

University of Bolton

**UBIR: University of Bolton Institutional Repository**

---

Educational Cybernetics: Journal Articles

Institute for Educational Cybernetics

---

2005

## From Primitives to patterns: a discussion paper.

David Griffiths

*University of Bolton*, [dai.griffiths.1@gmail.com](mailto:dai.griffiths.1@gmail.com)

---

### Digital Commons Citation

Griffiths, David. "From Primitives to patterns: a discussion paper.." (2005). *Educational Cybernetics: Journal Articles*. Paper 2.  
[http://digitalcommons.bolton.ac.uk/iec\\_journals/2](http://digitalcommons.bolton.ac.uk/iec_journals/2)

This Other is brought to you for free and open access by the Institute for Educational Cybernetics at UBIR: University of Bolton Institutional Repository. It has been accepted for inclusion in Educational Cybernetics: Journal Articles by an authorized administrator of UBIR: University of Bolton Institutional Repository. For more information, please contact [ubir@bolton.ac.uk](mailto:ubir@bolton.ac.uk).

# From Primitives to Patterns: a discussion paper.

David Griffiths, 10<sup>th</sup> July 2005

## **Introduction**

There is general agreement that it is important for the success of Learning Design that teachers are able work with Units of Learning, and you may like to look at my paper *The Role of Teachers in Editing and Authoring Units of Learning using IMS Learning Design*, a paper which gathers together much of the discussion on this topic in UNFOLD [1]. It is also clear that IMS LD is a complex specification, because it describes all the complexity of learning activities, and teachers do not want to, and should not be expected to, engage with the LD language itself. There are two levels of engagement with LD which we can envisage for teachers with no advanced LD skills

a) *Adapting*: identify existing UoLs which may be valuable, and adapt them to their needs

b) *Bricolage* (or more prosaically perhaps) *mix and match*: construct new UoLs by bringing together parts of other UoLs.

These two levels raise many issues of tool design (e.g how do you make it easy for a user to understand what a UoL does?), and systems development (e.g. how can you disaggregate a UoL so that the chunks are still useful).

If as educators we hope to get effective solutions for teachers using LD then we will have to be very clear the description of our needs, as systems developers do not usually have teaching experience or understanding of pedagogic issues. At the Braga CoP meeting there was there were discussions which suggested that the terminology which we use is not always clear, and this was borne out by the online discussion which followed, and is available at <http://moodle.learningnetworks.org/mod/forum/view.php?id=247>. Consequently it seemed useful to look at this and to seek to reach an agreed clarification of some points which have sometimes caused confusion to newcomers to the field, and to discuss some of the pending issues.

The terms which I'd like to look at are:

- pattern
- exemplar
- template
- primitive
- activity and activity structure
- nugget

## **Pattern**

As discussed in [1]the concept of patterns goes back to Alexander's original work on architecture, and his statement that “A pattern language gives each person who uses it, the power to create an infinite variety of new and unique buildings, just as his ordinary language gives him the power to create an infinite variety of sentences” [2] p. 167.

Each pattern addresses a problem and provides a solution, but Alexander's formulation stresses that the point of patterns is not to lead to automatic reuse, but

rather to support creativity. As McAndrew argued in the UNFOLD meeting in Barcelona, September 2004 [3], this implies that the descriptions used in patterns should not relieve teachers of pedagogic responsibility, but rather support their engagement. McAndrew, Goodyear and Dalziel[4] define Alexandrian patterns as follows (adapting from [5])

- i) A picture (showing an archetypal example of the pattern).
- ii) An introductory paragraph setting the context for the pattern (explaining how it helps to complete some larger patterns).
- iii) Problem headline, to give the essence of the problem in one or two sentences.
- iv) The body of the problem (its empirical background, evidence for its validity, examples of different ways the pattern can be manifested).
- v) The solution. Stated as an instruction, so that you know what to do to build the pattern.
- vi) A diagrammatic representation of the solution.
- vii) A paragraph linking the pattern to the smaller patterns which are needed to complete and embellish it.

The authors take this model and propose that a pattern language can be developed for learning, equivalent to that composed of the 258 patterns which Alexander established for architecture. These patterns consist of expository text, and below I reproduce one of the examples given:

<b>Pattern: COLLABORATIVE EVALUATION</b>
<b>Context:</b> A group of learners need to understand the principles behind a particular technique so that they can progress to become able to select particular implementations for others and to be able to take part in producing further examples themselves. Such learners need to develop an appreciation of the different forms available, the structure they have and why particular forms are suitable for some tasks.
<b>Body:</b> The contradictory challenges in this are the need to understand the structures that have been used alongside the need to see new ways to do things. The breadth of what is available needs to be examined alongside understanding how the software might apply when used in depth. It is important to balance individual views with group views and established positions from literature and other sources.
<b>Solution:</b> Building a collaborative evaluation enables the sharing of the work load and brings in the views of others to enable testing of consensus and variation in the depth that each individual may look at a particular example.
It is associated with patterns for LEARNING THROUGH DISCUSSION, COLLABORATIVE LEARNING and NETWORKED LEARNING PROGRAMME. It builds on patterns for DISCUSSION GROUPS, DISCUSSION ROLE, FACILITATOR, DISCURSIVE TASK, SEARCH, and CONSENSUS FORMING.

From McAndrew [4] Figure 8 Collaborative Evaluation as a Pattern

We should note that in this formulation a Learning Pattern is

- at a high level of abstraction
- consists of an expository text.
- part of a cascade, building on other patterns and in turn being associated with other patterns.

If we accept this view on the nature of patterns, how could they be used with Learning Design? A number of reflections present themselves.

- 1) A Learning Pattern is at a higher level of abstraction than a UoL, describing a problem in pedagogy, rather than a means to achieve a learning objective.
- 2) A Unit of Learning cannot describe the context for a Learning Pattern, still less constitute the pattern itself, but it could usefully illustrate the solution. The pattern

which it is associated with will in turn help the teacher to understand how the UoL could be used.

3) A Learning Pattern is part of a cascade of elements at different levels which provide materials for teachers to think with in developing a larger solutions. This suggests that illustrations to Learning Patterns expressed in LD will not all be at the level of UoL, but may at a lower level of granularity (e.g. an activity structure) or at a higher level (a course structure or an itinerary of courses).

4) As Griff Richards has pointed [6] out, we can distinguish between “PATTERNS (which we recognize in our environment) from DESIGNS (which are intentional change plans).” Architecture is a complex enterprise, but there is general agreement about most of its components. We do not find architects entering into fiery arguments about what constitutes a door or a window. In education, on the other hand, different traditions have substantially different views on what constitutes knowledge and learning, and how we can detect it. This suggests that distinguishing the patterns in the educational domain may be more complex than in the architectural domain, and perhaps more than one pattern language may be necessary in order to reflect the differing views of practitioners on the educational environment.

### ***Exemplar***

An exemplar Unit of Learning is one which provides an example of how to resolve a problem in Learning Design. The focus of the problem could be technical (for example, how do you use a QTI evaluation in a UoL) or illustrate a pedagogic approach (for example the well known Versailles Negotiation[7]). The exemplar is the UoL itself, complete and ready to run.

### ***Template***

A Learning Design template, like a template in any other aspect of computing, is a partly completed file to which the user can add data. A template can be made from any UoL, but it only makes sense to make a template from a useful and reusable UoL. Consequently a template may be considered to be a partly completed exemplar, with place holders for the information to be added by the user.

Templates offer a way of simplifying the production process, and making it more accessible to teachers and content specialists by eliminating the need for them to carry out the whole development process.

In UNFOLD we have documented templates as follows:

- A narrative, (free text description of the learning activities)
- A lesson plan (with no LD specific aspects)
- A worksheet (one step nearer to LD)
- A walk-through (Screen shots from UoL)
- The example UoL itself (an exemplar)
- A part completed UoL to be filled in.

The interface provided for filling in the completed UoL is not part of the template and it could be a general purpose editor, or a specialised application.

LD templates are sometimes described a formal representation of a pattern, by implication equating a pattern with an exemplar of best practice. In my discussion of patterns above I indicate why I do not think this is appropriate.

## **Primitive**

As mentioned in [1] the concept of primitives, was introduced into the UNFOLD discussion by Casey. The term draws on the computer science, where it is used to refer to “datatypes provided by a programming language as basic building blocks”[8]. Similarly in 3D design a primitive is a basic structure which can be combined with others and refined. When applied to pedagogy a primitive may be construed as an interactive event in a classroom, such as "discuss this text" or "research this topic on the web".

A pattern or exemplar intended to provide a solution to a problem, but a primitive simply identifies a basic element which may be useful in any context. In order to define primitives it is necessary to decide to look at the continuum of pedagogic practice, and to make a decision on how to divide the continuum into parts which make for effective discussions about practice, and then to implement these chunks as pieces of Learning Design code. Primitives are rougher, more tentative conceptions of pedagogy built on the day to day practice lecturers and teachers, rather than the result of an analysis of education from a particular theoretical perspective. The hope is that these structures can provide teachers with something recognisable which they can work with and build into UOLs.

## **Activity and activity structure**

Rob Koper in his presentation at the UNFOLD Braga CoP meeting[9] drew attention to the fact that activities are sometimes taken to mean “an opportunity for someone to do something”, so, for example a sport such as basketball could be referred to as an activity. Similarly, in the context of online education a chat environment or a conferencing system could be thought of as an activity which is available to users. Rob Koper clarified that in learning design an *activity* is understood in the sense in which it is used in psychology, that is to say “that which is done by the person”, while the context which provides the opportunity for this to happen is an *environment*.

In the IMS LD Best Practice Guide [7] *activity* is described as follows:

“An activity (learning-activity or support-activity) has a number of parts. They can have their own learning-objectives, prerequisites, and meta-data. Typically, they also have a reference to an environment which will contain the learning objects and/or services to be used in that activity. They also have an activity-description which is typically a reference to Web page which provides information, description, instructions about what the user should do in this activity. In some cases this is sufficient, and may be all that is needed for example to describe offline activities that are to be carried out. However, it typically tells the user what they should be doing with the resources contained in the associated environment. By tagging it separately from the resources in its environment, the runtime system can treat it differently, perhaps always keeping it available on a tab or menu for the duration of the activity.”

Activities can be grouped together into *activity structures*.

Because Learning Design separates Activities from Activity Structures and these from Roles and Resources, they are all potentially reusable components. The function of an activity structure is, however to be assigned to a role at a particular point in the learning process. As a result it is quite likely that an author of a UoL would like to reuse an activity structure in association with a specific role, and this creates more complexity in creating systems which support reuse of activity structures (without creating insurmountable difficulties)

## ***Nugget***

The concept of the *nugget* was introduced by the work of Southampton University in the Dialog+ project.

In her presentation on the Dialog Plus Toolkit[10] at the UNFOLD CoP meeting in Braga Gráinne Conole said that the project had arrived at its own definition of a learning activity, in consultation with practitioners (who provided the term 'nugget'). A detailed taxonomy of learning activities was produced building on previous work. For example the classification of approaches builds on Laurillard, Vygotsky etc. (but they can also be referred to by a label, such as Socio-cultural). Similarly Bloom's taxonomy of outcomes is used, together work with Sarah DeFreitas on learning activities and Laurillard's work on tools. Using the Dialog+ toolkit a practitioner can define a series of learning activities, and produces a plan for a lesson or part of a lesson. The sequence of learning activities is congruent with an *activity structure* in Learning Design, and because of this it has been possible to save work carried out in Dialog+ as a Learning Design activity structure.