A FRAMEWORK FOR DEVELOPMENT AND MANAGEMENT OF E-LESSONS IN E-LEARNING

Azita A. Bahrami
Department of Information Technology
Armstrong Atlantic State University
11935 Abercorn Street
Savannah, GA 31419

Keywords: E-lesson's model, E-lesson Cube, E-lesson's View, E-learning, and Ontology.

Abstract The use and or re-use of the existing e-lessons for the creation of new ones make the e-learning both time and cost effective. To accomplish this, however, requires the removal of some obstacles first. This paper presents a framework for that purpose. The progression of the concepts leading to the framework includes the introduction of a multi-dimensional e-lesson model that leads to the construction of an e-lesson cube. This cube is the backbone of an e-lesson warehouse, which in turn is the main component of the proposed framework.

1 INTRODUCTION

In general, any "lesson", λ, includes a subject, S (topic), designed for a specific audience, A, in a specific field, F, of study utilizing a delivery system, D, for conveyance of the lesson. This can be expressed as λ = (S, A, F, D) with the subject's value of $\omega = \{o_1, o_2, . . ., o_p\}$, where os are a set of related and organized objects (sub-topics). This makes a lesson a four dimensional object which carries the value of ω.

An e-lesson, however, is only three dimensional because there is only one and the same delivery system for all e-lessons, namely, the Internet. As a result, an e-lesson can be viewed as a 3-D object, λ = (S, A, F), with a value composed of a set of digitized objects. For example, an e-lesson teaching the Subject "DNA" to the audience of university freshman students majoring in the field of biology and using teaching objects of enzymes, chromosomes, and genes would be noted in the following general form,

λ_1=("DNA", "University Freshman", "Biology") and

ω_1 = {Enzymes, Chromosomes, Gene}.

Each digital object in ω is an integral part of the e-lesson and may be composed of digitized text, data, image, sound, video, etc. or some combinations of them.

In the e-learning arena, DNA, for example, can be taught to the following groups of students: (1) university freshman in biology, (2) Ph.D. students in biology, (3) university seniors in computer science, and (4) university seniors in forensic medicine (Table 1.) Naturally, there would be four different sets of values for the e-lessons suitable for these groups encompassing different breadth, depth, and emphasis. In explaining the point, suffice it to say that a university freshman in the field of biology may need the general and basic ideas about DNA whereas a Ph.D. student of the same field may need an in-depth study of DNA for the purpose of acquiring deep knowledge of DNA's functions. A university senior in the field of computer science studying DNA, however, may need to learn not the DNA's functions but the replication of DNA strands and the "patterns" that they carry. And lastly, a university senior student in the field of forensic medicine may study DNA from the view point of criminal investigations.

A. Bahrami A. (2005).
A FRAMEWORK FOR DEVELOPMENT AND MANAGEMENT OF E-LESSONS IN E-LEARNING.
In *Proceedings of the First International Conference on Web Information Systems and Technologies*, pages 504-509
DOI: 10.5220/0001226705040509

Table 1: A set of e-lessons.

λ_1 = ("DNA", "Univ. Freshman", "Biology")
λ_2 = ("DNA", "Univ. PhD students", "Biology")
λ_3 = ("DNA", "Univ. Seniors", "Computer Science")
λ_4 = ("DNA", "Univ. Seniors", "Forensic Medicine")

Let us assume that the four versions of the e-lesson on DNA in Table 1 are already developed and in use. Now the development of a new e-lesson on the same subject for the freshman students of medicine is being attempted. Can some parts of the values from the existing four e-lessons be used for this fifth group of students? To answer this question, it would be beneficial to first lay out the ground work for an e-lesson development.

- **Time and Cost.** The most time consuming and costly part of an e-lesson development is the creation of digital content for an e-lesson because an e-lesson requires the valuable time of a field/domain expert in both the initial design-and-development and the subsequent modifications and or re-writing of the lesson for different audience in the same or different fields. Therefore, using the existing e-lessons whose glitches, for the most part, are already removed can shorten and lessen the amount of time and cost necessary for development of a new e-lesson.

  Though the original e-lesson developer can be consulted, if necessary, for any modification, the required time and cost would still be less than developing an e-lesson from ground zero.

  Furthermore, providing links to the existing e-lessons inside the new e-lesson can substantially shorten the e-lesson itself or, rather, expand the e-lesson beyond the boundaries of the intended objectives of the present lesson.

- **Success Rate**. Using the existing e-lessons or parts of them whose content validities have inevitably been already tested would improve the chances of success for the new e-lesson, and thus the e-learning experience.

To sum up the points made, it may be stated that an e-lesson can be constructed time-and-cost effectively through using, re-using, and referencing the existing lessons. To actualize such idea, however, requires the removal of some obstacles first. The obstacles are: (a) how to find the e-lessons of interest and (b) how to utilize the found e-lessons.

An architecture has been presented by Siqueira et al (Siqueira et al, 2002) that resolves the first obstacle. This paper, however, presents a framework for the removal of the both obstacles making the use, re-use, and reference to other existing lessons possible. The foundation of the framework is investigated in section 2. The framework itself and the relevant discussion are covered in section 3. The conclusion and future research are included in section 4.

2 THE FOUNDATION OF THE FRAMEWORK

An e-lesson has the following properties:

(a) **Dimensions.** An e-lesson is a 3D object, $\lambda = (S, A, F)$.

(b) **Value.** An e-lesson has a value, $\omega = \{o_1, o_2, \ldots, o_p\}$.

(c) **Owner.** The owner of an e-lesson has the right to his/her intellectual property and is the authority for granting permission and profiting from the lease of his/her property, if so he/she chooses.

(d) **Host.** A host is the computer in which the e-lesson resides.

(e) **View.** The information, metadata, regarding an e-lesson, which is within the "content" section of a meta tag inside the HTML source, is considered to be the view of the e-lesson. The metadata contains the actual descriptive-words about the value of the e-lesson including keywords, terms, phrases, etc. The structure of metadata for e-lessons is aimed to become standardized (Anderson et al, 1999, Hermans et al, 1999, and Hodgines et al 2002) and the EU commission initiative on e-learning and IEEE are the two major forces behind this effort. In order for e-lesson developers to create a new e-lesson, they need to query the views, manipulate the views, and map the views, all of which are addressed below.

2.1 Querying the Views

An e-lesson designer needs to query metadata of existing e-lessons of interest related to the e-lessons he/she intends to develop. As the result of not yet having a set of standard in place for metadata, some relevant information fail to return. An example of such case is when the designer queries about "ER Model" and may not have any return because the existing e-lessons may have presented their descriptive-words for the same concept as "E-R Model", "Entity Relationship Model", "Database Top-down Model" or other variations.

Quite clearly, this is a problem needing to be solved. The solution to this major problem is to create ontology. The ontology would represent the metadata items of e-lessons and their relationships with one another in a thorough and formal fashion. In other words, ontology is more than a multi-faceted taxonomy because it includes the descriptive-words of the metadata, their complex relationships, and the rules about how to specify descriptive-words and relationships (Neches et al, 1991). One of the relationships that is established among the metadata items from the view point of ontology is the "concept hierarchy". For example, an e-lesson designer looks for a descriptive-word, such as "Database Design" and receives no return because none of the existing e-lessons contain these descriptive-words in its meta tag. Yet, the descriptive-words such as "Universal Relation Model" and "ER Model" that are two well known approaches in "Database Design" could be found. The hierarchical relationship among the three descriptive-words "Universal Relation Model", "ER Model" and "Database Design" is identified and kept in the ontology that includes these three descriptive-words. Naturally, due to the large number of possible values for each dimension of the e-lessons and their combinations, there is a need for a library of ontologies along with a library interface.

2.2 Manipulating the Views

To develop a new e-lesson, the developer needs to group and re-group the existing e-lessons along the three dimensions of S, A, and F in various combinations. This can be accomplished through "e-lesson cube" along with the needed set of operations.

2.2.1 E-lesson Cube and Needed Operations

A set of existing e-lessons and their values are given, Table 2. The assumption is that the set of possible values for dimension audience is: university freshman students (UF), university sophomore students (USP), university junior students (UJ), university senior students (US), university master students (MS), and university Ph.D. students (PhD). Also, the set of values for the field dimension is: business administration (BA), which includes both computer information systems (CIS) and management (MG), computational sciences (COS), which includes information technology (IT) and computer science (CS), and electrical engineering (EE), which includes computer engineering (CE), electronics (EL), and power (EP).

Table 2: A set of existing e-lessons and their values.

λ_1 = ("ER Model", "MS", "CIS") $\omega_1 = \{o^1{}_1, o^1{}_2, \ldots, o^1{}_{p1}\}$
λ_2 = ("ER Model", "US", "IT") $\omega_2 = \{o^2{}_1, o^2{}_2, \ldots, o^2{}_{p2}\}$
λ_3 = ("ER Model", "US", "CS") $\omega_3 = \{o^3{}_1, o^3{}_2, \ldots, o^3{}_{p3}\}$
λ_4= ("ER Model", "US", "CS") $\omega_4 = \{o^4{}_1, o^4{}_2, \ldots, o^4{}_{p4}\}$
λ_5= ("Router", "UJ", "CS") $\omega_5 = \{o^5{}_1, o^5{}_2, \ldots, o^5{}_{p5}\}$
λ_6 = ("Router", "USP", "CE") $\omega_6 = \{o^6{}_1, o^6{}_2, \ldots, o^6{}_{p6}\}$
λ_7 = ("Channels", "MS", "IT") $\omega_7 = \{o^7{}_1, o^7{}_2, \ldots, o^7{}_{p7}\}$
λ_8 = ("Universal Relation Model", "UJ", CS) $\omega_8 = \{o^8{}_1, o^8{}_2, \ldots, o^8{}_{p8}\}$
λ_9 = ("Universal Relation Model", "US", "MG") $\omega_9 = \{o^9{}_1, o^9{}_2, \ldots, o^9{}_{p9}\}$

Let us only consider those lessons for which the subject is "ER Model" (i.e. λ_1, λ_2, λ_3, and λ_4.) Since these e-lessons have the same subject, they may be

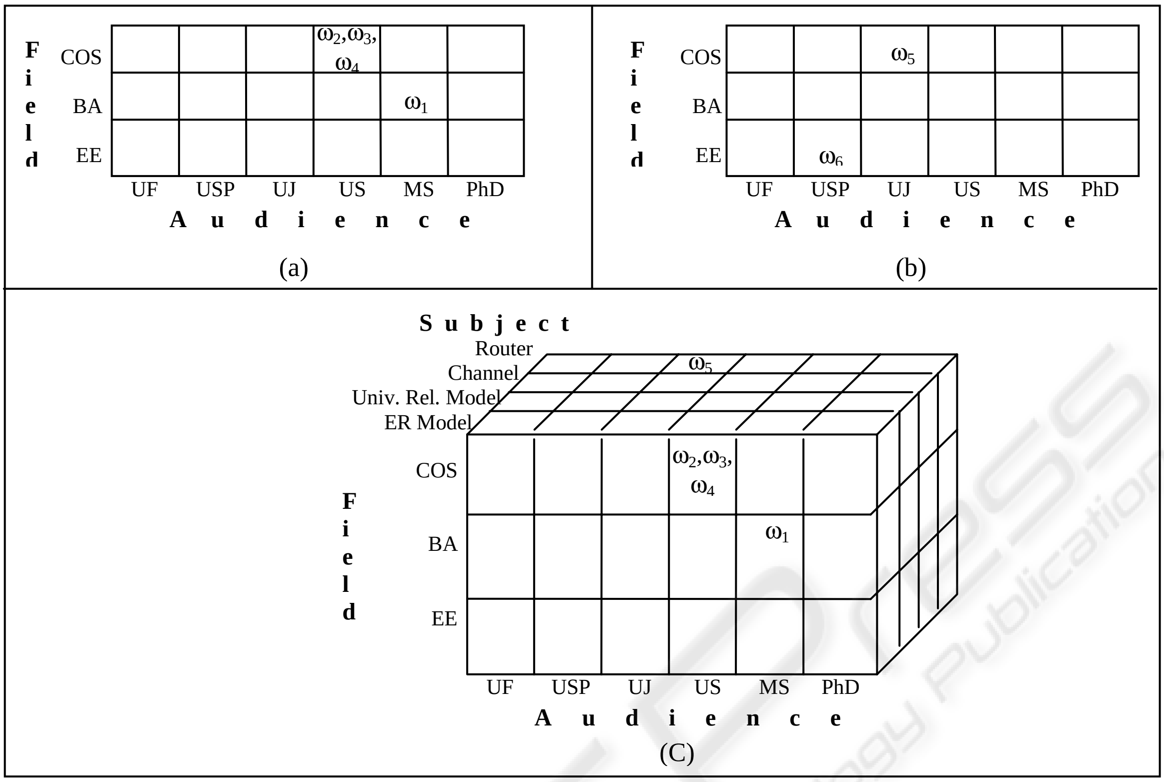

Figure 1: E-lesson Cube Formation: (a) 2-D model for those lessons of Table 2 with subject "ER Model", (b) 2-D model for those lessons of Table 2 with the subject "Router", and (c) the e-lesson cub for the lessons of Table 2.

shown in a two dimensional space, a table, Figure 1.a. The e-lessons for the subject "Router" (i.e. λ_5 and λ_6) are also shown in the table of Figure 1.b. Thus, for each subject a table can be produced. These tables collectively make an e-lesson cube, Figure 1.c. As a result, an e-lesson cube has three dimensions and a collection of values.

The question is through what specific operations the grouping and regrouping of the existing e-lessons are accomplished? The needed operations are as follow.

Roll-up operation. The values for any of the dimensions (S, A, F) can be collapsed to make a new value within the "concept hierarchy". For example, the values on the subject's dimension of the e-lesson cube of Figure 1.c can be collapsed into higher level subjects of "Database Design" and "Network". As a result, the tables for the subjects of "Universal Relation Model" and "E-R Model" are integrated to make a new table for the "Database Design" subject. The tables for the subjects of "Router" and "Channel" are also integrated to make the table for the "Network" subject.

Roll-down operation. The opposite of roll-up operation can be achieved by expanding one or more dimensions. For example, field's dimension may expand from (COS, BA, and EE) to (CS, IT, CIS, MG, CE, EL, and EP).

Slice operation. The e-lesson cube can be cut along only one of the three dimensions for a set of values on that dimension.

Dice operation. The e-lesson cube can be cut along more than one dimension for a set of values (on those dimensions). Dicing the e-lesson cube of Figure 1.c for Subject = (ER Model | Universal Relation Model) and Audience = (US | MS) creates a sub-cube for which S =(ER Model, Universal Relation Model), A = (US, MS), and F = (all the fields)).

Pivot operation. Any e-lesson cub, or results produced by any of the above operations can be rotated so that each dimension may become the focal point of observation.

The above operations can be delivered through a technology called On-line Analytical

Processing (OLAP) (Han et al, 2001, and Chaudhuri, 1997). For the ease of manipulation, the exiting e-lessons available to a developer may be stored in a data warehouse. Data warehouses are specifically designed and developed for multi-dimensional data and are used by OLAP technology. Therefore, a data warehouse is quite suitable for creating, storing, and manipulating the e-lesson cube model proposed in this paper.

2.3 Mapping the Views

The outcome of applying a group of operations on a warehoused e-lessons is a very large set of values for multiple e-lessons. The question is that how such a large volume of data may be used by a designer. To answer this question, it can be assumed that the set of lessons' values $\Omega = \{\omega_1, \omega_2. . ., \omega_n\}$ belonging to the set of e-lessons $L = \{\lambda_1, \lambda_2, . . . , \lambda_n \}$ is delivered by OLAP. Let the union of the digital objects of the e-lessons' values in Ω be $O = \{o_a, . . ., o_q\}$ and the union of the views (descriptive-words) for the lessons in L be $W = \{w_e, . . ., w_m\}$. There is a many-to-many relationship between the elements in O and W, which can be easily broken into two 1:n relationships and the results be stored in a relational database for the actual use by the designer.

3 THE FRAMEWORK

The proposed framework has six components shown in Figure 2. These components are, "E-lesson Warehouse", "OLAP Technology", "Library of Ontologies", "Library Interface" "Mapper", and "Controller". Each component is briefly described below.

E-lesson Warehouse. This is a warehouse of e-lessons collected by the designer composed of a set of smaller warehouses (**data marts**), each containing e-lessons of one subject only. The e-lessons of the warehouse are integrated from many heterogeneous data repositories that are either structured (databases) or unstructured (flat files) and are distributed.

OLAP Technology. This technology is capable of handling the operations "roll-up", "roll-down", "slice", "dice", and "pivot". For details of OLAP technology consult (Han et al, 2001, and Chaudhuri, 1997)

Library of Ontologies. This is a collection of ontologies used for different subjects in the subject domain.

Library Interface. This component has a dual function: (a) It helps the Controller by providing the descriptive-words related to the ones submitted by the designer through an OLAP command and (b) it facilitates the library's maintenance.

Mapper. This component provides a detailed mapping of the views and values on each other, using the sub-component **Indexer**, and stores the findings in a relational database. The creation, manipulation and maintenance of this database are also the function of the Mapper.

Controller. This is the component that (a) receives a query from the user, (b) examines the descriptive-words in the query for possible replacement using the "library of ontologies", (c) re-writing the user query, if it is necessary, and (d) Obtaining the answer to the query and returning it to the user. The Controller can also provide the user with any mapped views and values.

It was previously mentioned that to make the e-learning more time and cost effective, it is highly desirable to use, re-use, and reference existing e-lessons. It was also indicated that such an attempt requires the removal of two obstacles: (a) how to find the e-lessons of interest and (b) how to utilize the found e-lessons. The proposed framework of Figure 2, addresses both obstacles with solutions. To justify the solutions, let us discuss how the proposed framework provides for each obstacle.

How to find the e-lessons of interest. A designer wants to find all the e-lessons that teach, for example, the "Database Design" subject to a freshman audience. As soon as the "Controller" receives this request from the user, it consults the "library of ontologies" through the "library interface". The library determines the fact that the

"Database Design" subject is located in a. higher

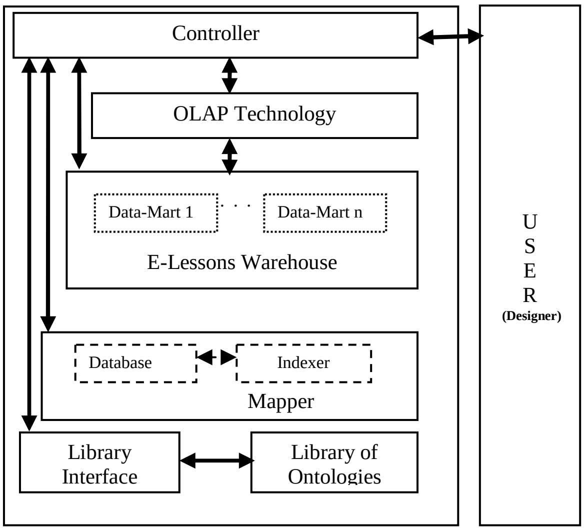

Figure 2: The Framework

level of "concept hierarchy" and is composed of a set of lower level concepts {ER model, Universal Relation model}. This set of concepts is communicated back to the Controller by the "library interface". The Controller modifies the user's query to include all three descriptive words "Database Design", ER model, and Universal Relation model. The new query is implemented by the OLAP that finds all the e-lessons of interest and returns them to the user through Controller.

How to utilize the found e-lessons. The mapping of the views onto the e-lessons' values, by the Mapper component of the proposed framework, makes it possible for the designer to use the e-lessons' values in a constructive way. The creation and maintenance of a database for the mapped values enable the designer to retrieve any relationship between the views and the values on-line. In fact, the Mapper provides a powerful filter for isolating the needed materials among high volumes of the e-lessons' values delivered by an OLAP operation.

4 CONCLUSION AND FUTURE RESEARCH

The proposed framework makes it possible for a designer to use, re-use and or reference the existing e-lessons. The great benefit of using the framework is that the time and cost of development a new e-lesson lowers, making the entire e-learning experience time and cost effective. The implementation of the proposed framework and the investigation of its behavior are the focus of the future research.

REFERENCES

Siqueira S. W. M., Braz M. H. L. B., and Melo R. N., "E-Learning Content Warehouse Architecture", Proceedings of the International Conference of WWW/Internet, pp. 739-742, Nov. 2002, Lisbon, Portugal.

Anderson, T, McArthur D., Griffin S., and Wason T., "IMS Learner Resource Meta-data Best Practices and Implementation Guide v.1", Revision: August 1999, http://www.imsglobal.org/metadata/mdbest01.html.

Hermans, H. J. H., Koper E. J. R., Loeffen A., Manderveld J. M., and Rusman E.M., "Reference Manual for Edubox-EML/XML binding 1.0/1.0 (Beta Version)", http://dspace.learningnetworks.org/retrieve/122/EML1.0-ReferenceManual.pdf

Hodgins W., (Chair), Duval E. (Technical Editor), "Draft Standard for Learning Object Metadata IEEE P1484.12.1/D6.4", July 2002, http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

Neches R., Fikes R., Finn T., Gruber T., Patil R., Senator T., and Swartout W.R., "Enabling technology for knowledge sharing", AI Magazine, Vol. 12 , Issue 3, pp. 36-56, 1991.

Han J., Kamber M., "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, 2001.

Chaudhuri S. and Dayal U., "An overview of Data Warehousing and OLAP Technology", ACM SIGMOD Records, No. 26, pp. 65-74, 1997.