Personalised E-Learning Through Learning Style Aware Adaptive Systems

By

John Canavan

A dissertation submitted to the University of Dublin,
in partial fulfillment of the requirements for the degree of
Master of Science in Computer Science

September 2004

Declaration

I declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Signed: ___________________

John Canavan

10th September 2004

Permission to lend and/or copy

I agree that Trinity College Library may lend or copy this dissertation upon request.

Signed: ____________________

John Canavan

10th September 2004

Acknowledgements

I would like to thank my project supervisor, Vincent Wade, for his direction and advice during the project. Thanks are due also to Ian O'Keeffe and Declan Dagger, from the KDEG research group, for all their support and assistance throughout the year.

Abstract

There are many factors that can influence the extent of learning. These would include factors such as the student's learning style and motivation for learning. In a learning environment, each individual student will have different requirements and characteristics. An important role of the educator is to recognise that their pedagogy and educational material must cater for the individual learner's characteristics and requirements. This is true for both the traditional classroom approach and for e-learning. There is a need to move away from the "one size fits all" paradigm and to offer learner's a personalised learning experience. This research investigates how personalised courses can be delivered to the learner in an adaptive environment. More specifically, it examines how learning style information can be integrated into an Adaptive Hypermedia System to offer increased personalisation.

Learning style can be defined as the attitudes and behaviours that determine an individual's preferred way of learning. Many different learning style models exist today. The learning style model chosen for the integration into an Adaptive Hypermedia System during this research is the Honey and Mumford Learning Style Model, an information processing model type. This report presents a review of learning styles along with a state of the art pertaining to three adaptive systems that offer some degree of pedagogical support for learning style. An extension of the Adaptive Personalised e-Learning Service (APeLS) framework is detailed to cater for the Honey and Mumford Learning Style Model. A prototype adaptive course was developed in conjunction with this report. A group of users evaluated this adaptive course offering some important feedback on the degree of personalisation perceived.

Table of Contents

Table of Figures

1. Introduction

1.1 Background

Individual students on any course will have different requirements and characteristics. A challenging part of an educator's role is to adopt their teaching or pedagogical style to suit the variety of students in a classroom environment. A designer of an e-learning course also has to make such provisions when developing a course. Today it is widely accepted that during the design and development of educational material attention must be focused on the learner's characteristics and requirements (Del Corso et al, 2002). There exists a need, therefore, to move away from the "one size fits all" paradigm and to develop methods whereby personalised courses are presented to learners which cater for these varying requirements and characteristics. If the preferred learning style of a learner is not taken into consideration during the design of a course this could cause a student's discomfort level to be great enough to hinder or prevent learning (Bruen et al, 2002). This project proposes to examine how Adaptive Hypermedia systems can support personalisation based on underlying learning style theory.

In order to capture a learner's preferred learning style a particular learning style model must be chosen as a basis for the course. Many different types of learning style models exist such as the information processing Honey & Mumford Learning Style Model (Sarrikoski, 2000). This learning style model and others are introduced in chapter two of this report. An initial problem to address when developing an e-learning experience incorporating learning style is which learning style model to choose. The first step in being able to provide a personalised approach to learning is to identify the individual characteristics of learners (De Bra et al, 2003a). Therefore, if a learning style model has been chosen to cater for the provision of individualisation of a course then information pertaining to the learner's preferred learning style must be captured by the system in order to facilitate personalisation.

Once the learning style model has been selected and a decision made on how best to capture a learner's particular learning style, the course designer must decide how to structure the course according to the particular learning style model. This is an important and difficult process and two key skills are required to successfully implement it effectively. The first skill requires that the course designer has the ability to devise a course based on sound pedagogical principles. The second key skill requires that the course designer has the ability to use current technology to deliver such a course. This project is concerned therefore with not just the technical aspect of delivering an e-learning course but also examines important pedagogical principles that should be considered when designing such a personalised course.

1.2 Project Goal and Objectives

The primary goal of this project is to investigate the integration of learning styles into an adaptive e-learning environment to provide a personalised learning experience. More specifically, the aim is to extend the framework of the APeLS e-learning service, introduced in chapter three of this report, to incorporate adaptation based on learning style knowledge pertaining to the user. To achieve this goal the area of learning and learning styles theory along with Adaptive Hypermedia was researched and documented. An adaptive e-learning course incorporating learning style consideration was developed. This course was based on the current adaptive SQL course implemented for undergraduate students at Trinity College Dublin. An evaluation of this course developed was also carried out through the analysis of user feedback.

The broad objectives for this project can be summarised as follows:

1. To review the area of learning styles and their potential integration into an Adaptive Hypermedia System.
2. To develop a framework for an Adaptive Hypermedia System to cater for personalised e-learning based on a suitable learning style model.
3. To evaluate the prototype system developed by analysing user feedback.

1.3 Roadmap

To develop a successful e-learning course it is important to build the course on sound pedagogical principles. A starting point in this project was to examine the area of learning and the current learning paradigms. Current popular learning styles and learning style models were investigated. Identifying the popular models was the first step, however a more important process was the selection of an appropriate learning style model as a basis for the adaptive e-learning course developed in conjunction with this report. In order to address the questions pertaining to learning and learning style models a review of current literature was carried out and documented in chapter two of this report.

As this project was concerned with developing an adaptive course a state of the art review was carried out on three particular adaptive systems. Particular interest was paid to answering questions such as: How did these systems model the learner and the domain? How did they present particular courses according to different pedagogy? How was learning style theory integrated? What technologies were used for their development? The results of this review are presented in chapter three of this report.

Following the state of the art review on Adaptive Hypermedia Systems the next logical question was: How can this knowledge of learning styles and Adaptive Hypermedia be utilised to design and develop a personalised adaptive course? It was decided to implement an adaptive e-learning course using the APeLS system. APeLS is discussed in chapter three of this report. The design and implementation of this framework is presented in chapters four and five respectively.

Following the development of an adaptive e-learning course incorporating learning style knowledge, the evaluation covered two areas. The first approach was to obtain user evaluation of the prototype SQL course developed for the implemented framework. Answers to questions pertaining to users opinions on the effectiveness of the personalised course were obtained. The second approach was the comparing and contrasting of the framework developed as part of this research against other adaptive systems researched. This evaluation is detailed in chapter six with the conclusions presented in chapter seven.

2. Background

2.1 Learning and Learning Paradigms

This chapter will examine the area of learning and introduce the popular learning paradigms. Learning has been defined as having taken place "when people can demonstrate that they know something that they did not know before (insights and realisations as well as facts) and/or when they can do something they could not do before" (Mumford, 1995). A person can learn in two substantially different ways (Honey et al, 1992). Sometimes a person will learn as a result of formal structured activities, for example learning by attending lectures or reading books. The second means is learning through experiences, often in an unconscious, ill defined way. The first type of learning, i.e. learning dedicated to the acquisition of knowledge, is both more familiar and more straightforward than experimental learning (Honey et al, 1992).

The two traditional learning paradigms in educational science were the behavioural and cognitive views of learning. The current paradigm today is the constructivist view (Sarrikoski et al, 2000). The behavioural view stated that the outcome of learning was change in observable behaviour and was mainly concerned with relations between attributes pertaining to the learner (such as intelligence, social background and abilities) measured quantitatively with outcomes such as course grades. The cognitive view of learning became popular in the 1960's and began to replace the behavioural view as researchers became more interested in the learning process itself examining areas such as reasoning, comprehension and problem solving. The basic idea of the constructive view examines "the learner's own active contribution to his learning process in a social context where the learner constructs his knowledge by combining new information and experiences with his existing knowledge structures" (Sarrikoski et al, 2000).

From the above definition of the constructive view of learning it is apparent that information, therefore, cannot be simply transferred from one person to another due to the fact that information will be interpreted differently by individual learners (Sarrikoski et al, 2000). David Kolb is one of the best known representatives of the constructive view of learning and amongst the contributions he made to this view is verifying that a learner will have his own learning style.

2.2 Learning Style

There are a large number of factors that can influence the extent of learning and some of these can be viewed in Figure 2.1. Learning style may be defined as "the attitudes and behaviours which determine an individual's preferred way of learning" (Honey et al, 1992). A student, for example who prefers practical experience, learning a new programming language may prefer to begin writing code immediately whereas another may favour reading up and studying the new language prior to writing any code. Most learners are unaware of their own learning style preferences but are vaguely aware of what they feel comfortable with, and learn more from, certain activities than others (Honey et al, 1992). While there is plenty of research on learning style there does not seem to be any agreement or acceptance of any one theory (Bruen et al, 2002). All learning style models presume that it is possible to measure a learner's learning through the use of psychometric instruments.

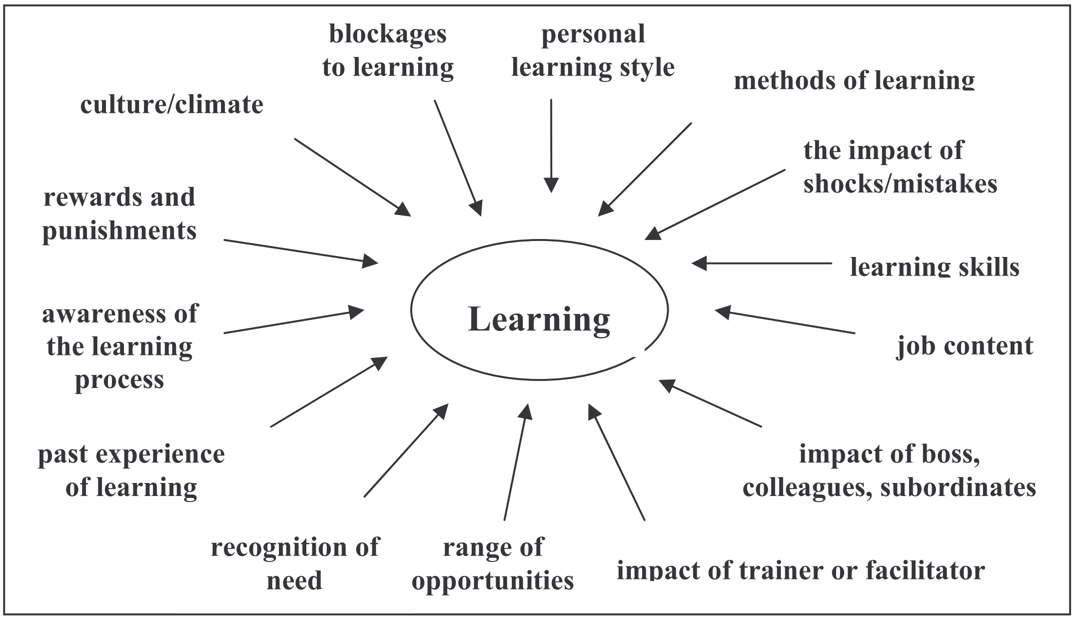

Figure 2.1 Factors that influence learning

2.3 Honey and Mumford Learning Style Model

Learning is a life-long process (Honey et al, 1992). This continuous process is like the coils of a spring (or a never-ending spiral) with each coil of the spring (or loop in the spiral) having four distinct stages. These four stages in the learning cycle presented by Honey and Mumford (1992), illustrated in Figure 2.2, are based on Kolb's work. Kolb uses different words to describe the stages and the four learning styles, but the "similarities between his model and ours [Honey & Mumford's] are greater than the differences" (Honey et al, 1992). A learner can start anywhere on the cycle and not necessarily at stage 1, for example, a person could start at stage 2 by acquiring some information and think about it before reaching some conclusions at stage 3 and then decide how to apply this knowledge at stage 4.

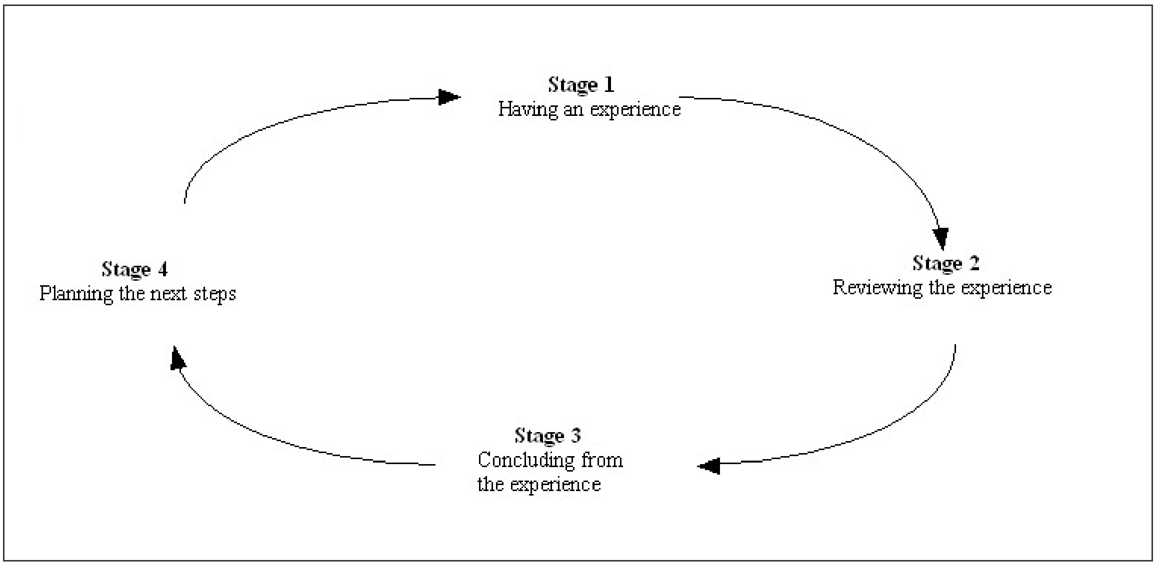

Figure 2.2 The Honey and Mumford learning stages

These four stages, shown in Figure 2.2, are introduced briefly below (Honey, 1994):

- Having an experience: The two types of experiences one can have are reactive (letting the experience come to you) and proactive (deliberately seeking the experience). Opportunities to learn from experience are greatly increased if the normal things that happen to us are supplemented by extra experiences we create.

- Reviewing the experience: If one learns from an experience it is vital to review what has happened.
- Concluding from the experience: This involves scanning the raw material from the review for conclusions, answers or lessons learned.
- Planning the next step: Planning involves translating some of the conclusions into a form where they can be put into action.

Each of the states are mutually supportive and none is a fully effective learning procedure on its own as each stage plays an important part in the total process (Honey, 1994). Most people develop preferences which give them a liking for certain stages over others however this preference may lead to a distortion of the learning process as greater emphasis is put on certain stages to the detriment of others (Honey et al, 1992). If a learner followed the learning cycle by giving sufficient attention to each stage of the cycle then he would increase his chances of effective learning (Mumford, 1995).

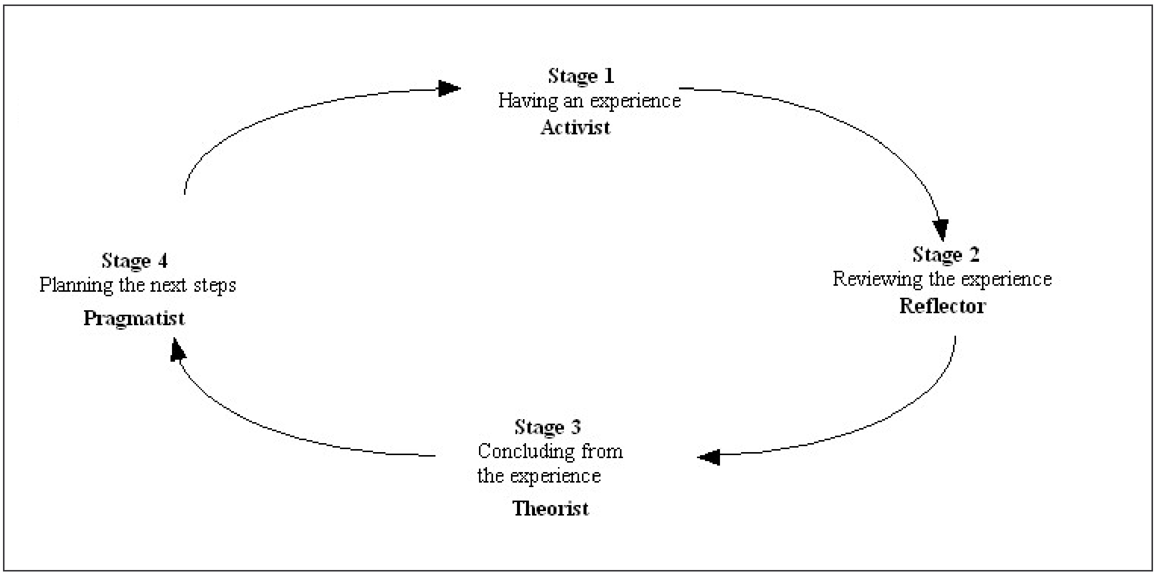

Figure 2.3 The Honey and Mumford learning styles

For each of the stages identified above, a particular learning style is associated with the stage – see Figure 2.3. The four learning styles identified are activist, reflector, theorist and pragmatist. Activities that are suited and not suited to these types of learners are listed in Appendix 4. These learning styles are briefly described below (Honey et al, 1992):

Activists: Learn from relatively short here-and-now tasks. They are open-minded, not sceptical, and this tends to make them more enthusiastic about anything new. They tend to act first and consider the consequences afterwards. Their days are filled with activity and they tackle problems by brainstorming. They like working with others but tend to hog the limelight. They learn best when: involved in new experiences, problems and opportunities; working with others in business games, team tasks, role-playing; being thrown in the deep end with a difficult task; chairing meetings, leading discussions.

Reflectors: Learn from activities when they are able to stand back, listen, and observe. They like collecting information and being given the opportunity to think about it. Their philosophy is to be cautious. They tend to adopt a low profile and have a slightly distant, tolerant, unruffled air about them. Reflectors learn best when: observing individuals or groups at work; they have the opportunity to review what has happened and think about what they have learned; producing analyses and reports doing tasks without tight deadlines.

Theorists: Adapt and integrate observations into complex but logically sound theories. They are interested in and absorb ideas even though they may be distant from current reality. They like to analyse and synthesise. They tend to be detached, analytical and their approach to problems is consistently logical. They learn best when: they are put in complex situations where they have to use their skills and knowledge; they are in structured situations with clear purpose; they are offered interesting ideas or concepts even though they are not immediately relevant; they have the chance to question and probe ideas behind things.

Pragmatists: Are keen on new ideas, theories and techniques to see if they work in practice. They like to get on with things and act quickly and confidently on ideas that attract them. They are essentially practical, down to earth people who like making practical decisions and solving problems. They learn best when: there is an obvious link between the topic and job; they have the chance to try out techniques with feedback e.g. role-playing; they are shown techniques with obvious advantages e.g. saving time; they are shown a model they can copy e.g. a film or a respected boss.

2.4 Honey and Mumford Learning Style Questionnaire

The Honey and Mumford Learning Style Questionnaire was developed to capture the learner's preferred learning style (Honey et al, 1992). This questionnaire has 80 questions, 20 for each style, and it typically takes between ten and fifteen minutes to complete. Unlike Kolb's learning style questionnaire, the Honey and Mumford questionnaire refrains from asking direct questions about how people learn, instead it probes general behavioural tendencies of the learner such as "I actively seek out new experiences" or "In discussions I like to get straight to the point". The questionnaire probes behavioural tendencies and not learning tendencies as it is believed that people do not consciously consider how they learn thus it is not helpful to ask questions that directly enquiring into this (Honey et al, 1992).

3. State of the Art

3.1 Introduction to Adaptive Hypermedia

Adaptive hypermedia is an emergent research domain and has been described as being on the crossroad of hypertext (hypermedia) and user modelling (Brusilovsky, 2001). A major disadvantage of traditional hypermedia applications was that they presented the same content and the same set of links to all users thus failing to satisfy the heterogeneous needs of many users. This is particularly true when dealing with e-learning where not only will learners have different levels of prior knowledge and different goals but they will also have various pedagogical preferences such as a preferred learning style. Adaptive hypermedia offers an alternative approach to the traditional "one-size-fits-all" approach.

Intelligent Tutoring Systems (ITS) adapt the content or presentation with the modification based on the interactions between the user and the system. The majority of the traditional ITSs tended to merge these models into a single system which, although capable of some levels of adaptivity, are very difficult to repurpose. Compared to the ITS Adaptive Hypermedia Systems (AHS) tend to provide a clearer separation of the learner and content models, however the narrative or pedagogical model is usually embedded in the content or the engine itself (Bruen et al, 2002).

AHS build a model of the goals, preferences and knowledge of each learner and uses this model throughout the interaction with the user in order to adapt to the needs of the learner (Brusilovsky, 2001). Popular adaptive techniques applied are adaptive content selection, adaptive navigation and adaptive presentation (Brusilovsky et al, 2002). The first technique allows the most relevant pieces of content to be selected based on learning goals for example. As the user navigates through the system the links may be manipulated to guide the user adaptively to the most relevant information items. The third technique allows the system to adapt the content retrieved and has some deep roots in the research on adaptive explanation and adaptive presentation in intelligent systems (Brusilovsky et al, 2002).

3.2 Examples of Adaptive Hypermedia Systems

Three following adaptive systems will be examined in the next section of this chapter:

- APeLS (Adaptive Personalised e-Learning Service) - Developed by the Knowledge and Data Engineering Group in the Department of Computer Science at Trinity College Dublin. The system is currently being used as part of an undergraduate course and has received extremely positive feedback during past usage (Clarke, 2003).
- AHA! (Adaptive Hypermedia Architecture) - Developed by researchers on the Department of Computer Science at Eindhoven University of Technology, Holland. It has been studied and experimented upon by several research groups in different countries (Stash et al, 2003).
- 3DE (Design, Development and Delivery Electronic Environment Educational Multimedia) - An environment for compiling custom courses for learners. It was developed as part of a research project within the European Union IST 5th Framework Programme. It would not be considered a true Adaptive Hypermedia System.

3.3 Comparisons of Three Adaptive Hypermedia Systems

The following six sub-sections will compare and contrast the three above systems under the following heading: General Architecture, Learner Modelling, Content Modelling, Methods of Adaptation, Pedagogic Support and Support for Learning Style. A final sub-section will provide a summary and analysis of this discussion

3.3.1 General Architecture

Most Adaptive Hypermedia Systems will generally have a separate learner and content model with the narrative model embedded in the content or adaptive engine itself. The approach used in APeLS is described as a multi-model approach with the adaptive engine being fed by these three models (Dagger et al, 2003). The adaptive engine will interpret and reconcile these models to produce a personalised course for the learner. The three main models in APeLS are the learner (see Learner Modelling sub-section), content (see Content Modelling sub-section) and narrative (see Pedagogic Support sub-section) models. The learner model contains modelled assumptions that represent the characteristics of the student that are important to the system (Conlon et al, 2002). For example, it will contain the learner's goals and possibly their learning style preference. The content model represents the learning resources within the system while the narrative (pedagogical) model represents the ways in which the content can be sequenced for the learner (Conlon et al, 2003). The interaction between these three models and the learner's interface with APeLS is illustrated in Figure 3.1.

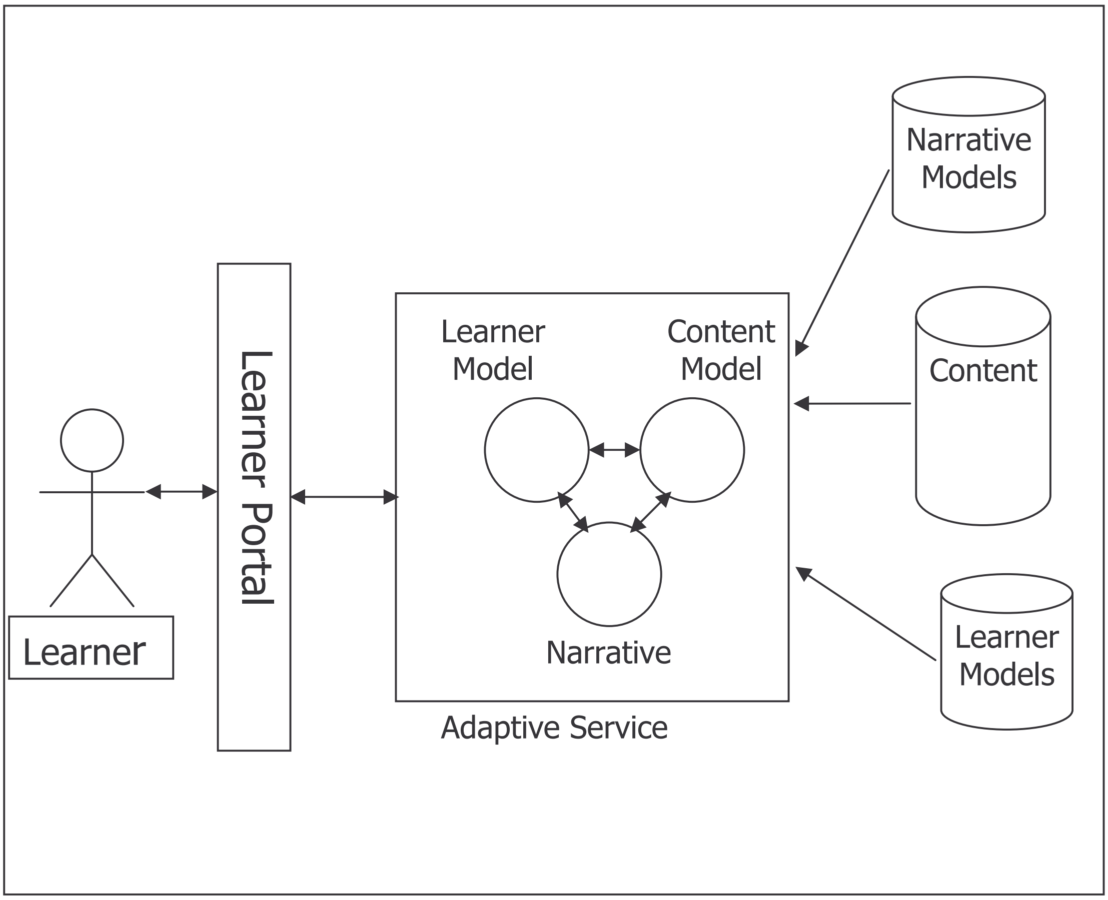

Figure 3.1 Model interaction in APeLS

AHA! contains one model less than APeLS, the two models being the user (see Learner Modelling sub-section) and domain/adaptation (see Content Modelling sub-section) models. The user model here is analogous to the learner model of APeLS in the sense that it maintains relevant information about the learner using the system. The domain/adaptation model contains the concepts taught, the relationship between these concepts along with the embedded adaptive logic. The domain/adaptation model is similar to the narrative model of APeLS, however its big disadvantage is the intertwining of the domain, content and adaptive techniques.

Similar to APeLS and AHA, 3DE maintains a model of the learner, or more precisely a profile of the learner. This profile maintains data pertaining to the learner's goals, competence and learning style. Unlike AHA! this profile is not updated as the learner moves through the course – it is generally only updated at the start of the session when the student selects the relevant learning goals and/or completes the learning style questionnaire (see Learner Modelling sub-section). The content elements are organised in a hierarchy of atom, content unit, composite unit and course (see Content Modelling sub-section). The content units of the 3DE system are analogous to the Learning Objects of APeLS. A tool at the heart of the 3DE project, called the custom course compiler, builds a customised course from the micromodule library taking into account the learners goals and learning style along with the prerequisites required (Sarrikoski et al., 2000).

All three systems make use of Open Source technologies such as Java, XML and MySQL. Some of these technologies have been used to develop tools to assist the author in the authoring process. The adaptive engine that APeLS runs on is written in Java. APeLS also makes use of other Java related technologies such as Java Server Pages (JSP) and JESS (Java Expert System Shell). JSPs are used to create the dynamic HTML while the rules engine employed is based on JESS. AHA! is delivered as Open Source software, written entirely in Java, using Servlets (De Bra et al, 2003). 3DE is also implemented using Java technologies such as Servlets.

XML is another key technology employed by the three systems. APeLS, for example, stores the personalised course model as an XML file before sending it through an XSL transformer.

It stores components of the content and learner models in the form of XML documents in the open source Xindice XML database. XML files or a MySQL database are employed for storing the domain/adaptation and user models of AHA! 3DE also makes use of XML, for example the metadata tool generates XML IMS compliant files which can be used to load the metadata on an IMS compliant database system and also to simplify the content exchange between different e-learning systems (Del Corso et al, 2003).

To aid the authoring process AHA! provides two Java applet tools, called the Graph Editor and the more low level Concept Editor, when preparing the concepts and associated rules of the domain/adaptation model. Along with providing essential tools such as the learning style questionnaire and the custom course compiler 3DE provides a metadata tool for the authors of content. The main purpose of this tool is to speed up and ease metadata insertion into each learning unit (Del Corso et al, 2003). For APeLS it is envisaged that course authors will design courses through a graphical interface with design support facilities (Dagger et al, 2003a). Such tools are currently being developed.

3.3.2 Learner Modelling

As the system consults the learner model to adapt the performance of the system to the learner’s characteristics these characteristics must be captured as accurately as possible. It must be worth remembering that bad guidance is worse than no guidance (De Bra, 1999). There are a number of sources of information that can help construct a learner model (Conlon, 2000). The system will acquire information about the learner and infer learning characteristics based on this data. The learner can provide this data or feedback to the system directly by submitting questionnaire answers for example. Indirect feedback can be given to the system from such sources as the results of exercises or problems solving tasks, alternatively the system may keep track of the learner’s browsing and update the model based on this. AHA!, for, example uses this technique.

The following three are some of the many techniques available for modelling the student – stereotype model, overlay model and combination model (Brusilovsky, 2001). The stereotype model classifies a new student into a particular category e.g. novice, intermediate or expert. This method is quick but not necessarily very accurate. With the overlay model approach a model of the student's knowledge is built up on a concept-by-concept basis and is updated as the user moves through the system. The overlay model is the most common type of user (learner) model in adaptive hypermedia applications (De Bra, 2002). The combination model is simply a combination of the stereotype and overlay models introduced above. Here the student is categorised by stereotype and the model is gradually modified as the student moves through the course.

Various techniques are been applied in APeLS for modelling the learner. APeLS, similar to 3DE, primarily relies on direct techniques i.e. querying the learner directly rather than indirect techniques. APeLS can store information in relation to prior knowledge and learning objectives along with pedagogical considerations, such as preferred learning style, pertaining to the learner. Examples of two learner controls developed for an APeLS course are the Rebuild Scope and Rebuild Style features (Clarke, 2003). The Rebuild Scope feature captures the learner's motivation factors, course aims and previous experience and it takes the form of a questionnaire. The learner can return to the Rebuild Scope questionnaire at any stage to answer the questions again in order to update the learner model and thus update the course content presented. The second feature, Rebuild Style, was added to capture the learner's learning style and it also took the form of a questionnaire. By completing the questionnaire the learner is classified as having a particular learning style as per Kolb's definitions of learning style. See Support of Learning Style sub-section for further information on this. These two techniques are just two of the techniques designed for APeLS in order to capture information for the learner model, other techniques are similar i.e. questionnaire presented to the learner in order to determine such information as goals, previous experience and learning style.

While APeLS primarily relies on the direct feedback technique to capture information for updating the learner model the approach taken by AHA! involves indirect techniques utilising the overlay model as discussed above. AHA! courses mainly consist of concepts which are used to represent topics of the application domain (De Bra et al, 2003). Any number of attributes can be associated with a concept, some of which have a meaning to the system for example "access" and some which have meaning for the author (and learner) for example "knowledge". All of these attributes that appear in the domain/adaptation model also appear in the user (learner) model thus applying the overlay model approach.

For each user of AHA! the user model consists of a value for each concept. The most common interpretation of this value is that the value means the knowledge level of the user with respect to the concept (De Bra et al, 2002). The adaptation rules within an AHA! course define how the user model is updated (De Bra et al, 2003). When the learner accesses a page generate rules in the domain/adaptation model (see Content Modelling sub-section below) causes the user model to be updated. The page is then adapted on the basis of the updated information in the user model. The author can define propagation rules which cause an increase or decrease in the value of other concepts which are associated with the concept accessed.

Within AHA! there is also a special concept called "personal" which is used to store learner related information such as name, email and password. This attribute can also be used, for example, to perform adaptation to ensure that the appropriate assignments are presented to students from different universities. The author can also define attributes here which reflect the learner's learning style, for example the attribute "KolbLM" may be included and it can have values such as "Reflector" or "Activist" (Stash et al, 2004).

For the 3DE application the direct feedback mechanism is employed to determine information pertaining to the learner. The learner must complete the Learning Style Questionnaire (LSQ) where thirty-six questions are presented in order to determine the individual's preferred learning style. The result of the LSQ is a set of four numbers, each representing one of the four basic styles from Honey and Mumford's Learning Style theory (i.e. an activist, reflector, theorist and pragmatist value), and this information is stored in the 3DE database (See Support of Learning Style sub-section). Also determined from the user's direct interaction with the system are the learning goals for the course. These goals are selected through a hierarchical selection of contents which are presented to registered users. A 3DE application can also maintain a personal portfolio containing a list of the active courses for the individual learner and from the portfolio page the user can continue a course from the last visited point.

3.3.3 Content Modelling

Even though these three adaptive applications handle content quite differently there are some similarities with all systems implementing in some way the notion of a hierarchy. The content in AHA! consists of a hierarchy of related concepts while 3DE implements a hierarchy of atom, content unit, composite unit and course. APeLS would not be considered truly hierarchical but it implements the concept of a candidate content group (CCG), as shown in Figure 3.2, that references smaller learning resources with each candidate piece of content differing on some axes e.g. learning style or prerequisites.

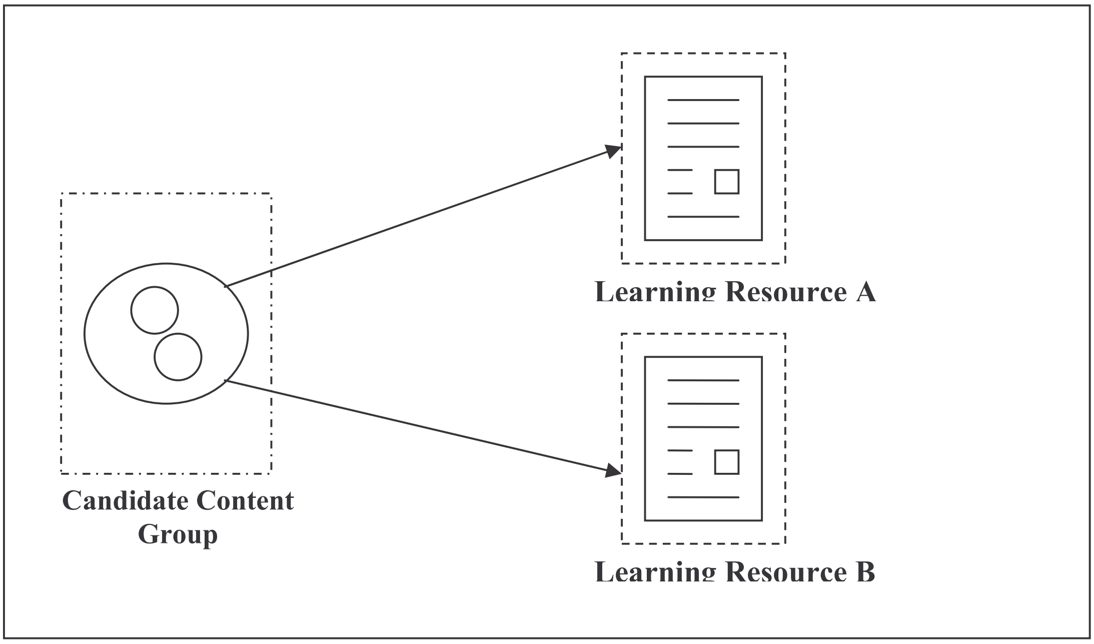

Figure 3.2 The CCG abstractly referencing the learning resource by identifier

The content model in APeLS represents the learning content which may be selected to be presented to the individual learner (Dagger et al, 2003). The abstraction used by APeLS is made possible through CCGs. The function of a CCG is to group together learning resources that are equivalent on some axis, for example concept taught. Each CCG has metadata associated with it to describe the role of the group and to identify the learning resources associated with it. There are many methodologies that can be applied to the grouping of learning resources such as learning style similarity or prerequisite knowledge similarity (Dagger et al, 2003a). The narrative therefore refers to the CCG rather than to the learning resource directly thus achieving a level of abstraction.

The learning resources themselves are known as learning objects (LOs). This separation of content from narrative facilitates the reuse of the LO. The potential reuse of the LO is related to its granularity – the smaller the granularity of the LO the greater the potential reuse of the LO (Conlon et al, 2002a). Each LO, which typically represents between two and fifteen minutes learning, is made up of two elements: learning content (e.g. piece of text or picture) and metadata to describe it. The descriptive metadata is important as it provides a means of searching for LOs in the content repository and also provides information to the adaptive engine when content needs to be selected for presentation to the learner.

AHA! does not provide the level of abstraction offered by APeLS and it does not have a content model in the same sense as APeLS. As discussed previously AHA! systems consist of concepts which are used to represent topics of the application domain. Provided with AHA! are two authoring tools, a Concept Editor and a Graph Editor, which allow the author to define the domain model for the application along with the adaptation model associated with it (De Bra et al, 2003a). Each concept defined has a set of generate rules and requirement rules. The generate rules indicate how page accesses generate user model updates while requirement rules define the set of actions that must be performed when the user accesses the (page/resource associated with the) concept. Once the domain model is created and all the concepts and relations have been defined, the author has to associate resources to concepts (De Bra et al, 2003a). In the simplest case each resource is a concept and vice versa. The resource or piece of content is typically an XHTML file and is referenced from the concept XML file through the `<resource>` tag.

The XHTML (or content/resource) pages created by the author also contain AHA! adaptation tags such as the `<if>` and `<object>` along with the conditional `<a>` (see Methods of Adaptation sub-section). Because these tags are "hard-coded" into the resource this does not encourage reuse of the resource. This is in sharp contrast with APeLS where the learning objects are of fine granularity and are free from such embedded adaptive logic thus facilitating reusability. The main goal of the multi-model approach used in APeLS is to separate the learning content from this adaptive linking logic or narrative (Conlon et al, 2002a).

The content in the 3DE environment is organised in a hierarchy of atoms, content units, composites units and course (Del Corso et al, 2003).

- The atom is an elementary resource such as a piece of text or a drawing. Usually there is no benefit in presenting a single learning object to the learner as it is only valid once it is presented in a context.
- A content unit (or micromodule) would be somewhat analogous to the Learning Object of APeLS. It is an elementary, logically indivisible composed of atoms, links and relations. The learner will spend approximately five to fifteen minutes on each content unit.
- A composite unit is a group of content units within a navigation structure and it would be analogous to a chapter of a book.
- A course is a set of composite units linked by their perquisites and organised into an orientation graph.

The method used by 3DE for modelling content is closer to the method used by APeLS than AHA! The content unit would be analogous to the Learning Object of APeLS. In the narrative of APeLS the concepts are sequenced in a particular way which would infer prerequisites, whereas the prerequisite information in 3DE is stored in the metadata of the content units, composites units. APeLS and 3DE are also similar in the sense that the course is formed (or compiled) whenever the user requires this action to occur. With APeLS a personalised course model is formed whereas 3DE utilises the custom course compiler to compile the customised course. Both of these actions take place when the learner requests it to occur, as discussed in the Learner Modelling sub-section, and can be invoked at any stage by the learner. AHA! is more focused on offering and prevention of access to certain content throughout the session rather than compiling an actual customised course based on the learner model.

3.3.4 Methods of Adaptation

Two distinct methods of adaptation implemented in Adaptive Hypermedia Systems are adaptive presentation and adaptive navigation support (Brusilovsky, 2001). Common techniques to achieve adaptive presentation would include page variants where different versions of the same page exist and conditional inclusion of fragments whereby a page is constructed from small fragments which are conditionally included or omitted (De Bra, 1999). Techniques implemented to achieve adaptive navigation support include direct guidance where the adaptive system determines the next most appropriate page, sorting of links from the most relevant to least relevant, link annotation, link hiding, link disabling and link removal (De Bra, 1999). These the two methods of adaptation discussed below.

APeLS provides adaptive presentation through candidacy. Candidacy can be implemented by the abstract grouping of learning resources with similar goals, objectives or learning style (Dagger et al, 2003a). Candidate selectors can be defined as the rule sets that choose a candidate from a candidate content group (Dagger et al, 2003a). Candidate selectors can be implemented in two modes – all (set of candidates that meet the minimum requirements) and best (single candidate that best fits the requirements). When there are several candidates selectors each one called refines the list of candidates with the last one called selecting the best candidate from the final list.

With AHA! the technique implemented for adaptive presentation is the conditional inclusion of fragments. It uses two techniques to carry this out, one with embedded fragments the other with objects (De Bra et al., 2003). Embedded fragments appear in the page and are included if an associated suitability expression based on the user (learner) model evaluates to be true. This mechanism has drawbacks however as it mixes the domain and adaptation models (De Bra et al., 2003). For greater flexibility the `<object>` tag mechanism should be used (De Bra et al, 2003a). When tags with objects of type “aha/text” are read, the id of the object, which is considered a concept, is examined and the access event associated with the concept is triggered to update the user model and determine if the fragment is to be shown (De Bra et al., 2003).

An interesting inclusion in version 3.0 of AHA! is the idea of stable presentations. Generally in adaptive applications the current state of the user model determines how the page is to be displayed each time the learner visits the page (De Bra et al., 2003). Due to updates of the user/learner model the page may look different each time the user visits. This may not be a desirable effect. With the current version of AHA! for each page the author can indicate the desired stability of the presentation, for example, a page may be configured as always adapted or always stable. Once the course is selected at runtime at the start of a session with APeLS it tends to be quite stable and therefore this requirement is not necessary. This requirement would not be necessary for 3DE either as once the course is compiled at the start of a session it will not change.

3DE also facilitates adaptive presentation with a selection technique closer to (but much simpler than) APeLS than AHA! Following registration with a 3DE course the learner will select learning goals based on a hierarchical selection of contents. The custom course compiler (CCC) will take these goals into account when building the customised course from the content repository. To create a course the CCC selects, for each educational objective defined, the micromodule that best matches the student's learning style and starting with the micromodule that does not have any prerequisite it follows the direction of the arcs based on the prerequisite metadata. The sequence of visited micromodules is the compiled custom course.

APeLS facilitates adaptive navigation through the sequencing of candidates in the narrative. The narrative's primary goal is to produce courses that are structured coherently and that satisfy the learner's goals in a way that engages the learner (Dagger et al, 2003a). It is possible to have several candidate narratives for a single course, for example a course may be structured so that the candidates have the same ethos, learning goals and require the same prior knowledge but differ in pedagogical approach (Conlon et al, 2002a). It would be possible, for example, to develop two narratives for a course, one which is case-based and another didactic. This methodology would support adaptive navigation by sequencing the material differently for each of the candidate narratives. At runtime the candidate narrative most suited to the learner would be selected.

A more traditional approach is taken by AHA! for its implementation of the adaptive navigation. When a page contains a conditional link (e.g. <a href=”tcd.xhtml” class=”conditional”>) the link is presented to the user in one of three colours depending on the current state of the user model. If the conditional expression evaluates to be true the link is shown in blue (unvisited) or purple (visited), however if the expression is false the link is shown in the same colour as the text without being underlined. The result is the hiding of unsuitable or undesired links (De Bra et al., 2003). The colour scheme associated with these links is configurable by the learner.

3DE does not really support adaptive navigation, however a micromodule has a list of prerequisites associated with it so there is an idea of navigation here. When the user selects the learning goals for the course the custom course compiler assembles relevant themes matching the learner’s style. It then assembles the path and presents a list of detailed prerequisites to the learner.

3.3.5 Pedagogic Support

Pedagogy is defined as the art and science of teaching (Clarke, 2003). Underlying any pedagogical adaptation there should be sound pedagogical principles - without this an e-learning system may suffer from polar problems of lost in hyperspace or learners feeling dictated to and constrained (Conlon et al, 2003b). There are many different pedagogical methods which can be applied to e-learning systems. Such methods, for example, may include didactic, case-based or problem-solving approaches. Different educators prefer different pedagogical approaches when teaching and some of these approaches are more suited to some learners than others. This section examines how APeLS, AHA! and 3DE provide support for different pedagogical approaches.

Through the sequencing of the narrative, courses can be generated that differ in ethos, learning goals, pedagogical approach or learner prior experience (Conlon et al, 2003a). For example, an author may create a course which presents the learner with a problem to solve at the beginning and by working through the problem various concepts will be presented to the user. Alternatively the same course could have the concepts sequenced an different way presenting the course to the learner applying a didactic approach.

The primary goal of the narrative is to produce personalised courses that are structured coherently and which cover the learner's goals. The narrative captures the logic behind the selection and delivery of a learning resource within the scope of the adaptive course (Dagger et al, 2003a). It allows the course author to separate the intelligence that performs the adaptation from the content and this increases the potential for reuse of the learning resources involved (Dagger et al, 2003a). Because the narrative refers to the abstract candidate content groups and not to individual learning objects the domain expert can create the narrative without being constrained by pedagogical issues. The most appropriate candidate from the group will be selected at runtime by the adaptive engine based on the learner model.

If the course author is concerned with pedagogical approach and wishes to write a narrative that encapsulates different pedagogical approaches then this is possible through writing several candidate narratives which achieve the same learning objectives. For example, there may be case where the author would prefer learners who have some previous knowledge of content to start off with a case-base example for this particular course whereas learners who do not have any previous knowledge to follow a didactic approach. In this scenario the author could prepare two candidate narratives which are structured and sequenced differently, one narrative to facilitate the case-based approach while the other would follow a didactic approach. When compiling a course APeLS calls the candidate selector to choose the most appropriate candidate narrative from the candidate narrative group for the course (Conlon et al, 2002a). This selection is based on the learner's metadata along with the narrative metadata repository. For example, if the learner's "preferredcoursestructure" is abstract the candidate selector will select the narrative that best matched this preference (Conlon et al, 2003a). The selection of the individual pieces of content from the candidate content groups must be carried out next, this is discussed briefly in the Support of Learning Style section.

The main pedagogical principle taken into account in both 3DE and AHA is learning style. The most recent version of AHA! has introduced functionality for the support of learning styles which will be discussed in the next sub-section. The power and flexibility that the narrative provides for creating a course according to various pedagogical approaches is not available with either AHA! or 3DE. With AHA! it is possible to design a separate course for a particular approach but this would be costly and take much time, therefore it is not very practical.

Amongst other things, the 3DE environment contains a pedagogical framework with the basic pedagogical principle taken into account is the learning style (Del Corso et al, 2003). The aim is to change teacher-learner relations from the 1-to-N paradigm to an N-to-1 model: each learner can select among several available teachers the one who best suits his learning style (Del Corso et al, 2002) – this is discussed further in the next section. In terms of pedagogy, the custom course compiler concentrates only on learning style along with the learning goals and prior knowledge of the learner.

3.3.6 Support of Learning Style

This section discusses how the various systems provide support for the learner's learning style. The ultimate aim of determining the learner's individual learning style is to facilitate further personalisation of the content. While there has been much research into learning style in the classical educational (classroom) setting there is less research into the application of learning styles in the new educational space created by the web (Stash et al, 2004).

Two key concepts of APeLS are candidacy and abstraction. Having multiple candidates that cover the same learning goal the candidate selector can select the most appropriate of these candidates based on information in the learner model. The criteria for selection can be set by the author of the course with one criterion possibly being learning style. The information in which the selection is based is characterised by the metadata. APeLS can deal with any learning style model, however the appropriate metadata would have to be included in the learner and content models. An appropriate selection mechanism would also have to be added to the narrative model.

A framework was developed for APeLS to apply the VARK (Visual, Auditory, Read/Write and Kinaesthetic) Learning Style Model when selecting content for the learner (Bruen et al, 2002). A learner, for example, with a VARK of V=2, A=1, R=1, K=3 represents a learner who is quite strong visually and enjoys the learning material to provide mechanisms for interaction. Expanding this example, if two candidates exist within a candidate content group the first having VARK values of 3:1:1:2 and the second having values of 1:0:2:1, in this case the first candidate would be more appropriate for this learner and would therefore be selected by the candidate selector.

An implementation of the integration of Kolb's Learning Style Model with APeLS has been developed (Clarke, 2003). Learners are presented with a questionnaire consisting of eighteen questions (with two possible answers to each question) in order to determine which of Kolb's learning styles are appropriate for that learner. Once the questionnaire answers are submitted the individual is classified as having a particular learning style as per Kolb's definitions of learning styles. The decision as to which learning object (candidate) to deliver to the learner is made at runtime by finding the best match based on the information in the learner model and the learning object ranking. Users of this course welcomed the fact that learning was further personalised but were disappointed with the lack of feedback from the system regarding their learning style classification (Clarke, 2003). This is one of the differences between the APeLS approach and the approach taken by the other two systems (see below).

Similar to APeLS, AHA! does not confine the author to using a particular learning style model e.g. the author may develop one course applying VARK and the next applying Kolb's Learning Style Inventory. The learning style information is stored within the relationships between the source concepts and destination concepts of the domain. For example, a destination concept called WritingApplets may have three source concepts such as AppletActivity, AppletExample and AppletExplanation, each representing a piece of content suitable for three different learning styles (De Bra, 2004). The author would define this "illustrates" concept relationship type using the Graph Author tool. It is important to note that the same strategy would have to be applied again if the author was working in a different domain i.e. no reuse is possible. This is not the case with APeLS or 3DE where information pertaining to learning style is stored in the metadata of the learning object thus facilitating reuse.

While many systems (such as APeLS and 3DE) access learners learning style through psychometric questionnaires AHA! does not take this approach. The developers of AHA! argue that there is a disadvantage to these tests as the learner is classified into a stereotypical group and assumptions made about their learning style are not updated during the learner's interactions with the system (Stash et al, 2004). This limitation is avoided in AHA! as it aims to infer the learner's learning style by their browsing behaviour.

With AHA! the learner may specify their preferred learning style by selecting the appropriate type from a drop down list. If the learner accesses the recommended concept, the system's confidence that the learner correctly defined his learning style increases. On the other hand, if the user accesses a non-desirable concept this confidence level will decrease – once it decreases below a particular threshold the system will ask the learner if he wishes to change learning style. If the user did not select a preferred learning style on the registration page the system can match the user with a learning style (similar to the mechanism described above) by observing their browsing. Unlike APeLS, the learner can find out at any stage what learning style the system has matched them with by inspecting the relevant attribute value of the user model through a separate screen. In this screen the learner may also change the selected value of the learning style attribute to another.

While any learning style model can be applied to APeLS and AHA! this is not the case with 3DE. Similar to the approaches applied to APeLS 3DE has adopted the constructivist view of learning. The learning style test implemented in 3DE is based on the Honey & Mumford Learning Style Model. A 3DE application will therefore be based entirely on this model, unlike APeLS where potentially any model can be applied.

The learning style test is much longer than the one applied in APeLS's case where the Kolb Learning Style Model was implemented. In 3DE's case there are thirty-six questions of the type "I get bored with routines" and "I like listening more than talking" (Sarrikoski et al, 2000). A four point scale ranging from 1 (I disagree) to 4 (I fully agree) is used. The learner profile is expressed by a normalised parameter for each of the styles (A, B, C, D) with each of the numbers ranging from 0 (low preference for style) to 100 (high preference). The learning styles of each micromodule are represented by the same parameters also. Each user is identified by a four-digit sequence of numbers representing the results for the styles A, B, C and D. Once the learner completes the Learning Style Questionnaire (LSQ) the results are stored in a database and the student also receives a written report of the results with some explanations. The learning style test results are used to match the 3DE courses with each student's personal learning style. However, before compiling a course, the student can choose whether to use the version of a course that best matches his learning style or whether to use another learning style. This feature is not present in APeLS and may prove to be useful as the learner may develop his weaker learning styles through this functionality (Sarrikoski et al, 2000).

Also present in the 3DE environment is a 'learning to learn' unit. The developers recognise that once a learner is aware of their learning style that they can use this knowledge to improve their learning skills. This unit can be accessed from any course and the student can get information about learning skills and techniques which may be suited to their learning style (Del Corso et al, 2003).

3.3.7 Summary and Analysis

This sub-section provides a summary of the above discussion where comparisons were made between the three systems. A brief critique of the techniques employed by the three systems is provided along with some proposed enhancements.

The architecture of the two Adaptive Hypermedia Systems discussed above is quite different. The common feature that both possess is a learner (user) model with 3DE also having this as part of its architecture. In the AHAM (Adaptive Hypermedia Application Model) reference model, a framework to express the functionality of any adaptive hypermedia system, the learner model is an overlay of the domain model and the adaptation model is another separate model (De Bra et al, 2002a). AHA! however intertwines the domain and adaptation models which is not ideal as reuse of content is difficult due to adaptation rules being included in the XHTML content. To realise its full potential AHA! should move towards a separation of these two models to conform to the AHAM reference model.

The main advantage of the three model architecture of APeLS, where there is the clear separation of the narrative, learner and content models, is facilitating the reuse of content. With 3DE the fact that content is arranged in a structured hierarchy also encourages reuse of content created. Both AHA! and 3DE provide authoring tools for the preparation of courses. Development of an authoring tool for the APeLS system is currently taking place. Authoring tools are essential in providing a mechanism for instructional designers to design their own courses independently of a system expert.

The most important element of an e-learning system is how precisely the system models the learner (Conlon et al, 2003). To update the learner model direct (or explicit) techniques along with indirect (explicit) techniques are used. Ideally a system should apply both of these techniques to maintain an up-to-date model of the learner, however it was noted that each system examined seemed to concentrate mainly on only one of these techniques. Direct techniques will always be necessary as the system will generally store some basic information about the learner such as name and password. It was noted that APeLS and 3DE rely heavily on direct techniques such as allowing the learner to select the relevant learning objectives. AHA! does not allow the learner to select course objectives but does provide a means of adapting the course on a page for page basis based on the current state of the learner model. An integral part of AHA! is the ability to update the learner model based on the learner’s browsing through the course. As explained previously, this is achieved through use of an overlay model approach. APeLS and 3DE however do not provide this functionality. While it may be possible to integrate such functionality into APeLS this would not be the case for 3DE as the course is complied at the start of the session and remains static thereafter unless the learner explicitly re-compiles a new course.

The method used for content modelling by APeLS and 3DE encourages reuse of material which is important as courses can be developed quicker and more cheaply if (some) content already exists. In order for this reuse to take place there must be an effective way of searching through this content and the use of descriptive metadata facilitates this. Authoring tools need to be provided to empower the instructional designer in the preparation of such learning objects. The metadata tool has been developed for the 3DE system which aims to speed up and ease metadata insertion in each learning unit (Del Corso et al, 2003). A similar type of authoring tool is currently being developed for APeLS to facilitate the organisation and manipulation of components pertaining to the various models. An advantage of the AHA! system over APeLS is the integration of various authoring tools with the current version as discussed above. The major disadvantage of AHA! is that it does not conform to the AHAM reference model as it intertwines adaptation with content.

The two common methods of adaptation presented were adaptive navigation support and adaptive presentation. Techniques were introduced as to how these types of adaptation could be performed. Key concepts such as candidacy and abstraction pertaining to APeLS were presented. Candidacy facilitates adaptive presentation by selecting the most relevant learning object. The sequencing of the abstract candidate content groups in the narrative facilitates the adaptive navigation. When examining AHA! it was noted that adaptation rules were placed inside of the XHTML page pointing out the same issue again whereby content and adaptation are interleaved. Also, regarding adaptive navigation, AHA! does provide functionality whereby desirable links and undesirable links are shown in different colours. This can be a useful technique as the learner is encouraged to follow the most appropriate path when working through a course. This technique however was not included as part of the APeLS or 3DE system. 3DE facilitated adaptive presentation using a similar, but simpler, mechanism to APeLS where by a custom course was selected once the learner requested a new course.

APeLS provides a more flexible and efficient approach when authoring courses which differ pedagogically. APeLS facilitates the construction of many candidate narratives which may differ in pedagogic approaches. The adaptive engine calls the candidate selector to choose the most appropriate candidate narrative (based on the learner model) from the candidate narrative group for the course. All three systems take into consideration different learning styles, however outside of this 3DE and AHA! offer little else in terms of pedagogical approach. With AHA! in particular it would be costly to develop another course which differed in pedagogical approach due to lack of support for content reuse. APeLS suffers from the fact that it is only recently that the development of graphical authoring tools to aid with the authoring process of the narrative began.

Learning style support is catered for in all systems. Both APeLS and AHA! allow the author to choose which particular learning style model to use when developing a course whereas 3DE constrains the author to use the Honey and Mumford learning style model. The fact that 3DE authors are forced to classify material according to the Honey and Mumford model ensures that authors can integrate any relevant 3DE material developed into their course as this has already been classified in terms of the Honey and Mumford model. This would not be the case for APeLS or AHA! as content developed could have been classified according to a different learning style as to the one the author is interested in. This may not be a big issue with APeLS as the appropriate metadata could be added into the content and learner models, however a selector for that learning style type would have to be written.

Both AHA! and 3DE can provide the learner with some feedback regarding their learning style. In AHA! the learner is allowed to change the learning style as represented in the learner model, while with 3DE the learner can recompile a course according to a different learning style than the one obtained from the Learning Style Questionnaire. APeLS does not provide such capabilities.

While APeLS and 3DE determine a learners preferred learning style through use of a psychometric questionnaire AHA! does not take this approach. AHA! has the ability to update the learner's preferred learning style through observing the learner's browsing through the course.

4. Design

4.1 Introduction

This chapter presents and justifies some of the key decisions made during the design phase of this project. Essentially the design involved extending the APeLS framework to offer learning style adaptation based on the Honey and Mumford Learning Style Model. Through carrying out the literature review of the three e-learning systems, as presented in chapter three of this report, an understanding of how best to integrate learning style theory into an adaptive system was gained. This chapter details how the APeLS framework was extended to produce personalised e-learning courses incorporating the Honey and Mumford Learning Style Model. Some of the features present in 3DE and AHA! were added to APeLS along with other new techniques. The purpose of this chapter, therefore, is to present the details pertaining to this extension of the APeLS system. Background information, such as how content was prepared along with how personalisation was achieved through content presentation, is discussed. The chapter concludes with design information pertaining to the content, learner and narrative course models based on the presented requirements.

A discussion on this decision to create a framework incorporating the learning styles presented in the Honey and Mumford Learning Style Model is presented below. As stated above, the implemented framework was designed to run on an extended APeLS system. APeLS, introduced in chapter three of this report, has been successfully used for other adaptive courses such the SQL course for undergraduate students at Trinity College Dublin. As APeLS provided the architecture for reconciling the learner, content and narrative models, an important task in the design of this extension was how to structure these three models. Following a primary discussion on the requirements for the implementation of this framework extension, a design structure for these three models is presented.

4.2 General Design Considerations

4.2.1 Decision Honey and Mumford Learning Style

Many different learning style models exist such as those based on instructional preferences (e.g. Dunn & Dunn), social interaction models (e.g. Perry), personality levels (e.g. Myers & Briggs) and information processing (e.g. Kolb, Honey & Mumford) (Sarrikoski, 2000). The Honey and Mumford learning style model is categorised as being an information processing model type or more specifically it is an information processing model based on experiential learning (Sarrikoski, 2000). The other models categorise the learner on the basis of less relevant aspects related to learning (e.g. senses and the environmental factors) whereas learning is mainly to do with perceiving and processing information (Sarrikoski, 2000). For this reason the Honey and Mumford learning style model was an appropriate model to choose.

This framework, developed in conjunction with this report, will concentrate on mechanisms to provide greater personalisation for the learner. A primary objective was the provision of personalisation through the selection of content that best matched the learner's preferred learning style. Content as represented by learning objects was categorised according to the four learning styles. For example, a piece of content which was best suited for an activist and pragmatist received a high value for in the metadata tags "activist" and "pragmatist" (and a lower value for "reflector" and "theorist"), therefore this piece of content would be selected for a learner who is categorised as being an activist or a pragmatist. This is further discussed in the Content Model sub-section later in this chapter.

Learning styles are themselves learned as people repeat strategies and tactics that they found successful in the past therefore certain behaviour patterns develop and become habitual (Honey et al, 1992). A person's learning style may change over time, for example a person who moved from a "quick fix" culture to an organisation that by the nature of the work was more reflective may experience increases in his reflector/theorist behaviour over time (Honey et al, 1992). Many people have deliberately set out to strengthen an underdeveloped style and as a result have become a more rounded learner (Honey et al, 1992).

Although, this framework concentrates on mechanisms to provide greater personalisation for the learner, a means is also provided to facilitate the development of a learner's weaker learning style. Feedback pertaining to the learner's learning style as determined through the learning style questionnaire is given to the learner. A learner, after receiving feedback showing strong tendencies towards the activist learning style, may wish the adaptive course to be re-compiled as if his preferred learning style is in fact theorist. Empowering the learner to carry out such an action provides a mechanism to develop a weaker learning style and assisting him in becoming a more rounded learner. The 3DE custom course compiler, which was discussed in the previous chapter, provides such functionality.

At 80 questions the Honey and Mumford Learning Style Questionnaire is quite long. Shortened versions of the questionnaire have been used and such a version may be more appropriate for this project. IBM used a shortened version, for example, to investigate the learning styles of 365 of their managers (Honey et al, 1992). Another shortened version, consisting of thirty-six questions, was devised as part of the 3DE project (Del Corso et al, 2003). It was decided to use this 3DE questionnaire for this course.

4.2.2 Content Preparation

Once a decision was made to use APeLS, the next choice was to decide on the subject matter of the course and how the content could be gathered and developed. It was decided to use the content developed for the aforementioned SQL course used by undergraduates at Trinity College Dublin. It was hoped that this content could be structured in such a way that it could cater for the four learning styles included as part of the Honey and Mumford Learning style model. On further examination however, it was noted that this pre-prepared content did not offer the wide range of activities needed to cover these four learning styles. For example, the activist enjoys being involved in group work and challenging tasks such as finding information. The pragmatist learns best through completing tasks that demonstrate obvious practical advantages and where there are immediate opportunities to implement what they have learned. Such activities were not incorporated into the SQL course used by the undergraduates, therefore, this content needed to be supplemented with such activities that would be appropriate for the four learning styles. This content and its metadata, along with the learner and narrative models, were changed and extended to cater for the Honey and Mumford Learning Style Model along with the additional APeLS features to be added. This revision and extension of the APeLS models is presented later in this chapter.

The next chapter presents a description of the type of activities that were prepared for the four learning styles, but some examples are provided here. Activists worked together in groups and communicate with each other through posting messages in a discussion area; they were expected to search for solutions to tasks in books and on the World Wide Web; they were encouraged to carryout practical activities such as logging into a MySQL database and executing some queries. Reflectors were provided with more examples to reflect upon; they had many short activities to complete and ideas for sample solutions were presented for further reflection. The content previously used in the current SQL course was most appropriate for theorists, this content however was enhanced with extra reading links and some practical tasks. For pragmatists the content was supplemented with links and references to commercial database systems currently used in industry such as Oracle, SQL Server and MySQL; pragmatists were also encouraged to carry out further practical tasks such as logging into a MySQL database and executing queries as they worked through the Data Manipulation section.

A subsection of the current SQL course was used as the basis for this course. The three sections covered were as follows: an introduction to database systems, the relational model and data manipulation (including the INSERT, UPDATE and DELETE commands). In the data manipulation section of the course a Case Study was provided whereby the students must work through a scenario where they have to manipulate data in a database system that has been prepared for a hypothetical customer. It was decided to make use of such as a case-based approach as it incorporated many of the activities that are suitable for all the four learning styles.

4.2.3 Content Presentation

When designing a personalised e-learning course there are many ways in which the course can be personalised. For example, it can be personalised according to the learner's learning goals and/or their preferred learning style. This section examines how this course will offer personalisation based on the learner's learning style and the learner's learning goals.

A learner's learning style was identified through their responses to the questions in the learning style questionnaire (see Learner Model sub-section presented in this chapter for more details). The learner's preferred learning style was determined by the highest score a learner achieved. For example, if a learner's activist value was 50, the reflector 40, the theorist 75 and the pragmatist 30 then this leaner would be classified as a theorist. If there is a substantial emphasis on one stage of the Honey and Mumford learning cycle (presented in chapter two), that emphasis ought to be congruent with the learner's preferred learning style (Mumford, 1995). Classifying the learner as a theorist therefore would mean presenting a theorist orientated course to the learner. As noted earlier in this chapter, learners can strengthen their underdeveloped learning style to become more rounded learners. It would be worthwhile, therefore, to give learners a learning experience that facilitates such development. A decision was made to present the first two sections of the prototype course (Database Background and The Relational Model) according to the learners dominate learning style. The third section of the course (Data Manipulation) had four objectives associated with it. It was decided to present each one of these objectives, i.e. the content associated with these objectives, according to a particular style. For example, an activist would be presented with the activist form of concept, the reflector form of concept two, the theorist form of concept three and the pragmatist form of concept four. This mechanism allows learners to experience other learning styles and assists them in their aim of becoming more rounded learners.

This prototype course developed offered personalisation to the learner by presenting a course that was structured to be congruent with their preferred learning style. The course also offered personalisation by allowing the learner to select what aspect or section of the course they wished to complete. The three sections that the learner can choose from are the three areas introduced previously i.e. Database Background, The Relational Model and Data Manipulation. Through completing a questionnaire, which is introduced in the Learner Model sub-section later in this chapter, the learner determines what concepts the course should cover.

4.2.4 User Features

A number of user features were required to provide feedback and to facilitate learner empowerment. To extend the concept discussed above in relation to becoming a more rounded learner, it was decided that the learner should be allowed to re-compile the course based on any selected learning style. For example, after completing the learning style questionnaire the leaner may be classified as an activist and is therefore presented with the activist course. However this learner may be interested in developing his theorist skills and may wish to be presented with the theorist course. To facilitate this type of learner empowerment an option is included to allow the learner to re-compile the course based on a selected learning style.

When creating a personalised course that takes into consideration learning style it is important to give learners some feedback regarding how the course makes use of learning style theory. One of the findings from an earlier adaptive course developed for APeLS was that users expressed their disappointment that the system did not provide any such information (Clarke, 2003). For this course it was decided to give the learners some relevant feedback, more specifically the feedback is in the form of the following:

- For each of the four learning styles (activist, reflector, theorist and pragmatist) the learner was classified as having a high, medium or low tendency (suitability) for that particular learning style. Along with this classification a brief explanation about the meaning of this classification was also be provided. Furthermore, an email link was included here to allow the learner to send comments based on their classification to the course designer.
- There was information included about how the course was compiled i.e. the first two sections were compiled according to the learner's highest learning style score while in the final section the content was presented to the learner pertaining to the four learning styles – see Content Presentation sub-section above.
- As part of the evaluation of a previous adaptive course developed with APeLS, a student stated that he would have liked to have known what section of the content in the course was suitable for a particular learning style (Clarke, 2003). It was decided,

therefore, that it would be of benefit to the learner to provide feedback through pointing out how the content was designed and how the content was structured to suit a particular learning style.

- General information about the Honey and Mumford Learning Style Model was also included.

4.3 Design of Models

4.3.1 Introduction

This section describes the design decisions pertaining to the learner, content and narrative models when designing this framework. These models incorporated in the APeLS system were introduced in chapter three of this report, however the standard learner, content and narrative models had to be extended to cater for learning styles and the some of the requirements discussed in the above section.

4.3.2 Learner Model

For this framework the standard learner model implemented for APeLS delivering the current adaptive SQL course to undergraduate students was extended. The standard features from the current course that was also present in this framework are as follows:

- A unique learner identifier
- The learner forename and surname.
- A list of competencies that the learner required. A simple questionnaire listing the three sections of the course has been implemented to obtain this information.

The new features to be included are listed below and described subsequently. These new features include:

- The answers to the learning goals questionnaire as supplied by the learner.
- The answers to the learning style questionnaire as supplied by the learner.
- The Honey and Mumford values for the learner pertaining to the four learning styles.
- A value that indicates how style information was incorporated to build the current personalised course.

From a previous adaptive course developed for APeLS, also incorporating learning styles, some learners suggested that they would prefer if the system could maintain the originally chosen options when they revisited the particular control (Clarke, 2003). For example, in that course there were two questionnaires the user had to complete. One of these questionnaires contained questions regarding the learner's competencies while the other was in relation to the learner's learning style. The learner could return to these questionnaires to revise the answers and then re-compile the course based on the updates. However, when returning to any of these two questionnaires the learner had to complete the questionnaire again as the system did not retain the previous answers. Users of the system expressed their dissatisfaction with this feature as, for example, if they only wished to make one change they would in fact have to complete the entire questionnaire again.

For this course it was decided to store the learner's answers to each of the questions in the two questionnaires as part of the learner model. The first questionnaire, as introduced above, contains only three questions regarding the learner's goals for the course. Essentially this questionnaire decides which, if any, of the three sections (Database Background, The Relational Model and Data Manipulation) of the course the learner wishes to complete. The second questionnaire, i.e. the Honey and Mumford Learning Style Questionnaire, determines the learner's learning style. This is a longer questionnaire with thirty-six questions. The version of the Honey and Mumford Learning Style Questionnaire implemented as part of this framework is the one developed as part of the 3DE project – for more on this questionnaire see chapter five. Because the learner model now stores these values, when the learner returns to the questionnaires the previous values saved will be displayed on the screen and the learner can change the required values and re-compile the course without having to answer every question in the questionnaire again.

The activist, reflector, theorist and pragmatist scores calculated based on the answers in the Learning Style Questionnaire are also stored as part of the learner model. These scores are important as they are examined by APeLS when deciding if a piece of content is suitable for a particular learner. For example, if a learner has a low activist score then learning objects that are most suitable for activists, i.e. where the activist value is high, would not be suitable for this learner. These four learning style scores for each learner range from 0 (low) to 100 (high).

The final addition to the learner model is a flag that indicates the style information that was used to compile the current course. As documented in the User Features sub-section above, a course will be built based on the learning style questionnaire results or by the user selecting one of the four learning style types from a drop down list. This information is stored essentially as a reminder to the learner so that he can ascertain whether the course was built based on his actual learning style or by a selected learning style.

4.3.3 Content Model

Similar to the learner model, the content model for this extension of APeLS has many features present in the current adaptive SQL course along with some additional features. The architecture described in chapter three will be maintained i.e. candidate content groups referencing the learning objects. Similar to the current course, therefore, each learning object has metadata describing it. The candidate content group has a list of members (i.e. learning objects associated with it) which are similar on the concept taught. Once again a list will be made of the main similarities and differences between the content model of this course and the standard content model implemented for the current SQL course.

The similar features to the current SQL course include the following:

- The content metadata includes general information such as identifier, title and location.
- Each candidate content group contains an identifier, a list of members and the competency taught.

The new features to be included are listed below and described subsequently. These new features include:

- The inclusion of Honey and Mumford information pertaining to each learning object.
- The introduction of prerequisite metadata pertaining to learning objects.

The most important new feature to be included is the addition of Honey and Mumford markup in the metadata of each learning object. It was decided that the most appropriate mechanism for classifying learning material was to use three classifications – strong, average and weak. For example, if a learning object was only suitable for an activist it would be given a strong rating (i.e. 100) for its activist value, if it was average the activist value would be given an average rating (75) or if it was not suitable for an activist the activist value would be given a weak value (25). This was the most straightforward mechanism of learning object classification for learning style.

The second important extension of a content model was the inclusion of prerequisite information. This is best explained with an example. A candidate content group which teaches concept X may have many members and not necessarily four i.e. not necessarily one for an activist, one for a reflector, one for a theorist and another for a pragmatist. Therefore, if the adaptive engine is selecting the appropriate candidate for an activist learner it examines the activist value for the each of the members. There may be more than one member with an average (75) or strong (100) activist rating. What happens in this scenario is that the selector selects all the suitable candidates, i.e. average or strong, and these candidates are then sorted based on the prerequisite information (see the Narrative Model sub-section for further information).

4.3.4 Narrative Model

The narrative is primarily responsible for constructing a course model of the personalised course. The narrative for this prototype course is structured in a similar manner to the narrative for the current SQL course. The most important difference is the addition of a Honey and Mumford candidate selector along with a generic candidate sorter that is also implemented as part of this framework – see below. Similar to the current course, the narrative will consist of a hierarchy of sections and concepts. Each section will have many concepts. For example, the Database Concepts section has a concept covering the advantages of database management systems and another for the disadvantages of such systems. There are three sections in this course. The first section (Database Background) contains nine concepts, the second section (The Relational Model) also contains nine with the final section

(Data Manipulation) containing four concepts. The list of these concepts can be seen in Appendix 1.

The narrative has two candidate selectors which play an important role in selecting the appropriate content to present. The first candidate selector retrieves all the members that cover the appropriate concept i.e. it retrieves all the learning object identifiers from the candidate content group that covers this concept (each candidate content group for this course covers one concept only). This is a standard selector concept selector implemented for APeLS so it will not be discussed further.

Once these members are retrieved a second candidate selector is invoked. This candidate selector has been custom written for this framework and is based on the Honey and Mumford metadata associated with the learning objects. It retrieves all relevant members from the original group of members selected by the first candidate – "relevant" here infers members that are congruent to the current learner's learning style. For example, there may be ten members that cover concept X and the role of the first candidate selector will be to return these. However, out of these ten members there may be only two of these members that are appropriate for the current learner's learning style, this second candidate selector will be responsible for selecting these two. "Appropriate" here infers that the learning object has an average (75) or strong (100) rating for the current learner's preferred learning style. A sorting mechanism that sorts candidates based on prerequisite information has also been designed and implemented for this framework. There are more details on this in the next chapter of this report where the framework implementation is presented.

The sequence of events, shown in Figure 4.1, that the narrative must execute for each of the concepts in the narrative model are as follows:

1. Check that the current concept is required by the learner i.e. that the current concept is included in the learner model as one of the learner's goals/competencies required.
2. If [1] is true, obtain the appropriate learning style for this learner from the learner model. Else if [1] is false go to [6].

3. Get all the relevant members (learning objects) that cover this concept from the candidate content group (CCG) responsible for this concept.
4. From the members returned in [3] retrieve the appropriate ones for this learner's learning style i.e. the candidates that have an average (75) or strong (100) value for the current learner's preferred learning style.
5. If the number of candidates returned in [4] is greater than 1 then sort these candidates based on prerequisite information from the candidate metadata and add them to the course model. Else add the one candidate to the course model.
6. Move on to the next concept in the narrative and return to [1].

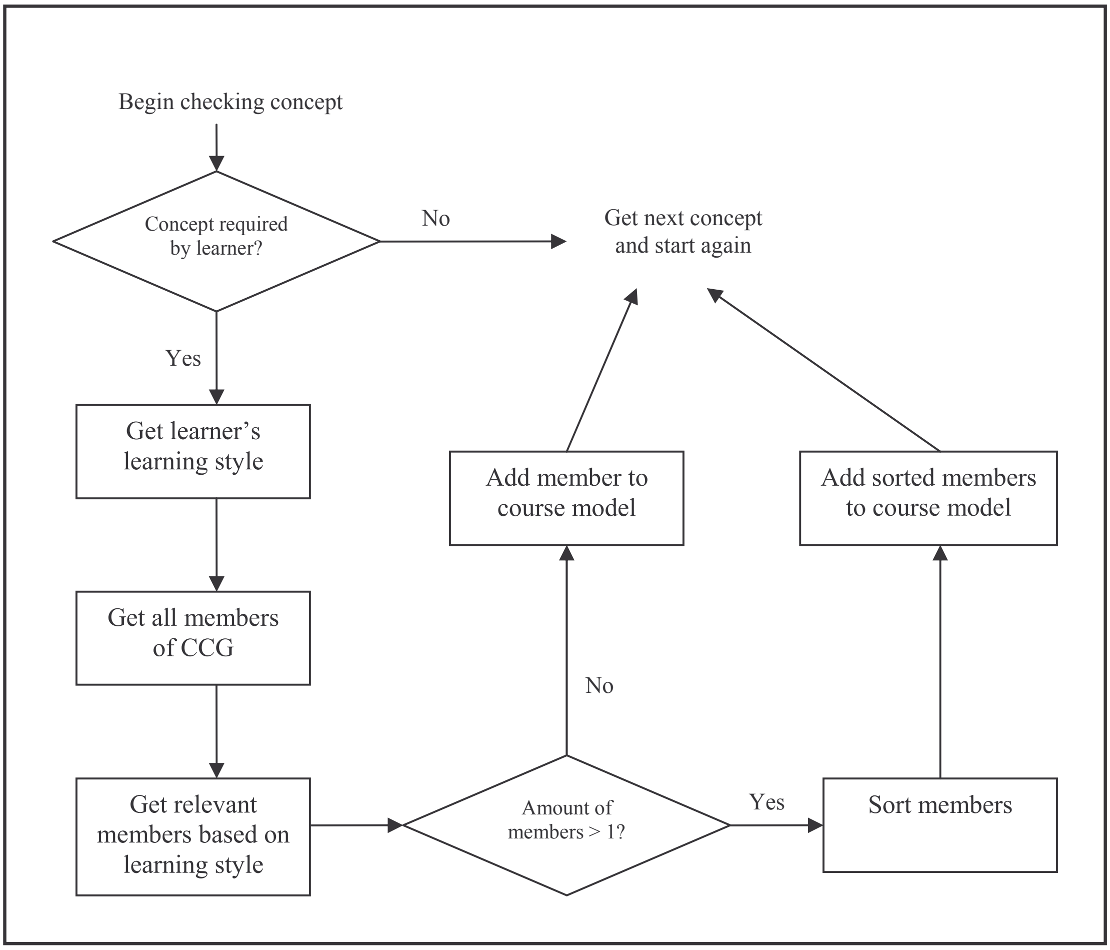

Figure 4.1 Sequence of events narrative executes for each concept

5. Implementation

5.1 Introduction

This chapter describes the implementation of the framework for the extension of APeLS to cater for the Honey and Mumford Learning Style Model. More specifically it demonstrates how the design requirements described in the previous chapter were implemented. There are three main sections in this chapter. The next section illustrates the type of content that is presented to learners as part of the prototype SQL course developed, taking into consideration the four different learning styles. The section following it describes the framework implementation whereby the three APeLS models were extended. The final section describes how the course is presented to the leaner.

A discussion on the key implementation aspects are detailed in this chapter with the following list providing a summary of these:

- Content was developed to incorporate activities appropriate for the four Honey and Mumford learning styles. Practical activities, such as executing SQL statements on a live database and case studies, were added to ensure that learners received more "hands on" experience.
- Four important extensions were added to the learner model of APeLS to incorporate information pertaining to the learner's learning style.
- The content model of APeLS was also extended. The learning object metadata was supplemented with information pertaining to the Honey and Mumford Learning Style Model. Prerequisite information for the sequencing of learning objects was also added.
- A new narrative was written for the SQL course. This narrative included functionality such as selecting learning objects based on the learner's preferred learning style and sorting candidates returned based on prerequisite information.
- The APeLS system was supplemented with user features (e.g. the ability to recompile a course based on the learner's preferred learning style) and user feedback (e.g. the

learner's rating for each of the learning styles following the completion of the learning style questionnaire).

5.2 Content Development

5.2.1 Introduction

As discussed in the previous chapter, content for the prototype course had to be developed to introduce activities that were suitable for the four Honey and Mumford learning styles. Described below is an overview of the content developed for the four different learning styles. It was decided to use the same (or similar) core material for all four learner types and to supplement this content with different learning resources such as activities, tasks and links to further information to cater for the learners with different learning styles. Therefore, instead of developing four bespoke courses, the reuse of content (core material) was encouraged and once this was developed the learning objects suitable for the four learning styles would augment this.

5.2.2 Activist

Throughout the SQL course the activist is only presented with a small amount of theory and content. Activists enjoy a wide range of different activities and this is what they are presented with over the course of the three sections. The activist must go and research a particular topic and complete tasks based on this research. For section 1 (Database Concepts) they must work in groups to complete one of three tasks. Following this group work they answer individual questions based on the work completed by the other two activist groups. In section 2 (The Relational Model) the activist completes individual tasks and may join in on a discussion based on these tasks with other activists. Through collaboration with others, the activist has ample opportunity to interact with and learn from fellow group members. Also in section 3 (Data Manipulation) of the course the activist is expected to research ways of manipulating data in a relational database.

By preparing questions for other group members to answer in section 1 of the course, the activists will feel that they have received some of the limelight they enjoy. The emphasis for the activist material is to allow them to search for particular solutions to problems rather than presenting all solutions to them in a didactic manner. The theoretical side to the content is kept to a minimum with emphasis on activity rather than theory. This is highlighted particularly in section 3 where the activist is expected to log into a database and to complete some data manipulation tasks on the data in this database while working through the course.

5.2.3 Reflector

Reflectors learn best when they are allowed to ponder over activities. The course has been designed to take this into consideration. Throughout each section reflectors are presented with short activities where they must consider what the appropriate solution would be. For example, half-way through section 1 (Database Concepts) of the course they are provided with a link to a webpage and asked if they can identify further features that a database system may have. On the page following this activity a list of possible features are provided with short explanations to provide the reflector with some extra information to reflect on. This technique of asking a question and providing ideas for a solution is employed quite often throughout the reflector's course – further examples are the "Relational Keys" activity and the final activity in section 2 (The Relational Model) of the course.

Reflectors like to stand back from events and observe. In section 3 (Data Manipulation) of the course plenty of examples of how to use the INSERT, UPDATE and DELETE commands in SQL are provided as part of the reflector experience. It is only at the end of each subsection that the reflector is then invited to log into the database and issue these commands i.e. the reflector is encouraged to go through the examples and when finished working through them they can log into the database to gain some practical experience. Because the reflector enjoys carrying out research they are encouraged to refer to the reading section for further reading material along with being provided links to various external World Wide Web pages throughout the course.

5.2.4 Theorist

Theorists enjoy being intellectually stretched. Their course tends to be the most theoretical of all the four courses. Theorists require that what is being offered to them is part of a system or model. In section 2 (The Relational Model) of the course they receive more background on the relational model theory and are also presented with a link to read Dr. Codd's famous research paper on the relational model.

As graphical representation suits the theorist style of learning, most of the graphics that had been prepared for this course are presented to the theorist in all three sections. The activities at the end of each section ensures that the theorist has gained a good grasp of the main concepts and models presented and they are encouraged to continue to the reading section to build on the knowledge they have gained. During the course they are not presented with, what they may perceive as, 'gimmicky' tasks such as executing commands on a database. Instead it is suggested to them that they may try out these commands on a live database if they wish to do so.

5.2.5 Pragmatist

Because the pragmatist wants to know the reason why they study, it is important to link the content of the course to real world situations. When introducing section 1 (Database Concepts) to the pragmatist the importance of studying relational database systems is immediately made obvious. Here the pragmatist is presented with facts and figures showing how pervasive relational database management systems are today. In sections 1, 2 and 3 there are links and activities pertaining to popular commercial databases systems such as Oracle and SQL Server along with the open source MySQL database system.

Similar to the activist, the pragmatist will enjoy the practical aspect of researching material and this is catered for through the provision of tasks with links to web pages where solutions can be found. Pragmatists enjoy being shown techniques that have obvious practical advantages. This fact was taken into consideration when designing the course and is demonstrated, for example, in the second sub-section of section 3 (Data Manipulation) where the benefits of views are introduced after the multi-row INSERT command. Because pragmatists learn best when they are given immediate opportunities to implement what they have learned during section 3 of the course they are asked to execute the SQL commands as they work through the course.

5.3 Implementation of Models

5.3.1 Learner Model

The implementation of the four new features added to the learner model, discussed in the previous chapter, is presented here. Figure 5.1 below shows a portion of an XML file that contains (part of) a sample model of a learner. The additions to the learner model, introduced in the design chapter, are captured in this XML document and will be discussed below. The first two of these new features involves the updating of the learner model following the learner submitting the answers to questionnaires in order to personalise the course. The user, it must be remembered, can return to these questionnaires at any stage of the course to reconsider and change answers they previously made. The course is then rebuilt to reflect changes to the learner's answering.

The first of these new features is the saving, in the learner model, of the learner's goals, or more specifically the learner's answers to the goals questionnaire. As stated in the previous chapter there are in total twenty-two concepts covered in the course. Nine of these concepts related to section one of the course (Database Concepts), a further nine to section two (The Relational Model) with section three (Data Manipulation) having four concepts. A goals questionnaire was designed and implemented as a Java Server Page (JSP) where the learner is asked whether or not he wishes to cover these three sections – there is one question for each section. When the learner submits their answers for this particular questionnaire this information is stored in the learner model where there is an element representing each question. These answers are stored as a boolean value in the learner model. In the example learner model XML file shown in Figure 5.1 the first `adaptivitytype` element, whose attribute value is `goals.answers.answer1`, contains an element `langstring` which has a value of `true` indicating that the learner's answer to question 1 in this questionnaire was "Yes". There are, in total three `adaptivitytype` elements that store the answers for the three questions i.e. one `adaptivitytype` element for each goal question. As discussed in the previous chapter, the advantage of storing these values is that when the user returns to the HTML page containing the questionnaire the previous answers are maintained. Therefore, the learner does not have to complete the entire questionnaire when revisiting it to change some of the answers. When this questionnaire is submitted the relevant concepts are added to the learner model. For example, if the user answered "Yes" to the first question and "No" to the other two, then the nine concepts associated with section one are stored in the learner model as competencies required. This mechanism of storing the concepts required, however, is part of the standard SQL course so will not be discussed further here.

The second new feature introduced in the previous chapter is the storing, in the learner model, of the answers pertaining to the second questionnaire. This second questionnaire, the Honey and Mumford Learning Style Questionnaire, is also implemented as a JSP. The thirty-six questions and the mechanisms for calculating the learner's activist, reflector, theorist and pragmatist scores were developed as part of the 3DE project (Del Corso, 2003). The questionnaire questions can be seen in Appendix 2. In this questionnaire there are 9 questions relating to each of the four learning styles. When the learner has completed the questionnaire the learner model is updated to store the learner's answer to each question. An example of this can be seen in Figure 5.1. The second `adaptivitytype` element, whose attribute value is `learningstyle.honey.answer3`, contains an element `langstring` which has a value of `0a` indicating that the learner's answer to question 3 in this questionnaire was "I totally disagree". A value of `0a` indicates "I totally disagree", `0b` indicates "I partly disagree", `1` indicates "I partly agree" and `2` indicates "I totally agree". There is an `adaptivitytype` element for each of the thirty-six questions. Again, the advantage of storing these values is when the user returns to adjust the answers to this questionnaire the previous answers are maintained.

The third new feature for the learner model is the storing of the activist, reflector, theorist and pragmatist score. In the learning style questionnaire, discussed above, selecting "I totally disagree" or "I partly disagree" contributes 0 to that particular learning style associated with the statement, "I partly agree" contributes 1 while "I totally agree" contributes 2. Once the user completes this questionnaire and submits the answers their learning style scores are calculated. These scores are then normalised, according to the 3DE mechanism, so that they range in value from 0 (low) to 100 (high). The learner model is then updated and these scores

```
<?xml version="1.0" ?>
<learner>
  <general>
    <identifier>learner1</identifier>
    <name>
      <surname>Smith</surname>
      <forename>Peter </forename>
    </name>
  </general>
    <educational>
      <adaptivity>
        <adaptivitytype name="goals.answers.answer1">
          <set type="ALL">
            <candidate>
              <langstring lang="en">true</langstring>
            </candidate>
          </set>
        </adaptivitytype>
               .
               .
        <adaptivitytype name="learningstyle.honey.answer3">
          <set type="ALL">
            <candidate>
              <langstring lang="en">0a</langstring>
            </candidate>
          </set>
        </adaptivitytype>
               .
               .
        <adaptivitytype name="learningstyle.honey.activist">
          <set type="ALL">
            <candidate>
              <langstring lang="en">80</langstring>
            </candidate>
          </set>
        </adaptivitytype>
               .
               .
        <adaptivitytype name="learningstyle.rebuild.userStyle">
          <set type="ALL">
            <candidate>
              <langstring lang="en">activist</langstring>
            </candidate>
          </set>
        </adaptivitytype>
     <educational>
<learner>
```

Figure 5.1 Sample XML document that stores a model of a learner

are stored in the XML file modelling the particular learner. An example of this can be seen in Figure 5.1. The third `adaptivitytype` element, whose attribute value is `learningstyle.honey.activist,` contains an element `langstring` which has a value of `80` indicating that the learner has a high activist score of 80. There is an `adaptivitytype` element for each of the four learning style scores. As discussed in the previous chapter, these scores play an important part role in the content selection.

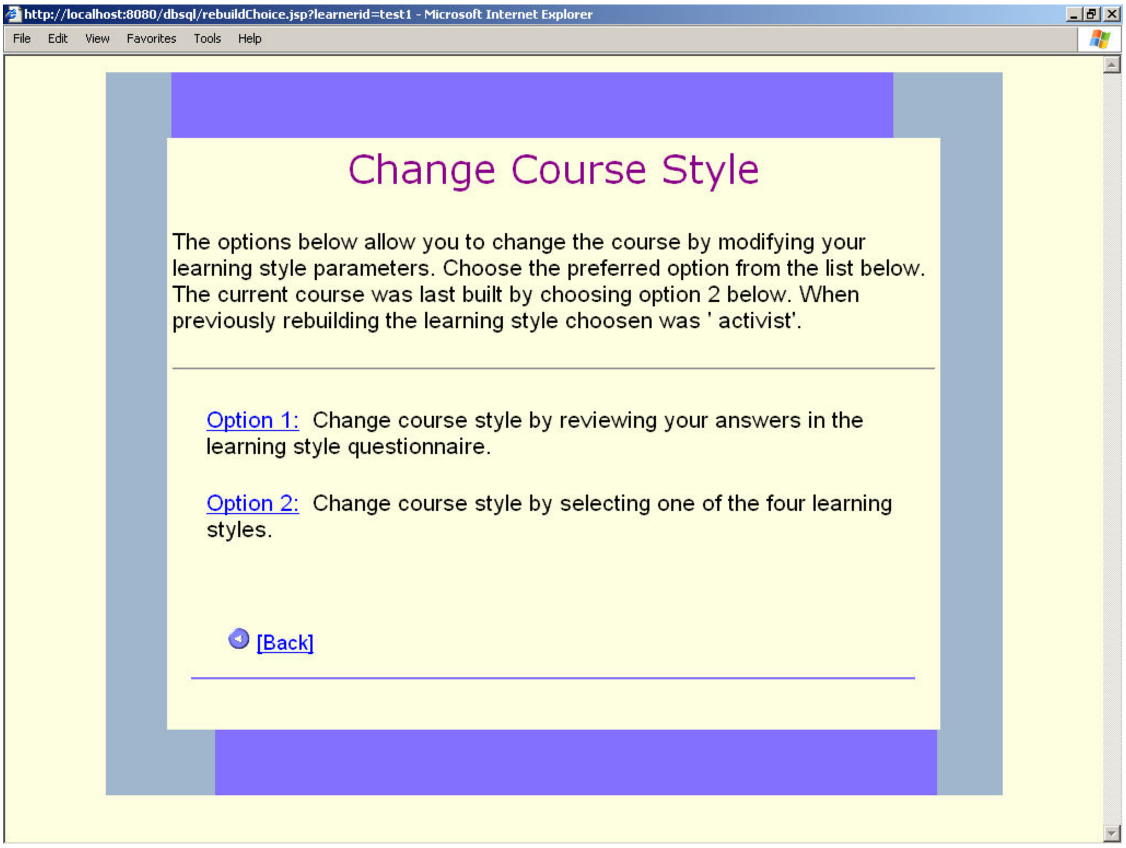

Figure 5.2 Screenshot providing user with information as to what learning style control was used to build the current course

The learner model is also extended to offer user feedback. This feature is a flag to indicate what learning style control was used when building the current course. In the previous chapter it was stated that a course's style is dictated by the learner's learning style questionnaire results or by the learner explicitly selecting a particular style from the list four of styles to build a course. The purpose, therefore, of this addition to the learner model is to offer some feedback to the user as to what style consideration was taken when previously building the course. In Figure 5.1 the final `adaptivitytype` element, whose attribute value is `learningstyle.rebuild.userStyle`, contains an element `langstring` which has a value of `activist`. This indicates that the previous time this course was built the learner explicitly requested that the style to be used was activist and not their preferred learning style as calculated in the learning style questionnaire. A screenshot of the feedback the user would get in this scenario is shown in Figure 5.2 – i.e. the section stating the following "The current course was last built by choosing option 2 below. When previously rebuilding the learning style chosen was 'activist'". If this `langstring` element were blank then that would indicate the preferred learning style, as calculated from the answers of the learning style questionnaire, should be used when rebuilding the course.

5.3.2 Content Model

This section describes the implementation of the content model for this framework. The principle framework implemented here is similar to the standard framework used for APeLS as presented in chapter three. This implementation, as stated in the previous chapter, is an extension of the content model implemented for APeLS as part of the current SQL course used by undergraduate students at Trinity College Dublin. This section will concentrate on these extensions rather that describing the entire content model implemented for APeLS – chapter three provides a discussion on the standard content model pertaining to APeLS. As described in the design chapter, there are two primary extensions to the content model for this framework. This aim of this section is to describe how these extensions have been implemented.

As presented in chapter three, each learning object developed for APeLS will have an associated XML file that contains the metadata for that learning object. This standard XML file describing the learning object will contain information, in the form of elements, such as title, unique identifier, keywords, language, description and location. The first key extension to the content model is the augmentation of this metadata with learning style information. An example of an extended version of the learning object metadata XML structure designed for this adaptive course can be seen in Figure 5.3. In this XML file there is a `supportedlearningstyle` element that contains information pertaining to learning style. As explained in the design chapter, each learning object is classified as having a strong (100), average (75) or weak (25) value for each of the learning styles. From the XML file in Figure 5.3, for example, the `pragmatist` value is `100` indicating that the learning object that this metadata refers to is strongly suited to a pragmatist. The other learning style elements pertaining to learning style i.e. `activist, reflector` and `theorist`, have low values of 25 indicating that this particular learning object is not suitable for an activist, reflector or theorist learner. This markup information plays an important role when selecting candidates (learning objects) that are appropriate for a certain learner with a particular learning style. If, for example, a learner is classified as being an activist then a learning object will only be included in this learner's course if it is rated as having an average (75) or strong (100) activist value – a learning object with a low (25) activist value would not be considered in this scenario.

The second extension to the content model is the inclusion of prerequisite information. This information is important as, for a certain concept, there may exist more than one appropriate learning object. If there is more than one learning object for a particular concept then some mechanism is needed to order them. This explains, therefore, why the prerequisite information is included. From the XML file in Figure 5.3 the presence of a `prerequisite` element should be noted. This particular learning object, whose `identifier` is `db1a04a`, has a `prerequisite` of `db1a04`. What this essentially means is that prior to this learning object being shown to the user, the learning object whose identifier is dba1a04 must be shown first. Another issue arises here. What happens if the prerequisite for learning object X is learning object A if the learner is an activist or learning object B if the learner is a pragmatist? To solve this scenario, if ever a learning object has more than one prerequisite, depending on learning style as introduced above, then more than one `prerequisite` element must be included in the metadata file. Each of the `prerequisite` elements must have an attribute of `type` that can have a value of `activist`, `reflector`, `theorist` or `pragmatist`, thus indicating explicitly what the appropriate prerequisite is for a learner with a particular learning style. For example, the following piece of markup indicates that the prerequisite for an activist is learning object A and that the prerequisite for a pragmatist is learning object B: `<prerequisite type=”activist”> A </prerequisite>` `<prerequisite type=”pragmatist”> B </prerequisite>`.

```
<?xml version="1.0" encoding="us-ascii" ?>
<pagelet>
  <general>
    <identifier>db1a04a</identifier>
    <title>DBMS Advantages</title>
    <keyword>DBMS Advantages</keyword>
    <keyword>Database management software advantages</keyword>
    <keyword>commercial products</keyword>
    <description>List of DBMS advantages</description>
    <language>en</language>
  </general>
  <pedagogical>
    <objectivestaught>
      <objective>
        db.concepts.background.dbms_advantages
      </objective>
    </objectivestaught>
    <prerequisite>db1a04</prerequisite>
    <supportedlearningstyle>
      <type style="honey and mumford">
        <activist>25</activist>
        <reflector>25</reflector>
        <theorist>25</theorist>
        <pragmatist>100</pragmatist>
      </type>
    </supportedlearningstyle>
    <semanticdensity />
    <displayarea />
  </pedagogical>
  <technical>
  <location>db1a04a.html</location>
  <format />
  <requirements>none</requirements>
  <size />
  </technical>
</pagelet>
```

Figure 5.3 Sample XML file storing learning object metadata

One other extension added, which is related to the idea of prerequisites, is to do with the fact that this course allow candidate content groups to have many members (learning objects) associated with it. With the APeLS framework used by current SQL course there is a one-to-one mapping between candidate content groups and learning objects i.e. each candidate content group has only one learning object associated with it. Therefore, with the standard course there is no need for the `prerequisite` element. An example of an XML file representing a candidate content group is shown in Figure 5.4. In this XML file notice that there are many `member` elements, sometimes however there may only be one if necessary. Notice also that the candidate content group has only one concept associated with it (in this case `db.concepts.datamodification.dml`). Each candidate content group in this framework will always have just one concept associated with it, similar to the current APeLS framework.

```
<?xml version="1.0" ?>
<candidategroup>
  <general>
    <identifier>cgdb2a00</identifier>
  </general>
  <members>
    <member>db2a00</member>
    <member>db2a00a</member>
    <member>db2a00b</member>
    <member>db2a00c</member>
    <member>db2a00d</member>
    <member>db2a00e</member>
    <member>db2a00f</member>
    <member>db2a00g</member>
    <member>db2a00h</member>
    <member>db2a00i</member>
  </members>
  <educational>
    <adaptivity>
      <adaptivitytype name="competencies.taught">
        <set type="ALL">
          <candidate>
            <langstring lang="en">
              db.concepts.datamodification.dml
            </langstring>
          </candidate>
        </set>
      </adaptivitytype>
    </adaptivity>
  </educational>
</candidategroup>
```

Figure 5.4 XML file representing a candidate content group

5.3.3 Narrative Model

The narrative model was implemented using Jess and Java, similar to the current implementation. The primary new features developed for this framework were the implementation of a custom candidate selector and a sorting mechanism for candidates. The custom candidate selector will return a list of appropriate candidates based on Honey and Mumford metadata of the candidates. If more than one candidate is returned from this custom candidate selector, the sorting mechanism will sort these candidates based on their prerequisite information.

The custom candidate selector is written in Java and is called from the Jess narrative. A list of candidates and the learner's preferred learning style are passed into the selector. The selector's role is to pick the most suitable candidates from the list based on the learner's learning style and return this new subset to the narrative. It inspects all the Honey and Mumford metadata for each candidate passed in to see if each candidate has an average (75) or high (100) value for the learner's learning style. If this check is positive then the candidate is added to the list of candidates to be returned to the narrative otherwise the candidate will not be considered appropriate for the learner.

Following from the above, if there is more than one candidate suitable, these candidates must be sorted based on prerequisite information to ensure they are presented in the correct order to the learner. The details pertaining to prerequisite information is described above in the Content Model sub-section. This sorter is also written in Java and is called from the JESS narrative. The list of sorted candidates can now be added in the correct order to the personalised course model.

The remainder of the narrative is structured using JESS similar to the way the standard framework is structured. One other custom function was written in Java and called from the JESS narrative. This function returns a string classifying the learner as an activist, reflector, theorist or pragmatist. It achieved this by querying the learner model to find their highest learning style score.

5.4 Remaining Implementation Details

When the personalised SQL course model has been built it needs to be transformed into a structure that the learner can access. This is achieved by sending the XML representation of the course through an XSL transformer to produce a number of Java Server Pages (JSPs). These JSPs provide the HTML form of the personalised course. The JSPs are served by the Jakarta Tomcat server. This mechanism is inline with the way the standard APeLS framework was implemented.

Many of the learner features, described in the previous chapter, are presented to the user as dynamic JSPs custom written for this framework. These features include the questionnaires displaying the learner's previous answers, the scoring page indicating to the learner how they are classified according to the Honey and Mumford Learning Style Model and the page indicating what learning style control (questionnaire or drop down list) was used to build the current course's style as shown in Figure 5.2.

Apart from the numerous JSPs and the custom JESS functions written in Java, many Java utility classes were written to encapsulate functionality that was required in different parts of the course. Functionality encapsulated in these utility classes included the following:

- A class to check the status of the radio boxes in the various questionnaires. It would return "checked" if the item was previously saved in a checked state by the learner. The learner model was queried to determine the learner's previously saved answers.
- A mechanism to process the learning style questionnaire and calculate the normalised Honey and Mumford scores. This was specified in the 3DE documentation and implemented as part of this framework.
- Functionality to get the state of the current learner model, mainly information regarding learning style classification and other additional features added to the learner model.

6. Evaluation

6.1 Introduction

The objectives for this research are presented in chapter one of this report. These objectives are in relation to learning style theory and Adaptive Hypermedia Systems along the implementation of an adaptive course based on this background knowledge. The final objective stated that the background research pertaining to learning style and Adaptive Hypermedia would be applied to develop an adaptive course for evaluation. This knowledge was subsequently applied during the design and implementation stages of this project and is documented in chapters four and five of this report. This chapter will therefore gauge the success of the implemented SQL course where this understanding of learning style and Adaptive Hypermedia Systems was applied. A group of volunteers from the Computer Science Department at Trinity College Dublin agreed to participate in this evaluation process. The results of this evaluation are presented next.

6.2 Course Evaluation

6.2.1 Evaluation Group

The evaluation group consisted of five volunteers from the Computer Science Department at Trinity College Dublin, all of whom had some previous database and SQL experience. These volunteers were either postgraduate students or research assistants. They were all known to this author. Each volunteer was emailed a link to the course with their login details together with basic instructions. Three of these volunteers had no prior knowledge of Adaptive Hypermedia Systems while two others had some knowledge of the current SQL course used by undergraduates at Trinity College Dublin.

6.2.2 Questionnaire

A questionnaire consisting of twenty statements which the user must respond to, having completed the course, was designed. The user had to state whether they totally agreed, partly agreed, partly disagreed or totally disagreed with each one of the statements. These twenty statements can be seen in Appendix 3 of this report. This questionnaire was designed to cover the following goals:

1. To establish the learner's views of learning styles in general and learning styles pertaining to the course content.
2. To determine user satisfaction regarding various features of the course.
3. To get feedback from user regarding the degree of course personalisation perceived.
4. To evaluate the appropriateness of the content in the three sections of the course.

Learning Styles Views

Statement 1: "Prior to taking this course I was aware of learning style theory"

The first five statements in the questionnaire cover the first stated objective above. The first statement, i.e., gauges whether or not the users have any prior knowledge of learning styles. It was assumed that users who had some previous knowledge would have a better understanding of how the course was offering increased personalisation based on learning style. It was found that majority (three) of the volunteers had some knowledge of learning style. Another user stated that he was aware that he had preferred way of learning but was not aware of learning style theory.

Statement 2: "My knowledge and awareness of learning styles (and learning style theory) has increased as a result of completing this course"

Statement two attempts to ascertain if the user's knowledge of learning style had increased as a result of completing this course. All volunteers agreed that their knowledge had increased as a result of completing the course, with three totally agreeing. One user, who had prior knowledge of learning styles, stated that it was just his understanding of the Honey and Mumford Learning Style Model that had increased and not necessarily learning style theory in general. This was inline with what was to be expected and it must be remembered that the purpose of the course was not to teach learning styles.

Learning style links provided allowed the user to understand how the course was incorporating learning style to increase personalisation. It seemed, therefore, that these links were read and understood by the users, both by those who had prior knowledge of learning styles and those who did not. One leaner however pointed out that perhaps this type of information was either not needed at all or else that students should be educated formally about the implications of learning styles rather than simply providing links to information about the topic. This author agrees that perhaps students should indeed be educated about learning styles as such education should ensure that learners would become more effective at learning as they would be aware of learning situations that are suited to them and those learning situations that tend not to be effective. The purpose of the links provided in this course was simply to provide background information to the learner, and as stated earlier, not necessarily to teach the learner about learning styles.

Statement 3: "My preferred approach to learning was calculated correctly through the learning style questionnaire"

The course offered feedback to the learner on how they were classified following the completion of the learning style questionnaire. Statement three attempts to measure learner satisfaction with this classification, that is, to see if they were content with their classification. In general users appeared to be satisfied with their classification with four members agreeing and only one disagreeing. One user commented that his low activist score and medium pragmatist score seemed to contradict each other, however this is not necessarily the case. Activists and pragmatists will process information the same (via active experimentation) but how they perceive information is different however – activists through concrete experience but pragmatists through abstract conceptualisation. Another user commented that he would have like to see the actual scoring as he was classified as having a high score for three learning styles but wanted to know how much they differed by.

Statement 4: "When rebuilding a new course by selecting a learning style type from the drop down list, the changes to the course content reflected my expectations"

The learners were encouraged to compile courses according to different learning styles. Statement four determines the user satisfaction with the content presented based on the learning style they selected. Learners could compare the content presented with the activities documented in the Honey and Mumford information section of the course. All five users either totally or partially agreed that the content presented was what they expected. One user commented that he was slightly confused with the mechanism for presenting content in The Data Manipulation section of the course where each concept was presented according to a different learning style. From this feedback, it may have been better just to present the content according to the dominant style and employ the mechanism of moving the learner through the various learning styles for a separate course or just as an option.

Statement 5: "There was an appropriate amount of information regarding learning style in the 'Learning Style Information' link"

Statement five was in relation to the amount and appropriateness of the learning style information available to the user of the system. Four users agreed (totally or partly) that there was an appropriate amount of such information available and this tended to be consistent with the responses to statement two regarding an increase in awareness of learning style. As stated above, one user felt that learners should be educated in learning style theory prior to taking the course or else this type of information should not be presented at all.

Course Features

The second objective for the questionnaire is to determine user satisfaction regarding various course features. During the design phase of this framework it was decide to include some extra features to provide user feedback (e.g. feedback regarding classification according to Honey and Mumford Learning Style) and user empowerment (e.g. ability to recompile a course based on any of the four learning styles). For part six of the questionnaire user satisfaction level with the seven features below was obtained.

Statement 6: "The following features of the course proved to be useful…"

1. Change course style with questionnaire

2. Change course style by selecting a particular style in drop down list
3. Change course scope with questionnaire
4. Feedback on the learning style questionnaire results
5. Information about Honey and Mumford Learning Style Model
6. How this course uses the Honey and Mumford learning style theory
7. Material in the Case Study link (including MySQL activities)
8. The Reading section

The satisfaction levels for these features can be seen in the Figure 6.1 below.

Feature	Total Agree	Partly agree	Partly Disagree	Totally Disagree
1	3	1	1	0
2	3	1	1	0
3	5	0	0	0
4	4	1	0	0
5	2	2	1	0
6	3	2	0	0
7*	4	0	0	0
8	2	2	1	0

Figure 6.1 Satisfaction levels with course features

* Only four users used feature 7

In general the results displayed in Figure 6.1 were encouraging, but the following are some comments made by users:

- The feature whereby you can rebuild the style course may only be useful for advanced learners.
- The rebuild facility took a little long to run. This was a technical problem not directly related to the course – it had to do with the server the course ran on.
- It was not obvious how the learner was classified on the learning style scoring page, perhaps this classification should have been in bold.

- The actual scoring from the Honey and Mumford Learning Questionnaire was not supplied.
- Some of the explanations, such as how the course uses the Honey and Mumford Learning Style Model, were too verbose. Others however, as stated above, commented that there was not enough information regarding this learning style model.
- More information pertaining to the learning resources should have been made available. For example, links are provided to websites and books but did not point out exactly where information could be obtained (e.g. chapter number, section of website etc) or what type of learner it would be most beneficial to. The type of guidance provided in the course would probably be more suited to activists rather than the other learners.

Course Personalisation

Statement 7: "The mechanism for changing course scope was intuitive to use"

Statement 8: "The mechanism for changing course style was intuitive to use"

Statement 9: "The practical activities pertaining to MySQL were beneficial"

Users found the mechanism for rebuilding the scope intuitive to use (statement seven) and gave a favourable response to the intuitiveness of the rebuild style facility (statement eight). Some users stated, regarding the learning style questionnaire, that they were not too sure how modifying their answers changed the style. However, there was feedback given on their scoring and this could have been used as a gauge. All of the four of the five users who logged into the MySQL database found this activity beneficial (statement nine).

Statement 10: "I understand how the course attempted to personalise content based on my learning style"

Statement 11: "I understand how the course attempted to personalise content based on my course goals"

Statement 12: "When a new course was built following learning style adjustments (made in Change Course Learning Style), the new course reflected these changes"

Statement13: "When a new course was built following scope adjustments (made in Change Course Scope), the new course reflected these changes"

The third objective pertaining to the questionnaire was explicitly related to the users views on the personalisation offered to them in the course. An initial concern when obtaining responses from users regarding the success of adaptation to learning style is that they may not understand how the course is attempting to offer this personalisation. From previous statements (e.g. statement six) it seemed that users were happy to use the various tools provided and to read about how this course offered personalisation based on learning style. It was not surprising, therefore, that four of the five users stated that they understood totally how the course attempted to offer personalisation based on learning style (statement ten). One user stated that he partly disagreed with the statement. Some users explicitly commented that thanks to the explanations provided regarding this adaptation that they were confident that the course met the requirements for learners with different learning styles. Users also responded positively to the rebuild style feature in statement twelve with some users stating that it was easy to see the difference between the styles due to the small size of the courses.

In relation to the personalisation based on goals (statements eleven and thirteen), this mechanism was quite straightforward with the three questions asked mapping to the three sections presented. Unsurprisingly there was a high level of satisfaction with this – all five users selected "I totally agree" for these two statements.

Content Appropriateness

Statement 14: "The content of section 1 of the course (i.e. Database Concepts) was appropriate"

Statement 15: "The adaptation to learning style in section 1 was appropriate"

Statement 16: "The content of section 2 of the course (i.e. The Relational Model) was appropriate"

Statement 17: "The adaptation to learning style in section 2 was appropriate"

Statement 18: "The content of section 3 of the course (i.e. Data Manipulation) was appropriate"

Statement 19: "The adaptation to learning style in section 3 was appropriate"

The final goal of the questionnaire was to find out what users thought of the course content and the adaptation provided for the various learning styles for each section. Statements fourteen to seventeen dealt with the first two sections of the course i.e. Database Background and The Relational Model sections, while the final two statements dealt with section three of the course i.e. Data Manipulation section. In general, from the feedback it seems that users were happier with the adaptation in sections one and two and with the content in section three.

In terms of content, sections one and two did not have as much content, particularly for activists and pragmatists, as section three had. One user commented that more content for these people should be provided. Some of the learning pertaining to activists and pragmatists was supposed to be carried out through group activities. However, due to the small numbers of volunteers available, it was not realistic to carry out these group activities. To evaluate these sections in terms of content, therefore, it would be necessary to have appropriate numbers working together in groups. In terms of adaptation, users, in general, found these two sections provided the appropriate adaptation as described in the relevant learning style sections i.e. the sections that explained how the content was structured for the various types of learners and also the Honey and Mumford Learning Style explanations.

For section three users were happy with the content but found the adaptation technique somewhat difficult to understand – this has been addressed above. One user stated that he found it difficult to understand the difference between the SQL in some of the links and the SQL as defined by MySQL. Perhaps some more explanations were needed here. Another felt that there were too many pages per concepts e.g. INSERT and perhaps that this could be broken down into smaller concepts similar to the mechanism used in the first two sections.

6.3 Impact on APeLS

Sub-section 3.3.7 of this report compares and analyses the three adaptive systems discussed in chapter three. This section examines and compares some of the changes made to APeLS during the course of this research with the AHA! and 3DE systems. For a comparison of APeLS with 3DE and AHA! refer the aforementioned sub-section.

The most obvious feature enhancement of APeLS was the implementation of a framework whereby this e-learning service can now provide learning style adaptation based on the Honey and Mumford Learning Style Model. Because of the clear separation of the three models associated with APeLS it tends to be more flexible than either AHA! or 3DE. AHA! intertwines the domain and adaptation models thus hindering reuse as, for example, adaptation rules are embedded in the XHTML content. 3DE, because of its architecture, does not cater for any other learning style model apart from the one devised by Honey and Mumford. During the design and implementation sections of this report it has been shown that the extension of the current APeLS framework to incorporate learning style adaptation was relatively straightforward. This extension of the APeLS framework is detailed in chapters four and five of this report.

Further direct mechanisms were added to the APeLS system for updating the learner model. This was evident, for example, in the two questionnaires where the learner's responses to the answers were stored. As discussed in sub-section 3.3.7 of this report APeLS and 3DE rely heavily on direct mechanisms for updating the learner model whereas AHA! relies mainly on indirect mechanisms. The mechanism added to APeLS whereby the learner's learning style is changed after their answers to the learning style questionnaire have been calculated could be classified as an indirect mechanism i.e. through answering questions pertaining to behaviour learning style inferences are made.

An important addition to this framework was the ability to provide feedback to learners based on their preferred learning style. As discussed previously in this report, it is important that learners become aware of their learning style to assist them in becoming more rounded learners. Similar to the mechanisms implemented for 3DE, presented in chapter three, a means whereby learners can compile courses based on any of the Honey and Mumford learning styles was incorporated into APeLS. The purpose of this was to allow learners to develop their weaker learning style(s), in an attempt to becoming more rounded and thus more effective learners. In order for learners to identify their preferred learning style, feedback was provided based on their responses to the Honey and Mumford Learning Style Questionnaire. Instead of providing their scoring for the four learning styles, APeLS now provides explanations as to what the learners scoring means for each of the styles. This is in contrast with the 3DE where learners are simply given their scores for the four learning styles with no explanations as to what these scores infer. General information pertaining to the Honey and Mumford Learning Style Model and how provisions were made when developing content for the SQL course (e.g. group work for activists) were also provided. Users of the system expressed that they found this feedback beneficial and their awareness of their own learning style and learning styles in general had increased.

Currently graphical tools are being developed for course designers using APeLS. It was noted in chapter three of this report that such graphical tools are used by course designers of 3DE and AHA! No such graphical tools were used during the development of this prototype SQL course. It was noted how time consuming it was preparing the content (learning object) and content metadata. The framework is now in place for APeLS to offer adaptation based on the Honey and Mumford Learning Style Model for any course. However, course designers would have to create the content and content metadata to run their course on APeLS. One benefit of developing such a framework over the AHA! mechanism is that current content can be reused as adaptation rules are not incorporated into the XHTML pages themselves. Therefore, if it was decided to incorporate a learning object developed as part of this prototype SQL course into another course, then no further changes to the XHTML page or the content metadata would have to be made. This however may not be the case with content developed for AHA! as, for example, a different overlay model may be used for the new course thus making it necessary to change the embedded adaptation rules.

6.4 Summary

This section provides a summary of user feedback following their completion of the prototype SQL course developed in conjunction with this report. In general learners increased their knowledge of learning style as a result of completing this course. It is important for learners to be aware of their learning style as they become aware of situations where they learn best. This results in more effective learning. One user of the system pointed out that learners should be educated on learning style theory rather than simply providing links to learning style information for them to read.

The features to give users feedback pertaining to how this course used learning style and also the tools provided to allow learners create courses according to a learning style of their choice proved popular. Some enhancements to these features were suggested by users and are listed above.

Users seemed to gain a sound understanding of how the course offered personalisation and responded favourably to the adaptation offered in the first two sections of the course which were presented according to the learner's dominant learning style. Users were a little confused about the mechanism of adaptation in the final section of the course whereby each concept was presented according to a different style. Perhaps this type of adaptation should have been an option whereby the learner could select to use or choose to ignore.

Users in general were more satisfied with the content presented in the final section of the course as it was more detailed and did allow for practical activity whereby they could log into a MySQL database. Some users felt that the content for the first two sections of the course would need to be increased.

7. Conclusion

7.1 Introduction

Three distinct objectives for this project were introduced in chapter one of this report. This chapter presents a short review of these objectives. Some future work that could be carried out is detailed in the concluding section of this chapter.

7.2 Review of Objectives

7.2.1 Objective 1

To review the area of learning styles and their potential integration into an Adaptive Hypermedia System.

An extensive state of the art review of Adaptive Hypermedia Systems was carried out and presented in chapter three of this report. Prior to this an introduction to learning styles and the Honey and Mumford Learning Style Model was presented in chapter two. It was decided to study the Honey and Mumford Learning Style Model as, discussed in chapter four, this learning style model is categorised as being an information processing model type. An important part of the state of the art research pertaining to Adaptive Hypermedia Systems was the investigation of how these systems currently offered support for pedagogic principles and learning style. It was noted that all three systems examined provided some support for learning style. The design and implementation presented in this report attempted to build on this knowledge.

7.2.2 Objective 2

To develop a framework for an Adaptive Hypermedia System to cater for personalised e-learning based on a suitable learning style model.

Following the examination of current learning style theory and how current adaptive systems offer support for learning style, the design phase for the extension of the APeLS system incorporating learning style support was completed. The current APeLS framework used for delivering the SQL course to undergraduate students did not incorporate any such learning style personalisation. It was decided to take a subset of the current course and extend the APeLS framework to incorporate learning style personalisation. Chapters four and five detail the development process covering the design and implementation pertaining to the extension of the APeLS framework to incorporate the Honey and Mumford Learning Style Model. This prototype extension of the APeLS system now has the ability to provide personalised courses based on the Honey and Mumford Learning Style Model.

Some features from current Adaptive Hypermedia Systems were included while other features were implemented as a result of feedback provided in the evaluation of a previous course incorporating learning style with APeLS. Other features were included as a result of background reading on learning style. For example, learners will become better all round learners (and thus more effective at learning) if they are exposed to different types of learning experiences suitable for the various types of learners. Therefore, for this reason the final section of the course was structured to allow learners to gain exposure to learning experiences suitable for the activist, reflector, pragmatist and the theorist. Also, a feature was provided to facilitate the building of a course according to a learner selected learning style.

7.2.3 Objective 3

To evaluate the prototype system developed by analysing user feedback.

Following the implementation the extension of APeLS a prototype SQL course was evaluated by a group of volunteers. These users offered to take the course and to provide feedback for evaluation purposes. The results of this evaluation are presented in chapter six of this report. Users expressed their satisfaction with the level of personalisation but also suggested some ways to improve the course and the offered features. The next section of this report details some future work which could be investigated as a development to this project. Ideas for future work arose from user feedback and from this author's experience during the design and implementation stages.

7.3 Future Work

Below is some suggested further work which could be completed as a development of this project:

- Decide on how best to display content to users who score high in more than one learning style. These learners will display characteristics associated with different types of learners e.g. activist and reflector.
- Some users found the learning style questionnaire quite long. Instead of requiring the user to fill in a learning style questionnaire, perhaps a learner's learning style could be inferred by the browsing behaviour. The AHA! system employs this mechanism.
- Marking up each learning object with learning style information may not be the best way to implement the system. There are two reasons for this. The first is that when a course is being built each learning object referenced in all the appropriate candidate content group has to be examined and this process can be quite slow. The second, and more important, reason is that it is difficult to markup such a small piece of information in terms of the four learning styles. Therefore, a better mechanism would be to have another level of indirection with the candidate content groups. Instead of the candidate content group referencing learning objects, it could reference a structured group of learning objects whereby the group is marked up with learning style information instead of the individual learning objects themselves.
- A decision would have to be made whether or not to explicitly inform students about learning style. If students are informed about how the course incorporates learning style then they should receive some formal education on learning style theory.

Appendix 1: Course Concept List

Below is the list of concepts covered by the adaptive SQL course developed in conjunction with this course. Each section has a certain number of concepts associated with it. The three sections are Database Background, The Relational Model and Data Manipulation.

Section 1: Database Background

- db.concepts.background.objectives
- db.concepts.background.introduction
- db.concepts.background.appplicationprograms
- db.concepts.background.features
- db.concepts.background.dbms_advantages
- db.concepts.background.dbms_disadvantages
- db.concepts.background.classification
- db.concepts.background.modeltypes
- db.concepts.background.end_activity

Section 2: The Relational Model

- db.concepts.relationalmodel.objectives
- db.concepts.relationalmodel.datamodel
- db.concepts.relationalmodel.terminology
- db.concepts.relationalmodel.keys
- db.concepts.relationalmodel.properties
- db.concepts.relationalmodel.integrity
- db.concepts.relationalmodel.relational_languages
- db.concepts.relationalmodel.SQL
- db.concepts.relationalmodel.end_activity

Section 3: Data Manipulation

- db.concepts.datamodification.dml
- db.concepts.datamodification.insert
- db.concepts.datamodification.update
- db.concepts.datamodification.delete

Appendix 2: 3DE Honey and Mumford Questionnaire

Below is a list of questions used to determine the Honey and Mumford learning style values for a learner. This questionnaire was developed as part of the 3DE project. The user must indicate their degree of agreement with each statement by selecting one of the following responses: I totally agree, I partly agree, I partly disagree, I totally disagree.

1. I am eager to test new ideas in practice.
2. I like to plan my work properly.
3. I do not believe in impulsive decisions.
4. I act spontaneously.
5. I believe mainly in practical facts.
6. I like the company of sociable people.
7. I want to see the connections between theory and practice immediately.
8. I tend to rely on principles and theories.
9. I prefer having lots of drafts before making the final version.
10. I usually say immediately what I think in order to achieve results quickly.
11. I do not comment before I've thought things through.
12. I get bored with routines.
13. I want to consider all information carefully before making any decisions.
14. I try to consider things in their logical context.
15. I find new experiences interesting.
16. I am always a very practical person.
17. I work out my thoughts before I express them.
18. I like to take several points of view into consideration before deciding my own.
19. Straight action is more typical for me than careful.
20. I tend to produce innovative ideas.
21. I tend to get straight to the point in the meetings.
22. I do not act without proper planning.
23. I like to work in detail before coming to a conclusion.
24. I always prefer a systematic way of working.
25. I work analytically when solving problems.

26. I do not hide my feelings.
27. I work effectively to see the practical results.
28. I avoid making hasty conclusions.
29. I am interested in putting ideas into practice.
30. I tend to organise my thoughts well.
31. I seek theoretical principles behind things and events.
32. I am usually the innovative person in the social situations.
33. I like talking more than listening.
34. Scientifically proved theories interest me.
35. I find it difficult to be spontaneous.
36. I am open to use any efficient method to reach results.

Appendix 3: Course Evaluation Questionnaire

The twenty statements below are what formed the questionnaire for the adaptive SQL course developed in conjunction with this report. The learner must indicate their degree of agreement with each statement by selecting one of the following responses: I totally agree, I partly agree, I partly disagree, I totally disagree. For each statement there was also a text area associated with it so that the learner could provide any extra comments.

1: Prior to taking this course I was aware of learning style theory.
2: My knowledge and awareness of learning styles (and learning style theory) has increased as a result of completing this course.
3: My preferred approach to learning was calculated correctly through the learning style questionnaire.
4: When rebuilding a new course by selecting a learning style type from the drop down list, the changes to the course content reflected my expectations.
5: There was an appropriate amount of information regarding learning style in the 'Learning Style Information' link.
6: The following features of the course proved to be useful:

(a) Rebuilding new course:

Change course style with questionnaire
Change course style by selecting a particular style in drop down list
Change course scope

(b) General Learning Style Information:

Feedback on the learning style questionnaire results
Information about Honey and Mumford Learning Style Model
How this course uses the Honey and Mumford learning style theory
Material in the Case Study link (including MySQL activities)
The Reading section

7: The mechanism for changing course scope was intuitive to use.
8: The mechanism for changing course style was intuitive to use.
9: The practical activities pertaining to MySQL were beneficial.

10: I understand how the course attempted to personalise content based on my learning style.
11: I understand how the course attempted to personalise content based on my course goals.
12: When a new course was built following learning style adjustments (made in Change Course Learning Style), the new course reflected these changes.
13: When a new course was built following scope adjustments (made in Change Course Scope), the new course reflected these changes.
14. The content of section 1 of the course (i.e. Database Concepts) was appropriate.
15. The adaptation to learning style in section 1 was appropriate.
16. The content of section 2 of the course (i.e. The Relational Model) was appropriate.
17. The adaptation to learning style in section 2 was appropriate.
18. The content of section 3 of the course (i.e. Data Manipulation) was appropriate.
19. The adaptation to learning style in section 3 was appropriate.

Appendix 4: Honey and Mumford Learning Styles

The following is the list of activities that are suitable and unsuitable for the four Honey and Mumford learning styles. This information was taken directly from Honey and Mumford (1992).

Activists

Activists learn best from activities where:-

- There are new experiences/problems/opportunities from which to learn
- They can engross themselves in short ‘here and now’ activities such as business games, competitive teamwork tasks, role-playing exercises.
- There is excitement/drama/crisis and things chop and change with a range of diverse activities to tackle.
- They have a lot of the limelight/high visibility, ie they can ‘chair’ meetings, lead discussions, give presentations.
- They are allowed to generate ideas without constraints of policy or structure or feasibility.
- They are thrown in at the deep end with a task they think is difficult, ie when set a challenge with inadequate resources and adverse conditions.
- They are involved with other people, ie bouncing ideas off them, solving problems as part of a team.
- It is appropriate to ‘have a go’.

Activists learn least from, and may react against, activities where:-

- Learning involves a passive role, ie listening to lectures, monologues, explanations, statements of how things should be done, reading and watching.
- They are asked to stand back and not be involved.
- They are required to assimilate, analyse and interpret lots of ‘messy’ data.
- They are required to engage in solitary work, ie reading, writing, thinking on their own.

- They are asked to assess beforehand what they will learn, and to appraise afterwards what they have learned.
- They are offered statements they see as ‘theoretical’, ie explanation of cause or background.
- They are asked to repeat essentially the same activity over and over again, ie when practising.
- They have precise instructions to follow with little room for manoeuvre.
- They are asked to do a thorough job, ie attend to detail, tie up loose ends, dot the is and cross the ts.

Reflectors

Reflectors learn best from activities where:-

- They are allowed or encouraged to watch/think/ponder over activities
- They are able to stand back from events and listen/observe, ie observing a group at work, taking a back seat in a meeting, watching a film or video.
- They are allowed to think before acting, to assimilate before commenting, ie time to prepare, a chance to read in advance a brief giving background data.
- They can carry out some painstaking research, ie investigate, assemble information, probe to get to the bottom of things.
- They have the opportunity to review what has happened, what they have learned.
- They are asked to produce carefully considered analyses and reports.
- They are helped to exchange views with other people without danger, ie by prior agreement, within a structural learning experience.
- They can reach a decision in their own time without the pressure and tight deadlines.

Reflectors learn least from, and may react against, activities where:-

- They are forced into the limelight, ie to act as leader/chairperson, to role-play in front of on-lookers.
- They are involved in situations which require action without planning.

- They are pitched into doing something without warning ie to produce an instant reaction, to produce off-the-top-of-the-head ideas.
- They are given insufficient data on which to base a conclusion.
- They are given cut and dried instructions of how things should be done.
- They are worried by time pressures or rushed from one activity to another.
- In the interests of expediency they have to make short cuts or do a superficial job.

Theorists

Theorists learn best from activities where:-

- What is being offered is part of a system, model, concept, theory.
- They have time to explore methodically the associations and interrelationships between ideas, events and situations.
- They have the chance to question and probe the basic methodology, assumptions or logic behind something, ie by taking part in a question and answer session, by checking a paper for inconsistencies.
- They are intellectually stretched, ie by analysing a complex situation, being tested in a tutorial session, by teaching who ask searching questions.
- They are in structured situations with a clear purpose.
- They can listen to or read about ideas and concepts that emphasise rationality or logic and are well argued/elegant/watertight.
- They can analyse and then generalise the reasons for success or failure.
- They are offered interesting ideas and concepts even though they are not immediately relevant.
- They are required to understand and participate in complex situations.

Theorists learn least from, and may react against, activities where:-

- They are pitchforked into being something without a context or apparent purpose.
- They had to participate in situations emphasising emotions and feelings.
- They are involved in unstructured activities where ambiguity and uncertainty are high, ie with open-ended problems, or sensitivity training.

- They are asked to act or decide without a basis in policy, principle or concept.
- They are faced with a hotchpotch of alternative/contradictory techniques/methods without exploring any depth, ie as on a 'once over lightly' course.
- They doubt that the subject matter is methodologically sound, ie where questionnaires haven't been validated, where there aren't any statistics to support an argument.
- They find the subject matter platitudinous, shallow or gimmicky.
- They feel themselves out of tune with other participants, ie when lot of Activists or people of lower intellectual calibre.

Pragmatists

Pragmatists learn best from activities where:-

- There is an obvious link between the subject matter and a problem or opportunity on the job.
- They are shown techniques for doing things with obvious practical advantages, ie how to save time, how to make a good first impression, how to deal with awkward people.
- They have the chance to try out and practice techniques with coaching/feedback from a credible expert, ie someone who is successful and can do the techniques themselves.
- They are exposed to a model they can emulate, ie a respected boss, a demonstration from someone with a proven track record, lot of demonstration from someone with a proven track record, lots of examples/anecdotes, a film showing how its done.
- They are given techniques currently applicable to their own job.
- They are given immediate opportunities to implement what they have learned.
- There is a high face validity in the learning activity, ie a good simulation, 'real' problems.
- They can concentrate on practical issues, ie drawing up action plans with an obvious end product, suggesting short cuts, giving tips.

Pragmatists learn least from, and may react against, activities where:-

- The learning is not related to an immediate need they recognise, ie they cannot see an immediate relevance/practical benefit.
- Organisers of the learning, or the event itself, seems distant from reality ie ‘ivory towered’, all theory and general principles, pure ‘chalk and talk’.
- There is no practice or clear guidelines on how to do it.
- They feel that people are going round in circles and not getting anywhere fast enough.
- There are political, managerial or personal obstacles to implementation.
- There is no apparent reward form the learning activity, ie more sales, shorter meetings, higher bonus, promotion.

References

Aroyo, L., De Bra. P., Houben, G. (2003) *Embedding Information Retrieval in Adaptive Hypermedia: IR meets AHA!* Proceedings of the AH2003 Workshop, Budapest, Hungary, 2003

Bruen, C., Conlan, O. (2002) *Adaptive ICT Support for learning Styles - A Development Framework for re-useable learning resources for different learning styles & requirements* ITTE 2002 Annual Conference of the Association of Information Technology for Teacher Education, 2002

Brusilovsky, P. (2001) *Adaptive Hypermedia* User Modelling and User-Adapted Interaction 11, pages 87-110, 2001

Clarke, L. (2003) *Adult Learning Through Adaptive Systems* Dissertation for Master of Science in IT in Education, Trinity College Dublin, 2003

Conlan, O., Hockemeyer, C., Wade, V., Albert, D. (2003b) *Metadata Driven Approaches to Facilitate Adaptivity in Personalized eLearning Systems* Journal of the Japanese Society for Information and Systems in Education, 2003

Conlan, O., Dagger, D., Wade, V. (2002) *Towards a Standards-based Approach to e-Learning Personalization using Reusable Learning Objects* E-Learn 2002, World Conference on E-Learning in Corporate, Government, Healthcare and Higher Education, Montreal, 2002

Conlan, O., Wade, V., Bruen, C., Gargan, M. (2002a) *Multi-Model, Metadata Driven Approach to Adaptive Hypermedia Services for Personalized eLearning* Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, Malaga, 2002

Conlan, O. (2000) *Novel Components for Supporting Adaptivity in Education Systems - Model-based Integration Approach* Multimedia 2000, Los Angeles, 2000

Dagger, D., Wade, V., Conlan, O. (2003) *Towards "anytime, anywhere" Learning: The Role and Realization of Dynamic Terminal Personalization in Adaptive eLearning* Ed-Media 2003, World Conference on Educational Multimedia, Hypermedia and Telecommunications, Hawaii, 2003

Dagger, D., Conlan, O., Wade, V. (2003a) *An Architecture for Candidacy in Adaptive eLearning Systems to Facilitate the Reuse of Learning Resources* E-Learn 2003, World Conference on E-Learning in Corporate, Government, Healthcare and Higher Education, Phoenix, 2003

De Bra, P (1999) *Pros and Cons of Adaptive Hypermedia in Web-based Education* Journal on CyberPsychology and Behaviour, Vol. 3, pages 71-77, Mary Ann Lievert Inc., 2000.

De Bra, P., Aerts, A., Smits, D., Stash, N. (2002) *AHA! Version 2.0 More Adaptation Flexibility for Authors* Proceedings of the AACE ELearn'2002 conference, pages 240-246, 2002

De Bra. P., Aerts, A., Smits, D., Stash, N. (2002a) *AHA! meets AHAM* Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, Springer LNCS 2347, pages 381-384, 2002

De Bra, P., Aerts, A. De Lange, B., Rousseau, B., Santic, T., Smits, D., Stash, N. (2003) *AHA! The Adaptive Hypermedia Architecture* Proceedings of the ACM Hypertext Conference, Nottingham, pages 81-84, 2003

De Bra, P., Stash, N., De Lange, B. (2003a) *AHA! Adding Adaptive Behaviour to Websites* Proceedings of the NLUUG Conference, 2003

Del Corso, D., Ovcin, S., Morrone, G. (2003) *A Teacher Friendly Environment to Foster Reusability and Learner-Centered Customization in the Development of Interactive Educational Packages* Frontiers in Education, 2003

Del Corso, D., Ovcin E, Morrone G., Gianesini, D. *The 3DE Custom Course Compiler: an engine to assemble custom web courses* FIE 2002 Conference, Boston, 2002

Del Corso, D., Ovcin, E., Salojarvi, S., Morrone, G., Gianesini, D., Kvist, T. (2001) *3DE: An Environment for the Development of Learner-Centered Custom Educational Packages* ASEE/IEEE Frontiers in Education Conference, Reno, NV, 2001

Honey, P., Mumford, A., *The Manual of Learning Styles*, Maidenhead, Berkshire, 1992

Honey, P., *Learning Log: A way to enhance learning from experience* Maidenhead, Berkshire, 1984

Mumford, A., *Effective Learning,* Institute of Personnel and Development, London, 1995

Sarrikoski, L., Salojarvi, S., Del Corso, D., Ovcin, E. (2000) *The 3DE: An Environment for the Development of Learner-Oriented Customised Educational Packages* ASEE/IEEE-Frontiers in Education Conference, 2000

Stash, N, Cristea, A., De Bra, P. (2004) *Authoring of Learning Style in Adaptive Hypermedia Problems and Solutions* WWW'04 Education Track, ACM, New York, 2004

Stash, N., De Bra, P. (2003) *Building Adaptive Presentation with AHA! 2.0* Proceedings of the PEG Conference, Saint Petersburg, June 2003