Using Graphs in Developing Educational Material

Thanassis Hadzilacos, Dimitris Kalles, Dionysis Karaiskakis, Maria Pouliopoulou
Hellenic Open University
Laboratory of Educational Material and Educational Methodology
Patras, Greece
{thh,kalles,karaisk,mpouliop}@eap.gr

Abstract. Distance learning institutions need to find a way to transplant the benefits of conventional tutoring practices into the development of digital content that is conducive to students' learning needs. Therein lie two great challenges: promote real distance learning effectively and, at the same time, try to accommodate the ability of humans to learn via collaboration. We have proposed the development of Learner's Open-and-Distance-Learning courses as both a theoretical model and an applied methodology to be one of our key priorities and describe how this concept co-evolves with web mining and institutional infrastructures.

Keywords: distance learning, interactive learning and teaching tools, authoring tools, modelling

1 Introduction

Developing an educational experience for a learner has at least two cornerstones: the existence of educational material and the organization of activities with that material. For example, a textbook is a repository of educational material. Reading it chapter by chapter is an educational activity. Consulting selected book parts when trying to solve an exercise is a totally different activity.

Meaningful educational experiences are usually based on the organization of carefully designed activities on quality educational material. The shrewd organization and the careful design necessarily cover some aspects of resource planning, such as how much time the learner is supposed to dedicate to the activity or, what is the sequence of activities that will best attain the educational goal. They also cover conventional aspects of design, such as the target audience and, the combination of tools to attain the goal. Detailed planning of learning activities, apart from the significant effort needed by the course designer, reduces the control students have over their own learning [1]. Learner support services [2] were proposed to provide individualized advice, but usually at a significant cost, especially in large scale applications and in ODL. Also note that educational experiences can be turned into educational material themselves. For example, watching a fellow student carry out an experiment in chemistry certainly produces an educational experience.
Furthermore indirect collaboration (based on observation, for example) can also significantly enhance the learning experience. Social Navigation [3] can be direct but

also indirect based i.e. on the traces of others. Those are quite significant in ODL where learners are supposed to have control on planning and implementing their learning but also in more informal learning settings like Communities of Practice.

In this contribution, we present a conceptual artefact, termed a **Learner's ODL course**, which, we claim, is a generic model that is suitable for accommodating the practices of the educational process, both solitary and collaborative, while still allowing room for developing new abstractions. Its real importance is in that it serves as a conceptual framework around which we attempt to integrate the technologies that are available to us, at any given time point.

We are careful to note that the educational process comprises of observable and explicitly initiated activities, as opposed to the learning process which is ad hoc and may or may not be a direct or indirect outcome of the educational process. After all, education does not necessarily result in measurable learning.

The rest of this paper is structured in five sections. We first briefly review the key stakeholders of the educational process. We then move to present a theoretical model of that process and argue why this model is a good springboard for the deployment of sophisticated data analysis applications (in the web mining context) that can spur the development of personalization services. We then discuss the practical issues of tool deployment and relate these issues to a large on-going application, before concluding by highlighting the context of an organization that is heavily investing in integrating its ICT infrastructures.

2 Background

Depending on how one views the educational process there are distinct components of it which become eminent during the observation. Even if each observer does in fact glimpse all components of the process, the emphasis is always on some key ones, which in turn may be different across observers.

A teacher, for example, usually views the educational process as a set of lectures to be delivered to an audience. Peripheral aspects of this view concern the distribution and grading of assignments and examinations. Another peripheral aspect, but also an easily overlooked one, is the personal improvement of a teacher's ability to deliver the same content over time, either by reflecting on the feedback of students or by collaborating with fellow tutors who are delivering the same course in parallel.

A learner, on the other hand, may or may not attend lectures. Attending lectures is only one of the activities that the learner has at his disposal. Studying, experimenting and collaborating are all activities that help hone a skill or develop knowledge about a subject. Informal communication and collaboration among peers is a key aspect of a learner's activities that a teacher may have little, if any, influence. In such collaboration views and homework solutions can be exchanged. Unless the teacher has explicitly designed an assignment to stimulate such communication, the indirect learning effects of the peer collaboration arrive by luck rather than by design.

Appreciating the difficulty of directly designing in detail such communication (and, then, by monitoring its implementation), one cannot fail to hypothesize that the a posteriori analysis of the peer collaboration process may lead to the identification of information nuggets of this process. Such nuggets can, as in any decision context, lead to the formulation of concise design advice for future exploitation. That, in turn, will be easier to disseminate to tutors for assistance and feedback purposes.

If we consider teachers and learners to operate at roughly the same level of education, we can move up one level and consider the educational system view. At that view, one deals with providing the educational material at a suitable scale for the student population and setting and monitoring quality issues in the delivery of education (i.e. scope of educational activities, depth and breadth of material, academic prerequisites across subjects, attendance logistics, etc.). Note that, at that level, the delivery mode of education (on-line, physical presence, etc.) is simply another component of that view.

Going a level down from teachers and learners one deals with educational material per se (books, instruments, software, etc.) and the development of blueprints or guides for using that material (solution manuals, demo software activities, etc.). At that level one would also address infrastructure issues.

For each of the above four views (and it should be obvious that the list is not exhausted here), it would be difficult to argue that they are unrelated. These views are not (and should not be) orthogonal, but they help focus the attention of people active in each level towards a common background of experience, expectations, and norms that allows for the smooth exchange of information within the boundaries of that view and across views. Still, with today's environments, it is easy to see that the two middle layers are the ones that offer the most potential for the emergence of communities of practice, mostly via the explicit sharing of experiences and via collaboration on the same task.

3 A Learner's ODL Course as a Model for the Educational Process

A graph-theoretic model of a Learner's ODL course is a computational model. It builds on top of some basic components which are elaborated below and it involves, at several points, activities of the stakeholders as described above.

A *learning object* is any piece of (multimedia) data or program whose purpose (intention) is to be used for learning. A learning object can be recursively defined as a set of learning objects. Examples of *learning objects* are the following: the text of Odyssey, MS Word, Sketchpad, a video lecture, a set of multiple choice questions, a

Euclidean geometry high school textbook, an MS Powerpoint presentation of organic compounds.

A *learning task* is a task whose purpose is learning. Examples of learning tasks are the following: read, solve an exercise, write a program, practice a musical instrument, draw a picture, design a database, make a summary, think over, correct, argue for/against.

A *learning activity* is an ordered pair: (learning object, learning task). Examples of learning activities are the following:

- Write a program to add two numbers (learning task) using a C++ compiler (learning object)
- Write down (type to the computer) what you hear (the learning object is a digitized dictation) and then check the spelling errors (in fact the learning object is the set {word processor, soundtrack, speller}).

A *learning environment* is a directed labeled multigraph (*LA*, *P*), where *LA* is a set (of vertices or, nodes) of learning activities and *P* is a bag (of edges) of labeled precedents. A multigraph is a “graph whose edges are unordered pairs of vertices, and the same pair of vertices can be connected by multiple edges” (Dictionary of Algorithms and Data Structures, National Institute of Standards and Technology (NIST), http://www.nist.gov/dads/). Examples of labeled edges are the following:

- From node LA5 to node LA15 “if you found LA5 very easy to do”
- From node LA5 to node LA100 “if you found LA5 very interesting”
- From node LA5 to node LA3 “if you did not manage to complete the task of LA5 satisfactorily”

A *reference node* is (a learning activity that is) connected to all other nodes via bidirectional (unlabeled) edges. Examples of *reference nodes* are the following:

- Dictionary (to look up a word or phrase)
- Calculator (to perform an arithmetic operation)
- On-line discussion (to communicate with a tutor or with fellow learners)

A *learning experience* (or, a *learning trip*) is a path (sequence of connected learning activities) on the learning environment graph.

A learner's note is a data structure attached to a specific node by a specific learner. A *learner's note* includes structured data fields (learner/user id, timestamp, access rights, etc.) and any (multimedia) data the learner chooses to attach (for example, files). Examples of *learner's notes* are the following:

- The list of adjectives asked for in example B1.
- A text that criticizes the effectiveness of the learning activity (node).
- A new soundtrack of the dictation (left by a student who found the pronunciation incomprehensible).
- A comparison or a synopsis of the past 10 notes left on the current learning activity (node).

A *learning environment communication system* is a communication system (such as email, discussion forum, etc.) with content consisting only of (pointers to) learner's notes. Examples of such content are the following:

- From a student to his teacher “Here is the list of adjectives asked for in LA5”.
- From a student to all other students “I found LA12 particularly useful, you can look up my comments in the note attached”.
- From a teacher to his students “Before attempting task LA112 read my note there”.

A *learning activity control block* is a snapshot of the usage of all the above in the context of a particular learner. It is a data structure containing (at least) the following fields:

- learner/user id
- timestamp
- (pointer to) learning object
- (pointer to) learning task
- (pointer to) learner’s note

A learning experience may well be a single-session path; for example, a learner dedicates a good solid hour to navigating the educational material along a particular line. A learning experience may also be a sequence of such paths; for example, we usually “remember” where we stopped studying (for a short or long break), and can resume from that point. A (metaphorically speaking) concatenation of such paths delivers a longer path that can still be a learning experience.

The graph-theoretic model also allows us to build in temporal information in the navigation paths. As a matter of fact, relative temporal information is inherently available in a path (sequence of node visits). Furthermore, the annotation of edges in terms of actual time spent in an activity before moving on to the next is a straightforward enhancement.

The detour ends here by noting that the above considerations simply suggest that, after we get the initial graph-theoretic model fixed, there exist a set of computational processes that will allow us to define arbitrarily complex layers of information based on the ground data. We elaborate on that in a following section.

4 Tool Deployment Issues

We start by noting that the theoretical model can be in principle implemented using rudimentary technology, such as hyper-linked files of conventional office-type applications, where educational assets can be grouped together in repository-type worksheets. Assets can then be drawn to compile learning activities. Such tools offer relatively smooth short learning curves for data collection and web publishing too.

As an example, Figure 1 shows how MS Excel could be used to design a learning environment. A learning activity is composed by an asset and by a learning task (allowing, of course, for some terseness in representation: when no task is shown for a text asset, the implicit task is “read”). Indentation can be used to designate priorities and preferences; this allows transitions between activities to be tagged (potentially) by attributes such as “was it interesting?”.

The divide-and-conquer approach ...			
Learning Task	**Learning Object**		
read	2.3.1 The divide-and-conquer approach		
		... the first two paragraphs	
write	Think how you would apply the above principle to		
read	2.3.1 The divide-and-conquer approach		
		... the next paragraph	
write	You might want to rethink your previous answer		
	Think about the following details		
exerc		How do you split in two a sequence that has an odd number of elements?	
exerc		How do you decide that a sub-problem is "small enough"?	
exerc			Is there an oprimal number of sequences?
read		2.1 Insertion sort	
read		2.2 Analyzing algorithms	
observe		Presentation by MIT OCW Algorithms Lecture 01	
read	2.3.1 The divide-and-conquer approach		
		... the next three paragraphs	
programming	Write a program for mergesort (do not test it)		
exerc	What kind of input do you think you need for testing?		
WWW		See an applet that demonstrates the mergesort algorithm	
WWW		See a collection of sorting algorithms	
exerc			Can you argue which of the above algorithms are divide-n-conquer?

Figure 1: A snapshot of a learning environment in MS Excel

After one settles on the issue of the implementation of the basic model, the issue of linkage with external resources must be addressed. Discussion rooms, and other related communication-oriented applications can be readily used to support the implementation of learner’s notes and of a learning environment communication system. At that point, one can opt to start integrating different technology offerings (having, of course, to address the overhead of inter-application communication) or adopting a generic platform approach that will allow for customization to retro-fit the implementation of the model as well [6, 16]. The latter approach can be more scalable (for example, portal offerings by commercial organizations) but the analysis to decide on such an investment may be too difficult to carry out effectively (hidden costs can surface quite easily and the steepness of the learning curve for developers may be expensive to estimate) [7, 10, 14]. Note that a need for development may be inevitable with any platform if one attempts to implement some relatively sophisticated objects (for example, the learning activity control block of the graph theoretic model earlier presented), even at the entry level.

However, there also exist some in-between approaches; in these approaches one may decide to use building blocks based on generic digital object identification schemes, such as DOI (http://www.doi.org) and expect that third-party providers (for example, a university) will supply the naming space, and couple these identification schemes with generic object ensemble builders, such as Fedora [8] or SCORM [4], which accommodate a disciplined format of digital object creation and manipulation.

As a matter of fact this is exactly the development roadmap for LAMS [10], which expects that activities will be structured around a *lesson* plan and that the support tools to implement these activities will be increasingly supplied by third parties. Incidentally perhaps, LAMS also seems to be the closest implementation of our graph formalism concept and one that explicitly foresees the linkage of collaboration activities within the educational process; moreover it indeed structures activities as tasks to be done with some resources. See Figure 2 for an example, of how LAMS implements the workflow described in Figure 1 (but also note that, since LAMS does not yet fully support branching, the only graph node transitions available are the ones from one node to the next; i.e. a strictly sequential experience).

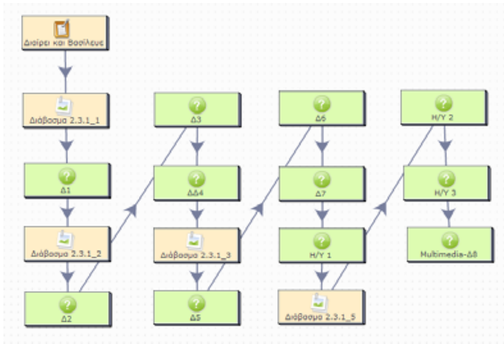

Figure 2: A snapshot of an activity workflow in LAMS

5 Web Mining With a Learner's ODL course

Cliques and connected components are usually employed as a means of demonstrating graph properties that are related to localization; here, we use *localization* as a metaphor to show that some *areas* of a graph may be very close neighbours in the sense that one has to venture explicitly outside this area through very specific paths. This is not a new concept and has been used in a very similar context in web site adaptation [13]. Automatically improving the organization and presentation of web sites based on data mining usage logs is a burgeoning scientific field and one of the approaches is based on the PageGather algorithm [12]. Therein, a clustering method, called *cluster mining*, is employed, which works on an input of user sessions, represented as sets of visited web pages (note the correspondence with learning experiences). PageGather then builds a graph by linking nodes (pages) with an edge whenever co-occurrence of these pages is detected across some user sessions. Page clusters (or, similar learning experiences) can then be defined using either cliques or connected components, with cliques considered to be more coherent and connected components considered to be faster to compute and easier to find.

There exist legitimate arguments about the computational cost of graph-based algorithms for inferring usage patterns [13]. However, if we can agree that our *aposteriori* analysis of the usage (by various users) of a Learner's ODL course will be used to improve its presentation and organization in a future version (thus, we do not focus on providing immediately customizable content), then these arguments are not related to our employing of the graph-based representation. Nevertheless, web usage mining is a complicated, of course, as it involves data pre-processing, pattern discovery and pattern analysis [15]. Data used for these procedures can be related to *content* (the real data in the Web Pages), *structure* (data describing the organization of the content), *usage* (data describing the pattern of usage in web pages) and *user profile* (data providing demographic information) [15]. Industrial reports (also based on anecdotal data) suggest that the data pre-processing can easily take up 80% to 95% of a project's time and resources [5].

The technical challenge is how to relate the relatively flat structure of web log files with the apparently deep structure of learning experiences (therein, we note again the introduction of cycles in experience paths). Our approach is to specify the course multi-graph in advance (*php* scripts interfacing to a *mySql* database were embedded in the course's *html* code). This approach is supported by the published experience in a similar project [11], where the difficulties of developing a data pre-processing environment are set out for a case study in a distance learning educational domain.

A coarse example of these concepts is shown below. Figure 3 demonstrates the course multi-graph structure, as specified by the course designer (actually, it is a view of the multi-graph where, for the sake of conciseness, we have only included learning activities). Figure 4 shows a learner's path during a single learning session in the course, with nodes being numbered according to the relative order of visit.

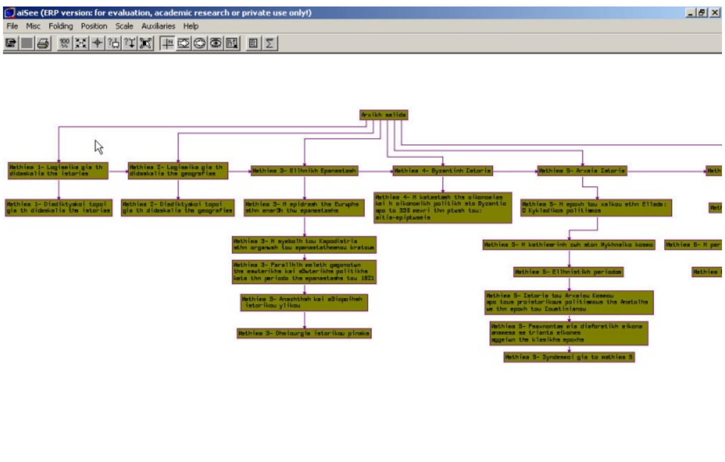

Figure 3: A view of the course multi-graph

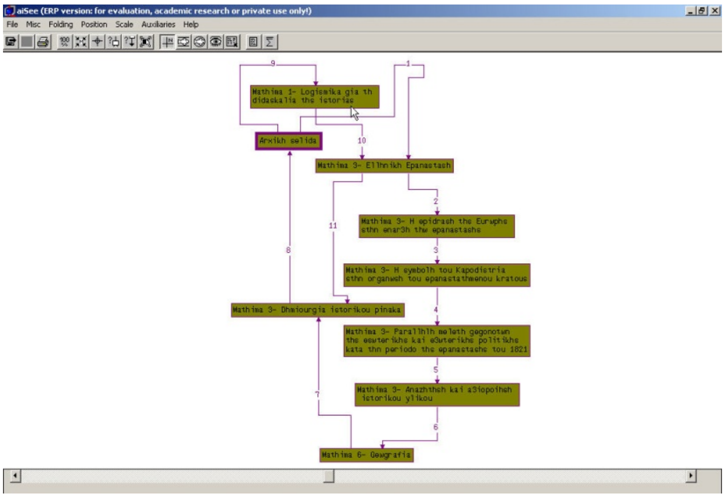

Figure 4: A visitor's path through the course

We also used a slight variant of the above mechanism to implement the note-passing mechanism between students and tutors (as described in Section 3). However, for this particular aspect of the Learner's ODL course, we are investigating the usefulness of open-source asynchronous discussion forum systems (and the extent to which they might accommodate the multi-graph specification as opposed to programming it from scratch).

The generalization of the above implementation to compute shared paths between collaborative (or non-) co-workers (students, tutors) is relatively easy. However, the visualization of those shared paths necessarily raises the issue of how to accommodate in the relatively limited estate of a computer screen the individual interactions of team members with the same material. While web usage mining applications are close to this problem, understanding which shared paths are meaningful and which are not will initially entail the close scrutiny by knowledgeable experts.

6 Conclusions

Like many other open universities, the Hellenic Open University (HOU) has gradually embarked on e-learning initiatives, spanning from virtual classrooms, to discussion forums and to the mass-scale development of complementary on-line material.

The HOU has lately completed a major transition to a common commercial portal platform and has initiated the installation and operation of an open-source digital asset management system as well as a commercial SCORM-compliant authoring tool. Deploying the newly-developed courses on that platform will allow for the production and sophisticated analysis of log files, according to the principles (and, mostly, to the ideas) outlined above. We are also experimenting with the possibility of developing path detection as a web-service to be provided by a third party at the course deployment level as opposed to on-line log file analysis.

The graph model was a necessary tool in our design approach because it helped model important aspects of the educational process and, then, seamlessly supported the semantic annotation of student activities while allowing us the convenience of knowing that graph-processing algorithms and software are available as a commodity.

Why did we *not* use a different model? Actually we did. The MS Excel example was our first implementation attempt at attracting fellow tutors to the didactical merits of explicitly stating learning tasks and expected time for related activities. Note that these very tutors may well be excellent when addressing an audience; it is their skills at developing distance learning material that we aim to further develop. So, the tabular Excel model was the easiest to communicate.

Thus, taking into account that we need to also address the needs of tutors with limited IT skills, the careful selection of tools for the initial compilation and development of learning activities is a key factor in our decisions. It turns out that we must really first lower the entry threshold for tutors in order to be able to realize benefits for the students. That threshold, in turn, has to do with both the development of content as well as the development of a collaborative conscience. The latter is necessary to reinforce the sense of belonging to an academic environment that our students (and, sometimes, our tutors) seem to desperately need and that our tutors may sometimes find difficult to re-invest in, since most of them are already part of a conventional environment.

In that sense, we believe that our key contribution is the bridging of design richness and implementation practicalities in the context of a very large scale project of distance learning digital educational material. We feel that similar situations will be common in the context of almost all organizations developing similar content.

Acknowledgements

Dionyssis Zafeiropoulos, Panagiota Mpekou and Spyros Papadakis have been instrumental in the set-up and functioning of the LAMS platform at our site. Ernie Ghiglione (of LAMS) provided several technical and development insights that have reinforced our commitment to the graph formalism for representing the activities of the educational process.

References

1. Evans Terry, Lockwood Fred. *Understanding Learners in Open and Distance Education.* London – Koggan Page Ltd, 1994.
2. Simpson Ormond. *Supporting Students in Online Open and Distance Education.* London – Koggan Page Ltd, 2000.
3. Höök Kristina, Benyon David, Munro Alan. *Designing Information Spaces: The Social Navigation Approach.* London – Springer, 2003.
4. ADL Technical Team, *Sharable Content Object Reference Model (SCORM) 2004 2nd Edition*, Advanced Distributed Learning, 2004.
5. H.A. Edelstein. “Pinning for gold in the Clickstream”, *Information Week,* March 12, 2001.
6. T. Hampel, and R. Keil-Slawik, “sTeam: Structuring Information in Team–Distributed Knowledge Management in Cooperative Learning Environments”, *Journal on Educational Resources in Computing*, 1 (2), Summer 2001, Article #3, pp. 1-27.
7. G. Henry. “Connexions: An Alternative Approach to Publishing”, *European Conference on Digital Libraries*, Bath (UK), September 2004.
8. C. Lagoze, S. Payette, E. Shin, and C. Wilper. “Fedora: An Architecture for Complex Objects and their Relationships”, *Journal of Digital Libraries,* special issue on *Complex Objects*, (to appear) 2005.
9. Learning Activity Management System, http://www.lamsinternational.com, 2007.
10. N. Mc Lean. “The Ecology of Repository Services: A Cosmic View”, *European Conference on Digital Libraries*, Bath (UK), September 2004.
11. C.G. Marquardt, K. Becker, and D.D. Ruiz. “A Pre-Processing Tool for Web Usage Mining in the Distance Education Domain”, *International Database Engineering and Applications Symposium*, 2004.
12. M. Perkowitz, and O. Etzioni, “Adaptive Web Sites”, *Communications of the ACM*, 43 (8), 2000.
13. D. Pierrakos, G. Paliouras, C. Papatheodorou, and C.D. Spyropoulos, “Web usage mining as a tool for personalization: a survey”, *User Modeling and User-Adapted Interaction*,

14 (4), 2003. 11. M. Smith, R. Rodgers, J. Walker, and, R.Tansley. “DSpace: a Year in the Life of an Open Source Digital Repository”, *European Conference on Digital Libraries*, Bath (UK), September 2004.

15. J. Srivastava, R. Cooley, M. Deshpande, and P.-N. Tan. “Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data”, *SIGKDD Explorations*, Volume 1, Issue 2, 2000.
16. A.R. Trindade, H. Carmo, and J. Bidarra, “Current Developments and Best Practice in Open and Distance Learning”, *International Review of Research in Open and Distance Learning*, 1 (1), 2000.
14. O.R. Zaiane. “Web Usage Mining for a Batter Web-Based Learning Environment”, *Conference on Advanced Technology for Education*, pp. 60-64, Banff, Alberta, Canada, June 27-28, 2001.