The Design of an Adaptive Multimedia Presentation System

Nick Rowe
Faculty of Technology
Bournemouth and Poole College
Bournemouth, UK
e-mail: nrowe@bpc.ac.uk

Abstract—**The nature of adaptive multimedia presentation system within an e-learning environment is considered and the ability of such a system to add and remove fragments of a presentation discussed. The advantages and drawbacks of using learning objects are described and the ability to map an object to additional data identified as a means to provide links to other objects. The practical design of an adaptive E-Learning system is then considered by first referencing two open source systems, (AHA and MOT), and then considering the requirements of a prototype system. The principle components of such a system are identified and the system structure outlined. Relationships between principle entities in a presentation database are then discussed with the emphasis on adapting the content of the presentation by providing more or less detail. Links between learning segments are then discussed leading to consideration of the different types of link and how this is likely to affect the user experience. Finally, future issues are discussed leading to a consideration of the topics that would need to be evaluated in the use of the prototype system**.

Keywords – multimedia; E-Learning; learning objects; adaptive; ontology; education.

I. INTRODUCTION

Firstly, the paper considers the nature of adaptive E-Learning with reference to the common components found in such systems. The methods of *adaptivity* are then considered and split into two discrete mechanisms: navigational and content-level adaption, the latter being considered in the context of learning objects. The advantages of fragmenting a presentation are investigated and the benefits of interoperability and reuse are weighed against educational context and independence. The concept of wrapping the learning object in meta-data is introduced and using this data to provide links between the objects explained. Different link relationships are then defined and pre-requisite and co-requisite relationships discussed in detail.

The practical design of a prototype E-Learning system is described where a multi-modal interface has been designed to allow different elements to be synchronized under one timeline to allow reinforcement of the content. The structure of prototype E-Learning system is then described and the design of entities and the relationships between them identified. In particular, the relationships between the learning objects, (called segments), link segment and pending question entities are discussed.

The mechanisms for adapting the presentation content are detailed and the formulation of an Adaptive Descriptor identified as a key element in the process of producing presentations with different levels of detail for different users. Next, the authorship of the content is approached with particular regard to populating the Segment Level Identity using a practical editing system for the segment. This editing system allows additional fragments of the presentation to be inserted or fragments of the presentation to be removed. By such action, different levels of detail can be created from a single segment.

Lastly, the structure of these learning segments are considered within the context of knowledge-based ontologies. Characteristics of these segment maps are explored and the salient relationships between linking segments identified; particularly in regard to pre- and co-requisite links. Conclusions are then given and characteristics of the prototype design itemised. Design advantages are listed with a view to being considered against the increased production time of these materials.

II. ADAPTIVE E-LEARNING

Generally, acknowledging the important relationship between individual learners and the material they are viewing has a long history. Shute and Towle, [10], note that the goal of aptitude-treatment interactions, (ATI), research is to provide information about learner characteristics that can be used to select the best learning environment for a particular student to optimise learning outcome. They go on to itemise four components of adaptive E-Learning:

- Content Model, including a knowledge map
- Learner Model, containing information about the user
- Instructional Model, concerned with the presentation of materials
- Adaptive Engine, which uses information from other models to drive the system

Systems can access the learner in terms of domain-dependent information and domain-independent information. The former gains knowledge of the learner through pre-tests and performance data and the latter keeps track of the cognitive abilities and personality traits of the individual. Systems concerned with adaptive instruction tend to base their *adaptivity* on assessments of emergent content knowledge or adjustments of material based on

 ISBN: 978-1-61208-195-3

learner styles. The latter is a less suitable criterion than cognitive abilities for making adaptive instructional decisions.

It is true to say that research into adaptive hypermedia is at the crossroads of multimedia presentation and user modeling. Brusilovsky, [2], defines such systems as giving a presentation that is adapted specifically to the user's knowledge of the subject and suggests a set of most relevant links to proceed further. The second part of the definition is really a type of navigational adaptivity where the learner is given a level of control of over what content to see. So, two distinct areas of adaption are created: content level adaption, often called adaptive presentation, and link level adaption, called adaptive navigational support.

One interesting area that Brusilovsky identifies is the requirement to manipulate a presentation in certain ways according to the user needs. The information is offered in the context of *canned text adaption* and suggests applications can insert and remove text, alter fragments, stretch text, sort fragments and dim fragments. If the concept is extended to multimedia applications then these presentations can be manipulated in a similar manner. The fragments can be manipulated via an adaptive engine. This leads to the practical realization that presentations need to be reduced to fragments to allow these elements to be manipulated. These fragments are generally termed *learning objects* and there has been plenty of research around their use.

A good example of adaptive navigational support offered by an application is AHA!, an open source adaptive hypermedia platform, [6]. The system uses adaptive linking to suggest content for the user. It makes use of prerequisite relationships between the learning objects to link related references ensuring that the user has the required knowledge base to understand a given link. In this manner the user makes decisions about the content they wish to learn.

III. LEARNING OBJECTS

The definition of a learning object is *any entity, digital or non-digital, which can be used, re-used and referenced during technology-supported learning*, [7]. Although the definition is easily understood and widely accepted, the advantages gained by splitting up a lesson into learning objects are somewhat controversial. One of the biggest benefits often sited are that these objects can be reused and repurposed, [1]. However, this interoperability and reusability may have been overstated in the past. McGreal, [9], points out the difficulties in taking a learning object and reusing it in a different environment. This is principally because it is difficult to create learning objects independent of the context it was made in. The likelihood is that the object bears the imprint of the ideology and culture it was produced in.

Consequently, it is difficult to standardize a learning object and an object-oriented approach, as applied to software environments. This is incongruous in the complex context of learning, especially when the learning material is based on narrow technical and specialized concepts. Despite the challenge, the concept persists driven by the joint goals of reuse and adaptivity.

However, whilst learning objects may not be easily reused, segments of a similar content may be referenced and linked to provide the user with additional information.

Boyle, [1], describes the learning object as a wrapper around this object. This wrapper describes the component structure of the object, and includes the descriptive metadata. The learning object is thus packaged in a standard container format. This packaged object can be stored in digital repositories. The metadata permits fast effective searches to retrieve learning objects suitable for a particular purpose. A direct link can be made to the idea of learning objectives in pedagogical theory. This mapping suggests that each learning object should be based on one learning objective or clear learning goal, which links back to the original definition.

The design of the learning objects should be considered carefully to ensure they have minimal bindings to other units, (as well as being as context-free as possible). Even Boyle, [1], admits that this decoupling of learning objects is a considerable challenge and notes that this may be at odds with providing rich, integrated learning experiences. One way round this problem is to create a compound object consisting of two or more independent learning objects that are linked to try to achieve a richness not available to a single object, whilst maintaining a significant basis for re-use.

IV. THE LINKING OF LEARNING OBJECTS

In fact, the linking of learning objects goes further than this and a particular syllabus may be defined as a linked series of these objects. Indeed, much of the research on developing E-Learning systems over the last five years has concentrated on these links. In the design of the open source adaptive hypermedia platform AHA!, (Adaptive Hypermedia Architecture), De Bra et al., [6], describe how the system has been designed to use adaptive linking to suggest content for the user. It uses, what they term, *prerequisite relationships* to link related references and form a path through material. The system is capable of selecting and presenting information content based on the user's previous actions which are processed and stored in a user model. The system selects and annotates the links in a way that guides the user towards the most relevant information. In this way, navigational adaptivity is provided and the system builds concept relationships between the objects.

Once the learning material has been segmented into individual learning objects, two aspects become important for the presentation of these materials. Firstly, a lesson can be considered to be a chosen sequential set of these segments and secondly that any segment presented may, to a lesser or greater degree, be connected to another segment in the learning repository. These two elements become essential to the development of any E-Learning system. Authoring a lesson to be presented becomes a process of choosing already available segments from the repository and creating new segments for areas not available. The presentation system then needs to be provided with a set of links to other relevant segments that the student may find useful and optional decide to view. The data in the repository needs to be mined to find the relevant links to each segment within the lesson.

Whilst pre-requisite links will allow the user to create a path through a series of segments, it is also useful to provide the user with the ability to view *co-requisite* links as well. These links to segments that are close in content to the viewed segment and provide the user with the ability to view segments in some way connected to the current segment to provide reinforcement and to see the same topic from a different perspective. Figure 1 shows how this can be implemented on a practical system.

Figure 1. An Adaptive Linking Dialog showing Pre- and Co-requisite Connections.

The linked segment entries are ordered by a strength metric, (in brackets). This allows the strength of the link to be graded with some indication of the closeness of the topic covered by the other segment. When linked segments are selected they can be played and are shown within the context of the lesson they were designed in. This context may well be different from the original segment, but as such provides the user with a fuller picture of topic.

V. THE PRACTICAL DESIGN OF AN E-LEARNING SYSTEM

In practice, realization of all these concepts gives rise to two distinct functions of any E-Learning system. These are the authorship of materials and delivery of these materials. Cristea et al., [4], describe an attempt to combine two hypermedia systems, authoring with MOT, (My Online Teacher), and delivery with AHA. MOT uses domain mapping to structure and organize the resources. It uses adaption rules to build an *assembly language* of adaption. Concept weights, (meta-data), are then used to alter the presentation and make it adapt to a particular user. These weights can represent different measurable aspects of a learning fragment like difficulty or importance.

A Common Adaption Format, (CAF), sits between the two systems to convert data from MOT into a form understood by AHA. This is expressed as an XML document. In this manner the systems attempt to establish a common platform and format for the representation of adaptive educational hypermedia: an extremely important goal if learning object re-use is to become a practical reality.

VI. THE DEVELOPMENT OF A PROTOTYPE E-LEARNING SYSTEM

With current computing power there is rarely a problem with delivering rich multimedia content. One of the main issues that requires careful consideration is the synchronization of the different media that go to make up the presentation. Languages like SMIL, [3], seek to remedy this by providing a language to allow multimedia components to be synchronized and presented together. Although the presentations produced this way are impressive, authorship is complex.

A prototype development system was designed, [5], with these requirements in mind allowing multiple units acting together to reinforce the overall delivery.

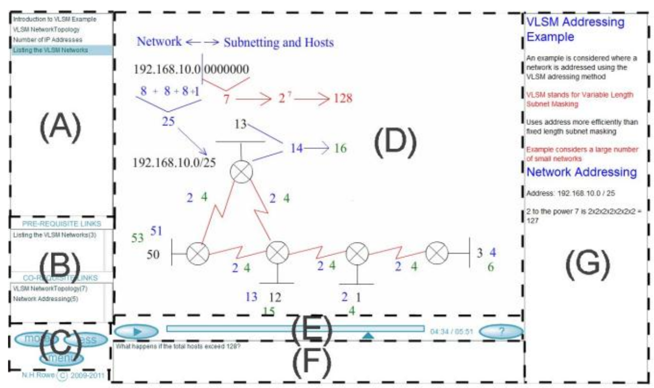


Figure 2. Screen Layout of a Multi-focus E-Learning System.

This has been developed and Figure 2 shows the screen layout of the enhanced system. Here, seven elements are synchronized to act from the same timeline. Element A is a list of segments, (learning objects), that form a lesson and forms a selectable table of contents for the presentation. B provides a selectable display of linked segments that can be optionally viewed by the user. C provides the user with buttons to increase or decrease the level of detail of the presentation. D is the viewing panel for the audio-visual material and E is the timeline control area. This section gives the temporal control of the presentation and also allows the user to ask textual questions. These can be accessed by the segment author and additional presentations created to answer the question. When an answer is published it can be viewed by all users in area F and thus this area is filled with frequently asked questions for the currently playing segment. When a question is selected, the solution presentation plays temporarily interrupting the current content for the span of the answer. Area G contains incrementally loading HTML, (iHTML). Here the content, text and images, is displayed and reveled in real-time, synchronized with the main presentation in a similar manner to subtitles. Each block, (or paragraph), of the HTML code is not displayed until a specific time is reached in the presentation.

With the publishing of answers to previously asked questions, during the life of the presentation more questions are likely to be asked and therefore the presentation matures over time. This provides more supplementary information useful to a learner viewing the presentation for the first time.

VII. The Structure of an E-Learning System

Once the decision in establishing the segment as the heart of an E-Learning system has been made, the rest of the system can be designed around it. Entities including the user and materials, to test the user knowledge, can be included in the E-Learning database.

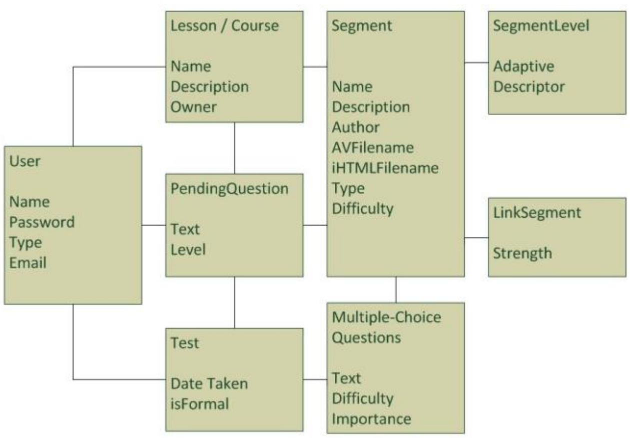


Figure 3. A Prototype E-Learning Structure.

In the development of the materials the educational concepts must be isolated from a lesson of a course and developed into learning objects. The syllabus of a lesson consists of an ordered set of concepts and a course is an ordered set of lesson. Each concept is formed into a segment. Initially, a segment contains audio-visual resources required for its presentation. This entity is then given a set of attributes including the name of the AV file that contains the presentation. This information can be accessed by independent engines that could be looking for links between segments. The algorithm to identify these links could be complex but the end result is simply to record the link between two segments in the *LinkSegment* entity and record a strength attribute that gives an indication of the strength of the link between the two segments. The *PendingQuestion* entity records the student question linked to a segment for access by the author of the segment to provide an answer. Once answered the question becomes another segment linked to the first with a maximum strength value.

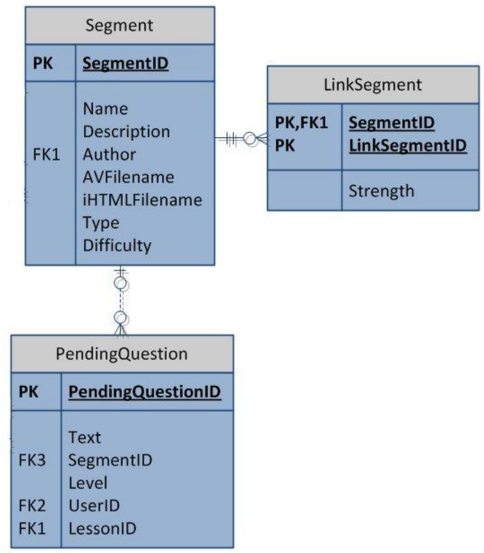
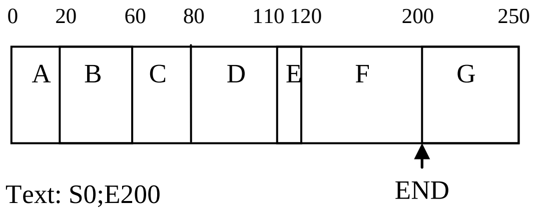


Figure 4. Relationships between a Learning *Segment*, *LinkSegment* and *PendingQuestion* entities.

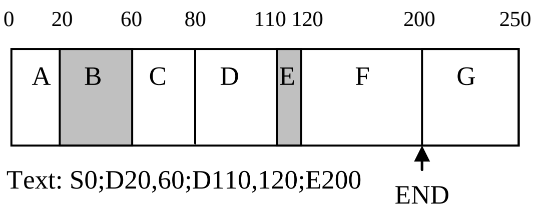
VIII. Adapting the E-Learning Presentation

To make a segment adapt to the user's needs during presentation the author must also determine parts of the AV presentation that will be viewed at different levels of detail. By providing these different levels each segment becomes adaptable. During a presentation, the user can be presented with the segment information at a preferred level of detail. The user can then alter this level to provide more or less detail during the presentation. The system can record these levels and change these levels based on other information in the database including the results to tests linked to the segment. Thus, the system adapts to the user needs by presenting the material at the correct level of detail.

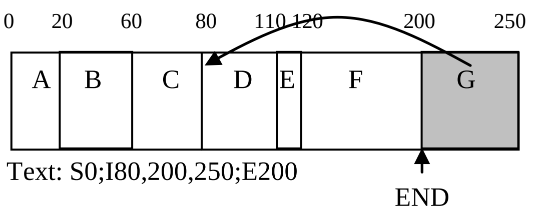
Thus, more or less detail can be created to a standard form and adaptively chosen for the user. A textual code is used to allow the system to piece together the presented form for the level chosen and acts as an adaptive descriptor for the segment. This is shown in Figure 5. Part (a) shows the media file being played as it was recorded from frame 0 to 200. The control text simply gives the end frame so that additional fragments are not played at the end of the file. Part (b) shows fragments of the media file being left out to create a less detailed presentation. Here, fragments B and E are left out of the presented sequence. The control text indicates which frames are to be removed. It also includes the end frame. Part (c) shows more detail being added to the presentation by substituting the larger fragment G in between C and D, (at time 80). Here, more detail can be added to specific parts of the file and therefore particular concepts are elaborated within the segment. These additional fragments are added to the end of the media file and can be additionally recorded at the time the presentation is made. The adaptive descriptor marks the frames to be removed and the frames to be substituted. Thus, a single media file is used for all levels of detail and adaptively presented by use of the set of descriptors for each segment at different levels.



(a) Normal level of detail, (as recorded). Segments A to F are played sequentially



(b) Less detail in presentation. Segments A, C, D and F are played sequentially



(c) More detail in presentation. Segments A, B, C, G, D, E and F are played sequentially

Figure 5. Three levels of detail from a single audio-visual fragment.

The relationship between *Segment* and the *SegmentLevel* entities is shown in Figure 6.

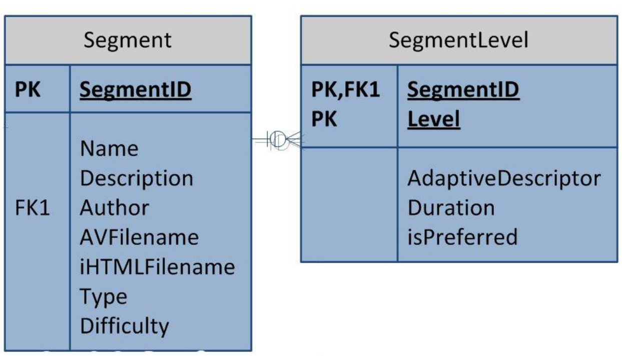


Figure 6. Relationship between a segment and levels of detail.

The attributes of the latter entity consist of the *adaptive descriptor*, a total value for duration of the segment at that level and a marker for the author's preferred starting level. The duration is calculated at the authorship stage in order that, when presented, the total duration of the lesson can be calculated easily by adding all the segment durations at each particular level.

The practical consequence of this editing process is to populate the Segment Level Entity as shown in Figure 7.

SegmentID	Level	AdaptiveDescriptor	Duration	isPrefered
1	0	S10;H20,0;H30,1;H80,2;I120,150,175;E150	165	0
1	1	S10;H20,0;D25,55;H80,2;E150	110	1
1	2	S10;H20,0;D25,55;H80,2;D85,110;E150	85	0
2	0	S5;E55	50	1
2	1	S5;D40,50;E55	40	0
3	0	S0;H10,0;H25,1;H115,2;E150	150	1
3	1	S0;H10,0;D15,45;H115,2;E150	120	0
4	0	S10;H25,0;H80,1;I115,150,165;E150	155	1
5	0	S5;H40,0;E145	140	1
7	0	S0;H1,0;H5.172,1;H14.994,2;H20.819,3;H26.383,4;E33...	33.3844	1
7	1	S0;H5.172,1;H14.994,2;H20.819,3;D21.916,33.266;E33...	22.0344	0
7	2	S0;H5.172,1;D10.84,33.266;E33.3844273	10.9584	0
8	0	S0;H16.718,0;H41,1;H64,2;E112.0389737	112.039	1
9	0	S0;H9.133,0;H13.035,1;H37.666,2;H81.136,3;H109.478...	260.806	0
9	1	S0;H9.133,0;H13.035,1;D14.341,97.541;H109.478,4;D1...	111.256	1

Figure 7. Display of data within the Segment Level Entity.

Here, in the left hand column, 9 segments are shown with an ID from 1 to 9. For each segment, levels are shown from 0 to a maximum value in the next column to the right. The middle column shows the adaptive descriptor with elements separated by a semi-colon. Duration, (in seconds), for each level of detail are given in the next column and it can be seen that the higher the level, the less time the content lasts for. This is because at each increasing level part of the presentation is removed according to the editing process used by the author that formed the data. The right hand side column shows which level is preferred by the author as a default level. This level may be changed by the user, and remembered by the database, but this will be the level viewed by this particular user the first time they encounter this content.

The highlighted second row of the table indicates the preferred level of segment ID 1. The segment will be shown at this level of detail by default. The user can reduce the level of detail, for example if the material is familiar to them, by selecting the 'Less' button. In this case, the segment will now only be 85 seconds long and a portion of the presentation will have been removed. If the user pressed the 'More' button instead, the level would decrease from 1 to 0 extending the presentation to 165 seconds. The difference in each level of the segment is contained in the adaptive descriptor and in each case as the level increased, part of the presentation was removed. For example, in the case of moving from level 1 to 2 an additional element 'D85,110;' is added which has the effect of removing 25 seconds from the presentation and therefore the duration reduces by 25 seconds. By editing the 25 seconds out of this level the author has made the decision that this portion of the presentation was not required at this level. The user can alter the level of detail, by selecting the more or less buttons, if this turns out to be important, but in this manner the presentation is less likely to contain unnecessary detail, but be pitched at the correct level.

IX. AUTHORSHIP OF AN ADAPTIVE PRESENTATION

Authorship of such a system relies on the choosing fragments on a temporal basis and marking sections to be excluded or included at a particular level. A particular lesson presentation is driven from a sequential set of segments. Each of these segments has additional data connected to the AV file and an adaptive descriptor allows these additional elements to be synchronized with the original AV file. Once the AV files have been collated, the author must mark on a time scale fragments of the segment to be deleted and inserted at a particular level. The adaptive descriptor is then created for each level and the *SegmentLevel* entity populated. The adaptive descriptor can also be used to mark times within the segment presentation to display the iHTML elements to allow them to synchronise with the main audio-visual element. Playback is then controlled by the adaptive descriptor with the playhead being moved ahead in real-time to skip over a section or moved to a completely different time to allow more information to be inserted.

Creating the descriptor consists of the author selecting the segment and this can be played in the system. In this creation mode a new set of editing functions are accessible to allow sections to be marked for insertion and deletion together with the ability to mark in time when iHTML paragraphs are to be displayed. A mechanism for doing this is shown in Figure 8.

These activities are done at each possible level and the author can create new levels, change the current level and mark a preferred level. This is used as a default level for a presentation if no additional information is in the database.

Generally, the original presentation will contain the maximum level of detail and new levels will delete selective sections to provide less detail, but the author may add addition sections with more detail if required. Once the process is complete the author publishes the segment and the adaptive descriptor is created for each level and the *SegmentLevel* entity populated for each level.

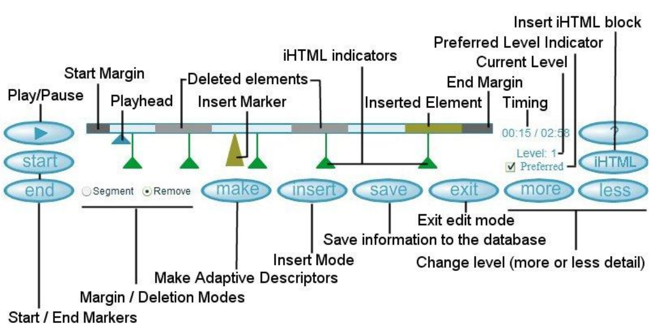


Figure 8. Editing functions for a segment.

The duration of each level is also stored so that the overall duration of a presentation can be calculated quickly. This value will change due to the choice of levels at each segment. The time bar reflects these changes in real-time as changes of level are made by the user.

X. THE STRUCTURE OF LEARNING SEGMENTS

There has been much work done on the structure and classification of knowledge. Since the Dewey Decimal system we have tried to position the vast and varied segments of knowledge into some kind structured network. Just like placing a book on a particular shelf in a library, ontologies seek to place nodes of knowledge onto a two dimensional map.

Holohan et al., [8], note that the combination of these ontologies with learning objects provide a powerful means to creating adaptable educational presentations. Authors can select and customise new or existing subject ontologies and employ a certain teaching and learning strategy in the generation of learning objects. They can also configure systems to offer strictly sequenced presentations to students, or to allow also varying degrees of free student navigation, based on the runtime incorporation of domain ontologies. Students in turn can take the generated courses in the preconfigured delivery environment, and this delivery is dynamically customised to the individual students' preferences and constantly monitored learning track.

There have been many different approaches to developing a structure and framework for this and mathematical systems have been and are being developed to map segments into a meaningful array providing information on the relative proximity of neighbouring segments. This, in turn, leads to providing an automatic metric for linked segment strength.

In this context, it is useful to define some characteristics of these segment maps. Figure 9 shows four types of relationships between learning segments and identifies different relationships between these objects.

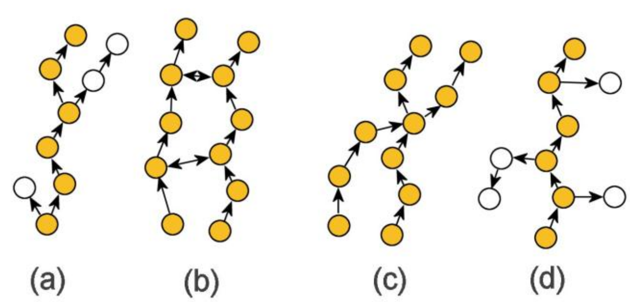


Figure 9. Relationships between segments in a knowledge database. (a) a lesson, (b) co-requisite linking, (c) twisting lesson paths and (d) question segments.

Part (a) shows that a lesson can be defined as a pathway through a set of segments, where the previous segment is a pre-requisite of the next. Branches off this path may exist, (and be given to the user as alternatives), but, at some stage, a decision is made about the "preferred" route. Part (b) shows co-requisite linking between segments belonging to different lesson paths. By giving the user a link to the other lesson path the idea of lesson cross-talk occurs, where a user jumps across to another lesson path. This can carry on indefinitely provided enough lesson paths contain links. Part (c) shows that lesson paths may cross each other and the segment belonging to both lessons may have an elevated status as a result. Lastly, part (d) shows the relationship between a lesson path and segments created to provide solutions to questions raised by these segments. Effectively, a single solution segment is an orphan as it does not belong to any independent lesson path. However, is very tightly bonded to the original segment and may be part of a path created by users asking further questions about solutions given. This is, in fact, how an on-line discussion forum already works. It can be noted too that pre-requisite links, (including solutions to questions), are unidirectional, while co-requisite links are bidirectional.

XI. Conclusions and Future Issues

A method for allowing different levels of detail of a single learning segment to be adaptively displayed has been described together with an interface to allow the user to optionally select linked segments and frequently asked questions. The user can also submit questions to be answered by the segment authors to add to this set. Different algorithms can then be experimented with to provide these links with an appropriate strength metric. The diversity of approach and context of closely related segments can only serve to provide the user with different learning approaches to similar subjects. By splitting the presentations in smaller segments the relevancy to linked segments is likely to increase. The downside is that it is unlikely that presentations will be produced by simply selecting segments from the database and will generally require a larger presentation to be split into smaller units based on pedagogy. In this manner, continuity between segments is preserved. However, if lessons are formed from a series of co-requisite segments selected from the database by some engine then problems with continuity may arise if the segments are created by different authors or different contexts. Even this aspect is not simple as discontinuity of delivery within a lesson is not always a bad thing. It is often used as a tool to assist concentration and refresh the learner.

The separation of longer audio-visual files into smaller segments, adding data to these segments and then marking sections to provide different levels of detail all add to the production time of a presentation. There needs to be clear benefits that offset this extra effort. These immediate benefits are likely to be felt by users of the system in following areas:

- the ability to ask questions,
- to gain access to answers for other users' questions,
- to be able to increase and decrease the content detail,
- the ability to branch to other linked segments,
- to allow engines to trawl through the learning database and create links between segments automatically.

The secondary benefits are likely to include the advantages of designing a lesson or course with access to a central coordinated digital repository of segments that are conceptually linked and the ability to use data held in the user model to make decisions about the level of detail required for a segment for a particular user. These benefits increase with an increase in the number of users: using the system to learn or to author segments.

A prototype design has been developed and the next stage is to evaluate its effectiveness with a view to determining if the time taken to produce the materials in this manner is justified.

References

[1] Boyle, T., 2003. Design Principles for Authoring Dynamic, Reusable Learning Objects. *Australian Journal of Educational Technology*.

[2] Brusilovsky, P., 2001, Adaptive Hypermedia. *User Modeling and User-Adapted Interaction* 11. 87-110.

[3] Bulterman,D.C.A, Rutledge, L., 2009. SMIL 3.0 Flexible Multimedia for Web, Mobile Devices and Daisy Talking Books. 2nd Ed. Berin:Springer-Verlog.

[4] Cristea, A.I., Smits, D., De Bra, P., 2005. Writing MOT, Reading AHA! - converting between an authoring and a delivery system for adaptive educational hypermedia. *A3EH Workshop*, AIED'05 (2005).

[5] Cutts, S., Davies, P., Newell, D. and Rowe, N., 2009. *Requirements for an Adaptive Multimedia Presentation System with Contextual Supplemental Support Media*, Proceedings of the MMEDIA 2009 Conference, Colmar, France.

[6] De Bra, P., Smits, D., Stash, N., 2006. Creating and Delivering Adaptive Courses with AHA! *Proceedings of the first European Conference on Technology Enhanced Learning*, EC-TEL 2006, Springer LNCS 4227, 21-33, Available from: http://aha.win.tue.nl/ publications.html. [Accessed 13 Feb 2012].

[7] IEEE. 2001. *IEEE Learning Technology Standards Committee* (LTSC) IEEE P1484.12 Learning Object Metadata Working Group; WG12 Home page.

[8] Holohan, E., Melia, M., McMullen, D., Pahl, C., 2005. Adaptive E-Learning Content Generation Based on Sematic Web Technology. *International Workshop on Applications of Sematic Web Technologies for E-Learning*, 12th International Conference on Artificial Intelligence in Education. 2005 Amsterdam. Netherlands

[9] McGreal, R. (Ed.), 2004. *Online Education Using Learning Objects*. London:Routledge, 59-70.

[10] Shute, V., Towle, B., 2003. Adaptive E-Learning. *Educational Psychologist*. 38(2), 105–114