V., Oksiuk OG // *In the context of information technology in the sphere of safety and defense.* - K.: - No. 1 (4). - P. 6-11.

Kravchenko Yu. V. (2009). Mathematical model of a complex social system / Kravchenko Yu.V., Lobanov AA, Oksiuk AG // *The material of the international scientific conference "Intellectual systems in making decidions and solving problems of the enumerative telecommunication* (ISDMCI'2009)". - Kherson: KhNTU, - P. 56-57.

Kravchenko Yu. V. (2003). Methodology of multicriterial discrete optimization of complex technical systems on matroid structures / Yu. V. Kravchenko, V. V. Afanasyev // *Scientific Journal* ІПМ в Eii. G. Є. Puhov vip. 22 - 1. - K.: IPM в E ім. G. Є. Pukhova - P. 73 - 78.

Kravchenko Yu. V. (2010). The model of representation of knowledge on the basis of predicates and fuzzy semantic networks / Kravchenko Yu. V., Oksiuk AG, Andrushchenko VN // *Materiels of the international scientific conference "Intellectual systems in making desicions and problems of the enumerative telecommunications"* (ISDMCI '2010). " - Kherson: KhNTU, - P. 156.

Tesla, J. M., Inosova, S. V., Timinskyy, O. G., & Iegorchenkov, O. V. (2010). Interpretation and use of the uncertainty principle in project management. *Management of Complex Systems*, *3*, 33-36.