A first exploration of an inductive analysis approach for detecting learning design patterns

Citation Details: Francis Brouns, Rob Koper, Jocelyn Manderveld, Jan van Bruggen, Peter Sloep, Peter van Rosmalen, Colin Tattersall and Hubert Vogten (2005). *A first exploration of an inductive analysis approach for detecting learning design patterns*. Journal of Interactive Media in Education (Advances in Learning Design. Special Issue, eds. Colin Tattersall, Rob Koper), 2005/03. ISSN:1365-893X [jime.open.ac.uk/2005/03].

Submission to Journal of Interactive Media in Education

A first exploration of an inductive analysis approach for detecting learning design patterns

Francis Brouns, Rob Koper, Jocelyn Manderveld, Jan van Bruggen, Peter Sloep, Peter van Rosmalen, Colin Tattersall and Hubert Vogten

Educational Technology Expertise Centre
Open Universiteit Nederland
PO Box 2960, NL-6401 DL Heerlen
The Netherlands
www.ou.nl

Abstract: One way to develop effective online courses is the use of learning design patterns, since patterns capture successful solutions. Pedagogical patterns are commonly created by human cognitive processing in "writer's workshops". We explore two ideas; first whether IMS Learning Design is suitable for detecting patterns in existing courses and secondly whether the use of inductive analyses is a suitable approach. We expect patterns to occur in the method section of a learning design, because here the process of teaching and learning is defined. We provide some suggestions for inductive techniques that could be applied to existing learning designs in order to detect patterns and discuss how the patterns could be used to create new learning designs. None of the suggested approaches are validated yet, but are intended as input for the ongoing discussion on patterns.

Keywords: learning design, pedagogical patterns, pattern detection, inductive analysis

1 Introduction

How do we develop effective online courses? According to Koper (2005) learning design rules capture learning design knowledge and can assist in developing the best suited learning design. A learning design rule describes which learning design method can be used in a particular learning situation with a certain probability of success. At least three different approaches can be followed to arrive at learning design rules: (1) designs based on instructional theory, (2) designs based on best practices and (3) designs based on patterns in best practices. These three approaches can be illustrated with the following three scenarios, all starting with the designer having some ideas about the course title, the learning objectives, the course content, the target group and the setting of the course.

In the first scenario, the designer starts by selecting a pedagogical approach that might be appropriate for the design task. Such an approach could be a personal theory or a formal instructional design theory like the ones described in e.g. Reigeluth (1999). The designer creates the course inspired by the prescriptions in the theory. These theories give only limited concrete examples how to solve specific problems in the design task, so many parameters are left to the designer to decide and fill in. When the theory is appropriately tested, well selected and well applied, chances are high that the course is effective.

In the second scenario, the designer does not want to read and study instructional design theories and models. Instead, the designer searches for a comparable successful example course, in the same or another subject field. When an example is found that exactly fulfils the needs, the course can be re-used. However, in most cases only one or more comparable examples can be found that can be used as a guide for structuring and developing the new course. When the examples are effective, well selected and well restructured, chances are high that the course is effective.

In the third scenario, the designer searches for so-called course patterns that can be used to develop the new course. A course pattern is an *abstraction of a set of* best practices to fulfil a recurrent design problem (Alexander, 1979). Such a pattern is a kind of template, including instructions and examples

that can be filled in to create the new course. Patterns can be chained like building blocks to create the new course.

The first two approaches are well understood and applied, but the latter is relatively new. Although the use of pedagogical patterns is not new, the patterns usually have been created via deductive technologies, i.e. in "writer's workshops" where experts get together and write down the pattern in a specific format based on their experience and knowledge (Bergin, Eckstein, Manns, Sharp and Voelter, 2005; E-LEN, 2004; Hernández Leo, Asensio Pérez and Dimitriadis, 2004; Jones and Stewart, 1999; McAndrew, Goodyear and Dalziel, 2005). These pedagogical patterns present an abstraction of the pattern in a specific format. In addition to this approach, we would like to explore inductive methods to detect patterns in existing courses. The condition for such an approach is that the courses must be structured in a machine interpretable way to detect the patterns in the data. The ideas explored in this article are whether the recent IMS Learning Design (IMS LD) specification (IMS, 2003; Koper and Olivier, 2004) and its predecessor Educational Modelling Language (EML, 2000; Koper and Manderveld, 2004) are suitable for the detection of re-usable course patterns and whether inductive analysis is a suitable approach in detecting patterns. After a short introduction of the literature on patterns, we will explore several techniques and methods that could be applied in pattern detection. Then we suggest how these patterns can be used to construct new learning designs.

2 Patterns

Alexander (1979) was one of the first to describe the principle of patterns based on his observations in the field of architecture. His ideas have been taken up by others; initially in artificial intelligence, programming and software engineering (Fowler, 1997; The Hillside Group, 2005), then increasingly in education (Avgeriou, Papsalouros, Retalis and Skordalakis, 2003; Bergin *et al.*, 2005; Jones *et al.*, 1999).

The most commonly found definition of a pattern states that a pattern is a model of a solution for a recurrent problem that can be used repeatedly in many contexts (Alexander, 1979; Bergin *et al.*, 2005; The Hillside Group, 2005). The solution is generic and abstract, but analysed from a set of best practices. A solution is considered to be a potential pattern only after one has noticed that similar solutions are applied to a recurrent problem (Alexander, 1979). This observed regularity then is taken as a pointer to analyse the data further in order to try and create an abstraction from it that can function as a pattern.

Pedagogical patterns capture effective learning designs, allow newcomers in education to learn from more experienced developers, provide solutions that can be applied in many circumstances, function as a communication medium and means and enhance knowledge management and knowledge transfer (Alexander, 1979; Bergin *et al.*, 2005; E-LEN, 2005; Hernández Leo *et al.*, 2004; Jones *et al.*, 1999; McAndrew *et al.*, 2005). For example the Collaborative Evaluation pattern (McAndrew *et al.*, 2005) describes a possible solution to teach students how to perform an evaluation. Other examples of pedagogical patterns can be found online at the websites of the Pedagogical Patterns [1] and the E-LEN [2] projects. These pedagogical patterns describe the pattern in a common format. The format includes the context and a description of the problem and a solution. The patterns are described at a rather abstract level and leave the actual implementation to the designer.

There are two aspects of patterns that are important. Firstly, the identification of learning design patterns in a set of units of learning, and secondly, the application of learning design patterns to create learning designs for new units of learning. In the following paragraphs we will take both aspects into account.

3 Methods for pattern detection

Several approaches to detect patterns in learning designs are possible. Learning designs can be inspected, visualised or analysed by various techniques. One can simply look at the learning designs, either as raw data or in a run-time environment, and try to determine patterns. However, this is a very

time-consuming and laborious process and imposes many restrictions. A better method is to try to automate (part of) the analysis.

A first simple method is to count the occurrence of IMS LD elements. We looked at the frequency of elements in a total of 34 distance learning competence based courses, developed for higher education institutes and the Open Universiteit Nederland. The courses were designed to be delivered via the Internet, some contained additional face-to-face components. Average size of the course was 100 to 200 hours. As expected, the frequency of IMS LD elements varied widely between courses, mainly because the courses addressed different domains and were not of identical size. To give some impression of the data, Table 1 lists the frequency of the IMS LD elements for a sample of 7 of the 34 courses.

	learning-activity	environment	learning-object	service	play	act	role-part
course 1	54	92	200	1	1	1	3
course 2	9	28	11	9	2	2	3
course 3	11	28	293	7	1	2	10
course 4	16	16	23	7	1	1	5
course 5	6	7	25	2	1	1	5
course 6	56	116	155	4	1	1	2
course 7	30	14	26	2	1	1	4

Table 1: Frequency of selected IMS LD elements for some of the courses

This example illustrates the complexity of pattern detection in learning designs. Only limited information can be gained from simple frequencies. It is more likely that frequency in combination with other aspects like order or type of elements provide more relevant information. For instance, when a learning design consists of several activities, it is interesting to know that the first one is always an introduction, and the last one provides self-assessment and so on. This implies that part of the patterns lies in the structure (order of elements) and part in the content (type of element, type of content).

Therefore natural language techniques like Latent Semantic Analysis (LSA), data, text and web mining that can analyse, classify, categorise, visualise data based on content combined with algorithms that are capable of analyzing structured, hierarchical data (like XML) might be better suited. Below we give a short description of LSA and introduce some XML analysis techniques.

3.1 Latent semantic analysis and indexing

LSA is a theory and a method to extract and find concepts and similarity of concepts in large bodies of text, relying on the meaning of words, terms and phrases in the contexts in which they are used (Landauer, Foltz and Laham, 1998). Concepts are found by analysing the material for the existence of specific terms or phrases in the context. The number of concepts is presented in frequency tables. Latent Semantic Indexing (LSI) and clustering techniques reduce the frequency table by retaining only the most significant concepts (Dom, 2000; Han *et al.*, 1998; Jain, Duin and Mao, 2000). LSA is particularly suitable for clustering and classification of documents and has been applied to a wide range of document types, including XML and web documents (Dom, 2000).

3.2 Analysis of XML

XML tags present the syntax in which to express the semantics. XML data are always hierarchical and presented in trees. The growing use of XML for documents and interchange of data resulted in numerous efforts to analyse XML documents in order to detect changes (Cobéna and Abiteboul, 2002;

Wang, DeWitt and Cai, 2003), similarities (Weis and Naumann, 2004; Zhao, Bhowmick, Mohania and Kambayashi, 2004) or to combine XML data into new XML datasets (Guha, Jagadish, Koudas, Srivastava and Yu, 2002).

XML could transformed to topic maps (XTM, 2003) or to graphs (Shash, Wang and Giugno, 2002).

Combinations of algorithms have been applied by Yoon *et al.* (2001) and Zhenchang (2002).

4 Where to find patterns in learning design?

4.1 Method or act as building blocks

We expect patterns to occur in the learning design *method* of a learning design, because the method specifies the teaching-learning process. In order to be as flexible as possible, we need to identify small re-usable components in the method section. A method comprises plays, acts and role-parts. In theory there are several solutions to identify patterns in learning design methods. First, the complete method section can be used to model a pattern. Especially when units of learning are small (e.g. a few study hours) this would be a good approach. When units of learning are larger (e.g. up to 300 study hours), smaller components must be identified in order to allow for enough flexibility. One can argue that these large units of learning should be built from smaller, independent units of learning. So, one approach could be always to base a pattern on a method section of a unit of learning and in the case of larger units of learning, first to split the unit of learning up into smaller, independent ones. Another approach is to build units of learning on *acts*. Acts are the smallest, independent sections within a method. To build a new unit of learning, several act patterns can be combined to create the new method. Even the content of two acts can be merged (under certain conditions) to create a new act.

We assume that these two approaches are the only two that can be used. Role-parts, the first lower level of aggregation, can be dependent on each other (e.g. a learning activity and a support activity can be related). The smallest independent, and thus re-usable, entity is the *act* within a method section. However, as can be seen from the previous Table 1, in some designs only a few acts are used per method. The use of acts is not independent of the type of pedagogy modelled. Typically in universities having open admission and individualized learning paths, like the Open Universiteit Nederland, acts are not used extensively. One of the major uses of an act is to synchronize the activities of the individual learners: for instance to model week or month blocks of work.

5 The process to identify patterns in learning design

Given the discussion above, the process of identifying patterns in learning designs can be summarized as follows. Patterns are expressed as a method or as an act. At least two units of learning are needed to identify a pattern (the input-set). These units of learning must be classified as best practice within a certain field, e.g. for a certain type of learning objectives. It is not necessary that the units of learning contain the same content, only structural similarities will be identified as a pattern. The method of a unit of learning contains references to roles, activities and environments. Additionally, a learning design does not contain the actual content, but refers to resources containing the textual descriptions. Therefore a transformation to resolve the references is needed before pattern detection can start. Also transformations may be needed to make the structures in the XML document of a similar form when alternative modelling approaches are possible. The patterns matching mechanism will first detect structural similarities in the XML structures of the input-set. The output will be a document that contains the common elements in the input-set. Then the pattern can be refined by adding additional relevant aspects that require a more extensive and complex processing of the document. Several of the above mentioned techniques could be applied to different stages of the analysis. XML tree comparison and regular expression techniques will likely be used for the patterns in the method, consisting of specific orders of elements, while text, data mining and LSA will be used to determine type of activity, environment and roles by examining titles and content resources.

6 Using learning design patterns

Learning design patterns can be used to create new learning designs. In order to accomplish this, templates need to be created based on the patterns. A formal and a human readable description of the pattern have to be created and elaborated. The pattern should be supplemented with examples and additional information detailing its use and intention. The pattern should not only consist of a description but include the learning design formalised in IMS LD as well. The latter serves as a template containing dummy example text to be replaced by the author. The patterns in the form of models must be available to course developers together with the IMS LD templates and instruction on how to complete the template.

Because learning design patterns are based on classes of units of learning that are considered best practice, it would be good when a description becomes available of the experience in practice together with the underlying units of learning. How effective, efficient and attractive were they in practice? In what setting were they applied? These data can help to determine which pattern is more suitable given the situation of the course to be developed.

7 Discussion

We offered some ideas on the suitability of the IMS LD specification for describing patterns in learning designs. Moreover we suggested using inductive analysis methods instead of a deductive approach, because with the advent of the IMS LD specification the former can be done on existing courses. These patterns are more specific than patterns described with deductive approaches, because they do not only describe the solution to a common problem, but also specify a template with IMS LD elements. This template, together with instructions and examples, should help designers to create new learning designs. The inductive analyses allow patterns to be induced from existing courses.

There are still many questions that need answering; a few are mentioned here. Are there sufficient courses available for this approach? LSA and data mining usually is performed on rather large datasets, but also has given results on small datasets. A growing number of IMS LD coded learning designs is available; there are the more than 34 courses at the Open Universiteit Nederland as mentioned above, plus a set of, at present, 36 courses that is available via the UNFOLD website. This number should increase rapidly with the new Moodle/LAMS export possibility. Can it be used to classify designs as good as well as bad practices, for example when user data, such as success or failure rates, completion time, etc are added to the analysis, or even with human classification of the design? Are acts the smallest independent units in learning designs? Are the templates sufficient for practitioners to develop new courses?

To find answers to these questions, the suggested approach needs to be validated. The techniques should be applied to existing courses to find out whether patterns can be detected. The patterns should then be translated into templates and author's instructions to form the basis for the development of new learning designs. The suitability of these templates should be validated by practitioners.

Acknowledgements: The authors wish to thank the management and staff of the Schloss Dagstuhl International Conference and Research Center for Computer Science for providing a pleasant, stimulating and well organised environment for the writing of this article.

8 References

The Hillside Group (2005). Accessed online on 19 Jul. 2005 at: http://hillside.net/

Alexander, C. (1979). *The Timeless Way of Building*. New York: Oxford University Press.

Avgeriou, P., Papsalouros, A., Retalis, S., and Skordalakis, M. (2003). Towards a Pattern Language for Learning Management Systems. *Educational Technology & Society*, 6 (2), 11-24.

Bergin, J., Eckstein, J., Manns, M., Sharp, H., and Voelter, M. (2005). *Pedagogical Patterns.* Accessed online on 19 Jul. 2005 at: http://www.pedagogicalpatterns.org

Cobéna, G., and Abiteboul, S. (2002). Detecting Changes in XML Documents. *Proceedings of the 18th International Conference on Data Engineering (ICDE'02)*, 26 February - 1 March, 2002 , San Jose, California. IEEE.

Dom, B. (2000). Pattern Recognition Meets the World Wide Web. *Proceedings of the International Conference on Pattern Recognition (ICPR'00)*, 3-8 September, 2000, Barcelona, Spain. IEEE.

E-LEN. (2004). *Design patterns and how to produce them.* Accessed online on19 Jul 2005 at: http://www2.tisip.no/E-LEN/outcomes.php

E-LEN (2005). *e-LEN project.* Accessed online on 19 Jul. 2005 at: http://www2.tisip.no/E-LEN/

EML (2000). *Educational Modelling Language.* Accessed online on 19 Jul. 2005 at: http://eml.ou.nl

Fowler, M. (1997). *Analysis Patterns: Reusable Object Models*. Boston: Addison-Wesley.

Guha, S., Jagadish, H. V., Koudas, N., Srivastava, D., and Yu, T. (2002). Approximate XML joins. *Proceedings of the 2002 ACM SIGMOD international conference on Management of data.* Madison, Wisconsin, ACM Press.

Han, E., Boley, D., Gini, M., Gross, R., Hastings, K., Karypis, G., Kumar, V., Mobasher, G., and Moore, J.. (1998). WebACE: A Web Agent for Document Categorization and Exploration. In: K. P. Sycara and M. Wooldridge (Eds.) *Proceedings of the 2nd International Conference on Autonomous Agents (Agents'98)*. ACM Press.

Hernández Leo, D., Asensio Pérez, J. I., and Dimitriadis, Y. A. (2004). IMS Learning Design Support for the Formalization of Collaborative Learning Patterns. *Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT'04*), 30 August - 1 September, 2004, Joensuu, Finland. IEEE.

IMS. (2003, January 20). *IMS Learning Design Information Model. Version 1.0 Final Specification*. Accessed online on 19 Jul. 2005 at: http://www.imsglobal.org/learningdesign/ldv1p0/imsld_infov1p0.html

Jain, A. K., Duin, R. P. W., and Mao, J. (2000). Statistical pattern recognition: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 (1), 4-37.

Jones, D., and Stewart, S. (1999). The case for patterns in online learning. In: P. De Bar and J. Legget (Eds.), *Proceedings of Webnet'99 Conference, Honolulu Hawaii*, 24-30 October 1999. Association for the Advancement of Computing in Education.

Koper, E. J. R., and Olivier, B. (2004). Representing the Learning Design of Units of Learning. *Educational Technology & Society, 7*(3), 97-111. Accessed online on 30 March 2005 at: http://ifets.ieee.org/periodical/7_3/10.pdf

Koper, R. (2005). An Introduction to Learning Design. In: R. Koper and C. Tattersall (Eds.). *Learning Design A Handbook on Modelling and Delivering Networked Education and Training* (pp. 3-20). Berlin Heidelberg: Springer-Verlag.

Koper, R., and Manderveld, J. (2004). Educational modelling language: modelling reusable, interoperable, rich and personalised units of learning. *British Journal of Educational Technology*, 35 (5), 537-552.

Landauer, T. K., Foltz, P. W., and Laham, D. (1998). Introduction to Latent Semantic Analysis. *Discourse Processes*, 25, 259-284.

McAndrew, P., Goodyear, P., & Dalziel, J. (2005). *Patterns, designs and activities: unifying descriptions of learning structures.* Accessed online on 19 Jul. 2005 at: http://kn.open.ac.uk/public/document.cfm?docid=5295

Reigeluth, C. M. (1999). *Instructional-design theories and models (volume II)*. London: Lawrence Erlbaum Associates.

Shash, D., Wang, J., and Giugno, R. (2002). *Algorithmics and Applications of Tree and Graph Searching.* Accessed online on 19 Jul. 2005 at: http://www.acm.org/sigmod/pods/tut/sh.ppt

Wang, Y., DeWitt, D. J., and Cai, J.-Y. (2003). X-Diff: An Effective Change Detection Algorithm for XML Documents. *Proceedings of the 19th International Conference on Data Engineering (ICDE'03)*, 5-8 March, 2003, Bangalore, India. IEEE.

Weis, M., and Naumann, F. (2004). Detecting duplicate objects in XML documents. *Proceedings of the 2004 international workshop on Information quality in information systems*, 18 June, 2004, Paris, France. ACM Press.

XTM (2003). *TopicMaps.Org.* Accessed online on 19 Jul. 2005 at: http://www.topicmaps.org/

Yoon, J. P., Raghavan, V., Chakilam, V., and Kerschberg, L. (2001). BitCube: A Three-Dimensional Bitmap Indexing for XML Documents. *Journal of Intelligent Information Systems*, 17 (2/3), 241-254.

Zhao, Q., Bhowmick, S. S., Mohania, M., and Kambayashi, Y. (2004). Discovering frequently changing structures from historical structural deltas of unordered XML. *Proceedings of the Thirteenth ACM conference on Information and knowledge management*, Washington, D.C., USA. ACM Press.

Zhenchang, X. (2002). *On Trees and Graphs.* Accessed online on 19 Jul. 2005 at: http://www.cs.ualberta.ca/~stroulia/Presentations/Trees_and_Graphs.ppt

9 Footnotes

[1] The Pedagogical Patterns project: http://www.pedagogicalpatterns.org/

[2] The E-LEN project: http://www2.tisip.no/E-LEN/