Educative Resource Patterns Presentation in a Model-based Instructional e-Learning System Design Environment

Habib M. Fardoun[1], Abdulfattah S. Mashat[2] and Daniyal M. Alghazzawi[1]

[1]Information Systems Department, King Abdulaziz University (KAU)
Jeddah, Saudi Arabia
[2]Department of Information Technology, King Abdulaziz University (KAU)
Jeddah, Saudi Arabia

Abstract. In this paper we discuss the resource patterns, its presentation and interaction in a model-based instructional e-Learning system development using eLearniXML notation. In order to achieve productivity and agility in instructional e-Learning system development the resources patterns are introduced. Also it is clarified how a presentation model is applied by using a set of interaction resource patterns elements. In this research work a catalogue of patterns is identified, documented and presented. These resource patterns are set of interaction elements, they are represented by using abstract interaction objects, and they are patterns because they appear again and again in instructional systems.

1 Introduction

The modelling of a learning process under Model-based Instructional e-Learning System Design Environment Mb-ISDE starts with the specification of the task and the domain models [1]. These models set the starting points for other models that will be described during this paper.

As our study is focused on e-Learning education and the creation of e-Learning systems, next to the educative department of the University of Castilla-La Mancha, Spain (UCLM) we started to look to find a mechanisms to design learning materials, and we propose a catalogue of patterns, called resource patterns, to create a technological activities that supports many of the traditional activities of teachers and students in the learning/teaching process. Where, in our methodology a resource pattern is a link between domain, task and presentation model. The approach presented in our research work is based on the belief that one of the most critical factors of successfully e-Learning with face-to-face learning is making situated and targeted, thus deliberate use of learning technology. Our purpose it that, by using these patterns the teacher will control his activities and how his students work. He has the opportunity to see (check) the software as an all, so he controls the teaching and learning environment which are to the way they work and to their knowledge.

This paper is part of a continuous research works, being presented in the last years

Fardoun H., Mashat A. and Alghazzawi D..
Educative Resource Patterns Presentation in a Model-based Instructional e-Learning System Design Environment.
DOI: 10.5220/0004095800260036
In *Proceedings of the 1st International Workshop on Interaction Design in Educational Environments* (IDEE-2012), pages 26-36
ISBN: 978-989-8565-17-4

to describe a new education system using eLearniXML notation, [1] [2] [3] [4]. Our focus, in this paper, is over the tasks the teacher has to carry out in order to leverage the learning experience of the students. To accomplish this goal, we define a set of tasks and concepts that provides the teacher with a set of learning objects that support either, individual and group learning [5]. These learning objects are described in terms of design patterns, and supported by model-driven environments encouraging use of models that are platform independent. The user interface of our approach is based on UsiXML framework specification language for user interface design [6]. This language allows us to describe user interfaces at different level of abstraction in order to specify the user interface of the different learning objects in terms of their functionality according to the task and domain models. And since, our applications are not general, UsiXML must be adapted. We are interested on e-Learning applications and for this specific domain we proposed modifications in UsiXML and a new modified language for Mb-ISDE, which is eLearniXML notation.

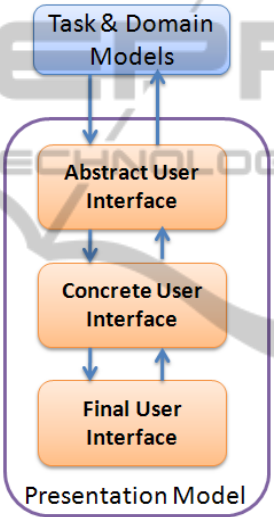

Fig. 1. Cameleon reference Framework, Cameleon Project [7].

Our methodology is based on three abstraction layers defined within the Cameleon project [7], see Fig. 1. These layers are the abstract, the concrete and the final user interface. By using interface models for the development of the system, instead of final code, eLearniXML contributes to more agile and flexible e-Learning system development, improving the quality of the generated systems in a systematic way. This method fosters the separation of concerns between the contents to be introduced to the students and the way they are presented. By doing so, portability and maintainability quality factors are greatly improved [2]. Moreover, this separation of concerns allows for a more systematic and clear content development process. In this sense, eLearniXML is aimed at the development of high quality instructional systems. By using a model-based approach, most of the facets in the development of this kind of systems are improved, in terms of flexibility and productivity. Therefore, our approach contributes to provide a more systematic approach to instructional systems

development, and it provides the basis to include the development of such systems according to the current trends in software of model-driven development.

The article is organized as follows: Section 1 exposes a catalogue of patterns for e-learning system development. Then, Section 3 exposes our three groups of resource patterns, presented in this paper, obtained from studies of the conduct of the teacher and his students in a classroom and, Section 4 describes in detail each one of these resources patterns. Section 5 describes how to develop the eLearniXML by Model-Based Instructional E-Learning System Design Environment (MB-ISDE). Finally, Section 5 exposes conclusions and future work

2 Resource Patterns Documentation

Patterns are used over a wide area of software design. Probably most famously, software design patterns were popularized and catalogued by [8], and these have had a significant influence on software design since their publication. Software design patterns address a particular part of software development that is below the level of software system design but above that of program design. Subsequently, patterns have been published that cover the entire scope of software development.

As mentioned earlier, one of the first incursions of the patterns in the field of computer science was in the programming community. The most important publication in this field was Design Patterns: Elements of Reusable Object-Oriented Software [8] which describes solutions to common problems in software design.

The elements that our patterns adapt is a combination between those proposed and described by [9] and GoF pattern [10] which is, in his case, a description of communicating objects and classes that are customized to solve a general design problem in a particular context. Each design pattern focuses on a particular educative problem or issue and is described using the following structure:

- Pattern Name and Classification: The pattern name is an identification that can be used to describe a design problem; it contains the name of the pattern and its classification. A handle we can use to describe a design problem and its solutions in a word or two. Both are important to have brief information of the pattern identification and to which group it belongs.
- Problem: Explain the problem and its context, a scenario that illustrates a design problem. What are the situations in which the design pattern can be applied? (When to apply the address? How can you recognize these situations? It could describe structures of user interfaces that have an inflexible design or a list of conditions that must be met before applying the pattern.
- Solution: The solution describes the elements of design, its relations responsibilities and collaborations. The solution describes neither a design nor a specific or particular implementation, because the pattern is like a template that can be used in different situations. Provides an abstract description of a design problem and how objects are arranged to fix. It allow the actors a clear description of the proposed solution (other patterns may be required to complete the full solution of the problem).

- Structure: a graphical representation of user interface models (at an abstract level), and the mapping process between them.
- Participants: the actors (teacher, student) that participate in the design of patterns and their responsibilities.
- Sample Code: Code fragments that illustrate how you might implement the pattern in XML, ELearniXML.
- Use When: a description of the situation in which the pattern can be used, what are the characteristics of the context, in terms of tasks, the actor/s.
- How: how the teacher and student in the pattern solve the problem. The scenario will help you understand the more abstract description of the pattern that follows.
- Why: the explanation of why a pattern must be used in a specific situation and the benefits it would suppose to the actors pattern). What are examples of poor designs that the pattern can

The consequences are the results of applying the pattern. Even though the consequences are between the lines when describing the design decisions, it is critical the evaluation of design alternatives to understand the costs and benefits of applying the pattern. The consequences are usually given in terms of space and time, although they may be including questions of language and implementation. From the point of view of user interface design it can be mentioned the impact on flexibility, usability and functionality in different systems.

2.1 How Resource Patterns should be Applied

Several of the patterns we present above are available in e-Learning systems; for example, a debate forum which is the implementation of a pattern presented in Communication Resource Patterns. The weakness is, however, that most systems only give the functionality of a debate forum, while the methodical knowledge and experience of how to use it in a learning situation is lacking. The result then must be the way to how to moderate an online discussion. The expertise, of online teachers, should instead be implemented into the system. Another weakness is often a lack of communication between online teachers, up until now everybody makes the same mistakes, instead of implementing the solutions into the system and so learning from each other.

To make the design process more effective we need to have tools to simplify the process. The tools should make it possible to develop systems with all the features and opportunities that we want, which is to include the four aspects of Fig. 3. Carstensen and Schmidt see flexibility as one specific challenge for CSCW-systems design; we have to establish basic building blocks and platforms so that the actors themselves can establish a CSCW system fulfilling their needs [11]. Resource patterns can be examples of building blocks to ensure such flexibility in e-Learning systems. Resource patterns can work as one kind of e-Learning design tool. These patterns are useful tools that make it easy to share the e-Learning expertise learned from past mistakes since resource patterns are archetypes on well-used solutions. Resource Patterns will build expertise of experienced online teachers into the system, and help novice online teachers and students learn how to work online see Fig. 2.

The usage of resource patterns will lead to a systematic approach to e-Learning design, where the teachers do the work of collecting the resource patterns and the students make use of it as a solution of the problem. Resource patterns will be useful and it will provide a common language for teacher and students. Resource patterns are a collection of tested teaching techniques methods. Some patterns will work in several categories, e.g. a pattern of discussion; will be useful to communicative as collaborative learning and problem solving.

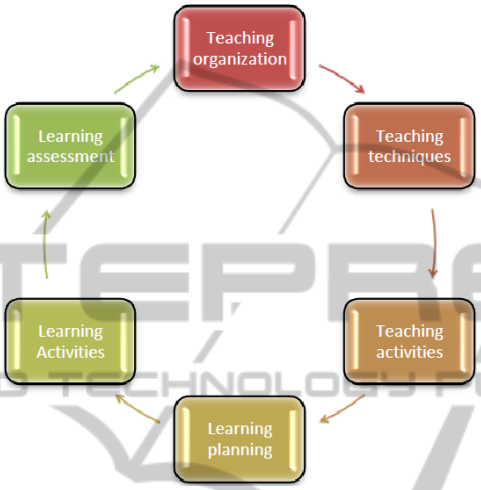

Fig. 2. The four focus points of our e-Learning System.

Next we present how resource patterns solve the found problems. Instructional design methodologies favour many different approaches. We, as e-Learning system designers could single out the nouns of a problem statement and define the corresponding instructional tasks and actions. The verbs in the statement would become the learning operations. Or we could concentrate on the responsibilities in our system. Or we could model the real world and translate all subjects and their actions into your design as e-Learning user interfaces. There is always disagreement on which approach works best and sometimes one approach is better suited to a problem, than the others. The last approach, modelling the teaching / learning world often works. However, when we are faced with a problem where our instructional system design ends up with no presentation as counterparts for the teaching/learning aspects, one might be in trouble if there is no guide. Such a guide is a resource pattern. When modelling e-Learning systems becomes difficult, a resource pattern helps us to identify the needed aspects - even if they are low level presentation like select properties, priority. If our strictly model the teaching/learning world, your design might not be flexible enough. For example, a instructional system that represents the use of a calendar, agenda and exercise is not found during the analysis phase. It's found later when one wants to make the design more flexible and reusable.

After finding what type of holes are it that sits in our design. Since we are looking for a resource pattern to fill these holes, we describe the problem as it relates to the three types of resource patterns; cooperation, coordination and communication. The

goal here is to identify and narrow down the list of possible resource patterns.

Once we have worked out what type of pattern we are looking for we need to delve into our design a little further and decide what is our primary intention is if that hole were to be filled. Taking an example, a teacher while he is presenting a lesson to his students, he pulls out a large amount of information of the lesson organization and planning as the information that each activity presents and ultimately wanted to convert this information to XML, keeping any hierarchical information and activities aspects that was present in the lesson. These hierarchies and activities would then be used to define the level of nesting in the xml structure.

The lessons retrieved would not always be the same and the xml structure would have to reflect this. Having ruled out using a persistence framework like hibernate or abates knowing that he could implement a much lighter solution, he then proceeded to create simple mapping files for the tables he was pulling out. This is where he found my hole. How do the structure and store this information in a manner that will reflect the final xml structure? To cut a long story short, the lesson organization planning he modelled with eLearniXML.

2.2 Relationships among Resource Patterns and others Pattern Languages

Our pattern language as a whole is part of a larger network of pattern languages. When describing the language, we have already mentioned areas like collaboration, communication and design support, that would well fit into the pattern language but which are not part of this thesis. Fig. 3., extends the layers of our pattern language with complementary pattern languages. We have linked our patterns to patterns from the Designing Interfaces language wherever the style of interface is important for better support of computer-mediated interaction. We also considered Jan Borcher‘s Pattern Approach to Interaction Design [12]. This book explains how patterns can support the construction of interactive devices.

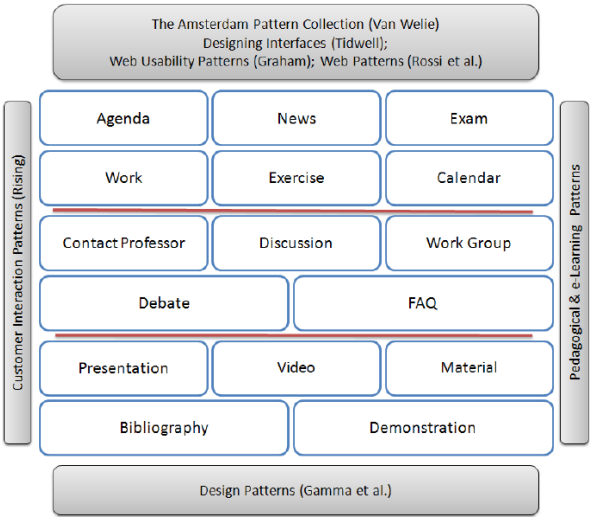

Fig. 3. Presentation of other languages related to our pattern language.

When talking about pattern languages that address technical problems related to the creation of collaborative systems. The History Patterns collection [13] describes practices for logging and restoring previous states of a system. Logging is an aspect that is relevant for the provision of awareness in collaborative systems. Teachers/students have to understand how a collaboration learning environment evolved over time to be able to situate their own tasks in the current context. Our Mb-ISDE interaction resource patterns in education should regulate the control between the student and the system, accommodate real-life tasks and their solution methods which are rich in feedback and provide interactive illustrations supporting conceptual understanding, learner controlled inspections and problem solving. Through the structured procedure capturing systems which offer simple devices, perceptually reflective learning on concrete items can be achieved.

3 Resource Patterns, eLearniXML and MB-ISDE

After identifying a set of interaction elements for the performance of an educative process, we grouped them into different groups, which we called interaction resources patters in education, depending on the type of service that these elements provide to solve a certain educative need or situation. Later we proceed to incorporate these resource patterns into an e-Learning system, to see how they would work, and which changes we must apply to make them provide the same service, as if this educative class is given face-to-face so students will not feel that anything is changed to them neither the lack of any regular service they are used to use.

3.1 Resource Patterns and eLearniXML

ELearniXML language which we propose as a graphical notation is an adapted XML language, based on UsiXML language, to specify e-Learning systems. That interactive learning system with different types of interaction and teaching techniques, modalities of use, and computing platforms can be described in the manner that preserves the design independently from peculiar characteristics of physical computing platform.

An eLearniXML specification is described using a series of models that represent the models most commonly used in the development of user interfaces based on models, and additionally some models that can represent possible transformations that can be applied on a model. ELearniXML is based on the principles of modelling MDA (Model-Driven Architecture) and Mb-UIDE process (Model-Based User Interface Development System) which we used to specify the steps and types of models in the development of user interfaces for interactive applications. In this paper, we present a pattern-based solution to address the model-based interface development using the eLearniXML notation. So we make use of patterns in the development process to help teacher and students in the teaching / learning process. Patterns are used when we want to write models using eLearniXML. A pattern is a general planning principle, which states a problem that may occur repeatedly in the environment that is helpful

again and again in eLearniXML. When we build classes, lessons, and topics patterns are used to represent the used tasks and relationships between them. Patterns can be gathered using eLearniXML, see Fig. 4.

We presented this solution of eLearniXML with educative patterns, because after our study and research we haven‘t found any other notation that can be easer to the user in the e-Learning environment, in this case the teacher by creating the content and organizing it, and the student in the other side by the student to absorb this information and interact with it.

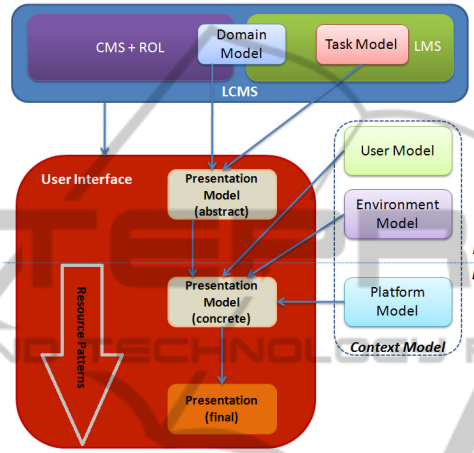

Fig. 4. Mapping between components of an e-Learning system and Model-Based Models, and resource patterns distribution in eLearniXML.

The starting point for the modelling process, of an educative topic by applying it CAT techniques, are the specification of both task and domain models. These models are the cornerstones for the development of the rest of models. Therefore, our interest in the requirements of the application is first focused on the tasks the teacher will carry out by using our system, and the concepts that will be presented to the students so they can perform their tasks. The way, in which these concepts and tasks are presented, to provide a better learning experience will be designed in next papers.

3.2 Using Resource Patterns in Our Mb-ISDE

In our Mb-ISDE methodology, eLearniXML, we are considering two main cornerstones: models and e-Learning components. These elements are integrated in a common framework; it is anticipated in the Fig. 5. as depicted in that Figure, models and e-Learning components (resource patterns) can be combined in order to provide a more homogeneous ISD development environment. In last papers [1],[2],[5], during the process of the research work we presented the learning technique approach we adopt and the needed tasks to perform them, as the resource patterns required to carry out this work. And finally we present the development process of this system which is the model-based development approach following an instructional system design

process which is a specified ADDIE model. Our Mb-ISDE study a model-based development of instructional e-Learning systems, it proposes a set of models of different themes and different level of abstraction, including, the task, domain models, presentation. The actors use eLearniXML notations, in a higher level of abstraction, to specify these models or declarative descriptions of the interface. The identified resource patterns and the used models are distributed throughout the development of the instructional system design process. While the resource patterns are reflected in during the presentation models as it is shown in the patterns catalogue, in the above section. The final relation between Mb-ISDE, the models and the resource patterns is represented in Fig. 4.

4 Conclusions

We started this paper by making a brief overview of the used technologies in e-Learning. From one side we did that, to remember the way in which they work and what actions and services they give the user. And from the other side, as an introduction to the second section of this paper, to see if they cover all or part of the needed actions by the teacher and student during a learning / teaching process.

As a result of our study of a normal day in a face-to-face classroom of an educational level, randomly elected, we found that there is a set of interactive elements in the learning / teaching environment that are being used repeatedly during the educative process. We collect these elements and divided them into several groups (coordination, communication and cooperation), depending on the type of activity that they provide its services. We convert these elements from face-to-face education to e-Learning one, and we call them interactive resource patterns in education [4]. These patterns, resource patterns, are related between each other, and they are used with other patterns to provide the needs of a certain educative action. For that we presented a tree that presents the existed relation between these patterns to make the user understand better the way in which he can make use of these patterns depending on the educative action he wants to perform. These relations it helps the teacher while he wants to create the teaching process, and gives him an idea about which patterns he must use. Once we have defined these patterns and the relation they have between each other‘s, we incorporate them to the eLearniXML methodology and we found that these patterns are the perfect elements that provide the mapping between the eLearniXML models (task, domain and presentation) so that they can be specified correctly for an e-Learning system. And finally, by including these patterns into the Mb-ISDE approach, we found that its released is done easier near to the models are developed more systematically and in an instructional way which enrich our objective of the development of learning / teaching systems.

We conclude with this paper, that these patterns are just set of huge number of interactive elements that can be used in the creating of learning / teaching process. It depends, between other reasons, on the needs of the lesson, topic to be taught, as on the available resources. The work presented in this paper of the relation between patterns, based-models development and instructional design process demonstrate the

capacity of creating an e-Learning system, which contemplate all the used actions in face-to-face classroom and can be performed in any platform.

References

1. Fardoun, H., Montero, F., Jaquero, V., 2009a. Diseño de sistemas de e-Learning para el soporte de nuevas técnicas de enseñanza. Proc. of X Congreso Internacional de Interacción Persona-Ordenador Interacción'2009 (Barcelona, Spain), 2009.
2. Fardoun, H., Montero, F., Jaquero, V., 2008. eLearniXML: Hacia el Desarrollo de Sistemas de e-Learning Basados en Modelos. Proc. of IX Congreso Internacional de Interacción Persona-Ordenador Interacción'2008 (Albacete, Spain), pp. 351-360. ISBN: 978-84-9732-596-7.
3. Fardoun, H., Montero, F., Jaquero, V., 2009b. eLearniXML: Towards a model-based approach for the development of e-Learning systems considering quality. Advances in Engineering Software 40, 12 (December 2009), 1297-1305. DOI= http://dx.doi.org/10.1016/j.advengsoft.2009.01.019
4. Fardoun, H., Montero, F., Jaquero, V., 2010. Designing e-Learning Systems to Support new Teaching Techniques. Journal of Computer Science and Engineering Volume 2, Issue 2. August 2010. ISSN: 2043-9091
5. Fardoun, H., 2011. PhD Thesis. ElearniXML: towards a model-based approach for the development of e-learning systems. University Castilla-La Mancha.
6. Limbourg, Vanderdonckt, J., Michotte, B., Bouillon, L., & López Jaquero, V., 2005. UsiXML: a Lan-guage Supporting Multi-Path Development of User Interfaces. En 9th IFIP Working Conference on Engineering for Human-Computer Interaction. EHCI-DSVIS'2004 (págs. 200-220). Springer-Verlag.
7. Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L., & Vanderdonckt, J., 2003. A Unifying Reference Framework for Multi-Target User Interfaces. Interacting with Computers. Vol. 15 (3), 289-308
8. Gamma, E., H., R., Johnson, R., & Vlissides, J. 1995. Design Patterns: Elements of Reusable Object-Oriented Software. Addison Wesley.
9. Van Welie, M. 2007. Interaction Design Patterns. http://www.welie.com/patterns/index.html
10. Cechich, A., & Moore, R. 1999. GoF: A Formal Specification of Gof Design Patterns.
11. Carstensen, P., & Schmidt, K. 2002. Computer supported cooperative work: New challenges to systems design. in: "Handbook of Human Factors". K. Itoh (ed.).
12. Borchers, J. 2001. A Pattern Approach to Interaction Design. John Wiley and Sons Ltd.
13. Anderson, F. 2000. A collection of history patterns. . En N. F. In Harrison, Pattern Languages of Program Design 4, (págs. pp. 263–297). Reading, MA, USA: Addison-Wesley.
14. Alexander, C., Ishikawa, S., & Silverstein, M., 1977. A Pattern Language: Towns, Buildings, Construction. New York: NY: Oxford University Press. Alexander, Ishikawa, & Silverstein, 1977;
15. Avgeriou, P., Papasalouros, A., Retalis, S., & Skordalakis, M. 2000. Towards a Pattern Language for Learning Management Systems. Journal of Educational Technology & Society, 6 (2), 11-24.
16. Chen, C.-T., Cheng, Y.-C., & Hsieh, C.-Y. 2007. Towards a pattern language approach to establishing personal authoring environments in e-learning. Sixth Lasted International Conference. Chamonix, France.
17. Anacleto, J. C., Neto, A. T., & Almeida Neris, V. P. 2009. Cog-Learn: An e-Learning Pattern Language for Web-based Learning Design. eLearn Maganzne.

18. Reeves, T. 1997. Established and emerging evaluation paradigms for instructional design. En Instructional development paradigms (págs. 163–178). Englewood Cliffs, NJ: Educational Technology: In C. Dills & A. Romiszowski (Eds.).
19. Nielsen, J. 1995. Card Sorting to Discover the Users' Model of the Information Space. Obtained from: http://www.useit.com/papers/sun/cardsort.html