

Discussion of Collaborative Learning Designs, Languages, Models and Tools – LICEF-Télé-université’s Contribution

Gilbert Paquette

CICE Research Chair, LORNET network leader, LICEF, Télé-université
gilbert.paquette@licef.telug.uqam.ca

Abstract

This short paper summarizes work at LICEF on Educational Modeling Languages, focusing on the design of collaborative units of learning.

1. Previous work

Work on EML has started at the LICEF research center in 1992 with the design of an instructional design system called AGD. From it, graphic modeling tools have evolved, recently to the MOT+LD specialized graphic language. In parallel, an instructional design methodology was developed and embedded in new Web-based tools such as ADISA. The method is now being adapted to a new version called MISA-LD. During the last year, we have started to build a IDLD repository of learning design patterns and examples.

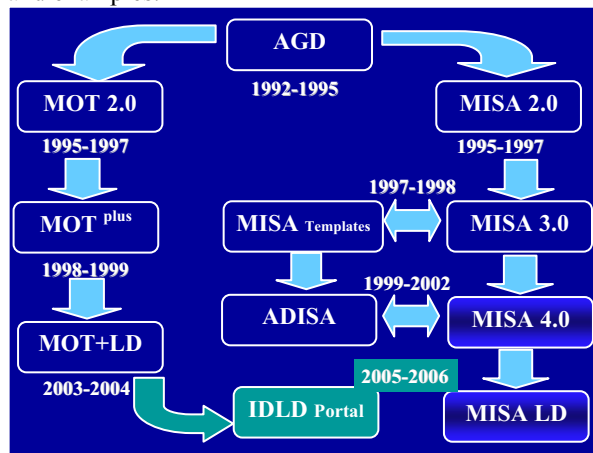


Figure 1 – EML work at LICEF

2. MISA and the IMS-LD specification

The IMS-LD specification leaves open the choice of instructional methods and modeling tools that can support designers in the process of building learning designs, especially for collaborative scenarios.

Extensive research and development in the last thirty years have led to a host of ID methods. As shown in [1] the MISA Learning Systems Engineering Method is especially well suited to help designers build IMS-LD compliant Units of Learning, because it provides a well founded educational modeling language at the crossroads of instructional design, software engineering and knowledge engineering.

In [2], we have shown that the IMS-LD specification and the MISA method complement well each other. The IMS-LD specification provides a standardized machine-readable representation of a learning design, whereas MISA proposes a systemic and mostly graphic method to design and implement such learning designs. MISA helps develop learning designs by specifying four models for knowledge and competency model, pedagogical strategy, learning material structure and delivery processes.

Still there are differences to take in account beyond terminology. IMS-LD covers well and integrates better both the pedagogical and delivery models. It permits a more developed multi-actor representation than MISA, but it is poorer for knowledge/competency [3] and learning material association to method components. Finally, some ambiguities for collaborative design must be clarified in the specification and natural graphic representation must be provided to choose between collaboration rules and tackle the inherent complexity of the specification.

3. Graphical Learning Design

Using a graphical representation technique and a modeling tool like MOT+LD, concepts, procedures and principles are used to describe all IMS-LD level-A components (see figure 2) as well as their relationships. We have experimented such a graphical language to be closer to instructional designers, than software engineering graphical languages like UML or text-based editors like RELOAD, while still enabling an automatic translation from graphical designs into machine-readable IMS-LD XML files.

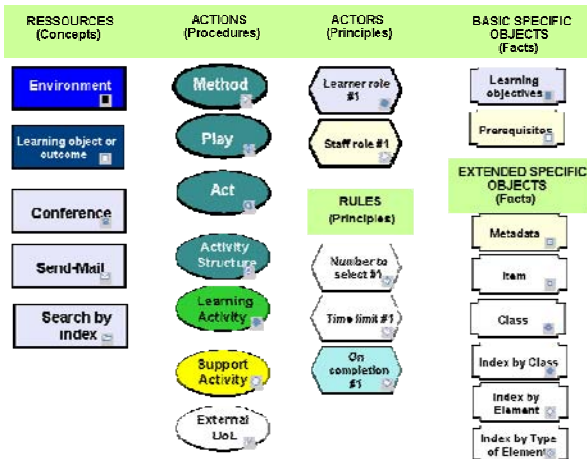


Figure 2 – Basic MOT+LD graphic symbols

4. Collaborative design support

One of the most interesting features in IMS-LD is the possibility to synchronize actors in multi-actor process-based scenarios. From level A, to levels B and C, the specification adds more and more design capabilities for design and run-time support of the actors.

One example is the Versailles unit-of-learning we have modeled graphically in MOT+LD. It involves students organized into six teams. Each team must discuss in private on-line meetings, then negotiate with the other teams in a set of forum and finally all together. Such a complex scenario requires sound design and good computer support at run-time. Using the IMS-LD specification and a graphic tool like MOT+L relieves the designer from taking care of all the run time technical details. This example shows very well the value of IMS-LD but still, we have found a few weaknesses where improvements should be made to the specification [4].

The representation of collaborative activities works well if the granularity of the interactions between actors is large, as in the Versailles scenario. If the granularity is smaller, as in educational games, the representation might become either superficial (black box effect) or cumbersome if all small events need to be described.

Another difficulty is a certain form of ambiguity in the representation of collaborative activities. For example, if N actors are involved in role-parts within the same activity, there is yet no vocabulary to distinguish between situations where they work separately at different times, or together at the same time. In MISA, we have a build a set of collaboration rules that could help improve the specification.

Finally, IMS-LD has been built to facilitate run-time implementation. It is strongly influenced by the theatre play-act-scene metaphor. In a play, actors proceed sequentially through the acts, while acting in parallel within acts or activity structures. This approach can be implemented in computers rather easily compared to situations such as project-based learning, where the flow of activities involve many returns to previous activities or phases in the project, where new activities or resources are added at run time and alternative activities or resources cannot be planned at design time. These more open learning scenarios are harder to manage even if the design embeds sophisticated control and support rules.

5. Conclusions

IMS-LD provides the basic functionalities to build collaborative scenarios but graphic design tools are essential to cope with the inherent complexity. Text-based hierarchical tools like RELOAD should be used when the basic level A design has been completed enough and exported to it to add the more technical elements such as items addresses, metadata, properties and conditions, notifications. This two-step approach correspond to different designers' competencies.

We should keep improving on IMS-LD (standards are hard to impose) Important tasks are to:

- simplify the graphic languages;
- extend the EML with basic collaborative templates and controlled vocabularies to remove ambiguities;
- revise the specification in 2-3 years when enough experience is achieved.

6. References

- [1] Paquette, Gilbert. (2003) Instructional Engineering for Network-Based Learning. Pfeiffer/Wiley Publishing Co.
- [2] Paquette, G., De la Teja, I., Léonard, M., Lundgren-Cayrol, K. and Marino, O. (2005). How to use an Instructional Engineering Method and a Modelling Tool. In R. Koper & C. Tattersall (Eds.). Learning Design - A Handbook on Modelling and Delivering Networked Education and Training, Springer Verlag, pp. 161-184
- [3] Paquette, G., M. Léonard, K. Lundgren-Cayrol, S. Mihaila and D. Gareau. (2006) Learning Design based on Graphical Knowledge-Modeling, Journal of Educational technology and Society ET&S, Special issue on Learning Design, January 2006.
- [4] Paquette, G., and O. Marino. (2006 to be published) Learning Objects, Collaborative Learning Designs and Knowledge Representation. Technology, Instruction., Cognition and Learning, Vol. 3, Old City Publishing, Inc.