PERSPECTIVES AND ASPECTS IN POEML
Supporting Adaptation in Learning Processes

Manuel Caeiro, Luis Anido and Martin Llamas
Department of Telematic Engineering, University of Vigo, Vigo, Spain

Keywords: Model-driven engineering, Process-based Modeling, e-Learning, Educational modeling languages.

Abstract: Adaptation is a key feature of e-learning systems. The educational process performed by the support of e-learning systems should be carried out taking into account the cognitive and other particular characteristics of the students, of the teachers and of the environments. However, this adaptation is not a simple process. It implies the conjunction of technical and pedagogical issues. This paper introduces a solution to support adaptation in the context of an Educational Modeling Languages (EML). EMLs have been conceived to allow us the modeling of learning units, enabling the expression of different pedagogical approaches. Eventually, these models are intended to control and support the development of the corresponding learning units.

1 INTRODUCTION

During the last years there have been several initiatives trying to support e-learning based on a model-oriented approach. *Instructional Engineering* has been proposed by (Paquette, 2004) as ”a method that supports the analysis, the creation, the production, and the delivery of a learning system, integrating the concepts, the processes, and the principles of instructional design, software engineering, and knowledge engineering”. A main result of this initiative is a proposal for a modeling language in accordance with the previous description. Similarly, *Educational Modeling Languages* (EMLs) have been proposed recently as process-based modeling languages to describe learning units and support their execution (Vignollet et al., 2006). More specifically, *IMS Learning Design* (IMS-LD) is the current *de-facto* EML standard (Koper et al., 2003). This language has been developed as a computational EML that can be used by suitable *Learning Management Systems* (LMSs) to support, control and manage the development of e-learning experiences (Koper and Olivier, 2004). A review of these and other modeling languages from a graphical point of view has been recently performed in (Botturi and Stubbs, 2007).

Despite the existence of numerous initiatives and proposals, the support of e-learning based on the model-oriented approach has not succeeded, yet. A main problem in this approach is the difficulty to create models of learning units by final users. The achievement of a single language that enables to create models in accordance with different pedagogical approaches is feasible (at least following a process-based approach, as IMS-LD), but it is not very usable.

This paper introduces a new EML to simplify the modeling of learning units named as *Perspective-oriented EML* (PoEML). Following a basic separation-of-concerns idea, PoEML approaches the modeling of learning units through several perspectives and aspects. Each perspective and aspect involves the modeling of a part of the whole learning unit in a way (almost) separated from the other perspectives. As a result, these separation into perspectives brings many benefits. Particularly, this paper is focused on the description of the support of adaptation based on perspectives and aspects.

2 POEML

This section introduces a new EML devoted to increase the expressiveness of existing EMLs while observing important capabilities such as adaptability. It has been adopted a *separation of concerns* approach. Basically, the modeling of educational units is not attained as a single and whole stuff, as in existing EMLs, but the learning units are observed from several perspectives that are modeled separately . Following this approach the new EML has been named as *Perspective-oriented EML* (PoEML). Next sections

Caeiro M., Anido L. and Llamas M. (2009).
PERSPECTIVES AND ASPECTS IN POEML - Supporting Adaptation in Learning Processes.
In *Proceedings of the 4th International Conference on Software and Data Technologies*, pages 226-231
DOI: 10.5220/0002258202260231

describe the PoEML language at abstract and structural levels.

2.1 Foundations

The key strategy for the development of PoEML has been the *separation-of-concerns* principle. *Separation-of-concerns* is a long-standing idea that simply means that a large problem is easier to manage and solve if it can be broken down into parts and each part can be approached separately. It is an important design approach in many areas, such as software design (Parnas, 1979), used to facilitate the understanding, design and management of complex systems. In addition, UML is a modeling language for software engineering where different diagrams are proposed to model different issues: use cases, analysis, design, etc. A similar example in other domain is the architectural plans, which follow a separation of concerns for the development of buildings. In the learning domain there are also some proposals in which a certain kind or separation of concerns is provided, such as (Strijbos et al., 2004), where learning units are considered through several orthogonal axis. Anyway, as long as we know, PoEML is the first attempt that takes the separation of concerns as driven development principle.

Another important foundation of PoEML is *Activity Theory* (Engestrom et al., 1999). This theory has been used to analyze the issues involved in learning units and to identify an appropriate separation of concerns (Caeiro-Rodríguez et al., 2007). The *Activity Theory* has guided us in the *separation-of-concerns* in learning units towards the identification of 13 perspectives and 4 aspects. They have already been described in a previous document (Caeiro-Rodríguez, 2007).

2.2 The PoEML Conceptual Model

The PoEML conceptual model describes the main concepts of the language and the relations among them. From a conceptual point of view PoEML is mainly characterized by two important issues: the adoption of a special kind of task concept as basic building block and its hierarchical and structured nature. In addition, there is one root element with special components. Next sections describe these important features.

2.2.1 The Basic Concept

PoEML is a process-based language, as other EMLs. Nevertheless, the basic concept of PoEML is not the task, but the *Educational Scenario* (ES). There are two main differences that distinguish the PoEML ES and the typical EML task concepts:

- The Goal is recognized as a first-class entity in PoEML. It includes its own identifier, sub-elements and control information, as if it is compulsory or optional, input/output parameters, etc. By the contrary, in existing EMLs goals are represented just as a textual description that informs participants about what they have to do at each task. In this way, there is no control information associated with the goal. Actually, this control information is available in the models of learning units, but in the relations among tasks. More important, this information is completely interleaved with the specification of the learning unit structure. In this way, it is very difficult and complex to change the goals of a learning unit without changing the structure.
- Second, the ES includes not just elements (namely: Goals, Roles, Environments, Data Elements), but also specifications related with the control, management and coordination of such elements. Particularly, an ES includes specifications to support the modeling of the issues such as the assignment of permissions to participants, the notification of events, the invocation of operations, the ordering among Sub-ESs and the temporal plan of Sub-ESs. The control involved in these specifications affects only to the elements included in the ES.

In this way, the ES provides a completely encapsulated model that facilitates a separation of concerns approach to the modeling of learning units. On the one hand, an ES does not make reference to any issue outside of its own contained elements. On the other hand, each one of these elements and specifications is considered at a different perspective. In practice, the model of an ES is made up by the sub-models of its Goals, Roles, Environments, Data Elements and specifications of Authorization, Awareness, Interaction, Order and Temporal. In addition all these elements and specifications are modeled as first-class entities, each one of them involving its own unique identifier, sub-elements and control information. This model is completetely different from the solution adopted on IMS-LD, where the several concerns are interleaved.

Figure 1 illustrates this ES concept. It is shown how an ES can include several elements: Goals (at least 1), Roles, Environments, Data Elements and Causal Descriptions; and specifications: Awareness, Authorization, Interaction, Order and Temporal. It is important to stress that the modeling of each one of these elements and specifications can be performed using different PoEML packages, namely: Functional, Participants, Environments, Data, Authorization, Awareness, Interaction, Order and Temporal. In the figure, it is also shown that Goals, Roles and Environments can also include Data Elements.

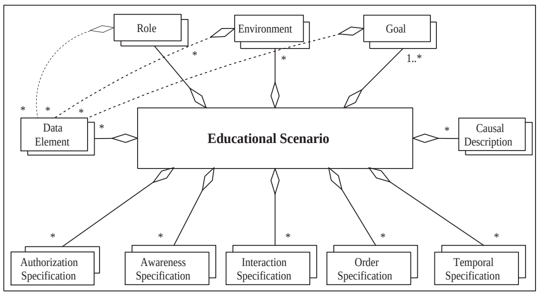

Figure 1: The *ES* elements and structure.

2.2.2 The Hierarchical and Structured Modeling of Learning Units

This conceptual view of PoEML is completed by considering the hierarchical aggregation of ESs to model learning units. Basically, any model of learning unit is made up by several ESs that are aggregated in a hierarchical way. As it is stated in the previous section, each ES includes its own Goals, Roles, Environments, Data Elements and Specifications. The hierarchical aggregation of ESs indicates that a certain ES (parent-ES) can be made up by several ESs (sub-ESs or children ESs). This hierarchical structure of the learning unit can be specified using the PoEML Structural package.

In addition to the hierarchical structure, a learning unit can require the establishment of relations between Goals, Roles, Data Elements, sub-ESs and Environments. These relationships have been identified in the separation-of-concerns analysis as particular flows:

- The *Functional Flow* involves the relations that can be established among Goals included in different ESs. These relations can be of two different types:
  - Completion relations. They can be used to indicate what conditions over the state of some Goals have to be satisfied to complete a certain Goal (generally this Goal is contained in a parent-ES and the other Goals in its sub-ESs). For example, to obtain the car driven license is required to pass the theoretical test and to have succeeded on the practical exam. Another example, to pass the theoretical test it is required to pass at least three of the four parts in which the test is divided.
  - Attempt relations. These relations can be used to indicate what conditions over the state of some Goals have to be satisfied to enable the attempt of other Goal (generally all these Goals are contained in sub-ESs of the same parent-ES). For example, the practical exam can be attempted when the theoretical test has been passed.
- The *Participants Flow* involves the relations that can be established among Roles included in different ESs. They are used to represent the transfer of participants among Roles. For example, the assignment of learners to a Role ”team”, mada up by 4 learners.
- The *Data Flow* involves the relations that can be established among Data Elements, indicating the transfer of data among them. For example: the grade obtained in the assessment of a questionnaire has to be transferred to the next ES and also maintained in the transcript of the learner.
- The *Control Flow* involves the relations that can be established among the sub-ESs of a certain ES. Basically, this flow indicates what sub-ES has to/can be done at a certain moment. It can be determined in two different ways:
  - In relation with the state of performance of other ESs. For example, when the learner pass the theoretical exam she can initiate any of the practices assigned. Then, when the learner finishes all the practices, she has to perform a practical exam.
  - In accordance with a temporal planification. For example, the presentation of the subject is going to be produced on March 11 at 5pm. Another example, the final exam has to be performed in less than 1 hour.
- There is not any *Environment Flow*, as the contents of the Environments cannot be transferred, but it is possible to establish relations between Environments. For example, the lab used to perform the final practice exam has to be the same lab in which the learners performed all the practices.

We would like to stress that the modeling of each one of these issues can be performed using a different PoEML package, producing as a result a specific submodel. In addition, this modeling is done in a very structured way, as the relations have to be established among elements included in the same ES or in its sub-ESs.

This hierarchical and structured approach is represented in Figure 2. There is as a main block, representing the Root ES, that contains two blocks representing sub-ESs: ES A and ES B. In addition, ES A contains three further sub-ESs: ES A.1, ES A.2 and ES A.3. The figure shows this structural arrangement and the relations established between Goals, Roles and Data Elements. It is quite obvious that the view in a single picture of all these issues is very confusing and complex. The PoEML separation on several perspectives provides a clearer view, enabling us to observe the different issues and relations in separated sub-models.

2.2.3 The Root ES

To complete this conceptual model it is necessary to introduce the concept of Root ES. The Root ES represents the global learning unit. It can include the same elements than any other ES. In addition, as it represents the global learning unit, it can also include further elements that capture the connections of the learning unit with the ”outside world”. This outside world is composed by real resources, tools, participants, environments and organizations that exist outside of the learning unit. For example, a learning unit model should enable to indicate that a certain PDF document has to be used. The connections between these external elements and the model of learning unit are established in the Root ES and they can be referenced from any sub-ES of the Root ES, independently of its depth level in the hierarchical ESs' structure. The specification of tools and organizations involves special features and it is performed in specific perspectives.

In this way, from a conceptual point of view, an ES is an encapsulated entity including all the elements that made up it and the specifications that define its structure and behavior. Furthermore, a root-ES is a hierarchical and structured model that can represent any piece of learning at different levels of aggregation, from simple lessons to complete curricula, and that includes all the connections with the ”outside world” required to perform its ”enactment”.

3 ADAPTATION

Adaptation is a central subject in e-learning and it can be considered in many different ways. In general, e-learning adaptation is focused on the optimization of certain measures (e.g., learning time, economic costs, user satisfaction) in order to improve the effectiveness and efficiency of the educational process. This section consider adaptation in the general e-learning domain firstly and then in PoEML.

3.1 Adaptation in e-Learning

Traditionally, adaptive hypermedia has captured the efforts in the e-learning adaptation domain (Henze and Nejdl, 2004), focusing on the student and on the contents (Brusilovsky, 2000) (Weber and Specht, 1997) (Bra et al., 2003). These proposals are based on the model of a single learner that is provided with contents in accordance to her/his characteristics and progress (determined through the grades obtained in tests and questionnaires). Nevertheless, this is just one of the approaches that can be followed to achieve learning. As in conventional face-to-face educational settings, many current e-learning systems are build on the basis of different pedagogical aproaches, such as constructivist or social ones. These systems provide functionalities that allow learners to communicate with other learners and teachers, to play, to experiment, to perform authentic activities, etc. Therefore, in this kind of e-learning systems, a broader view of adaptation should be supported, taking into account not just the contents and learner characteristics, but all the elements involved: participants, communications, activities, context, etc.

The search of a general solution to support adaptation in e-learning needs to take into account the variety of issues and approaches already considered. An interesting classification framework to analyze adaptation in e-learing is described in (Specht and Burgos, 2007) based on four main questionsis:

- *What parts or components in the learning process are adapted?* This question focuses on the part of an adaptive application that is adapted. Examples can be to personalize the pace of the instruction, adaptation of content presentations, the sequencing of contents, etc.
- *What information does the system use for adaptation?* In most proposals a learner model is the basis for the adaptation, taking into accout her/his knowledge, preferences, interests or cognitive capabilities. Nevertheless there are examples where the adaptation takes place to the learner setting, accessing devices, etc. Especially in social software the information for adaptation can come out of external information resources, collective logging information, or even contextual sensor information like the location of a learner.
- *How does the system gather the information to adapt to?* There are a variety of methods to collect information about learners to adapt to. Mainly implicit and explicit methods like described in works from user modeling can be distinguished.
- *Why does the system adapt?* This question mainly

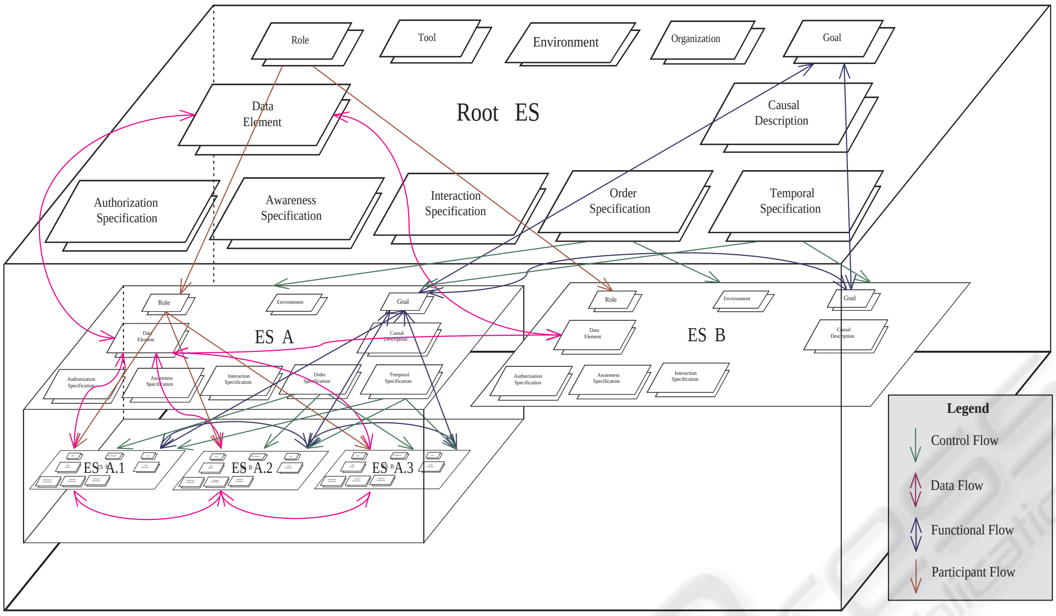

Figure 2: The modeling of learning units using an hierarchical aggregation of *ESs*.

focuses on the pedagogical models behind the adaptation. Classical educational hypermedia systems mainly adapt according for compensation of knowledge deficits, ergonomic reasons, or adapt to learning styles for an easier introduction into a topic.

3.2 Supporting Adaptation in PoEML

The support of adaptation in EMLs has been a main topic. IMS-LD is able to manage different types of adaptation (Burgos et al., 2006): Learning flow, content adaptation, evaluation and interactive problem solving support are best supported. Group adaptation is supported via administrative tools for user grouping and group properties and modification of a course on-the-fly can be partially implemented based on current runtime environment restrictions. Also interface-based adaptation is possible as long as the modifications are made inside the Unit of Learning and not in the player tool itself. Nevertheless, the provided solutions are complex because the models of learning units in IMS-LD have mane interdependences.

The support of adaptation in PoEML has been arranged in accordance with the previous four questions. In this way, the adaptations can be modeled in clear and simple ways. Basically, there is a direct relatioship among the language structure and the considered questions that facilitate the specification:

- *What parts or components in the learning process are adapted?* In PoEML these parts and components are given by the several perspectives (except the causal perspective). Each one of the perspectives involves the structures and control that can be modified.
- *What information does the system use for adaptation?* The information used for adaptation is included in the four aspects. Depending of the aspect it can be used data elements, events or decisions.
- *How does the system gather the information to adapt to?* In the case of data the gathering of information is performed using the data perspective. In the case of events, it has to be used the awareness perspective. In case of decisions, the way in which information is collected is managed by external tools (e.g., a voting application) that can be managed by the tools perspective.
- *Why does the system adapt?* The rationale about the adaptation can be maintained in the causal perspective.

4 CONCLUSIONS

The main approach adopted in PoEML is the separation-of-concerns. As in other engineering disciplines, the management of complexity is a key point to facilitate the solution of problems and the separation-of-concerns is a well-known technique to tackle with it. This strategy enables us to focus the attention on one perspective at each time, while abstracting from others. In this way, PoEML is not provided as a single set of elements, constraints and relationships, but it is made up by several sub-languages (packages), each one of them involving particular elements, constraints and relationships focused on a certain perspective.

The paper devotes a special attention to the support of the variety of adaptation. The separation-of-concerns approach adopted in PoEML is particularly appropriate to translate the general ideas managed about adaptation in e-learning to EMLs. In relation with other EMLs, uniquely IMS-LD has considered the support of adaptation. At this point, the solution adopted in PoEML is much more simple and powerful.

ACKNOWLEDGEMENTS

This work has been funded by the Spanish *Ministerio de Educacion y Ciencia* under grant TIN2007-68125-C02-02, and by the Galician *Conselleria de Innovacion e Industria* under grant PGIDIT06PXIB32 2270PR. It is also supported by the eContentplus programme ECP 2007 EDU 417008 (www.aspect-project.org), a multiannual Community programme to make digital content in Europe more accessible, usable and exploitable.

REFERENCES

Botturi, L. and Stubbs, T., editors (2007). *Handbook of Visual Languages for Instructional Design: Theories and Practices*. Idea Group Inc. (IGI), Hershey, PA, USA.

Bra, P. D., Aerts, A., Berden, B., de Lange, B., Rousseau, B., Santic, T., Smits, D., and Stash, N. (2003). Aha! the adaptive hypermedia architecture. In *HYPERTEXT '03: Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, pages 81–84, New York, NY, USA. ACM.

Brusilovsky, P. (2000). Adaptive hypermedia: From intelligent tutoring systems to web-based education. In *ITS '00: Proceedings of the 5th International Conference on Intelligent Tutoring Systems*, pages 1–7, London, UK. Springer-Verlag.

Burgos, D., Tattersall, C., and Koper, R. (2006). Representing adaptive elearning strategies in ims learning design. In *International Workshop in Learning Networks for Lifelong Competence Development, TEN-Competence Conference*, Sofia, Bulgary.

Caeiro-Rodríguez, M. (2007). *Handbook of Visual Languages for Instructional Design: Theories and Practices*, chapter PoEML: A Separation-of-Concerns Proposal to Instructional Design, pages 185–209. Information Science Reference.

Caeiro-Rodríguez, M., Marcelino, M. J., Llamas-Nistal, M., Anido-Rifón, L. E., and Mendes, A. J. (2007). Supporting the modeling of flexible educational units poeml: A separation of concerns approach. *Journal of Universal ComputerScience*, 13(7):980–990.

Engestrom, Y., Miettinen, R., and Punamaki, R.-L., editors (1999). *Perspectives in activity theory*. Cambridge University Press, New York, USA.

Henze, N. and Nejdl, W. (2004). A logical characterization of adaptive educational hypermedia. *Hypermedia*, 10(1):77–113.

Koper, R. and Olivier, B. (2004). Representing the learning design of units of learning. *Educational Technology and Society*, 7(3):97–111.

Koper, R., Olivier, B., and Anderson, T., editors (2003). *IMS Learing Design Information Model*. IMS Global Consortium, New York, NY, USA.

Paquette, G., editor (2004). *Instructional Engineering in Networked Environments*. Pfeiffer, San Francisco, CA, USA.

Parnas, D. L. (1979). *On the criteria to be used in decomposing systems into modules*, pages 139–150. Yourdon Press, Upper Saddle River, NJ, USA.

Specht, M. and Burgos, D. (2007). Modeling adaptive educational methods with ims learning design. *Journal of Interactive Media in Education (Adaptation and IMS Learning Design Special Issue)*, (8).

Strijbos, J. W., Martens, R. L., and Jochems, W. M. G. (2004). Designing for interaction: six steps to designing computer-supported group-based learning. *Comput. Educ.*, 42(4):403–424.

Vignollet, L., David, J.-P., Ferraris, C., Martel, C., and Lejeune, A. (2006). Comparing educational modeling languages on a case study. In *ICALT '06: Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies*, pages 1149–1151, Washington, DC, USA. IEEE Computer Society.

Weber, G. and Specht, M. (1997). User modeling and adaptive navigation support in www-based tutoring systems. In *Proceedings of the 6th international conference on user modeling*, pages 289–300.