LEARNING TECHNOLOGY IN THE EUROPEAN COMMUNITIES

PROCEEDINGS OF THE DELTA CONFERENCE ON RESEARCH AND DEVELOPMENT
THE HAGUE, 18–19 OCTOBER 1990

PRECISE CONSORTIUM FOR THE COMMISSION OF THE EUROPEAN COMMUNITIES DG XIII: TELECOMMUNICATIONS, INFORMATION INDUSTRIES AND INNOVATION

R ACADEMIC PUBLISHERS
ON THE CONCEPT OF REUSABILITY IN EDUCATIONAL DESIGN

G. OLIAMPO, A. CHIOCCARIELLO, M. TAVELLA and G. TRENTIN
Istituto per le Tecnologie Didattiche del CNR,
Via All'Opera Pia, 11
16145 Genoa,
Italy

ABSTRACT. The concept of reusability, which is already well established in several branches of engineering, is still little understood and practised in the field of educational software. This paper is based on the work carried out within the DELTA project ESM_BASE (Educational Systems based on Multimedia Databases). It makes an attempt to model the concept of reusability in the field of education by discussing the basic requirements for the actual reusability of multimedia educational components. The paper describes also the major architectural choices made by the project and some of the results achieved to date.

1. The need for reusability in multimedia courseware

The amount of effort which goes into producing high quality multimedia material is usually very large and the level of skills required very high. This is the reason why it is extremely important for the future of the field of the application of multimedia interactive technology to education to find efficient ways to share the efforts of courseware development. One very appealing approach is that of reusability. The word reusability evokes the idea of adapting and combining already existing reusable components of learning material instead of starting every new authoring effort from scratch.

The reasons for reusability may be summarized as follows:
- reducing the development cost
- improving the quality
- being able to give quick answers to educational needs. In fact the concept of reusability goes hand in hand with the concept of rapid prototyping.

In connection with the concept of reusability it is interesting to mention the hypothesis that a new wider market is going to develop for educational software involving the distribution of content-oriented rather than objective-oriented learning material [2]. In fact traditional courseware has a specific educational objective and is oriented toward a well defined target population, while a pool of reusable learning components could be related only to a specific content domain and not to specific educational objectives or users [6]. This independence from educational objectives and target populations makes it possible to think of a database of reusable learning components as a resource useful for developing many different educational applications related to the same content domain or flexible applications which can tailor the learning process on the specific need of a user [5].
2. Reusability in other fields

The concept of reusability is widely known and practised in several fields of engineering (even if often they do not use the term reusability). Civil, mechanical and electronic engineering have a long tradition of designing by using existing and commercially available components. Catalogues of components are commonly available in those fields. Sometimes these components may be very complex as in the case of microchips in electronic engineering.

In the field of software engineering, reusability is a more recent phenomenon. The first attempt of reusability was based on functional abstraction and the first practical tools to implement this concept were libraries of subroutines. The next step was data abstraction which supported a variety of linguistic tools (classes, clusters, modules, packages...). The most recent achievement has been object-oriented programming which, by enhancing data abstraction with the possibility of inheritance, provides a sound mechanism for supporting reusability.

In the field of education, reusability is little more than a dream. It is true that there are already some examples of reusing educational material. A typical case is the re-purposing of existing videodiscs, i.e. adapting a videodisc to meet an objective which is different from the original by building a new delivery software. However cases such as these implement only a very restricted concept of reusability as it will become clearer below.

It is true that in building the concept of reusability in education several ideas may be borrowed from the field of software development and from object-oriented programming. However, the specific characteristics of the fields of multimedia and education require that, to some extent, the concept of reusability be remodelled. To clarify this point it is useful to make comparisons with the field of software development. Classes such as collection or linked list [4] can be thought of as objects which are reasonably independent from any specific application and it is highly likely that they will be reused in many different contexts. Unfortunately no equivalent situation can be found in the field of education where the degree of complexity is much higher. In education, most concepts are not completely understood and formalized; the concept of correctness is much more blurred than in the field of software development; and no equivalent for the concept of class, which is the basis of the object-oriented approach, has yet been found. Not only do all these factors make it almost impossible to define context-independent components but, even within the same learning domain, it is very difficult to define components which can be easily retrieved, modified and reused by different applications. Another barrier to reusability, which has much more serious implications in this field than in the field of software development, results from linguistic differences which can make a potentially interesting component very difficult to retrieve and to use. Finally the present lack of standards for multimedia documents is a further obstacle to reusability which must be overcome.

3. Reusability in educational design

Some of the obstacles to reusability are at least partially of a psychological nature. "The not invented here syndrome is well known. The implication of this in practice is that reusable components should bring a significant advantage over home-brewed ones in terms of quality, ease of access and cost. A marginal advantage will not be enough to convince authors to use somebody else's mousetrap rather than invent their own." (adapted from [4]).

Following are some of the questions that need to be answered to approach the actual advantageous reusability of learning components:
What is a reusable educational component?

In the jargon of ESM_BASE multimedia reusable components are referred to as Units of Learning Material (ULM). It is quite obvious that the definition of ULM it is a critical aspect which deeply affects its actual reusability.

ULMs can be thought of as the basic blocks of multimedia learning material used by authors to develop new educational applications. ULMs can be taken out from existing educational applications, may be composed from existing fragments of material or may be built from scratch. A ULM is an abstraction which collects content segments of various media and type, hides their differences and offers a common and consistent interface in order to accomplish fruition and/or authoring activities [3]. All the multimedia aspects of the learning material are encapsulated in the definition of ULM.

ULMs can be simple monomedia objects (referred to as atoms), or complex object composed of other (simpler) ULMs. They may be of two types:

- **Multimedia Sequences (MS).** A MS corresponds to the delivery of a multimedia message to the user possibly comprising texts, still frames, dynamic sequences and digitized audio. No interaction is included in a MS, but the standard one provided by the delivery system (quit a MS, alter the velocity of delivery, replay, etc.)

- **Facilities.** A facility is any kind of (possibly interactive) program to be used by the author or by the learner to perform a specific educational function (a test, a simulation environment etc.)

In this exploratory phase of DELTA the emphasis is mostly on Multimedia Sequences. The glue which binds together different ULMs in a composite learning sequence has to be provided by authors using the facilities offered by a specific authoring environment. To ease the process of composition of ULMs into a learning sequence, each ULM is endowed with an EventTable where the set of the events (such as mouse clicking in particular zones, keystrokes, timeouts, etc.) visible by external applications are defined.

The internal structure of ULMs, which are complex multimedia objects, is based on the use of layouts. Layouts are abstract objects which provide the structure of ULMs, i.e. the relation between their atomic components. The same layout can be instantiated on different contents. The possibility to share the same layout among many different ULMs represents another interesting aspect of reusability. More details about the structure of ULMs can be found in [7].

How to access reusable components

"The best reusable components in the world are useless if nobody knows they exist, if it takes a long time to obtain them or if they cost too much. The practical success of reusability techniques requires the development of an adequate database of software components, which may be searched by appropriate keywords so that a potential user will find out quickly whether some existing component satisfies a particular need" [4].
As a further enhancement, "network services should also be available, allowing electronic ordering and immediate downloading of selected components" [4]. These statements, which refer to the field of software development apply equally well to the field of courseware development.

We must therefore think in terms of databases of reusable components. In the jargon of ESM_BASE we refer to such databases as DBLMs (DataBases of Learning Material). In a DBLM there is room for ULMs (which are undoubtedly the most important inhabitants of a DBLM) and for other entities as well, such as knowledge structures, contexts, ulm layouts, which represent less obvious aspects of reusability.

In the remainder of this section some elements of the conceptual structure of a DBLM will be discussed with major focus on the retrieval of ULMs.

5.1. HOW TO QUALIFY A ULM IN A DBLM

Authors, when looking for a given ULM, must be able to express their needs in terms of parameters which are meaningful and natural to them. Therefore ULMs must be qualified with suitable attribute and accessory information which help the retrieval process. It is obvious that each ULM must have a set of objective, cold attributes such as name, source, media involved, language, and so on, which deserve little discussion. The cognitive and pedagogical point of view is much more interesting. ESM_BASE has made an analysis of the author needs which has lead to the following approach to the pedagogical qualification of a ULM. Each ULM may be qualified under three complementary points of view:

- its content
- its possible contexts (i.e. the contexts in which it has been actually used or for which it has been built).
- its relation to the context (i.e. the functions performed by the ULM in a context or those features of the ULM becoming activated in a given context). Typically a context is a course in which a given ULM is used.

The explicit presence of contexts, though undesirable from the standpoint of complexity, is an essential component of the conceptual schema of a DBLM since very little can be said of a ULM which is context independent.

The content description of a ULM has been kept informal and loosely structured since it is obviously impossible to provide a complete formal semantic description for the content of ULMs due to the huge wealth of information contained in a typical ULM (think of a complex piece of text or of a picture or even worse of a video!). Besides, a formal or in any case deeply structured content description of ULMs would be of little or no use to authors whose basic need is to retrieve quickly and easily a set of potentially interesting ULMs rather than the ULM which solves a given problem. For these reasons it has been decided to describe the content of ULMs simply by sets of keywords and/or comments.

Context descriptions may be deeply structured and include information of a structural and pedagogical nature such as the content structure referred to by a given context (i.e. by a given course). This structural information is typically not only fairly manageable (being the result of an explicit design effort of the context author), but it is also extremely useful to the database user. In fact it gives a more complete meaning to ULMs by providing an idea of the function performed by a ULM in a given context. Moreover, it fosters another aspect of reusability related to contexts. This means that the author of a new course may reuse some of the features of a given context, the most significant possibility being that of reusing a content structure already existing in a DBLM.

Finally it should be noted that most of the pedagogical description of a ULM logically belongs to the relation of the ULM to its context(s). In fact pedagogical attributes are not absolute features of ULMs but depend on the way a ULM is used in a given context. As
an example, the function of a ULM may be an *educational explanation* in a given context and *motivation* in another one.

Figure 1 provides a metaphorical representation of the description of a ULM in a database. It should be clear that some ULMs may have very little or no context in their qualification. In this case their metaphorical description lies on the content-axis. In most cases a component has, at least partially, all the three qualifications and therefore its representation lies somewhere in the Cartesian space.

![Figure 1. A metaphor for the qualification of a ULM in a database.](image-url)

5.2. INTEGRATING QUERIES AND NAVIGATION

The above discussion refers to the possibility of making queries on a DBLM. In fact the author may look for potentially interesting ULMs by specifying aspects of their content, contexts and relation to context. However the traditional query mechanism might not be sufficient to cope with the needs of the authoring process and might need to be integrated with a navigation mechanism. For example, when a component X has been retrieved by an author via a query on the database, it may be very helpful to reach the next useful component Y by navigating a link network. This is especially useful if it is highly likely that Y is directly linked, or at least very close, to X. In fact, in this case, even if the author does not exactly know what he/she wants, through navigation he/she may reach something that is helpful. In other words, to build a really efficient mechanism of access, we have to merge the concepts of database and hypermedia. The information in the database should be heavily cross-referenced by links, the majority of which should be created and maintained by the system, rather than being explicitly maintained manually. This automatic linkage-creating mechanism will be crucial in producing databases of highly reusable material.

Figure 2 shows, in principle, the structure of a ULM. A ULM consists of a multimedia component, the description of its content, the description of the contexts where the ULM is used, the description of its relation to those contexts and, finally, a set of links to support navigation.
6. Modifying reusable components

Each author or user of a DBLM should be able to modify existing ULMs to adapt them to his/her actual needs. This could simply be considered as an editing problem, easily dealt with by a good multimedia editor. However, we believe that another factor, which deeply affects the structure of a DBLM, should be mentioned. At any given moment, a user may want to personalize some of the ULMs by adapting them to his/her needs, and then later refer to this personal version of the DBLM, which now consists of the original version enhanced by the modifications he/she has made. Since a DBLM can be accessed by different authors having different personalization needs, some provision has to be made to support personalized views of the same DBLM. The solution adopted by ESM_BASE consists in organizing a DBLM as a hierarchy of views endowed with some kind of inheritance mechanism (see fig. 2). In object-oriented programming the inheritance mechanism provides the basis for personalization. Fig. 3 refers to a mechanism which is rather different from the class inheritance typical of object-oriented programming. Here each personalized view inherits the knowledge of the ULMs in the parent view. However, the user has the right to modify some of them or add new ones to them.
Figure 3. A DBLM as a hierarchy of views

Theoretically, there is no limit to the depth and width of the hierarchy. The conceptual scheme of each view is inherited from the root and, for practical reasons, it is assumed that it cannot be altered by the view authors. Several problems may arise from this structure such as the integrity and the internal consistency of a view with respect to modifications being made in other views and the unnecessary duplication of learning material stored in more than one view. These problems require suitable criteria and rules for an efficient management of a DBLM intended as a collection of views.

In the present implementation of the DBLM shell, the inheritance mechanism can be controlled by the user who can specify the objects to be inherited by a view. Moreover the user may choose between two different inheritance modes, namely copy mode and reference mode. When using the reference mode, any modification to the parent view also affects its sons. It follows that there is the need for concertation among the authors of different views, to prevent unexpected modifications and consequent inconsistencies occurring in descendant views. The copy mode, on the other hand, duplicates the objects providing protection against the above problems.

This possibility of control over the inheritance mechanism is fairly general and allows to implement different management techniques some of which are listed below:
- a pure inheritance based on the reference mode
- a version mechanism based on the concept of making impossible any modification to a view after its release. This mechanism allows inheritance while preventing possible inconsistencies. The main drawback is that modifications are only allowed through the creation of a new version of a view.
- a pure copy mode
- a combination of pure inheritance and copy (and paste). Here, though inheritance might be considered as the default mechanism, facilities are provided for the author of a view to specify which sections of the ancestor databases have to be copied, and therefore duplicated, inside his/her view.
7. Minimizing linguistic obstacles

This factor is typical of the field of education and has no equivalent in the field of software. In most western countries language is not a serious obstacle to software reusability, however educational material is usually dependent on a specific language and often on a specific national culture.

Differences in languages and cultures should not be a major obstacle to reusing the same design effort. This is especially important for European countries, which in order to obtain a market which is large enough, must think in terms of a European rather than a national market. The multilingual aspects of a DBLM can be thought of as belonging to two different groups:

a) problems concerning the availability of multiple versions of the same ULM oriented to learners with different linguistic and cultural backgrounds
b) problems concerning the access to a DBLM by authors with different linguistic backgrounds.

7.1. AVAILABILITY OF MULTIPLE VERSIONS OF THE SAME ULM

ESM_BASE has assumed that a ULM, when delivered to its final user, should refer to a specific language and to a specific culture. However, when designed, it should be considered as a broad structure which can host equivalent multilingual or multicultural elements. This means that, at the time of delivery, a ULM should be tailored to the specific linguistic background of the user by a sort of dynamic binding process (Fig. 4).

Figure 4. Tailoring a ULM for a specific user

This approach has some implications at the level of the detailed architecture of a DBLM. As already mentioned, a ULM is generally a complex multimedia entity composed of basic monomedia entities referred to as atoms. Multilinguality has been rooted in the very definition of the atom which is structured as a set of equivalent monolingual elements in different languages.

It is intended that the element appropriate to the specific linguistic background of a user will dynamically selected at delivery time. If the atom is language independent in nature, the set collapses to a single element set.

It is interesting to note that if a user is familiar with more than one language with different levels of proficiency, he/she may access a richer body of material. The delivery mechanism of the DBLM will use the language that the user knows best to tailor the required ULM, and only in the case that some element is not available in the user's first language, will use elements in a language of a lower priority.

This solution has interesting implications on the design process, which could be carried out by the author in a specific language and be extended almost automatically to all the languages represented in the database. Of course, to make it possible, provision has to be
7.2. ACCESS TO A DBLM BY AUTHORS WITH DIFFERENT LINGUISTIC BACKGROUNDS

Authors of different mother tongues should be able to use a DBLM in their own language. In particular they should be able to easily access ULMs not available in their mother language in order to look for potentially interesting material. To achieve this objective not only must a DBLM be adaptive to the user’s mother tongue by choosing among different sets of commands and messages, but it also must be able to convert language dependent queries expressed by authors into language independent ones. This means that the values of the attributes of the entities in the database referred to by authors’ queries must also be converted from the original language to suitable internal forms.

In the present implementation this is obtained by introducing between a DBLM and its client applications a software layer which takes into account the linguistic dependencies and translates the requests to a DBLM into an internal, language independent form.

8. OTHER FACTORS INFLUENCING REUSABILITY

In this section some obvious, but very important factors influencing the actual reusability of a database of ULMs will be briefly discussed.

8.1. LOGICAL AND PEDAGOGICAL FEATURES OF ULMs

To be reusable, a ULM must first be usable, not perfect but usable. This means that a component should exhibit properties like understandability, correctness, quality, etc. ULMs should exhibit a good level of coherence. This means that different conceptual, independent or loosely coupled aspects should not be present within the same ULM as this makes it difficult to use without modification and difficult to manage from a logical point of view.

Finally each ULM should ideally exhibit a unique communicative, informative or educational function (ranging from very simple to very complex) and perform it completely. The completeness of the educational function allows the author to use a component without needing to retrieve another complementary component.

8.2. INTEROPERABILITY

This property refers to the possibility of exchanging multimedia material between different hardware/software environments and the possibility of linking components or applications which were not originally built to live together.

The solution to the first problem is dependent on document interchange standards for multimedia documents. At the moment, comprehensive proposals are still to come. ODA/ODIF, POSTSCRIPT, RTF (Rich Text Format), VEX (Video Extension for X-Window) are all existing, de facto or proposed standards which are mostly oriented to textual or graphic material or, as in the case of VEX, try to extend existing standards to multimedia documents. An ISO commission is presently working on this problem which is crucial to the process of sharing multimedia material.

The second problem is more complex since, at the moment, it is not realistic to think of standards or a uniform discipline accepted by application developers (which would also exclude most of the existing applications). Therefore specific research is needed to define protocols and tools for interoperability.
8.3. PROPERTIES OF A DATABASE OF REUSABLE BLOCKS

A user should be able to find in a pool or database of reusable components a significant portion (hopefully the majority) of the material he/she actually needs. A DBLM should not be only a marginal help, but a substantial help to the work of the author. A database may be oriented either to a specific content domain and to several classes of users or to several domains and to a specific target population. In any case the author should only need to add a little glue to link together the blocks found in the database. In other words a database should be as rich and as complete as possible with respect to a given domain.

9. Conclusions

From the above discussion it should be clear that the concept of reusability for educational software has several complementary components which make it very complex to put into practice. From the conceptual standpoint, many of these components are not yet fully clear and require further investigation both on the technical and on the pedagogical side. However we already have enough both conceptual and technological instruments, to start applying the concept of reusability to real world cases. Concrete experiences will be the best tool for refining existing concepts and methodologies.

At the present moment, the ideas presented in this paper have been translated into a prototype of a DBLM shell which has been used for building an actual DBLM in the field of Plate Tectonics. This database is a non-toy application which contains approximately 1,000 ULMs most of which have been built from existing material. In particular many of the ULMs are derived from material on a videodisc built by the authors for a previous project on seismic education. The process of construction of the database has provided tremendous feedback on the architecture of the database. A better definition of the architecture of ULMs, improved structures to allow for multilingual ULM design, a clarification of the role of contexts and of the relation of ULMs to contexts are some of the most important results of this experimental phase.

By linking the DBLM to specific authoring environments and using it for authoring, it is very likely that new important feedback will become available. This experimental work will only be partially completed within the framework of the DELTA exploratory phase and will also require the investigation of possible domain dependencies in the structure of a DBLM as well as further details concerning the market distribution of DBLMs.

10. References


