Key words – personalization, adaptive learning, e-learning, learning objects, learning style, learning profile

Maria ZAJAC[*]
Marcin PIEKARCZYK[**]

LEARNING STYLES BASED ADAPTIVE ENGINE (LSAE) AS A PERSONALIZATION TOOL IN VLE

The paper deals with a problem of ensuring high efficiency of learning in Virtual Learning Environment. One of the crucial factors in that process is personalized approach to the learner needs and expectations. Psychological theory of learning indicates that everyone has his/her individual learning style, which influences the way new skills and knowledge are acquired In order to prepare adequate form of learning material the idea of Reusable Learning Objects can be applied. Learning objects containing small pieces of information stored in various forms (e.g. text, graphics, video clips) create the repository called a knowledge base. The task of Activity Monitoring Unit is to trace and register learner's behaviour performed during the learning process and to generate the description of learner's profile as the input data for a course management unit called Dynamic Assembly Engine. In the paper the idea of such adaptive engine implemented in Learning Content Management System (LCMS) will be described.

[*] Pedagogical University, Computer Science Department, ul. Podchorazych 2, 30-084 Krakow, mzajac@ap.krakow.pl
[**] Pedagogical University, Computer Science Department, ul. Podchorazych 2, 30-084 Krakow, marp@ap.krakow.pl

1. INTRODUCTION

Along with growing popularity of e-learning solutions the question of its quality and efficiency becomes more important than ever. There are various ways of assuring these values, but probably the most promising is personalization of learning process. One of the most commonly indicated advantages of online learning is free access in asynchronous learning to online resources at any time and any place. This immanent e-learning feature is quite often indicated as a way of personalization although, in authors' opinion, it should be called individualization. It refers only to the "learning conditions", but the way of presenting learning material and performing learning activities remains the same for all the learners, whereas in fact everyone has his or her own individual learning preferences. In traditional classroom a good teacher can monitor the behaviour of his/her students and change the teaching methods, sometimes even on the spot, in order to get possibly best results of the work. In virtual learning environment (VLE) such adaptations are usually impossible. When knowledge is distributed in an automated way necessary adaptation features must be included in a learning system.

During the past decades many researchers have tried to solve the problem of personalised learning content delivery. A number of intelligent tutoring systems (ITS) as well adaptive hypermedia systems (AHS) have been elaborated, but they normally functioned in local environment.

As Brusilovsky spots in [1] "typically in a class lead by one of the authors of the adaptive system". It was expected that the Web with its powerful technologies will help to move such solutions from labs to real classrooms. Yet, instead learning in online environment has been dominated by Learning Management Systems (LMS) in which, the achievements in the fields of AH and ITS had not been applied. Why did it happen? Probably the main reason is that these systems usually support just one function (teaching one particular subject for instance), whereas modern universities expect the powerful integrated systems that are able to serve for all their educational needs. The question is then whether the attempts should lead to extending already existing ITS or AH solutions to make them more universal or whether the best solutions from those fields should be transferred to Learning Management Systems.

Another important handicap of AH or ITS is that they are nor shareable, whereas in contemporary VLEes this feature has become almost a requirement. That is one of the reasons why the authors of this paper decided to extend the possibilities of e-learning platform already used at the university (e-sgh.pl) instead of implementing a totally new, independent system. The details of this solution will be given in the following chapters.

2. THE ROLE OF LSAE

No matter how the distribution of knowledge is organized (AH, ITS or LMS) everyone, who tends to fulfil personalized approach, has to answer following questions:

1. *Which features of learner's profile will be taken into consideration?*

and

2. *How to collect them?*

With regard to the first question the commonly implemented solutions are based on theory of learning styles and the way of gathering necessary data is a questionnaire available on the e-learning platform. That is why it becomes a very important issue to include at least some features of personalization directly in LMS (Learning Management Systems) or in the knowledge base used by LMS. It can help to create online learning courses well adjusted to learners' individual needs and preferences. It means that the structure of a typical LMS should be "enriched" by an additional subsystem (engine), which can choose – according to some predefined decision rules – appropriate learning objects from the repository, join them and create a course. This system has been called Learning Styles Based Adaptive Engine (LSAE). It should be able to fulfil three main tasks, which can be defined as:

1. Collecting the data concerning student's learning profile;
2. Creating the set of input factors describing expected form and structure of learning content being delivered to the student;
3. Designing the structure of knowledge base, in particular the methodology of dividing learning content into *learning objects* that joined together can create a complete learning unit.

3. CHARACTERISTICS OF THE INTELLIGENT LMS

Classical approach to automatic learning content delivery using RLOs database illustrates figure 1. The structure of a typical LMS (Learning Management System) has been enriched by an additional subsystem (generator), which can choose – according to some predefined decision rules – appropriate learning objects from the repository, join them and create a course.

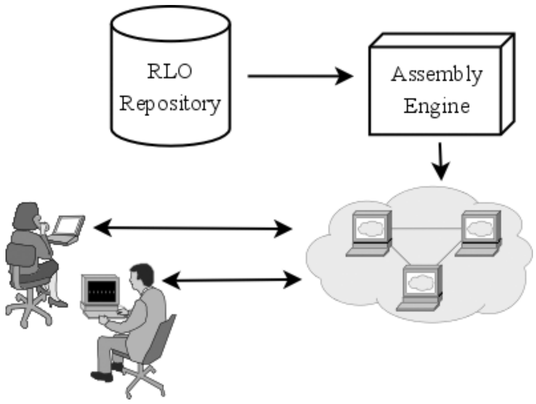

Figure 1. Basic model of learning content delivery using learning objects

This solution has two important advantages, which are the ability to utilize shareable resources and automation of creating the content for learning process. But there is also an essential handicap of the system because it does not take into consideration one's individual learning preferences. The Assembly Engine creates a course joining all the learning objects related to the topic. The learner usually receives learning materials prepared in a uniformed way. He or she has no influence on the final structure of the content being delivered. In order to ensure that the course has been prepared according to students' needs and expectations it is necessary to involve them directly in the process of creating a course. The idea of the system described in this paper is based on this foundation. The key issue is to add the user's interaction as an integral part of the course creation process without loosing automatic content delivery effect.

The changes should be introduced depending on the activity of a particular user. The appropriate system component has to monitor the factors like for instance: the order and the type of resources being used, the amount of references to them or even the number of recurrences to the previously used items. Figure 2 shows the idea of a system improved by introducing the Activity Monitoring Unit (AMU), which is responsible for collecting these data. Its presence in the general system structure makes it possible to create the feedback loop and in consequence, to construct highly personalized content in a fully automatic manner.

Data circulation within the system is completely closed. Moreover, this solution allows to improve systematically the level of compatibility between the way the learner absorbs new knowledge and the way it is delivered. It can be achieved by the regular modifications of the learning path represented by the chain of the features describing user's profile (UP). More details can be find in chapters 4 and 5. The Dynamic Assembly Engine has to merge selected LOs in order to meet relations requirements defined by UP chain. At the same time, this unit has to take into consideration compulsory mutual relations between particular course components imposed by learning objectives as well as by the course syllabus. Figure 2 shows the information flow in the system.

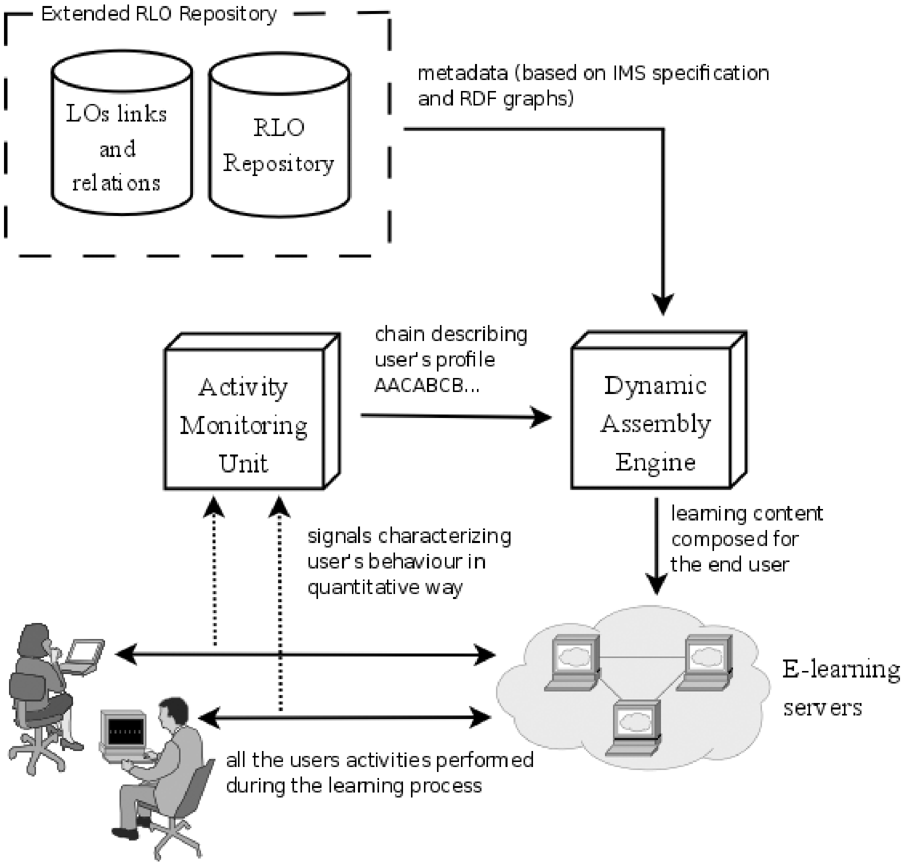

Figure 2. The structure of the learning content delivery system with embedded activity monitoring mechanism and the information flow

4. COLLECTING NECESSARY DATA

Psychology of learning indicates that everyone has his or her own learning style. There are several ways of defining those styles depending on the criteria taken into consideration. Richard Felder in [2] describes, that when the background for classification is Carl Jung's theory of personality types (extraverts, sensory, thinkers and judgers) 16 different learning styles are usually named and measured. Meyers-Briggs Type Indicator (MBTI) [6] is a well known example of inventory used for that purpose. Alternatively Hermann Brain Dominance

Instrument (HBDI) classifies learners preferences for thinking in four different modes based on the task-specialized functioning of the physical brain (left brain, cerebral, left brain, limbic, right brain, cerebral, right brain, limbic).

In conjunction with Howard Gardner's Theory of Multiple Intelligences Memletic Learning Styles Inventory has been prepared [5]. It distinguishes 7 different learning styles corresponding with various types of intelligence (see fig. 3). It is a *verbal (linguistic) learning style*, *a visual*, *aural*, *logical* and *physical* and with regard to our relations with others participants of learning process also *solitary* and *social learning styles* have to be taken into account. There is a significant difference between all the others previously mentioned definitions and the last one. While the outcome of a typical learning styles inventory is normally one dominating learning style (or, in some cases, two of them) Memletic LSI gives the information about the extent to what each of seven learning styles taken into account is used by a particular learner. Such approach allows the learning designers to prepare various forms of learning objects combined with versatile activities that involve different styles and therefore enable more efficient learning. This concept of personalization gives the backgrounds to the intelligent Engine, which is the subject of the paper.

4.1 KS-TIW QUESTIONNAIRE

As it was already mentioned most commonly used means of collecting data concerning user's profile is a questionnaire. There are lots of such tools prepared in electronic version which can be quite easily included into the e-learning platform. For the purpose of the system described in this paper a questionnaire based on Howard Gardner's Multiple Intelligence theory and Memletics Learning Styles Inventory has been elaborated. It must be clearly underlined that it is not simply a translation from the English version but a model built on the same backgrounds. Learning styles are strongly dependent on cultural and educational context, which means that the questions must correspond with one's educational experience and the conditions he or she was grown up and therefore cannot be directly transferred from the other environment. The questionnaire has the acronym KS-TIW from its Polish name[1] which can be translated into English as Learning Styles Questionnaire based on Multiple Intelligences Theory. It consists of 70 questions divided into 7 groups related to 7 learning styles being recognized. Its role in the system is to bring the information about possible learning styles of the potential learners. Each person is represented by the set of 7 values from the range 0-20, which illustrate the "involvement" of every recognized learning style in one's learning process. The results can also be presented in a graphic form. Figure 3 shows chosen graphs based on KS-TIW data. In figure 3a we can spot the dominance of physical learning style, which means that this person prefers "learning by doing". As the social dimension for this learner has also high value probably the group work will be more appropriate then individual studying. Figure 3b shows slightly different preferences – we can presume that although still "learning by doing" is also effective for this person verbal delivery of knowledge (e.g. descriptions and explanations) both in written and in aural form are even more important. Such information can be really helpful in construction of sample courses.

Figure 3. Graphical visualization of learning styles (a) dominance of social and physical learning styles, (b) dominance of verbal, aural and physical learning styles

During the research study the questionnaire data filled in by 220 students have been collected. The first step then was to find some similarities among them, which would allow to distinguish several "sample" profiles that can serve as a background for creating the input data for Dynamic Assembly Engine responsible for preparing the learning content adjusted to each individual learner. As there are no simple rules that would enable finding such subsets of learners it was decided to use some artificial intelligence techniques. Actually two steps approach was undertaken. During the first phase cluster analysis was used in order to divide the population of 220 learners into several clusters. Each cluster should represent a different learning profile. As the number of possible clusters was unknown the agglomeration method was used. Various types of linkage and different possible metrics have been tested. Figure 4 shows some cluster dendrograms illustrating clustering results by complete linkage and three chosen metrics: Euclidean, exponent metric (generalized Euclidean distance, r=4, p=2) and Manhattan.

[1] KS-TIW = Kwestionariusz Stylów oparty na Teorii Inteligencji Wielorakich

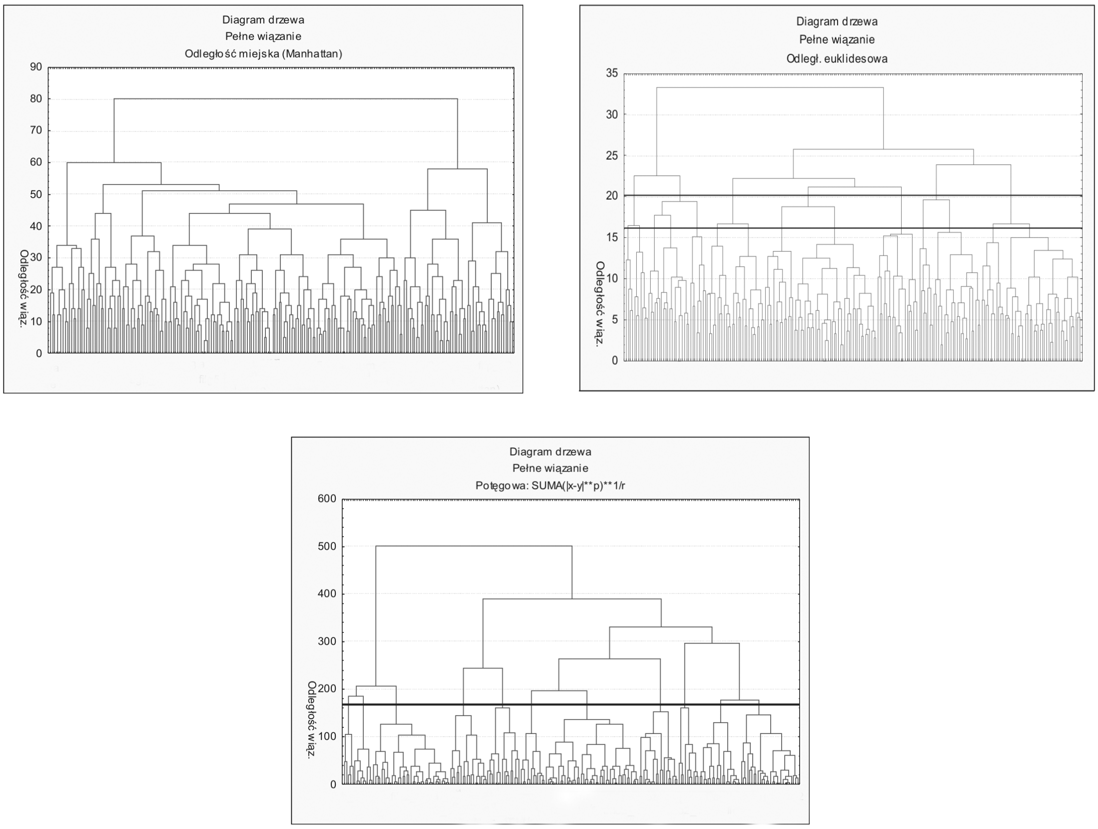

Figure 4. Cluster dendrograms - agglomeration method, complete linkage (a) Manhattan metric (b) Euclidean metric (c) exponent metric (generalized Euclidean distance), r=4, p=2

Other forms of linkage appeared to be completely inappropriate. As far as the choice of metric goes exponent metric with r and p parameters' values respectively 4 and 2 seem to be most suitable. By other metrics either the number of clusters is too small, or the differences between their size are too big (see indicated division lines).

4.2 INPUT DATA BASED ON THE COURSE STRUCTURE

The questionnaires are the easiest way of collecting the data but they have also some significant disadvantages. First of all a learning process is delayed from the very beginning as it must be preceded by filling in the questionnaire. Such imperative can be understood by some users as unnecessary waste of their time, which means they can fill it in without necessary attention, simply in order to pass it through. In consequence the data colleted in this way is not only useless but sometimes even unreliable as it cannot reflect the existing user's preferences. In order to avoid collecting such misleading information another solutions should be considered. The most natural alternative seems to be the idea of gathering necessary data during the learning process. In other words, the learning profile is built and improved step by step during the learner’s work with the course. Then the most important question concerns the tool responsible for tracking the learning process. Different solutions can be found in literature. Those more advanced, which focus on creating the image reflecting learning preferences and not only on pure demographical data like age, sex or school level for instance, usually are based on the use of artificial intelligence. Quite interesting series of experiments was performed by the team from ISISTAN Research Institute in Buenos Aires, Argentina. The researchers tried out several approaches; they implemented feed-forward neural networks, Bayesian networks and genetic algorithm for recognizing and measuring learners’ preferences. Their works are described respectively in [3, 9 and 10] However, no matter what tool is used for gathering the data, it is necessary to define which features are traced and how to create the input data of them.

For the purpose of the system described in this paper a few steps approach has been applied. These steps could be defined as follows:

1. the significant parts of a course structure as well as possible activities are indicated – all of them are called Learning Objects (LO),
2. the events accompanying previously indicated elements are defined,
3. the rating scale that allows to “measure” learners behaviour while these particular events happen is established.

The structure of a particular course depends on the subject being taught but nevertheless some typical elements constitute the core part of each course. These objects can be divided into several categories according to the roles they play in a course. Two different groups of roles have been distinguished. The first one corresponds with a goal particular LO serves in a course – it has been called “logical”. The first category in this group is called *learning content delivery* and it includes such LOs as: core course content, additional explanations, hints and tips, examples, additional resources etc.). Logical role also refers to *knowledge consolidation* (exercises, tasks, problems to be solved), *assessment tools* like tests and quizzes, as well as *activity tools* like chats and forums. The other group – called “physical” refers to the form the individual learning objects can have (e.g. an excerpt of text, a graphics, an audio or video recording). Following expressions describe indicated set of categories and Learning Objects linked to them. These various types of objects are called components as they constitute a course vector defined below. For each category of elements its descriptors should be defined. They have a qualitative character and can take values A to E which mean respectively: A – very often/very much, B – often/much, C – sometimes/average, D – rarely/little, E – never/none. A singular component is described as S_{ji}

where:

$i = 1 \ldots L$ – number of the objects' category (role)

L – total number of categories

$j = 1 \ldots k_i$ – number of the object in one category

k_i – number of features in category "i"

Components S_i create a course vector C^{vec} which can be defined as follows:

$$C^{vec} = \begin{bmatrix} S_I \\ S_{II} \\ S_{III} \\ S_{IV} \\ S_V \end{bmatrix}$$

where:

$S_i = \lfloor S_{i1}, S_{i2}, \ldots, S_{ik_i} \rfloor$

$i \in \{I, II, III, IV, V\}$ (at present five different categories have been distinguished)

$\forall_i \forall_j S_{ij} \in \{A, B, C, D, E\}$

The collection of chains representing all the elements of course structure used for that particular content constitutes the input for DAE, which is responsible for choosing appropriate learning objects from the repository (base of knowledge) and combining them into a personalised course. Unfortunately the components of a course vector do not describe all the features such a course must have as they do not define "the amount" of particular types of objects (like number of difficult or simple tasks for instance). As it is almost impossible to create an exhaustive set of rules that can be applied in order to create such a course another way of supplying necessary information must be found. In this case genetic algorithms seem to be helpful. DAE generates a population of possible courses and each of them is next compared with the primary course vector (C^{vec}). New generation of possible courses inherits the genes from those courses, which indicate the structure most similar to the primary vector C^{vec}. This "similarity" is measured by comparison of each chain of course features (course vector components) with a sample vector C^{vec}. Two important aspects are taken into account in this comparison. The first one is the number of different features in both chains (Hamming distance) and the second one is the "total cost" of changes to be introduced in compared vector (C^{can}). Following formulas describe this process:

x, y – two features chains to be compared;

$x = (x_1, x_2, \ldots, x_k)$ $\quad y = (y_1, y_2, \ldots, y_k)$

$x_i, y_i \in \{A, B, C, D, E\}$

$V = (V_A = 5, V_B = 4, V_C = 3, V_D = 2, V_E = 1)$ – a set of value functions used for transferring the data;

$D^i_{Hm} = \left| V_{x_i} - V_{y_i} \right|$ – the "value" of the distance between the symbols on "i" position of both chains;

$D_{Hm} = \sum_i D^i_{Hm}$ – modified Hamming distance (including the "cost" of change on "i" in the whole vector C^{can};

As it was already mentioned C^{vec} is an input course vector. C^{can} refers to a vector defining the structure of course „candidate" generated at a particular level of evolution. The total cost f_c which has to be paid in order to convert $C^{can} \to C^{vec}$ (or the other way round) can be derived from a following formula:

$$f_c = J^T \cdot GD_{Hm}(C^{vec}, C^{can})$$

where:

$J = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$ – matrix of ones, $L \times 1$ (L=5 in this case)

while

$$GD_{Hm}(C^{vec}, C^{can}) = \begin{bmatrix} D_{Hm}(C^{vec}_I, C^{can}_I) \\ \cdots \\ D_{Hm}(C^{vec}_V, C^{can}_V) \end{bmatrix} = \begin{bmatrix} D^1_{Hm}(C^{vec}_{I1}, C^{can}_{I1}) + \ldots + D^{k_1}_{Hm}(C^{vec}_{Ik_1}, C^{can}_{Ik_1}) \\ \cdots \\ D^1_{Hm}(C^{vec}_{V1}, C^{can}_{V1}) + \ldots + D^{k_5}_{Hm}(C^{vec}_{Vk_5}, C^{can}_{Vk_5}) \end{bmatrix}$$

5. TRACKING THE USERS' ACTIVITIES AND MODIFYING THE CHAIN OF PREFERENCES

Dynamic Assembly Engine composes the course content according to the information about learner's profile delivered by the AMU (Activity Monitoring Unit), which produces the appropriate chains on the base of two sets of data. The first one usually comes from the questionnaire. The character of this information is quantitative – it consists of the numbers representing "involvement" of each recognized learning style therefore it has to be transferred into qualitative description corresponding to the course structure elements and their attributes. It must be stressed that this initial chain of user's description is indispensable for the system to start its work and therefore, when it cannot base on the results of a questionnaire (they have not been collected or are misleading) a preliminary set of preferences must be taken a priori. The values in the input chain can be averaged for instance (a chain contains only letters C) or chosen randomly.

The second set of information is the description of user's profile used at the previous step of learning process and stored in the system. The role of a mechanism compiling these two types of measures can be fulfilled by an appropriate fuzzy controller. As the output of such controller an updated fuzzy controlling signal (reference) is generated and send to DAE.

6. PREPARING THE CONTENT OF THE REPOSITORY

The indispensable condition for efficient work of the algorithm described above is appropriate structure of data stored in the repository. In other words it is necessary to describe what type of learning objects it should contain. Their form, type and size have already been defined when constructing the input chain described above (chapter 5). But apart from these values we also need to know the relationships between elements. Which of them can be combined or linked together on the one hand and, which of them imply the necessity of the others if that is the case. The IEEE Learning Object Metadata (LOM)[2] standard offers some commonly used items for description of typical learning objects grouped into several categories. The IMS learning design specification [4] allows adding to this description some pieces of necessary information concerning pedagogical aspects and learning objectives of learning objects. And last but not least, the idea of RDF (Resource Description Framework) [8] graphs proposed by W3C's Semantic Web Activity seems to be useful for describing the relationships between those objects. In general, such graphs consist of entities and properties. In this context the entities correspond to LOs and properties can be understood as relations between them as well as their characteristics and attributes.

7. CONCLUSIONS

---

[2] http://ltsc.ieee.org/wg12/

The problem of personalization in online learning remains in focus of attention of many researchers nowadays. There have been various attempts undertaken but only some of the solutions are used for teaching real courses. Sophisticated Web-based AH systems as well as IT systems are often oriented on one type of tasks, like quizzes or assessments for instance, and therefore cannot be used for other purposes. Moreover, their content is not shareable and that is also a real obstacle, which blocks their popularization. In this paper another approach has been presented. The authors decided to implement some personalization tools directly in LMS already used at the university. Personalization is based on learning styles theory and appropriate questionnaire has been adapted to the virtual environment. Collected initial data, as well as the information regarding the learner's activities performed during the learning process are stored in the system and then converted by a fuzzy controller into a chain for Dynamic Assembly Engine responsible for creating a personalized course content. The system is now in its experimental phase and it is foreseen that when it passes the simulation tests it will be implemented on the university e-learning platform.

REFERENCES

[1] Brusilovsky, P., *A Distributed Architecture for Adaptive and Intelligent Learning Management Systems*, In R. Calvo and m. Grandbastien (Eds.), Intelligent Management Systems Workshop, Supplementary Proceedings of Artificial Intelligence in Education, 2003, pp.106-114.

[2] Felder, R.M, *Matters of Style*, ASEE Prism, 6, 4, 1996, pp.18–23, http://www.ncsu.edu/felder-public/Papers/LS-Prism.htm

[3] García, P., Amandi, A., Schiaffino, S., and Campo, M., 2006. *Evaluating Bayesian networks' precision for detecting students' learning styles*, Computers and Education, 49, 3, 2007, pp. 794-808

[4] IEEE LOM, http://ltsc.ieee.org/wg12/

[5] Memletic Learning Styles Inventory (http://www.learning-styles-online.com) (viewed: 30.05.2007)

[6] Myers, I.B., McCaulley, M.H., Quenk, N.L., Hammer, A.L., *MBTI Manual: A Guide to the Development and Use of the Myers-Briggs Type Indicator*, 3rd Edition. Consulting Psychologists Press, Palo Alto, CA, 1998

[7] Paramythis, A., Loidl-Reisinger, S., *Adaptive Learning Environments and e-Learning Standards*, Electronic Journal on e-Learning, 2, 1, 2004, pp. 181-194.

[8] Resource Description Framework http://www.w3.org/RDF/

[9] Villaverde, J. E., Godoy, D., Amandi, A., *Learning styles' recognition in e-learning environments with feed-forward neural networks*, Journal of Computer Assisted Learning, 22, 3, 2006, pp. 197-206.

[10] Yannibelli, V., Godoy, D., Amandi, A., *Interactive Learning Environments*, 14, 1, 2006, pp. 55-78.

[11] Zajac, M., Piekarczyk, M., *Algorytmy metod uczenia się w systemie komputerowego wspomagania nauczania w oparciu o metodologię Reusable Learning Objects (RLO)*, Informatyka Teoretyczna i Stosowana, 3, 5, 2003, pp.167-177.

[12] Zajac, M., *Reusable Learning Objects – czyli jak efektywnie przygotowywać materiały do kształcenia online*, E-mentor – Dwumiesięcznik Szkoły Głównej Handlowej w Warszawie, 3, 2004, http://www.e-mentor.edu.pl