# Learning Objects: issues to do with obtaining and labelling

Elizabeth Guest

**Introduction**

There is little consensus as to exactly what a learning object is, mainly because the concept is so wide-ranging. The term appears to have originated with the use of web-based media for distance learning. Petrinjak and Graham (2004) offer two definitions culled from the literature:

(1) Learning Objects are defined as any entity, digital or non-digital, which can be used, re-used or referenced during technology-supported learning. Examples of technology-supported learning include computer-based training systems, interactive learning environments, intelligent computer-aided instruction systems, distance learning systems, and collaborative learning environments. Examples of Learning Objects include multi-media content, instructional content, learning objectives, instructional software and software tools, and persons, organisations, or events referenced during technology-supported learning.

(2) Any reusable digital resource that is encapsulated in a lesson or assemblage of lessons grouped into units, modules, courses, and even programmes.

The idea of reusing teaching material is not new: we all do it. The difference is that the concept of a "learning object" both formalises this process, and provides a means by which learning objects can be stored in a repository and shared. Thus with this model, teaching material will not longer reside in the filing cabinets and on the computers of teaching staff, available only to the member of staff in question, but in a more central location available to anyone.

The rationale for such a change is mainly economic (Wiley, 2003): if material is shared between staff, and can be found easily, there should be less duplication of material. There should be less effort in developing teaching material if lecturers can choose a number of relevant learning objects and assemble them into a lecture, or material for a tutorial, or even a whole module. An additional benefit is that students can be allowed flexibility in what and how they learn: lessons can be personalised to meet their requirements.

This kind of approach is currently used in the corporate sector where companies that deliver training can quickly tailor a course to the client's needs (Collis and Strijker, 2004). This scenario is different from the traditional academic scene in that:

(a) The courses are better funded, and the course fees are much higher
(b) The courses are generally shorter
(c) The courses tend to be on very specific specialised subjects
(d) The aim is to build knowledge and specific skills (such as programming in a certain language to a required standard). Skills taught at university are more generic to enable students to switch to a different programming language, for example, with relative ease.

There have been attempts to incorporate this paradigm into the university sector (Porter, 2001) and the paradigm is showing some promise, but in this case there are several issues that need to be solved.

**Issues**

Collis and Strijker (2004) describe a learning object lifecycle. Objects are first obtained, or made. Then they have to be labelled in such a way that if someone is looking for material on

that topic, they can be found easily. Labelled objects are added to a repository, which may be in-house or in a wider repository. The repository can be searched and when a suitable learning object is found, it can be modified for use, or used as it is. Experience in using it may also result in modifications. These modifications are made and the object is added to the repository either as a new object, or to replace one that has been improved through the experience of using it or the need to update it. A diagram of this lifecycle is given in figure 1.

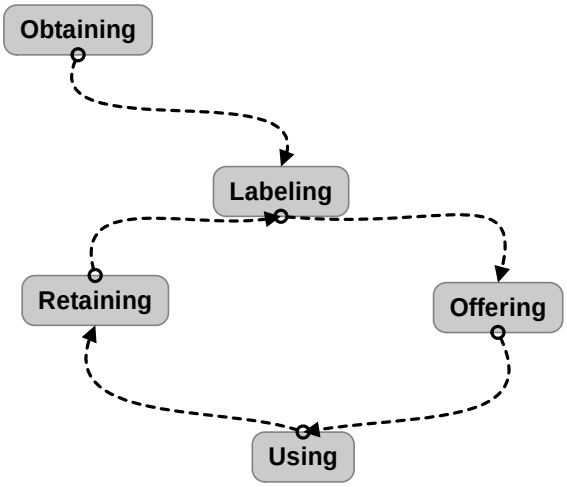


**Figure 1: The learning object lifecycle. Modified from Collis and Strijker (2004)**

There are issues involved with each of these steps, but in this paper only the issues to do with obtaining and labelling will be discussed as these pose the greatest problems within this cycle.

**Obtaining**

There are several issues to do with obtaining or making learning objects. The main issue is to do with *granularity*, which describes how encompassing a learning object is. Is a learning object a whole module, a class session within a module, or part of a session? Should it consist of a single concept, or several? Should it come with an online quiz, or should this be a different learning object? Is it instructional material, or material for discussion in class, or should these go together in a single learning object? Learning objects can be any of these. The user should be able to choose either a ready made lecture, or tutorial, or choose from a set of components to construct a lecture or tutorial.

Other issues are to do with pre-requisites and learning philosophies. For example, if we consider textbooks on the same subject, there will be broad agreement on which topics depend on other topics, but it is not possible to make a new book by simply selecting chapters from different books (even if it were legal). Each author has their own ideas on which topics should be included, the order in which to cover them and how to present them. By cutting and pasting chapters there will be overlap in the material covered and the books will not hang together well. There are likely to be similar problems with learning objects. Generally when we build a set of course materials we modify material found in books, papers, and the internet to suit our own styles, learning philosophies, and the students to whom the material will be presented. The use of a collection of learning objects may make a course seem a bit disconnected. However, if done well, and if the pre-requisites are followed, the variety in style may actually help students because different learning philosophies will suit different students.

Some theories and techniques are shared between subjects, but students will need examples specific to their interests and knowledge. For example eigenvalue analysis, used to work out resonance in engineering structures, is also used extensively in machine vision and also occurs

in natural language processing. The principles are the same, but this would not be obvious to the average student, who would struggle to apply it to their own specific interest area. It is essential that this material is presented in an appropriate context.

This kind of issue makes it hard to share learning objects. But perhaps the adaptation is not too hard. One could envisage an advanced system where the explanation of the method is core, but where appropriate examples are inserted according to requirements.

**Labelling**

The labelling of learning objects so that once they have been added to a repository, they can be found again, is problematic. Much research has been done in the area of metadata tagging for learning objects (Petrinjak and Graham, 2004) but this is still an unsolved problem. The IMS Global Learning Consortium is attempting to develop a tagging system and the intention is that this will enable people to tag their objects in a consistent way so that objects can be shared.

There are several problems with this. There is already a widely accepted labelling system in libraries. However, I am often surprised at the location the library categorises them in. I suspect that whatever system somebody comes up with, there will be similar problems when non-expert users attempt to use or apply it. Somehow this has to be made easy for the creators of the learning objects.

Collis and Strijker (2004) suggest solving this problem by means of ontologies. The idea is to give people an ontology from which to choose their metadata. However, for this to work, considrable collaboration would be required to agree on a common ontology. This is because it is very difficult to define a mapping between two different ontologies, even when the subject matter is the same. Indeed, finding a mapping between ontologies automatically is an unsolved problem. So it is unlikely that groups of people working within the same area will build similar ontologies without working closely together, with resulting difficulties in finding learning objects within a repository or in merging two repositories.

Anderson (2003) argues that building a cross-disciplinary repository of learning objects is hard because different disciplines have different kinds of knowledge and reason about their knowledge in different ways. For example there is a huge difference between physics and the social sciences. This also makes it hard for people across disciplines to agree on a labelling (and on what the labels mean), and then to use them consistently. It may be that cross-disciplinary repositories are not needed, but I think it would be good to work towards this because there will be overlap between areas. For example, there is a lot of overlap between physics and maths, and some overlap between maths and philosophy (in the area of logic, for example). Our knowledge does not fit into neat boxes, however much we would like it to.

**Discussion**

The concept of “learning objects” reminds me of object orientated programming. This is a paradigm where some computer programming code is packaged up into an object in order to make re-use of this code in different applications easy. At first this paradigm did not deliver what it promised, but now it is a powerful programming technique and all programming environments now incorporate a library of objects.

It has taken at least 20 years to achieve this and some of the problems were similar to those that currently exist with learning objects. At first, there was no consensus as to what an object should be. Should it be low level or high level? Today, this varies depending on the application: Integrated Development Environments have libraries of low level objects, but libraries of higher level objects exist for 3D computer graphics – and in fact you can choose between a high level library or a low level library depending on what you want to do. Another problem was in making

the objects versatile enough to handle a variety of applications. At first people kept writing their own objects because the library ones didn't have the procedures that they wanted to use.

There is still an issue with categorising objects, and I often end up looking through lists of objects and their procedures in order to choose the one I want to use. Since programming is a fairly specialised subject with its own jargon, some categorisation is straightforward so that objects can be grouped under these headings – but this can still result in long lists.

Learning Objects seem to be at the start of a similar process. It would be useful if we could construct our courses, modules, and materials for teaching sessions from libraries of tried and tested learning objects. But there are problems in bringing this to fruition, and I suspect that it will take years for these problems to be resolved.

**Elizabeth Guest (Interactive Systems, Innovation North)**

**References**

Anderson, J A. (2003) The Challenges of Collaborative Knowledge. In S. R. Acker (ed) *Learning Objects: Contexts and Connections*. The Ohio State University.

Collis, B, and Strijker, A. (2004) Technology and Human Issues in Reusing Learning Objects. *Journal of Interactive Media in Education* 4.

Petrinjak, A, and Graham, R. (2004) Creating Learning Objects from Pre-Authored Course Materials: Semantic Structure of Learning Objects - Design and Technology. *Canadian Journal of Learning and Technology* 30.

Porter, D. (2001) Object-Lessons from the Web: Implications for Instructional Development. In G. Farell (ed) *The Changing Faces of Virtual Education*. Commonwealth of Learning (COL).

Wiley, D. (2003) The Coming Collision between Automated Instruction and Social Constructivism. In S. R. Acker (ed) *Learning Objects: Contexts and Connections*. The Ohio State University.