SALO
Sharable Auto-adaptive Learning Object

Ignacio Gutiérrez Menéndez, Mª del Puerto Paule Ruiz and Juan Ramón Pérez Pérez
Universidad de Oviedo, Dpto. Computer Science, Oviedo, Spain

Keywords: e-Learning, Dynamic languages, Adaptation.

Abstract: New pervasive computing scenarios such as ubiquitous learning (u-learning) and mobile learning (m-learning) demands real time adaptation. Getting such adaptation couldn't be possible with the current technologies. It is necessary a change in the development of pervasive e-learning systems using dynamic technologies and including them in both: platforms and e-learning content specifications. In this paper we define and develop the concept of Sharable Auto-Adaptive Learning Object (SALO) like a Learning Object which includes content and describe its behaviour thanks to dynamic languages. Such features allow it to change and include new resources and behaviour, in a dynamic way and using the user's context at 'anytime, anywhere, from any device'.

1 INTRODUCTION

The reuse of electronic resources is a complex matter that should guarantee more attractive, effective, efficient and accessible learning experiences for students (Campbell, 2003). Ubiquitous learning allows students to learn in any place and moment and with any device (Abarca et al., 2006). A context-aware approach is needed because each learner has unique needs and context, but also provides focused and structured learning (Farrell et al., 2004).

The interoperability in the educational content has allowed the creation of the learning objects (IEEE, 2002) (Willey, 2000). The use of *open standards* in course construction has brought about Content specifications such as Sharable Content Object Reference Model (SCORM; http://www.adlnet.gov), IMS Learning Design (IMS-LD; http://www.imsglobal.org) and IMS Common Cartridge (IMS-CC; http://www.imsglobal.org/cc/).

The specifications work with an object model that allows serializing the learning objects, independently of the tool or application which they have been created or processed. However, the actual specifications present, under our point of view, a limitation: the impossibility of storing or serializing the objects behaviour. Stored and distributed learning objects are just content containers without any own behaviour. Actual Learning Management System (LMS) must provide the behaviour.

From our point of view, this way of implementing the behaviour makes impossible an adaptation in real time, restricting the creation of ubiquity learning (Weiser, 1993), mobile learning and context-aware learning systems. Our proposal follows changing this approach. The aim is to generate learning objects with data and behaviour. This idea is not new, in (Bailey et al., 2002) appears the concept of the Fundamental Open Hypermedia Model (FOHM) and in (Zouaq et al., 2008) presents an ontology-based approach for the dynamic generation of learning knowledge objects (LKO).

We have developed and implemented a SALO's (Sharable Auto-adaptive Learning Objects), objects which are fully autonomous, where both content and behaviour are stored. In order to achieve this approximation, it is necessary to use dynamic adaptation (Yang et al., 2002), which allows the possibility of adapting the learning objects to the context in which they are being executed.

2 RELATED WORK

2.1 Adaptive Educational Systems

In general, the Adaptive Educational Hypermedia Systems (Brusilovsky, 2001) are based on the use of Adaptive hypermedia techniques (Brusilovsky,

Gutiérrez Menéndez I., del Puerto Paule Ruiz M. and Ramón Pérez Pérez J..
SALO - Sharable Auto-adaptive Learning Object.
DOI: 10.5220/0003303103870390
In *Proceedings of the 7th International Conference on Web Information Systems and Technologies* (WEBIST-2011), pages 387-390
ISBN: 978-989-8425-51-5

1996) to solve a number of the problems associated with the use of educational hypermedia. Different ways of implementing the adaptation has been used in educational systems (Paule et al., 2008):

1. Use of rules that activate one another when certain actions happen, these rules can also create new rules, e.g.: AHA! (http://aha.win.tue.nl) and MOT (http://www.dcs.warwick.ac.uk/~acristea/mot.html) are both systems that belong in this first category.

2. The second kind is the creation of proprietary objects with a particular format created for each application. AuldLinky (Michaelides et al., 2001) is a system that belongs to this category.

3. In (Cristea et al., 2007) the authors describe two new representation languages that emerged in the process of adaptation: a common format for defining the static material, CAF, and an extended adaptation language for the description of the dynamic behaviour, LAG.

In general, the use of rules is important to support the adaptive capabilities of the system; however, the adaptive applications generate a content that cannot be reused by other systems. Regarding the creation of proprietary objects for each application, sharing these objects among different applications is difficult since these objects do not follow any standard format. The use of LAG is a step forward in this field, although it is not compatible with the existing specifications on the market.

2.2 Adaptation in the e-Learning Content Specifications

Merging adaptive hypermedia with e-learning seeks to adapt e-learning information according to the user needs and context. The specifications use the sequencing in order to get the adaptation, SCORM has the SN (Sequencing & Navigation) which by means of rules and depending on user model it allows the LMS to show the right content. The decision taken by the sequencing is done by a static engine and a number of rules that are executed according to some data returned by the learning object.

The learning objects are mere data containers, but without any related behaviour, leaving this behaviour to the LMS, so developers need to build a set of sequencing rules and an API defined by the content specification. It would be a great advance to provide learning objects with a complex and independent behaviour from the platform. With this goal, LMS just needs to care about the content administration task, releasing it from the task of managing object relationships.

3 OUR PROPOSAL: SHARABLE AUTO-ADAPTIVE LEARNING OBJECTS

We consider that an auto-adaptive learning object must have the following features:

1. Own content according to the learners' needs and context
2. Own behaviour allowing:
   - To be sufficiently aware of the learners' context
   - To be able to take full advantage of pervasive computing
3. To be reused many times independent of software and platforms versions
4. To be interoperable; compatible with the content specifications and with current LMSs

The first three features require dynamic adaptation. In order to get dynamic adaptation, we need to use reflective and reflection techniques (Pattie, 1987) and those techniques are offered currently by dynamic programming languages (Ortín et al., 2005). From our point of view, it is necessary the addition of dynamic features to a LMS to get context-aware, ubiquity and multi-mode e-learning systems. These features implies many changes in design and development of actual learning platforms, however these changes are compatible with the actual e-learning frameworks such as OKI(http://www.okiproject.org), ELF (http://www.elframework.org) or IMS Abstract Framework(http://www.imsglobal.org/af/), because these frameworks define a set of services which are independent of the programming language.

3.1 Our Proposal: Sharable Auto-adaptive Learning Object (SALO)

We define as a Shareable Auto-Adaptive Learning Object (SALO) as a *learning object which is able to describe its own behaviour, being independent from the LMS, adapting itself to the context and being reusable for other e-learning systems.*

A SALO contains data and behaviour (Figure 1), packed both of them with SCORM specification, which makes it independent of the LMS.

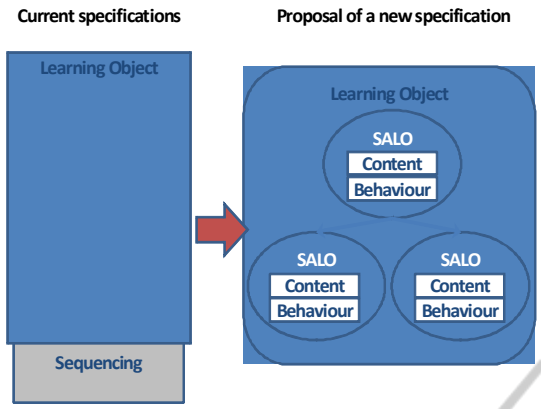

Figure 1: Transition from LO to SALO.

The creation process of a SALO is formed by an editing process plus a run time process and it is divided into four parts:

- To choose the content: pdf, html, or any kind of educational resource.

- Addition of behaviour: It consists on a set of techniques and methods of adaptive hypermedia and behaviour rules, which define the instructional design of the LO and they are implemented in Ruby. Teacher has an edit tool in order to include such techniques and rules into a SALO.

- Packing: SALO's learning content is packed in a similar way as SCORM does, but also our process packs its behaviour. Later on, content and behaviour will be used by the dynamic engine inside a LMS to make the user adaptation possible.

- Adaptive environment: It is done by the LMS and the adaptive engine. The adaptive engine loads the content and behaviour of the LO in memory, starting its execution. The engine also processes the SALO´s learning content and returns it to the LMS. Once the interaction between SALO and the user has finished, a terminate request is gotten and according to the object's behaviour the next SALO is selected and the process begins again. If the user decides to quit the course, the state is stored in the system.

The engine has two main functions: load and run the SALO. Firstly the loading process connects to a repository and it will take every data and resources building a current LO. Also, it will take the behaviour which will be transformed, using reflective techniques, into methods that will be attached to the LO previously built producing a SALO in memory. Secondly, the SALO is executed and it uses its new methods to change its view to the user.

The previously described process adapts the SALO to the user at runtime and can be modified just adding/removal behaviour in the repository. At this point it is interesting to highlight that every addition we have done to the SCORM specification, to create a SALO, is compatible with the current SCORM format, because, this extra information will be ignored by the current LMSs.

3.1.1 Preliminary Results

We have tested the implementation of a SALO with 20 learners of University of Oviedo. The goal of this test is to show that the adaptation technologies proposed is feasible.

For the test every user had to complete two courses - with and without adaptation - and they must be done twice each one. The subjects chosen for the courses have been related to "golf" and "basics of programming".

The results, which are not conclusive, shows that most learners appreciate the combination of adaptation technologies adopted and the support offered by the implementation because they notice as significant visual changes between the adapted course and the course without adaptation. These results go in parallel with other similar studies done by other researchers in the Adaptive Hypermedia Educational Systems.

4 CONCLUSIONS AND FUTURE WORK

Nowadays, researches related to ubiquity and context-aware e-learning systems are in great expansion. These systems need adaptation to the current user context in real time. Dynamically typed programming languages are able to include dynamic adaptation to the e-learning content, as it has been shown in this paper, however using dynamic technologies means a change in design, develop and implementation of the current LMS´s.

A SALO changes the meaning of the current LO's, giving them behaviour and making them independent of the LMS´s where they are being run. They are compatible with current content specifications like SCORM, and also with LMs´s like Moodle. Nowadays there are LMS´s implemented with dynamic languages like Moodle, implemented in PHP. This kind of implementation of LMS makes easier to add the adaptation engine which allows the SALO's execution.

Currently we are exploring the possibility of creating new adaptive strategies in runtime according to a user model and how to do the

connection between the user model and its translation in adaptive rules.

ACKNOWLEDGEMENTS

This work has been partially funded by the Department of Science and Technology (Spain) under the National Program for Research, Development and Innovation; project TIN2008-00276.

REFERENCES

Abarca, M. G., Alarcon, R. A., Barria, R., Fuller, D.(2006): Context-Based e-Learning Composition and Adaptation, *Mapping OWL-DL into ORM/RIDL, Herrero et al. (Eds.): OTM Workshops, LNCS 4278*, pp. 1976 – 1985.

Bailey, C., Hall, W., Millard, D. E. & Weal, M. J. (2002): Towards Open Adaptive Hypermedia. In Proceedings of *2nd International Conference Adaptive Hypermedia and Adaptive Web-Based Systems, LNCS 2347, pp 36-46.*

Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. *User Modelling and User Adapted Interaction, 6 (23), 87-129.*

Brusilovsky, P. (2001). Adaptive Educational Hypermedia, Proceedings of the *Tenth International PEG Conference, Digital Media Institute, Tampere University of Technology,* Tampere, Finland.

Campbell, L. M. (2003) Engaging with the learning object economy. In A. Littlejohn (Ed.), Reusing online resources: A sustainable approach to e-Learning, London*: Kogan Page*.

Cristea, A., Smits. D., De Bra, P. (2007), Towards a generic adaptive hypermedia platform: a conversion case study. *Journal of Digital Information (JoDI), Special Issue on Personalisation of Computing & Services, Vol 8, No 3*.

Farrell, R. G., Liburd, S. D., Thomas, J.C.(2004), Dynamic assembly of learning objects, Proceedings of the *13th international World Wide Web conference, pp. 162-169.*

IEEE. (2002), Learning Technology Standards Committee (PDF), Draft Standard for Learning Object Metadata (2002). *IEEE Standard 1484.12.1*, New York: Institute of Electrical and Electronics Engineers. Retrieved Sept 1, 2010, from http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf.

Michaelides, D. T., Millard, D. E., Weal, M. J. and De Roure, D., C. (2001) Auld Linky: A Contextual Open Hypermedia Link Server, In Proceedings of the *7th Workshop on Open Hypermedia Systems*, ACM Hypertext 2001 Conference Aarhus, Denmark.

Ortín, F., Díez, D (2005) Designing an Adaptable Heterogeneous Abstract Machine by means of Reflection. *Elsevier Information and Software Technology, 47 (2), 81-94.*

Pattie Maes. (1987). Computational Reflection. *PhD. Thesis. Laboratory for Inteligence Artificial.* Vrije Universiteit Brussel.

Paule, M. P., Fernández, M. J., Ortín, F., Pérez, J. R. (2008). Adaptation in current e-learning systems. *Elsevier Computer Standards and Interfaces, 30 (1-2)* 62-70.

Weiser, M. (1993), Some Computer Science Issue in Ubiquitous Computing. *Communications of the ACM.* 36(7) 74-84.

Wiley, D. A. (2000) Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. En D. A. Wiley (Ed.), The Instructional Use of Learning Objects. Retrieved Sept 1, 2010, from *http://reusability.org/read/chapters/wiley.doc.*

Yang, Z., Cheng, B. H. C., Stirewalt R. E. K, Sowell, J., S. Sadjadi, M., McKinley, P. K. (2002), An Aspect Oriented Approach to Dynamic Adaptation WOSS '02, *Charleston, SC, USA.*

Zouaq, A., Nkambou, R., Frasson, C. (2008): Bridging the Gap between ITS and eLearning: Towards Learning Knowledge Objects. *Proc. Ninth Int'l Conf. Intelligent Tutoring Systems (ITS '08), pp. 448-457.*