MAKING LEARNING DESIGNS IN LAYERS: THE CADMOS APPROACH

Mary Katsamani
Department of Digital Systems, University of Piraeus
M.Karaoli & A.Dimitriou 80, 18534
Piraeus, Greece
marykatsamani@gmail.com

Symeon Retalis
Department of Digital Systems, University of Piraeus
M.Karaoli & A.Dimitriou 80, 18534
Piraeus, Greece
retal@unipi.gr

ABSTRACT

There are various tools that teachers can use today in order to design a learning unit in a formal and reusable way. This paper gives an overview of the various features that current learning design tools support and discusses the required ones that will make these tools more appealing to the teachers. Factors that discourage teachers from using these tools are the lack of both user friendly graphical notation and the non support of a layered approach to learning design. Thus this paper proposes to build learning design tools following the concept of the "separation of concerns" that stems from the principles of web engineering. A graphical, IMS-LD level A/B compliant learning design tool, called CADMOS, which supports the "separation of concerns" concept in learning design, is being presented.

KEYWORDS

Learning design, learning design tools, separation of concerns concept, CADMOS model

1. INTRODUCTION

A "learning design" (LD) is the outcome of a design process which is both art and science. A teacher, using his/her experience and creativity, may create a learning design for a simple activity, for a course lasting one or a few hours, for a course lasting a few weeks or even months or for a curriculum, meaning a whole year teaching programme [Britain, 2004; Goodyear, 2005]. Teachers are now seeking not only models such as the Caladine's Learning Activities Model (LAM), the AUTC, the T5 and more recently the ELM, IMS LD and LAMS model [Botturi & Stubbs, 2008; Brown, 2006], that could guide them during the learning design process but also tools that could make this process easy and error prone. Research studies stress that it is very helpful for teachers (i.e. teachers as designers) to create learning designs by "dragging and dropping" activities, putting the items in a sequence and linking them to learning objects and learning tools [Laurillard, 2008; Paquette et al., 2006].

As Lejeune et al. (2009) pointed out, teachers will make a choice among the available tools based on criteria such as:

- Comprehensibility/usability
- Pedagogical neutrality
- Flexibility
- Interoperability

A teacher, with the aid of a learning design tool is called to orchestrate the learning activities that s/he thinks the students should perform in order to accomplish the desired learning objectives following the principles of a learning strategy. S/he might also need to specify the learning objects, tools and services that

will be related to these activities. The teacher should also be able to determine in which order the students should perform the activities and any conditions, preconditions or rules that might exist. Additionally there must be flexibility when creating the leaning design. A teacher should be able to revise the design and add activities if s/he thinks that so far hasn't been fulfilled the scope of the course or remove an activity if s/he thinks that eventually it doesn't provide something to the learning process or change the rules or the execution order of the activities.

The aim of this paper is to present a new approach in learning design which is supported by a graphical LD editor named CADMOS which meets the aforementioned challenges at the design phase, i.e. to allow educational practitioners to build, revise and visualize the desired learning designs, without the need of technical skills. This new approach is based on the principles of "separation of concerns" which is a well known technique in web engineering [Papasalouros et al., 2004; Rossi et al., 2008]. The structure of the paper is as follows. First we make a brief comparative analysis between learning design languages and learning design tools in order to highlight the requirements from a new LD approach. Then we present the underlying philosophy based on which CADMOS tool was developed. CADMOS tool was evaluated by learning designers, most of whom were experienced teachers who attended a MSc course on e-learning design. This evaluation process and its evaluation findings are presented in the paper. Finally, the paper ends with some concluding remarks and the plans for future work.

2. LEARNING DESIGN LANGUAGES vs VISUAL LEARNING DESIGN TOOLS

Learning design languages and visual learning design tools have been proposed to aid teachers during the learning design task. On the one hand the LD languages (e.g. E2ML, coUML, PCeL, PoEML) advocate the use of a specific notation (symbols and rules) for creating a design without the support of a specialized learning design tool teacher. On the other hand, the existing LD tools such as COMPENDIUM, MOT+, LAMS, COLLAGE and OPENGLM are based on specific design principles and philosophy [De Vries et al., 2006]. Obviously, languages and tools have advantages and disadvantages that determine their widespread usage.

Taking for granted that teachers need usable means for creating their learning designs, we have studied LD tools and languages and drew some conclusions about their characteristics which teachers might find useful when choosing among them. These design characteristics had been followed when creating the CADMOS LD tool that is presented in this paper.

2.1 Learning Design Languages

Learning design languages (e.g. E2ML, coUML, PCeL, PoEML) offer a specific notation style for creating a design without the support of a specialized learning design tool. Often the teacher faces problems in creating a learning design using the symbols and rules of a LD language because there are no supportive tools that could offer guidance. Table 1 briefly shows the results from a comparison of the four most popular languages according to the following criteria suggested by Luca Botturi et al. (2006): user skills, formalization, stratification, elaboration, perspective and notation.

Table 1. Comparison of learning design languages

CRITERIA	E2ML	PCeL	coUML	POEML
User Skills	Novice	Medium	Medium	Expertise
Formalization	Semi-Formal	Semi-Formal	Semi-Formal	Formal
Stratification	Flat	Layered	Layered	Layered
Elaboration	Conceptual	Conceptual	Conceptual	Implementation
Perspective	Multiple	Single	Multiple	Multiple
Notation	Visual	Visual	Visual	Visual

From the above comparison we notice that E2ML is the only language that is suitable for teachers with basic knowledge of learning design and basic computer skills. PCeL and coUML demand intermediate skills, while POEML demands users who know well the language. E2ML, PCeL and coUML are semi-formal

languages while POEML is formal. From the point of stratification, only E2ML is flat, which means that the learning activities are presented only from one point of view and not in different layers as the other languages do. POEML is also the language that helps the user design the learning activities in detail, while the other languages are more conceptual. On the other hand, PCeL is a language that presents the activities only from one perspective.

2.2 Visual Learning Design Tools

As opposed to learning design languages, learning design tools offer a structured environment that guides teachers during the design process. A specific category of tools which are becoming very popular provide graphical interfaces that facilitate teachers to create their lessons, usually through a drag-and-drop process. Such LD tools are COMPENDIUM, MOT+, LAMS, OPENGLM and COLLAGE each of which are based on specific design principles and philosophy [De Vries et al., 2006]. Table 2 briefly shows the results from the comparison of five popular LD tools according to the following criteria: user skills, guidance, templates/design patterns, export IMS-LD A, B, C, editing of a ready-made IMS-LD package.

Table 2. Comparison of graphical learning design tools

CRITERIA	COMPENDIUM	MOT+	COLLAGE	LAMS	OPENGLM
User Skills	Medium	Expertise	Medium	Medium	Medium
Guidance	-	-	+	-	+
Layered Approach	-	-	-	-	-
Templates / Design Patterns	+	-	+	-	+
Exports IMS-LD A,B,C	-	IMS-LD LEVEL A	IMS-LD LEVEL A	IMS-LD LEVEL A	+
Editing of IMS-LDs	-	-	-	-	+

The comparison in Table 2 shows that MOT+ must be used by teachers with specialized knowledge of the learning design (LD) standards. In particular MOT+ demands its users to know well the underlying modeling approach. On the contrary LAMS, COMPENDIUM, COLLAGE and OPENGLM can be used by teachers with intermediate level knowledge of the learning design (LD) standards.

COLLAGE and OPENGLM offer guidance to the teacher during the design process. In COLLAGE the teacher decides which learning strategy to use (in the format of learning activity flow patterns) and then follows a set of steps on how to organize the course. In OPENGLM the user chooses to add a new activity in the design and the tool guides him/her to edit specific properties.

COMPENDIUM, COLLAGE and OPENGLM have embedded learning design patterns that offer guidance to an inexperienced teacher. MOT+, COLLAGE and LAMS allow the user to export a design into a IMS-LD Level A format, while OPENGLM creates IMS LD level A, B and C compatible designs. COMPENDIUM is not IMS LD compliant. Finally, only OPENGLM offers the teacher the possibility to import an existing IMS-LD lesson for editing.

From the above comparisons, one can identify a set of criteria that an "ideal" LD editor should meet. A LD editor should

1. be appealing and usable to novice designers who do not have specialized knowledge of the learning design (LD) specification (User skills)
2. guide the teacher during the design process (Guidance)
3. contain templates or patterns, (Templates/Design Patterns)
4. allow the teacher to design learning activities from different perspectives and in different layers (Stratification, Perspective)
5. have visual notation (Graphical Notation)
6. have a formal underlying model (Formalization)
7. extract designs according to the IMS-LD specification (Exports IMS-LDs)
8. allow editing of IMS-LD compatible learning designs (Editing of IMS-LDs)

One idea for creating a new tool that might meet most (if not all) of the above criteria is to adopt principles from other disciplines such as programming and web engineering. The proposed LD approach is based on the concept of "separation of concerns". In the next sections, we are presenting this approach which is also supported by a graphical LD editor called CADMOS visual LD editor. We are not advocating that CADMOS is an ideal tool. Actually, in its current version, it supports only IMS-LD level A & B designs. The motivation

of our work is to investigate whether the concept of “separation of concerns” in learning design helps teachers during the LD task.

3. CREATING A LEARNING DESIGN WITH THE CADMOS APPROACH

CADMOS (CoursewAre Development Methodology for Open instructional Systems) approach proposes the “separation of concerns” for the design of learning process. This concept stems from software engineering and lately web engineering [Papasalouros et al., 2004; Rossi et al., 2008]. It argues that the designer builds the design in layers, and creates two design sub-models: the conceptual and the flow model. In the LD domain these models concern the following:

- The Conceptual Model is related to the learning activities that students and teachers will be engaged in during the instructional process of a specific subject. Each learning activity may be simple or composite. The composite learning activity consists of two or more simple learning activities but is addressed only to one role, e.g. student, teacher. Additionally in this model are defined the type of the resources that correspond to these activities.
- The Flow Model contains the navigational patterns (orchestration) of the learning activities.

3.1 Conceptual Model

When a teacher creates a new unit of learning, s/he at first has to define some metadata, like the title, the learning objectives, the prerequisites and the roles of this unit of learning. Then s/he starts to create the learning design at a conceptual level. Fig. 1 shows the conceptual model of a LD named “Programming Fundamentals”. The symbols that this model uses and their meaning are cited in Table 3. At the conceptual model the teacher has to specify the activities (composite or simple ones) of the LD.

For each composite activity there are some properties that need to be defined:

- Title: a title of the composite activity
- Description: a description about the composite activity
- Role: the actor that will be engaged in this activity (student or teacher)

For each simple activity the properties that need to be defined are:

- Title: a title for the simple activity
- Description: a description of the simple activity
- Role: the actor that will be engaged in this activity (student or teacher)
- Type: the type of the simple activity such as informative, theory, example, assessment, feedback, scaffold, simulation-modeling, and communicative
- Learning goal: the learning goal that corresponds to this activity
- Prerequisite: the prerequisite that a student must fulfil in order to complete this activity

Then the teacher should connect each simple activity with a resource. S/he can relate several specific resources to a learning activity. For each resource the properties that must be defined are:

- Title: a title for the resource
- Description: a description for the resource
- Type: the type of this resource, e.g. hypertext, audio, video, assessment, forum, and quiz
- Editor: the editor of the resource
- Copyright: free/proprietary
- Learning Object : the file or web site that corresponds to this resource

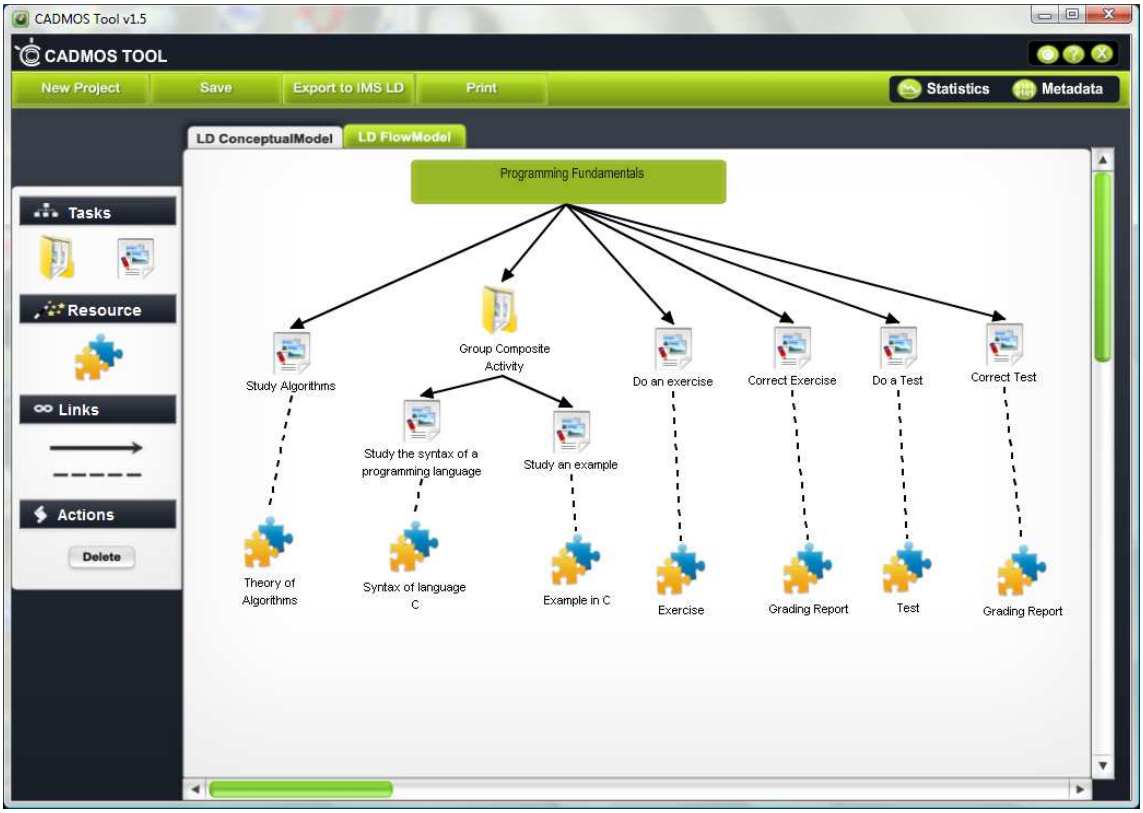

Figure 1. Learning Design Conceptual Model of the unit of learning entitled “Programming Fundamentals”

Table 3. The meaning of the icons in the conceptual model

Course name	Composite Activity	Simple Activity	Resource

3.2. Flow Model

After having created the conceptual model, the teacher moves to the next layer, i.e. the flow model, which concerns the orchestration of the learning activities. The flow model can be created automatically by putting the activities of the conceptual model, the one after the other as specified in the conceptual model from left to the right. Fig. 2 shows the model that is created assuming that there are no rules in the navigational between the different activities. This means that the student can do them in a linear way (linear navigational pattern). As we can see there are three different swim lanes, one for the student’s activities one for the teacher’s activities and one for a specified group’s activities. The idea of swim lanes can also be found in the Compendium LD tool [Conole, 2008]. In case the teacher has specified a composite activity in the conceptual model, in the flow model the activities that are part of this composite activity are represented grouped (i.e. inside a rectangular). In Table 4 are cited the symbols that the flow model uses and their meaning.

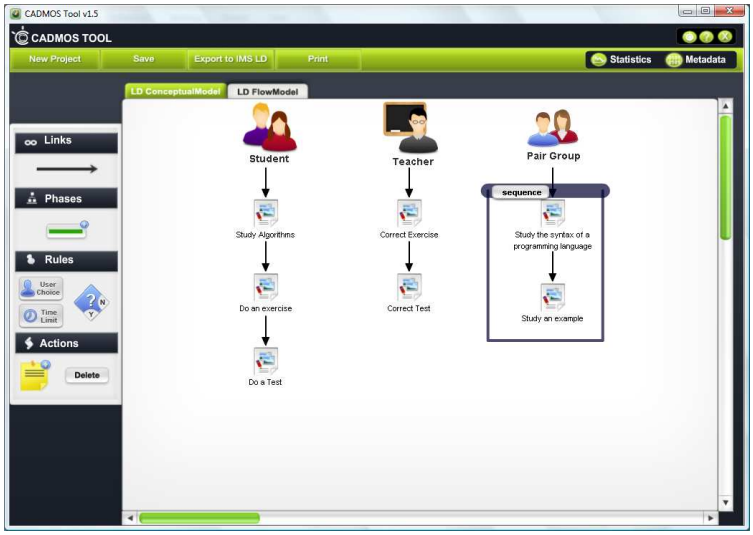

Figure 2. Learning Design Flow Model of the unit of learning entitled “Programming Fundamentals”

Table 4. The meaning of the icons in the flow model

Student	Teacher	Pair Group	Simple Task
			

The teacher needs to specify the navigation rules. As we can see in the toolbar, at the left of the design area, the “Rules” element consists of three different types of rules.

- The “User Choice” rule shows that a specific activity will be completed when its actor wants to.
- The “Time Limit” rule shows that an activity should be completed by a specific time.
- The score condition rule helps the designer define which activity will be done if the score of an activity is above a threshold or not.

In the current version 1.5 of CADMOS apart from the aforementioned rules the designer may add “comments” next to the learning tasks, if s/he wants to explain other rules. In the future more rules will be added to the tool. Also the teacher could divide the learning tasks into “phases”. The idea of dividing the flow in sections can be found in the representation of learning flow patterns [Hernández-Leo et al., 2010]. Fig.3 shows the above flow model divided into phases and having rules.

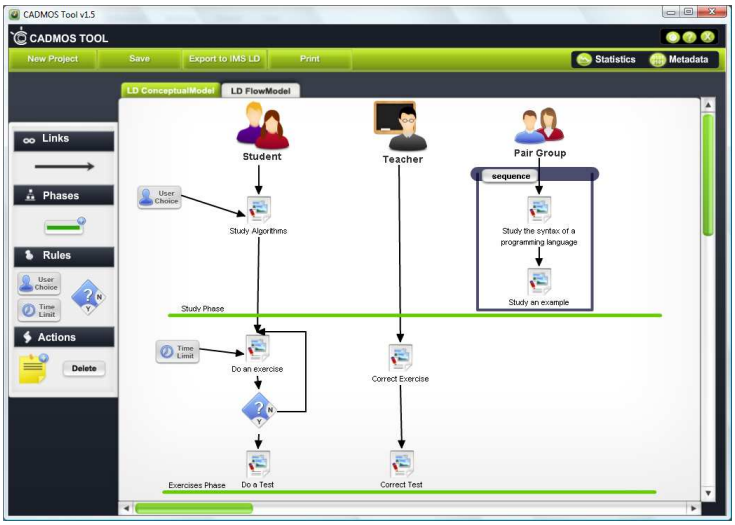

Figure 3. Learning Design Flow Model of the unit of learning entitled “Programming Fundamentals”, with phases and rules

There are some advantages from the use of this layered approach. In the conceptual model a teacher can change a learning object (resource) which is linked to a learning activity, without changing anything to the flow model. On the other hand if s/he adds or deletes a learning activity in the conceptual model, the flow model must be created from the beginning. Also, in case a teacher wants to add navigational rules in the flow model, s/he can do that keeping intact the conceptual model. Thus with the same set of learning activities, teachers can produce variations of learning activity flows according to their own instructional philosophy and learning context. For example, one teacher might decide that the students must study the theory before trying to do practice or the student may not be allowed to take the final assignment, before having performed all the suggested learning activities (e.g. study theory and examples and submit assignments). For the time being CADMOS tool does not support IMS LD level C.

CADMOS tool has been created in order to be used by novice learning designers, i.e. teachers with basic computer skills and knowledge of learning standards. The whole design process supported by the CADMOS tool is considered to be incremental. CADMOS has a specific meta-model which allows the teacher to save the learning design in the CADMOS's format (.cdm) or export it as an IMS/LD level B manifest file which can be reused by other IMS/LD editors or players. The tool also offers the possibility to import an existing IMS/LD level A/B manifest file and represent it in its own model. Thus, a teacher can also reuse an existing IMS LD learning design.

4. USER EVALUATION

With the aid of the current version of the CADMOS tool, the proposed approach was evaluated. Thirty six (36) MSc students who attended an MSc course on elearning design participated. Twenty five (25) of them were teachers. Twenty (20) of them were high school teachers (17 of them teach informatics, 1 teaches English) and five (5) of them were elementary school teachers. Five of them didn't have experience in learning design, while the rest of them had used one learning design tool at least. The majority of the students had used LAMS, MyUdutu and Dialog Plus. The evaluation process was completed in two phases:

Phase 1: During this phase we explained the philosophy of the CADMOS approach and functionality of the tool via a live demonstration in a laboratory. Students spent some time experimenting with the tool. At the end of this short face to face session we gave the students a lesson plan from environmental education in high school which is suggested by the Greek Pedagogical Institute and we asked them to create the equivalent learning design in CADMOS. The learning goals, the prerequisites and the flow of the learning activities were prescribed in narrative form and the students were asked to represent it in CADMOS. This session lasted almost one hour and the students posed their questions to the research group mostly about the functionality of the tool.

Phase 2: During this phase the students were given the CADMOS tool and a prescribed lesson plan in a narrative format taken from the curriculum of history of the last grade of primary school (also a suggestion by the Greek Pedagogical Institute). The subject of this lesson plan was the historic facts about Alexander the great which was familiar to all participants. Participants were asked to implement this lesson plan using the CADMOS approach and the tool. Then, they were asked to create a learning design from their own teaching practice. The duration of this phase was one week. During this week participants could pose questions or make comments about the approach and software bugs in a forum. After the completion of these tasks they answered an on-line questionnaire of 25 questions. With 22 close type questions of this questionnaire we tried to measure the quality criteria such as completeness, pedagogical expressiveness, flexibility, personalization, usability, and compatibility [14]. Three questions were open type.

In general the comments were positive. The majority of the participants (70%) claimed that were satisfied from both the approach and the tool. All of them said that the use of CADMOS was simple and that were able to easily and quickly complete the learning design. 69% of them stated that were highly satisfied with the guidance that was provided to them during the learning design process and only four (4) argued that the tool didn't have all the desirable facilities in order to design a lesson. The most important remark was that all of the students-designers said that the design approach via the two visual LD modelling views was very helpful. The creation of the conceptual model (100%) and the modification of the flow model (97%) were claimed to be simple and easy. The 86% of the participants claimed that the presence of ready-to-use design templates would have helped them while 83% of them stated that they appreciated the fact that they could reuse existing learning designs.

The participants made their positive and negative remarks in the three open-type questions. On the one hand many of them said that the tool must offer more options for characterising the types of learning

activities and learning objects. Furthermore, there were some feedback comments about the usability of the tool, like the lack of the undo feature, copy & paste functionality and few others.

All of the participants mentioned that CADMOS is a useful and easy-to-learn approach and that any teacher, with no specific knowledge in learning design and with basic computer skills, can apply it. The majority of them gave positive comments about the interface of the tool. Of course, as expected, they mentioned some software bugs during the use of the tool which are about to be fixed.

5. CONCLUDING REMARKS

This paper presented a learning design approach which is supported by the CADMOS graphical learning design tool. It is intended to be used by educational practitioners with basic computer skills. It allows the easy creation of learning designs in two layers (conceptual and flow). Although CADMOS tool is in a steady version, further experimentation is undergoing. We believe that it will be a great challenge for us to offer teachers the possibility in creating complex learning designs that will have rules and pre-specified templates of flows of learning activities (e.g. in the form of flow patterns). Thus the next big step is to enrich the meta-model of our approach so that it could offer more flexibility to the educational practitioners and be compatible with IMS/LD level C, which means rules, conditions and notifications. We will also add ready-to-use design templates based on known learning strategies (e.g. brainstorming, TPS).

REFERENCES

Botturi L., Stubbs T. (2008): Handbook of visual languages in instructional design: theories and pratices, *PA: Informing Science Reference*, Hershey.

Botturi L., Derntl M., Boot, E. and Figl K. (2006) A Classification Framework for Educational Modeling Languages in Instructional Design. *The 6th IEE International Conference on Advanced Learning Technologies (ICALT-06)*, (pp.1216-1220) Kerkrade, The Netherlands

Brown, C., 2006, Developing familiarity with learning design tools through subject analysis, *Journal of Learning Design*, Vol. 1, No. 2, pp. 10-20.

Britain A. (2004): A review of learning design: concept, specifications and tools, *A report for the JISC E-learning pedagogy programme*, May 2004, URL: http://www.jisc.ac.uk/uploaded_documents/ACF83C.doc.

Conole, G, 2008, Capturing practice: the role of mediating artefacts in learning design, Handbook of Research on Learning Design and Learning Objects: Issues, Applications and Technologies, in L. Lockyer, S. Bennett, S. Agostinho, and B Harper (Eds), *PA: IGI Global*, Hersey, pp.187-207.

De Vries, F., Tattersall, C. and Koper, R., 2006, Future developments of IMS learning design tooling, *Educational Technology & Society*, Vol. 9, No.1, pp. 9-12.

Goodyear, P., 2005: Educational design and networked learning: patterns, pattern languages and design practice, *Australasian Journal of Educational Technology*, Vol. 21, No 1, pp 82-101.

Hernández-Leo, D., Jorrín-Abellán, I.M., Villasclaras, E.D., Asensio-Pérez, J.I., Dimitriadis, Y., 2010, A multicase study for the evaluation of a collaborative learning pattern-based visual design approach, *Journal of Visual Languages and Computing*, Vol. 21, No. 6, pp. 313–331.

Laurillard, D., 2008, Enabling faculty to move forward the agenda: tools for learning design, *Open Learning Initiative Conference*, Carnegie Mellon University.

Lejeune, A., Ney, M., Weinberger, A., Pedaste, M., Bollen, L., Hovardas, T., Hoppe, U., & de Jong, T., 2009, Learning Activity Spaces: Towards flexibility in learning design, *Paper presented at IEEE ICALT 2009 conference*, Riga, Latvia, July 14-18.

Neumann, S., and Oberhuemer, P., 2009, User evaluation of a graphical modelling tool for IMS learning design, *Proceedings of the 8th International Conference on Advances in Web Based Learning (ICWL '009)*, Marc Spaniol, Qing Li, Ralf Klamma, and Rynson W. Lau (Eds.). Springer-Verlag, Berlin, Heidelberg, pp.287-296.

Paquette, G., Léonard, M., Ludgren-Cayrol, K., Mihaila, S., & Gareau, D., 2006, Learning design based on graphical knowledge modelling, educational technology & society, *Special issue on Learning Design*, January 2006, pp. 97-112.

Papasalouros, A., Retalis, S. and N Papaspyrou, N., 2004, Semantic description of educational adaptive hypermedia based on a conceptual model, *Educational Technology & Society*, Vol. 7, Issue 4.

Rossi, G., Pastor, O., Schwabe, D., Olsina, L. (editors) (2008): *Web engineering modelling and implementing web application*s, Springer, 2008