Towards a Common Graphical Language for Learning Flows: Transforming BPEL to IMS Learning Design Level A Representations

Pythagoras Karampiperis and Demetrios Sampson

Department of Technology Education and Digital Systems, University of Piraeus, Greece

&

Advanced e-Services for the Knowledge Society Research Unit, Informatics and Telematics Institute, Centre for Research and Technology Hellas, Greece

{pythk, sampson}@iti.gr

Abstract

The need for e-learning systems that support a diverse set of pedagogical requirements has been identified as an important issue in web-based education. Until now, significant R&D effort has been devoted aiming towards web-based educational systems tailored to specific pedagogical approaches. As a response to pedagogical concerns towards standardization and interoperability needs, IMS Learning Design (IMS LD) specification was introduced. Nevertheless, despite the wide adoption of the IMS LD specification still a common language for graphically representing learning flows is missing. This is due to the fact that although the IMS LD specification provides the means for technically representing learning flows (that is, sequences of activities), it does not provide guidance on how these flows could be represented in a human understandable way. As a result, there exist several authoring tools for designing learning activities that export content packages conformant with IMS Learning Design, but these tools are using different representations of the learning process. On the other hand, there exist standards for creating human understandable graphical representations of processes (i.e. the Business Process Modeling Notation standard), that could be useful in providing common representations of learning flows. In this paper we examine the ability of using BPMN as a common representation notation for learning flows modeled using the Business Process Execution Language (BPEL) and present an algorithm for transforming BPEL Workflows to IMS Learning Design Level A learning flows.

1. Introduction

During the last years, several web-based educational systems have been proposed aiming to address specific pedagogical approaches (e.g. MindTools [1] addressing constructive learning, ActiveMath [2] addressing problem-based learning etc.). The main drawback of those systems is that they are closed, self-contained systems that cannot inter-exchange either educational content or activities. Additionally, the supported content and learning scenarios are a-priori designed to serve and support a specific pedagogical approach. As a result they are non-flexible in supporting different pedagogical approaches and they require extensive redesign effort in order to be used in different domains. On the other hand, several Learning Management Systems already exist (e.g. Blackboard, WebCT, Lotus Learning Space, learn eXact etc.) delivering web-based courses with limited pedagogical flexibility, but are able to share/exchange learning content in the form of learning objects.

As a response to pedagogical concerns towards standardization and interoperability needs, IMS Learning Design specification [3] was introduced, offering a standardized way to associate educational content, activities and actors in a learning scenario, enabling the inter-exchange of not only content, but also pedagogical scenarios. Nevertheless, despite the wide adoption of the IMS Learning Design specification, still a common language for graphically representing learning flows is missing. This is due to the fact that although the IMS LD specification provides the means for technically representing learning flows (that is, sequences of activities), it does not provide guidance on how these flows could be represented in a human understandable way. As a result, there exist several authoring tools for designing learning activities (e.g. ASK-LDT [4], LAMS [5], MOT+ [6]) that export content packages conformant with IMS Learning Design, but these tools are using different representations of the learning process.

On the other hand, there exist standards for creating human understandable graphical representations of processes (i.e. the Business Process Modelling Notation standard [7]), that could be useful in providing common interoperable representations of learning flows.

In this paper we examine the ability of using BPMN as a common representation notation for learning flows modelled using the Business Process Execution Language (BPEL) [8] and present an algorithm for transforming BPEL Workflows to IMS Learning Design Level A learning flows.

2. Representing Learning Flows with IMS Learning Design Specification

In the IMS Learning Design specification an activity can be formally defined as a triple containing the content that is delivered by an educational system, the actors participating in the learning activity (such as the learner or a group of learners, the tutor etc.) and their corresponding interactions. These interactions include three types, namely, interactions with the learning content, interactions with the educational environment and interactions between the participating actors. Following the notation of the IMS LD, the different participating actors are called roles and they are divided in two main classes, namely the Learner Role and the Support Role. These roles can be sub-typed allowing the definition of refined learner and support roles. Furthermore, the IMS LD formulation provides a notation schema for the description of activities taken place in a specific Environment. These activities can be Learning or Support Activities..

To represent the learning flow (that is, the sequence of activities performed by each role), the IMS LD notation language uses the Act element. An act represents a logical categorization of a set of activities. In each act, several roles can participate by performing a Rolepart. Each Rolepart represents the activities performed by the corresponding role in a specific act and contains an Activity Structure, which represents the sequence of the performed activities. An activity structure can use a nested structure of activities and/or other activity structures defining the branching of the learning flow.

BPMN Graphical Element	Description	BPMN Graphical Notation	IMS LD Component
Pool	A Pool represents an actor (role) in a learning process.		Role
Lane	A Lane is a horizontal sub-partition within a Pool, for logically organizing and categorizing activities.		Act RopePart
Activity	An activity represents a working item (task) that one or more actors (roles) of the learning process perform.		Activity
Sequence Flow	A Sequence Flow is used to show the order that activities will be performed in a learning process.		Flow of activities represented as nested Activity Structures
Gateway	A Gateway is used to control the divergence and convergence of Sequence Flow. Thus, it will determine branching, forking, merging, and joining of paths. Internal Markers will indicate the type of behavior control.		Flow of activities represented as nested Activity Structures depending on the type of the Activity Structure (sequence of selection)

Table 1 Mapping of BPMN Core Graphical Elements to IMS Learning Design Core Components

The Business Process Modeling Notation (BPMN) [7] standard provides the means for creating human understandable graphical representations of processes, and as such, can be used for defining graphical representations of educational processes (learning flows) modeled with the IMS LD specification. Table 1 presents the mapping of BPMN core graphical elements with the IMS LD core elements.

3. Business Process Modeling Notation and Business Process Execution Language

As presented in Table 1, BPMN can be used to represent graphically a learning flow that has been modeled using IMS Learning Design specification. However, due to the fact that the sequencing information in an IMS LD document is modeled through the use of nested activity structures, the transformation of an IMS LD learning flow to a representation according to BPMN (and vice versa) is not a straightforward process. To overcome this, another intermediate model is required with the following key characteristics:

- This model should be low level (represented in XML), so as to be able to be converted to and/or retrieved from the XML representation of the IMS LD specification.
- The elements of this model should directly map to BPMN graphical design elements, so that the transfer from the XML representation to the graphical representation (and vice versa) would be straightforward.

A model that covers the above mentioned requirements is the Business Process Execution Language (BPEL) [8]. BPEL is an XML based language that represents a process, and can be directly mapped to BPMN graphical design elements [9]. When mapping a BPMN diagram to BPEL, a decision must be made as to the basic structure of the BPEL document. That is, will the BPEL format be based on the BPEL graph structure (the flow element) or the BPEL block structure (the sequence element)? This choice will affect how much of the BPMN Sequence Flow will map to BPEL link elements. Using a block structure as the foundation for mapping, link elements are only used when there exist parallel activities. Using the graph structure as the foundation for mapping, the Sequence Flow will map to link elements contained within a flow element.

The main advantage of the mapping of BPMN to BPEL's graph structure (rather than BPEL's block structure) is that the result of the mapping defines a directed graph that can be transformed to the IMS Learning Design notation language. Such an approach bares the potential to improve interoperability between high level IMS Learning Design tools. A proposed algorithm for transforming BPEL workflows to IMS LD level A learning flows, is presented in the following section.

4. Algorithm for Transforming BPEL Workflows to IMS LD Level A Learning Flows

Let us call Activity Graph, a directed graph that represents the BPEL workflow, consisting of nodes

(corresponding to workflow activities) and directed links (corresponding to the flow between two activities), with the following additional definitions.

Term	Definition
Source of a Link and Target of a Link	The starting activity, whereas, as *target* of a link we define the activity that follows the source one.
Root Node	Any node which is the source of more than one links contained in the Activity Graph.
Split Node	Any node which is the source of more than one links contained in the Activity Graph
End Node	Any node which is the target of one or more links contained in the Activity Graph and there isn't any link with this node as a source
End Split Node	Any Split Node with all children End Nodes

Table 2 Definition of Algorithm Terms

The transformation algorithm then, consists of the following steps:

Step A: Calculate Sequences

1. Starting from each *End Node*, go through the reverse of the links defined and calculate all activity sequences until reaching a *Split Node*
2. For each sequence of activities found:
   a) Define an activity structure with type equal to sequence, containing all the activities of the sequence found in reverse order
   b) Delete all the links between the activities of this specific sequence
   c) Delete the activities contained in this sequence from the Activity Graph
   d) Replace the target of the link between the relevant *Split Node* and the *Root Node* of this sequence, with the defined activity structure

Step B: Calculate Selections

1. Find all the *End Split Nodes* of the Activity Graph
2. For each one of them:
   a. Define an activity structure with type equal to selection, containing all the children of this *End Split Node*
   b. Delete all the links between this *End Split Node* and its children
   c. Delete all the children of this *End Split Node* from the Activity Graph
   d. Set the defined activity structure as a child to this *End Split Node*

Step C: Termination

1. If Activity Graph contains 1 Node
   then
      a. Define a Rolepart with reference to the remaining node (activity or activity structure)
      b. Terminate
   else Repeat from Step A

5. Conclusions

In this paper we discussed the problem of learning activities graphical design representations used by existing IMS Learning Design compliant authoring tools, and examined the ability of using BPMN as a common representation notation for learning flows modeled using

the Business Process Execution Language (BPEL). Then, we presented an algorithm for transforming BPEL Workflows to IMS Learning Design Level A learning flows.

6. Acknowledgements

7. References

[1] Jonassen, D. H., Hernandez-Serrano, J., Choi, I. (2000) "Integrating constructivism and learning technologies". In J. M. Spector and T. M. Anderson (Eds.), *Integrated and holistic perspectives on learning, instruction and technology: Understanding complexity,* Dordrecht: Kluwer, pp. 103-128.

[2] Libbrecht, P. (2004) "Authoring Web Content in ActiveMath: From Developer Tools and Further". In Proc. of the *3rd International Conference on Adaptive Hypermedia and Adaptive Web-based Systems (AH2004)*, Workshop on Authoring of Adaptive and Adaptable Educational Hypermedia, Eidhoven, Netherlands, pp. 455-460.

[3] IMS Global Consortium (2003) "Learning Design Specification Version 1.0", Available online at: http://www.imsglobal.org/learningdesign/index.cfm, last visited 15 November 2006.

[4] Sampson, D., Karampiperis, P., Zervas, P. (2005) "ASK-LDT: A Web-Based Learning Scenarios Authoring Environment based on IMS Learning Design", *International Journal on Advanced Technology for Learning*, vol. 2(4), pp. 207-215.

[5] Dalziel, J. R. (2003) "Implementing Learning Design: The Learning Activity Management System (LAMS)", In Proc. of the *20th Annual Conference of the Australian Society for Computers in Learning in Tertiary Education*, pp. 593-596.

[6] Paquette, G., Teja, I., Leonard, Mi., Lundgren-Cayrol, K., Marino, O (2005) "An Instructional Engineering Method and Design Tool" In R. Koper and C. Tattersall (Eds.), *Learning Design: A Handbook on Modelling and Delivering Networked Education and Training*, Springer-Verlag, pp. 161-184.

[7] OASIS (2003) "Business Process Execution Language for Web Services" Available online at: ftp://www6.software.ibm.com/software/developer/library/ws-bpel.pdf, last visited 15 November 2006.

[8] Business Process Management Initiative (2006) "Business Process Modeling Notation Specification" Available online at: http://www.bpmn.org/Documents/OMG%20Final%20 Adopted%20BPMN%201-0%20Spec%2006-02-01.pdf, last visited 15 November 2006.

[9] White, S. A. (2006) "Using BPMN to Model a BPEL Process", Available online at: http://www.bpmn.org/Documents/Mapping%20BPMN%20to%2 0BPEL%20Example.pdf, last visited 15 November 2006.