# Reusable Project-Based Learning Blocks for HE

Cleo Sgouropoulou [1], Maria Skiadelli[2]

[1]Department of Informatics, Technological Educational Institute of Athens, Ag. Spyridonos, 12210 Egaleo, Greece, csgouro@teiath.gr. [2]Natural Language Processing Laboratory, National Technical University of Athens, Iroon Polytechniou 9, 15780 Zografou, Greece, skiadeli@central.ntua.gr.

**Keywords**
Project Based Learning, IMS LD specification, Learning Design, patterns

## 1. EXECUTIVE SUMMARY

The goal of this paper is to present an e-learning design model for the Project-based learning (PBL) approach based on the IMS Learning Design (LD) specification. The model can serve as a template for the design of complete units of learning (based on the PBL approach) or as a modular part of more complex units of learning.

### 1.1. Background

Project-based learning is a learning approach for classroom activity that shifts away from the classroom practices of short, isolated, teacher-centered lessons and instead emphasizes learning activities that are long-term, interdisciplinary, student-centered, and integrated with real world issues and practices. PBL is considered mostly effective since it recognizes students' inherent drive to learn, their capability to do important work, and their need to be taken seriously by putting them at the center of the learning process. The project work is central rather than peripheral to the curriculum, highlighting provocative issues or questions that lead students to in-depth exploration of authentic and important topics.

Learning design is a relatively new but rapidly developing area of e-learning. The key idea behind learning design is the sequencing of learning activity. Using learning design methodologies teachers in fact manage to sequence or otherwise structure carefully and deliberately activities in a learning workflow to promote more effective learning. One major advantage of learning design is that it also enables teachers to record 'learning designs', often called learning scenarios, for sharing and re-use in the future.

### 1.2. Approach

The IMS LD specification is expressed as an XML binding which cannot be easily used by non-technical users such as teachers. Special tools have been created for this reason, in order to facilitate the learning design process. However, still the process of transforming a lesson plan from a simple narrative to a complete learning design is not trivial and requires a deep knowledge of the specification itself.

Using the IMS LD specification we have modeled the PBL educational model at a general level. The resulting design aspires to serve as a template for teachers who want to design their own PBL units of learning. Our work is based to some extent on pattern theory: basic modules can be considered as patterns that can be combined in many different ways by educators. The whole design of the PBL model can be considered as a pattern as well, that can contribute to the design of larger Units of Learning, which consist of various learning models.

The current paper illustrates an explicit methodology on how to configure and produce PBL-activities based on our PBL design, by presenting a solid example applied in the higher educational context.

## 2. INTRODUCTION

Within recent years great effort has been put into the introduction of innovative instructional methods in Higher Education that incorporate the use of technology and cultivate skills considered as valuable for the learners of our era. The introduction of such methods has made the design of the learning process by instructors more imperative than it used to be when traditional methods were used.

The IMS Learning Design (LD) is a specification created to standardize and facilitate the learning design process, whilst allowing produced designs, often called learning scenarios, to be shared and reused (Koper et al., 2004). In IMS LD the structure of the learning scenario is separated from the learning materials and services. Materials can then be reused within the same or different scenarios. The scenarios can also be reused and new materials can be added. The IMS LD specification is expressed as an XML binding which cannot be easily used by non-technical users such as teachers. Special tools have been created for this reason, which facilitate the learning design process (e.g. the RELOAD editor). However, the process of transforming a lesson plan from a simple narrative to a complete learning design is still not trivial and requires a deep knowledge of the specification itself. We claim that by providing teachers with pre-designed learning scenarios based on a concrete learning approach, we give them the opportunity to create their own learning designs in less time (accelerate the process) and allow them to focus on adjusting the core model to their own specific needs, adding their own resources and services.

Our modeling work has been based on the Project Based Learning (PBL) paradigm, as being one of the most promising methods for addressing the requirement for working knowledge development. Research in the field of applied PBL has revealed several important findings (Thomas, 1999). PBL, in comparison to other instructional methods, has value for enhancing the quality of learning by promoting active learning (application of knowledge in novel, problem-solving contexts). PBL can be considered as an effective method for learners to develop complex skills, i.e. planning, communicating, problem-solving and decision making. From a learners and educators' point of view, PBL is a more popular method of instruction than traditional methods.

Using the IMS LD specification we have modeled a PBL learning scenario at a generalized level. Our goal was this model to serve as a template for instructors who want to design their own PBL units of learning. We based our approach on pattern theory: basic modules can be considered as patterns that can be combined in many different ways by educators. The current paper is organized in two parts. The first part provides an overview of how the basic PBL model was designed and developed, outlining the steps from design requirements identification to template specification, whereas the second part illustrates in more detail the modeling work, including our approach towards patterns meant to be used in PBL scenarios, as well as directions for actual usage of templates.

## 3. DESIGNING THE PBL LEARNING MODEL

In this part we present the PBL model design methodology, starting with the statement of the design requirements and continuing with the description of the design process and products for a concrete PBL scenario, applied in our Departments' higher educational context. Note, that although the example presented is based on PBL, the methodology can be applied for the development of similar models based on other instructional methods, as well.

### 3.1. Design requirements

The overall design process started with the definition of a set of requirements that the resulting model had to fulfill. Our final model had to be:

1. *General purpose* - The model should be neutral to any specific educational context: subject area, discipline, student age or level of expertise.
2. *Extensible-modifiable* - The model should be incomplete in the sense that not all of the specification elements should be defined e.g. explicit learning objects. In this sense, the model is not ready to be executed by a LD engine. However, due to its incompleteness, teachers are capable of extending the model, adding their own resources, activities, material, services, etc.

3. *Modular* - The model should comprise various reusable modules e.g. Negotiation, Presentation, etc., that can be assembled in different ways to implement a specific PBL learning scenario.
4. *Graphically presentable* - Since the LD specification uses an XML binding, it is difficult to be used by non-technical users. Therefore we use the MOT+ graphical modeling editor to model our design at an abstract level. XML code is automatically generated and can be imported in some LD modeling tools i.e. the RELOAD editor, for further processing.
5. *Capable of supporting the blended learning paradigm* - Both face-to-face and on-line learning methods should be combined to implement the proposed activities. The teacher herself should be able to decide which activities should be electronically supported.

These requirements guided the entire design process and the final model was checked against them for validity.

## 3.2. The design process

***PBL scenario description***

The PBL scenario we elaborated is intended for application in a higher education course framework, however, it is not tied to a specific course, subject or discipline. Students are requested to undertake a project before taking the final exam at the end of the semester. An information delivery module (lectures series) of the course takes place in parallel with the project. The assessment criteria of the students' performance are known beforehand. Students undertaking the project will have to form groups of 3 to 5 persons and finish their work within an 8-10 week period (a semester usually lasts about 13 weeks). Each group works on a different subject, independently of each other. The scenario foresees, however, certain points (at time and place) of collaboration among groups. The subject of each project can be proposed either by the instructor or by the students' group and normally gets unanimously approved after negotiation. The instructor of the course is also assisted by a number of teaching assistants, who supervise the activities of the groups and provide technical support as regards the on-line components of the scenario. The whole team of instructor and assistants will be called hereafter the "staff team", whereas the groups of students will be called "project teams".

***First level narrative of the scenario (in natural language)***

We started by writing the narrative of the project scenario in natural language without following a certain template. Our goal is to produce a script that incorporates all the necessary information for the staff team that has no contact with the scenario designers (our team). The narrative is divided into sequential steps attempting to describe how things should happen in reality, including advice, experience and hints to the staff team that is going to run the scenario during the semester. The scenario is based on the blended learning paradigm i.e. there are parts to be carried out as face-to-face activities and parts that are to be performed as on-line (web-based) activities. Although recommendation is provided, it is up to the staff team to make the final decision on delivery mode of the individual activities, taking into account the following quote (Dernlt et al. 2004): "…in blended learning scenarios most interactions take place face – to –face with the on-line acting as a backbone for the transmission and sharing of material, for its preparation and organization for administration and for intermediate sharing of ideas and experiences…"

***Second level narrative of the scenario (in natural language using LD terminology)***

After producing the complete narrative we processed it in accordance with the IMS LD terminology. This task involves the identification of *Acts*, *Activity structures*, *Activities*, *Roles* and *Environments*(IMS Learning Design, 2005). Since our scenario does not relate to a specific course, subject or discipline we did not have to specify any learning objects, learning objectives, prerequisites, etc. This is part of the instantiation phase, where the details of the projects will be identified and finalized. For each Act, Activity structure and Activity there is a description of what is going to happen, which Roles contribute to the Act and which services (if any) are used. All the useful information included in the first level narrative e.g. advice to the teacher, recommendation,

explanation, is shared among the descriptions of the identified IMS LD elements. The narrative produced at this level is part of the template accompanying the final graphical model.

### *Graphical model*

A graphical model gives a better overview of the activity flow diagram that cannot be easily extracted from the narrative. Both the model and the narrative are necessary in order to run the scenario. As mentioned earlier, the designed PBL scenario is to be followed by people, not run by a machine; consequently there is a lot of useful information that should be conveyed to them along with the graphical representation.

MOT+ is an object-oriented modeling tool intended to express various fields of knowledge as graphic knowledge models (Paquette et al., 2005). It can also be used as a graphical IMS LD editor providing all the elements of the IMS LD information model and a set of semantic links for the representation of their relationships. The designed model can be exported in XML according to the IMS LD XML binding. This is an important reason why we opt for using MOT+ instead of another general purpose modeling tool. The IMS LD Best practice guide recommends the use of UML for the modeling of the narratives. However, these diagrams must then be transformed manually to the XML format, which in general can hardly be done by non-technical users.

Apart from the direct use of the IMS LD elements two other features of MOT+ were considered as attractive and useful for our work:

- The use of sub-models, that enables the designer to place part of the model on separate sub-models that interconnect with each other. This feature facilitates the creation of modular models, reduces complexity, enhances clarity and permits the design of larger models. A sub-model may also contain other sub-models (nested sub-models).
- The "copy-by-reference" feature, providing the designer with the opportunity to define once and then reuse elements wherever needed. This feature expresses to some extent the IMS LD reference philosophy.

The next section discusses the actual implementation of the PBL model in detail.

### *Scenario Template*

The graphical representation of our narrative constitutes a template and not a complete IMS LD Unit of Learning (UoL). We claim that the purpose of a template is to express one or more well established educational methods in a generative way (e.g. PBL) and serve as the basis for the creation of various UoLs that make use of these particular educational methods.

The final template is characterized by the following:

- the *Role* hierarchy tree is shallow and may be extended
- the *Environment* tree has no learning objects neither learning services
- the *Activities* tree is incomplete: activity structures may need to be further analyzed or not analyzed at all. Some activities may be missing.

It is actually due to this "incompleteness" that the produced model can be called "a template" since it is not related to any particular discipline, tool, service or learning material and it can be custom-tailored by teachers and instructional designers, according to their needs.

The following (Figure 1) is a tree representation of part of the IMS LD specification in which it is indicated which nodes should be partially or not defined at all, in order for a resulting model to adhere to the template definition.

## 4. IMPLEMENTING THE MODEL AND CREATING THE REUSABLE TEMPLATE

This section provides a detailed overview on the actual implementation process of the graphical model for our PBL scenario, as well as on its use as a template.

Our scenario foresees 3 phases: *Preliminary Activities*, *Main Phase* and *Assessment* (a lot of similarity with the example presented in Dertl et al. 2004). Each of the phases is represented by an Act element in our model. The scenario refers to a long-term project. Acts work as points of synchronization in IMS LD; upon completion of an Act all Roles must have finished all Activities

before proceeding to the next Act. Acts need to be serialized (executed one after another). Each Act consists of several Activity structures that they fall in one of the following categories:

- They are further analyzed forming a sub-tree whose nodes are either Activity structures or Activities according to the IMS LD information model.
- They are no further analyzed.
- They are instances of patterns.

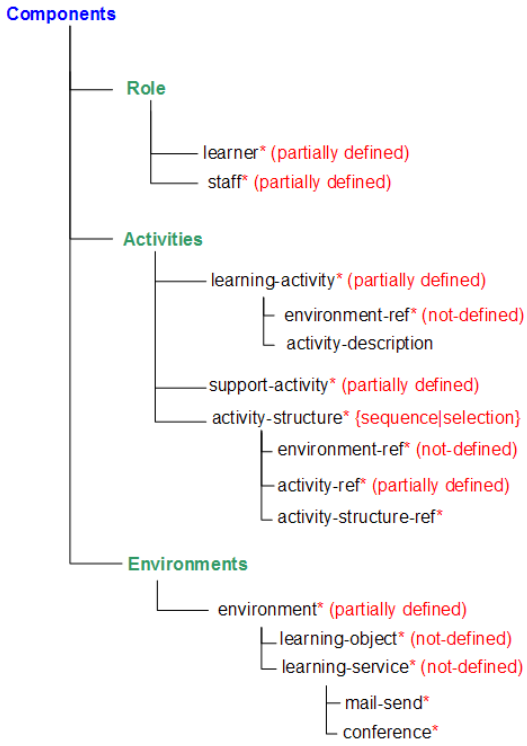


**Figure 1. Template tree representation of IMS LD**

Figure 2 illustrates the graphical model of the *Preliminary Activities* phase of our scenario, for which significant work was carried out for the definition of necessary reusable modeling components, as for the expressions of appropriate relationships.

## 4.1. Definition of sub-models

***Role sub-model***

We created a shallow Role hierarchy that contains only the basic roles needed for our PBL scenario. All roles are divided in staff roles and learner roles according to the IMS-LD specification and are placed in two separate tree structures. Role hierarchies may be modified by instructional designers and teachers. More roles may be added, others can be removed. We, however, recommend keeping roles that are used inside the pattern models. All roles used in models are copied as references from the role sub-model.

***Services sub-model***

The services sub-model contains all the services to support the on-line components of the PBL blended scenario. Services may be offered by means of an integrated environment or by individual tools e.g. “the mailer offers the mail service”. A designer may add more services or remove (or simply not use) certain services. All these services constitute a workspace that may have various descendants. In our design we have foreseen two of them: the “team workspace” which is shared by the members of a project team and the “class workspace” that is shared by all project teams. The final implementation will include several “team workspaces”, since each project team has its own

team workspace. These can be considered as instantiations of the "team workspace". The class workspace is shared by all project teams and it is a central meeting point for the whole course unit. In other designs one might include a student workspace or a staff team workspace, etc. All workspaces contain the set of services that appear in this model or subsets of this set.

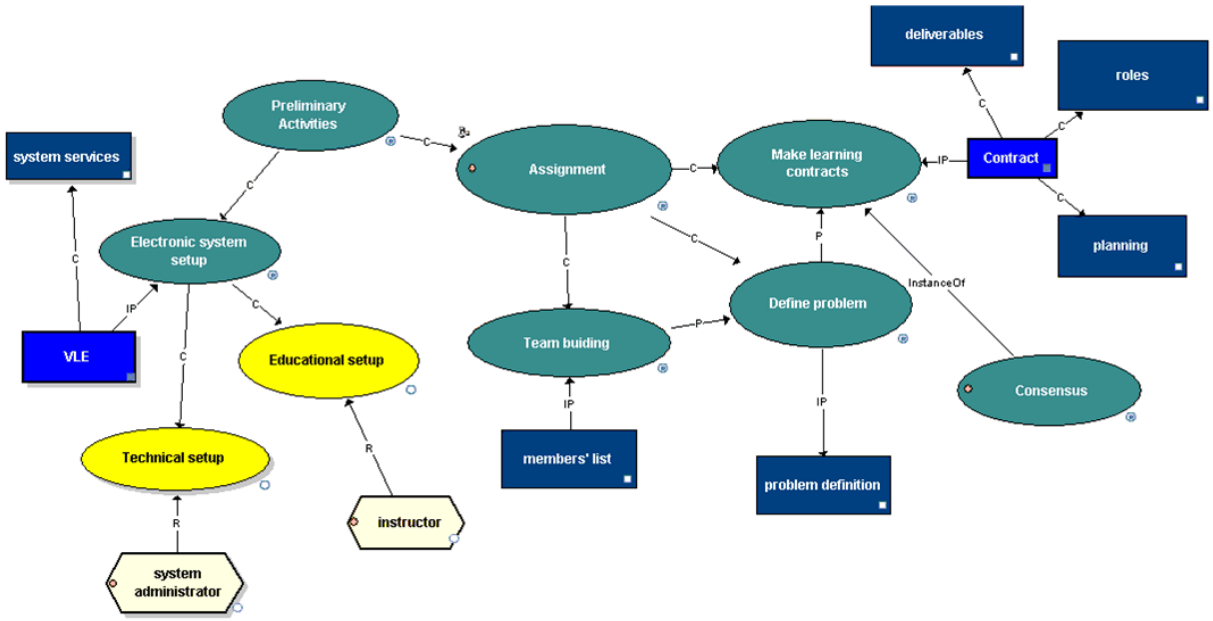


**Figure 2. The *Preliminary Activities* Act**

## 4.2. Pattern identification and design

While writing the narrative of the PBL scenario, we noticed that there are several reoccurring situations that can be modeled in a similar manner. For instance, the cycle "propose-negotiate-synthesize", appears constantly in a PBL scenario whenever a decision has to be taken and therefore consensus is required. No matter if the members of the team have to choose their leader, make up their minds about the project milestones, or assign roles to individual members, they always have to follow the abovementioned activity cycle. These situations are not identical; there are a number of parameters that may vary (subject, time, place, outcome, etc.) but they all share a common core that can be identified as a good common practice or solution wherever the situation of reaching consensus arises. Therefore, we claim that this activity cycle constitutes a pattern that can be called the "Consensus" pattern.

There are many definitions for patterns starting from the original definition that Alexander (1979) gave: "A pattern is a solution to a problem that occurs in various contexts and can be reused ever and ever". In the learning domain the definition of the Pedagogical Patterns Project seems more prevailing: "patterns are designed to capture best practices in a specific domain" or "capture expert knowledge of the practice of teaching and learning…"

***Pattern Identification***

The identification of patterns is a process called "pattern mining". It is difficult to identify patterns because it is hard to define what is a pattern and what it is not a pattern. Pattern identification has mostly to do with intuition and although there is no good methodology for pattern identification, there is a set of characteristics that identified patterns should possess, namely: *Encapsulation* and *Abstraction*, *Openness* and *Variability*, *Generativity*, *Composability* and *Equilibrium* (Appleton, 2000). We attempted to identify patterns in our scenario according to the following criteria:

- Patterns must represent situations, procedures or problem solutions that preferably appear more than once in a PBL scenario.
- They must be applicable to all kind of project setups, independently of the number of participants, duration period, or other parameters that may vary.

- In our model not everything ought to be a pattern. Only some of the Activity structures of the model have been identified as patterns, and some others make use of patterns.
- Patterns should bear some pedagogical value, e.g. Activities related to the system setup by technical staff are not considered as patterns
- They must be analyzable in order to represent a solution. This means that they must be roots of an Activity structure sub-tree. Activities or Activity structures that are no further analyzed cannot be patterns. (Later we will see that some of the activities or activity structures that have not been analyzed in our model have been placed in the list of the candidate patterns).
- They are not identical to tools or services that may also be reusable. For example there is no pattern called "chat" or "online discussion". These are services offered by the on-line system and although they may often appear in PBL scenarios, they do not constitute a practice or solution to a problem. Patterns can make use of such services, but they are more complex human activities.

***Patterns and Candidate Patterns***

After performing the analysis of our PBL scenario we ended up with six patterns and a list of candidate patterns. The patterns are: *Accomplishment*, *Approval*, *Consensus* (Figure 3), *Feedback*, *Presentation* and *Supervision*. We claim that these patterns can be useful in every PBL scenario. The fact that most of the patterns presented here (or their variations) are also found in other learning patterns repositories [e-Len patterns (Goodyear et al. 2004), the Pattern repository (Derntl, 2005), etc.] strengthens our approach. The modeling language is perhaps different, the proposed solutions may vary, but the stated problem is more or less the same, which enforces the idea that there is a set of common problems/situations that keep on appearing in learning scenarios and that learning design can be considerably improved and facilitated by the use of predefined solutions (patterns) to those problems. The list of patterns as well as the list of the candidate patterns may be extended by other designers and instructors whilst more PBL scenarios are being modeled using the same methodology that we used for our scenario.

Candidate patterns refer to a number of Activity Structures in our model that may be considered as patterns in a future attempt. Most of them are already met in other learning pattern repositories and/or represent a problem that seems to be recurring in PBL scenarios. The reason why we did not proceed with describing them as full patterns was mainly due to time restrictions to propose a good solution to the problem that they represent. We preferred to focus on a limited set of patterns, whose solution was already mature and they turned out to appear more than once in the current PBL scenario. The Activity Structures that form the candidate patterns list are the following: *Resolve conflicts*, *Make learning contracts*, *Team Building* and *Team leadership*.

There is no really good methodology for designing patterns, as well (Appleton, 2000). We designed our patterns according to a few basic principles:

- We tried to design patterns that encapsulate experience, expertise and common sense.
- We wanted our patterns to be less descriptive and more prescriptive. Our goal was to model human activities and give parameterized solutions that can be further explored.
- Our patterns cannot be played in their current state by a computer learning system, but they can act as guidelines for helping teachers and educational designers to design their own courses implementing good practices.
- We deliberately avoided the use of specific tools or services inside the patterns. For instance in the *Consensus* pattern there is a need for a team workspace, a general place where the negotiation, presentation, etc and other activities of the pattern will take place. However, it is up to the teacher to decide the exact media that will support the on-line activities.
- Some patterns have no roles attached to them, for instance the *Presentation* pattern, does not include either staff or learner roles. It turned out from our analysis that both learners and staff may present various topics at various steps of a PBL scenario. In the scenario presented in this paper, each group of students presents the final solution/results of its project, at the end of the Main Phase. In another project setup the instructor may decide to present the details of the project at the preliminary phase.

- Patterns may contain other patterns: example the *Presentation* and the *Accomplishment* patterns both contain the *Feedback* pattern.

Not all of the activities inside the pattern need to be executed. As an example, the *Consensus* pattern the *Vote* activity is an alternative choice when the synthesis phase does not lead to any decision.

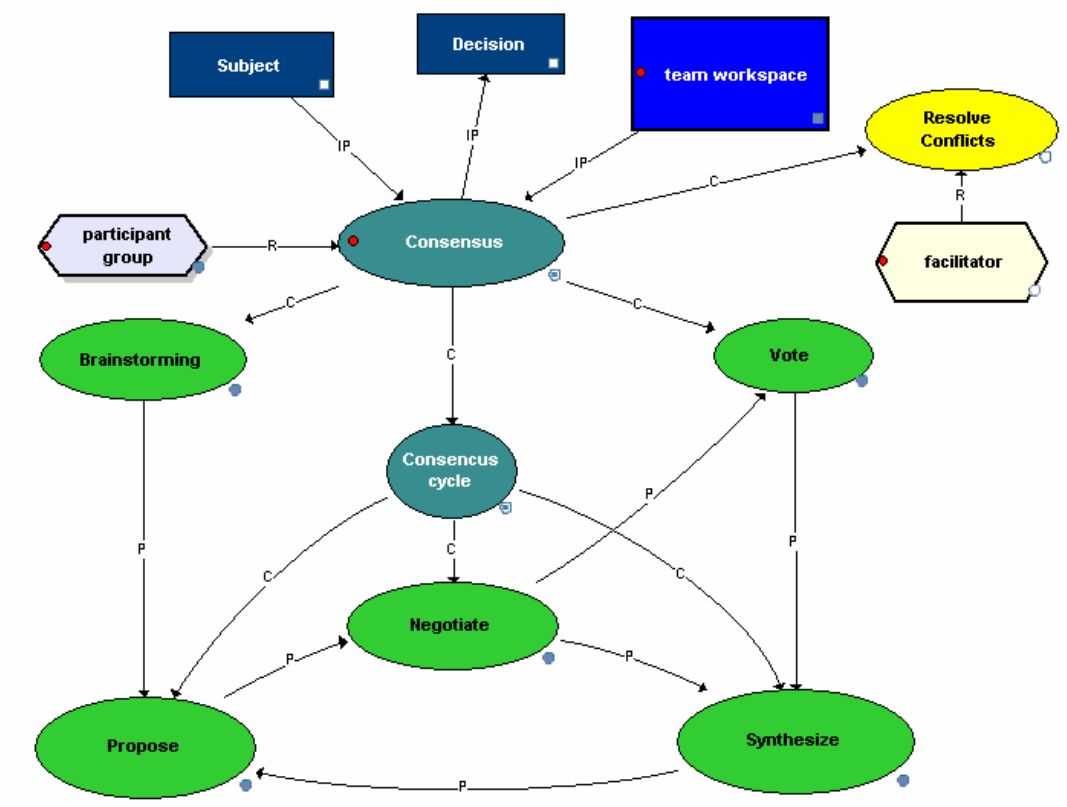


**Figure 3. The *Consensus* pattern**

### *Relationships expression*

Regarding the modeling of relationships between the several components in the different Act and patterns definition, we encountered specific problems with the MOT+ editor, summarized below:

1. *The lack of instantiation* - Patterns are modeled in the MOT+ environment in separate definition sub-models. Patterns in our model act as classes according to the object – oriented paradigm: they need to be instantiated. The idea of reusability of a pattern is expressed via instantiation. If an Activity structure is identified as a certain pattern, then we say that this Activity structure is *an instance of* that pattern. One or more Activity structures may be instances of the same pattern i.e. the "select leader" and "select milestones" Activity structures are both instances of the *Consensus* pattern. Because the IMS-LD model and the MOT+ editor, lack the "instance-of" kind of relationship, we modeled it not as a direct link but using the following method: let's suppose that an activity structure called AS has been identified as an instance of a pattern P that has a root node called N. The pattern P model is placed in a separate sub-model, from which we copy by reference the N node to the model where AS is placed. Then we create an un-typed link (offered by MOT+) that starts from the copy of N and stops to AS and give the name *InstanceOf* to that link.
2. *The lack of recursion/iteration* - There are some activities that may be carried out recursively. For instance, in the *Consensus* pattern the cycle "propose-negotiate-synthesize" may be carried out for an arbitrary number of times until consensus is reached and a decision is taken. In order to model the iteration of activities, we introduce a new Activity structure and give it a suitable name like, for instance, "consensus cycle" to indicate the

iteration. Then we link every activity that is part of this cycle with a *C* link with the newly created Activity structure and a *P* kind of link with the other Activities of the cycle to indicate the order of execution.

3. The lack of generalization/specification relationships - In many cases there is a need to define an *IsA* relationship between two elements. In the *Services* sub-model the "team workspace" and the "class workspace" both inherit the services of the "workspace" environment, but they are meant to be used for different purpose i.e. in different activity diagrams. We use the un-typed kind of link offered by MOT+ to model such kind of relationships giving it a suitable name *IsA*.

***Using the graphical model as a template***

The current version of MOT+, allows models to be saved as templates (having the .gab extension) and then new models can be opened using one of these templates. The teacher or educational designer that will use our template to design his/her own PBL scenario has the following options:

- Open a new model using the existing template and decide which parts will be executed on-line, and which face-to-face. Copy and paste the on-line parts of the model in new models, add placeholders for learning objects and services, then export the model in IMS LD XML format. Edit the resulting xml files (better use the RELOAD editor for this purpose) to add recourses and services and create the corresponding zipped packages. Play the exported packages with an IMS-LD player, whenever needed to support the on-line activities of the PBL scenario.
- Open a new model using the existing template and create another play in the same model that will represent another PBL scenario to be used for another project setup (for instance a project of shorter duration or larger groups of students, etc.) Create new acts, and activity structures based on the patterns, role hierarchies and services available from the definition sub-models of the template. Decide which parts will be executed on-line and which face to face and follow the same steps as above.

## 5. CONCLUSION AND FUTURE WORK

In this paper we tried to illustrate our methodology on how to express a complete PBL scenario in terms of an IMS LD model template that makes use of patterns. We explained how this model template was created, starting from a detailed narrative, and what problems were faced while modeling this narrative using the IMS-LD information model in the MOT+ editor. Since our model is based on patterns, we made an extensive reference to pattern identification and discussed the principles we followed for their design in our model. In the future we plan to explore other modeling environments that have some of the features that are inherent to pattern theory (like instantiation, specification/generalization, etc.) We will also try to extend our pattern list, analyzing more patterns and produce several PBL scenarios to prove their validity in various learning situations. We have, however, initiated a case study that uses our current scenario in a real higher education course, and have started collecting valuable feedback about how useful and helpful it can be to instructors and learners.

## 6. REFERENCES


Alexander, C. (1979). *The timeless way of building*. New York: Oxford University Press.

Appleton, B. (2000). *Patterns and Software: Essential Concepts and Terminology*. Retrieved September 2, 2007, from http://www.enteract.com/~bradapp/docs/patterns-intro.html

Derntl, M. (2005). *Patterns for Person-Centered e-Learning*. University of Vienna, Austria: PhD Thesis.

Derntl, M., Motschig-Pitrik, R. (2004). Patterns for Blended Person-Centered Learning: Strategy, Concepts, Experiences and Evaluation. *ACM Symposium on Applied Computing*.

Goodyear, P., Avgeriou, P., Baggetun, R., Bartoluzzi, S., Retalis, S., Ronteltap, F. & Rusman, E. (2004). Towards a pattern language for networked learning. *In Banks, S., Goodyear, P., Hodgson, V., Jones, C., Lally, V., McConnell, D., & Steeples, C. (Eds.), Proceedings of the Networked Learning*, Lancaster: Lancaster University, UK, 449-455.

E-LEN project website (2005), Retrieved March 31, 2008, from http://www2.tisip.no/E-LEN

IMS Learning Design (2005). *Information Model, Best Practice and Implementation Guide, Version 1.0 Final Specification IMS Global Learning Consortium Inc*. Retrieved January 26, 2008, from http://www.imsglobal.org/learningdesign/

Koper, R. & Olivier, B. (2004). Representing the Learning Design of Units of Learning. *Educational Technology & Society, 7, 3*, 97-111.

Paquette, G. et al. (2005). *An instructional engineering method and tool for the design of units of learning*, in R. Koper & C.Tattersall (Eds.), *Learning design: A Handbook on Modelling and Delivering Networked Education and Training* ,BerlinHeidelberg: Springer 2005, 161–183.

Paquette, G., Léonard, M., Ludgren-Cayrol, K., Mihaila, S. & Gareau, D. (2005). Learning Design based on Graphical Modelling. *Educational Technology & Society*.

Patterns Repository website (2008). Retrieved March 7, 2008, from http://elearn.pri.univie.ac.at/patterns/

Pedagogical Patterns (2008). Retrieved March 7, 2008, from http://www.pedagogicalpatterns.org/

Reload (2005). Reusable eLearning Object Authoring & Delivery Project, Retrieved on October 27, 2007 from http://www.reload.ac.uk/

Thomas, J. W., Mergendoller, J. R., and Michaelson, A. (1999). *Project-based learning: A handbook for middle and high school teachers*. Novato, CA: The Buck Institute for Education.