A Metadata Driven Approach to Educational Adaptive Hypermedia

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

Niamh Connor
Department of Computer Science
Trinity College Dublin

1999

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: ___________________________

Niamh Connor

September 1999

Permission to lend and/or copy

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

Signed: ___________________________

Niamh Connor

September 1999

Summary

The objective of this dissertation is to develop a teaching tool which extends the benefits of Web learning by providing increased flexibility for both students and administrators. The dissertation will investigate the technical and pedagogic requirements for courseware customisation to support constructivist based learning, design and develop a demonstrator capable of runtime customisation of WWW courseware based on individual student profiles and evaluate the complexity in supporting and administering such systems.

Acknowledgements

I would like to thank my supervisor, Vincent Wade for his encouragement and guidance, Owen Conlan for his valuable suggestions and all the members of the Computer Science department who have provided support during the year.

Table of Contents

Illustrations

Chapter 1

INTRODUCTION

Introduction

The role of education in society is undergoing dramatic change. Technology is emerging as an essential tool in the delivery of a 'mass' system of education. The rapid and pervasive impact of the World Wide Web has prompted widespread interest in the application of web technologies in the educational domain. This dissertation attempts to investigate a metadata strategy to support flexible, individualised courseware on the World Wide Web.

This introductory chapter outlines the influences shaping the changes in the role of education and the direction in which the delivery of education is likely to proceed. A description of the objectives that this dissertation proposes to achieve are followed by an overview of the structure of the dissertation.

Education and Technology

In the past the success of an economy has variably depended on the availability and cost of labour or raw material or capital. With the decline in heavy manufacturing industry, the automation of work and the 'global triumph' [Appleyard 99] of computer culture, many types of working-class employment no longer exist. 'Brain, not brawn, is now the most important world commodity ' [Appleyard 99]. This post-technological era will obey an economic order in which knowledge is the key resource [Drucker 94].

The control and application of information will become the determining factor in the efficacy of the economy. An assured supply of highly educated workers is essential to ensure that the economy can sustain sufficient productivity. Lifelong learning is the key to sufficient productivity in this economic order [Drucker 94].

A society based on knowledge will force the population to cultivate a habit of continuous learning [Wedekind et al 98] over a lifetime to ensure that they are not excluded from the new cognitive elite [Appleyard 99].

Lifelong learning is already becoming a mainstream expectation in business, industry and government [Reeves 98]. The delivery of education to a majority of the population over a lifetime will represent a significant challenge in terms of administration and management [Wade et al 97]. As the foundation of the knowledge society, education is set to become a central social and political concern [Drucker 94].

The application of learning is dependent on our evolving understanding of how the neurophysiological process of learning occurs. Learning in the brain is believed to occur by the connection of massses of neurons [Russell, Norvig 95]. An organism represents its experience by the pattern of activated neurons in the brain. Long term memory is developed by establishing repeated connections among masses of neurons [Hunt 97]. This implies that for learning to occur pathways among the masses of neurons must be built and activated [Hunt 97]. This pattern of activated neurons is the representation of the world in the human mind and is linked to the actual world by a mental model. A mental model 'is simply an internal representation of a person's current understanding of something ' [Galitz 97].

Mental models may be conceptual or specific [Barker et al 1998]. Specific models are limited to knowledge about a specific task. For example, a student can learn how to use an Oracle database system. They may learn to retrieve data from tables and alter table definitions. The mental model the learner has constructed is specific to using a particular database in a rigid manner.

Conceptual models are generic and increase the scope of application of knowledge and encourage the development of a critically aware thinker who, as a mature learner, can apply flexible and resourceful strategies to the acquistion of new knowledge.
A user who learns how to use a number of databases may develop a generic model of the concept of databases. This would entail the development of an understanding of the potential application of database systems which is not constrained to specific instances of database software. A generic model represents a deeper understanding of the subject in a higher level context that its use in limited situations.

In a neurophysiological context, the specific model represents an isolated pattern of neurons whose connection to other neuronal pathways is limited. A generic model indicates a pattern of neurons which has substantial interconnection with other major pathways within the neural network. Constructivism as an educational theory is compatible with this model of learning [Atherton 99].

The understanding of how learning occurs necessarily informs the practice of teaching and the strategies applied to maximise the effectiveness of learning. The focus has shifted from teaching strategies to learning strategies showing the new role of the learner as the centre of the educative process. The learner is in control of her own learning and is therefore responsible for her own learning. The function of educators is to provide more than the transmission of information. Students must be helped and encouraged to learn by connecting the acquired information with their experiential knowledge which is represented by the network of activated neurons in the brain.

The emphasis must migrate from directly teaching students to indirectly assisting students to build and activate these mental pathways [Hunt 97]. This is done by creating opportunities which facilitate reflection. Reflection is thinking about something in relation to what you already know, thinking about one's thinking [Hunt 97]. Reflection allows you to situate the new information in the context of the connected knowledge in your brain.

Traditional methods of teaching will not be abandoned but their application will be informed by the idea that it is crucial to distinguish between the acquistion of information and the acquisition of knowledge.

Traditional lecturing is recognised to be useful as a theatre of transcription [Hunt 97] but is no longer romanticised [Reeves 98] as the optimal medium for teaching. Learning does not occur until the student innately constructs his own knowledge by actively reflecting on or discussing the transcripted information. Information gained from a lecture is not knowledge because it is a passive mode of learning and the knowledge is not contextutalised and connected in the mind of the student [Hunt 97]. Traditional approaches require conceptual development and the refinement of mental models to eventuate outside of lecture times [Mc Naught 97] , [Wade et al 97]. Nothing can happen until the learner is actively engaged [Hunt 97] whether in a lecture theatre or sitting in front of a PC.

The potential benefits of technology have always depended on the cost of its introduction. Technology is becoming faster, more efficient, more powerful and at the same time cheaper. The amalgamation of computers and telecommunications is simultaneously decreasing the cost of communicating while increasing the amount of information that can be exchanged. Distance is becoming irrelevant as the provision of communication becomes location independent [Cairncross 95]. Technology now offers the opportunity to provide facilities for learning without regard to time, place or computing platform [Hunt 97].

Education needs technology. The management and administration of mass lifelong education in the post-technological knowledge-based society is not feasible without technology [Wade et al 97]. The new approaches to learning requiring individualised strategies is possible traditionally, only in one to one or small tutorial sessions. The teaching resources required for such a strategy are unattainable. The alternative is to apply these strategies using technology. The benefits of transformed pedagogical strategies cannot be realised without technology. The benefits of technology cannot be realised without transformed pedagogical strategies [Reeves 98] , [Drucker 94]. Technology is an essential vehicle for pedagogical change [Reeves 98] but its application must be driven by educational need [Tuckey 92].
Technology can give the student independence and control in terms of when, where and how learning occurs. This requires maturity in the way a student learns. Technology has proven much more effective in third level than in schools [Oppenheimer 97]. Students must make decisions and evaluate their own progress [Hunt 97].

Activities can be designed to encourage students to expend effort in thinking about the activity they are undertaking and applying knowledge in different contexts thereby actively constructing and refining their own knowledge [Hunt 97]. The more control a student has, the higher the order of thinking skills they will use and the more generic the mental model they will develop. The requirements for maturity on behalf of the student rises in parallel with the order of thinking skills employed.

An article in the Economist argues that the consequences of the communications revolution will be as profound as the discovery of electricity. Effects such as the building of skyscrapers allowed by the lift and the release of women to industry and the consequential reshaping of the family could never have been foreseen when electricity was discovered [Cairncross 97]. The evolution of ways to apply new technologies are gradual and unpredictable [Cairncross 97]. We cannot predict the way in which the use of technology in education will evolve, but the effects will be as pervasive as they are unpredictable.

Objectives of Dissertation

The primary objective of this dissertation is to investigate the use of metadata to support web-based adaptive, educational courseware. In the interest of achieving this objective a number of sub-objectives have been identified. The dissertation will commence with an investigation of educational theory for web based education. The use of adaptive courseware will be surveyed with an emphasis on the educational domain. The current educational metadata standards and approaches will be reviewed. The dissertation will then propose an architecture to support metadata driven adaptive educational courseware and develop an implementation to demonstrate the concept.

Overview of Dissertation

Chapter 1

Introduction. This chapter presents the motivation for the dissertation, outlines the objectives which the dissertation aims to achieve and describes the dissertation structure.

Chapter 2

Educational theory. This chapter reviews the development of educational theory and proposes a number of technology supported approaches with are compliant with constructivist principles of learning.

Chapter 3

Metadata Standards and Approaches. This chapter reviews the standardisation efforts that are ongoing in the area of educational metadata.

Chapter 4

Design. This chapter describes the design process which was undertaken and proposes an architecture for adaptivity in educational courseware using metadata.

Chapter 5

Implementation. This chapter describes the implementation that was developed to demonstrate the use of metadata in Educational Adaptive Hypermedia.

Chapter 6

Evaluation and Conclusion. This chapter evaluates the dissertation in view of the stated objectives and presents the consclusions reached.

Summary

This chapter has described the changing role of education in society and the emergence of technology as an essential element in the delivery of a ‘mass’ system of education. The objectives which this dissertation proposes to achieve were outlined and the chapter concluded with a description of the structure of the dissertation.

Chapter 2

EDUCATIONAL THEORY

Introduction

This chapter charts the development of educational theory by describing the three key educational theories of Behaviourism, Cognitivism and Constructivism. A critique of the implications of Constructivism concludes with a review of the techniques that can be used in web based educational systems to support a constructivist pedagogy.

The development of educational theory

Behaviourism

There is only reality. Knowledge is discovered.

The primary proponents of Behaviourism were Pavlov in Russia and Thorndike, Watson and particularly B.F. Skinner in the United States [Atherton 99]. Behaviourism stems from the view that psychology should concern itself with the behaviour of human beings rather than mental phenomena about which no concrete knowledge is possible. This supposes that objective knowledge about the external environment is possible.

Experimentation with animals was used to make generalisations about human behaviour. Such a generalisation is reflex learning or Classical Conditioning in which an unconditioned stimulus is presented together with a conditioned stimulus such that the unconditioned response is eventually produced on the presentation of the conditioned stimulus alone, thus becoming a conditioned response [Atherton 99].

A well known behaviourist experiment involves the presentation of food to a dog. The presentation of food produces salivation. If food is given to a dog with the sound of a bell, eventually the dog will salivate when hearing the bell alone. The dog has learned to associate the bell with food. This response is conditioned.

Operant conditioning demonstrates that if a behaviour results in a reinforcing consequence, the probability of the repetition of that behaviour is increased. Positive reinforcement is more effective than negative reinforcement, for example criticism or punishment.

In the context of education, the behaviourist learner is a passive recipient of knowledge. This knowledge necessarily induces correct behaviour. Learning is then the acqusition of this objective knowledge through rehearsal and correction [Tuckey 92]. The emphasis in teaching is on the reinforcement of correct behaviour. The enviroment presents a stimulus or antecedent that will prompt a particular response. If the response is correct, a favourable consequence will ensue to reinforce the appropriateness of that response. Learning has been achieved when that response is a consistent consequence of the presentment of that antecedent [Newby 96].

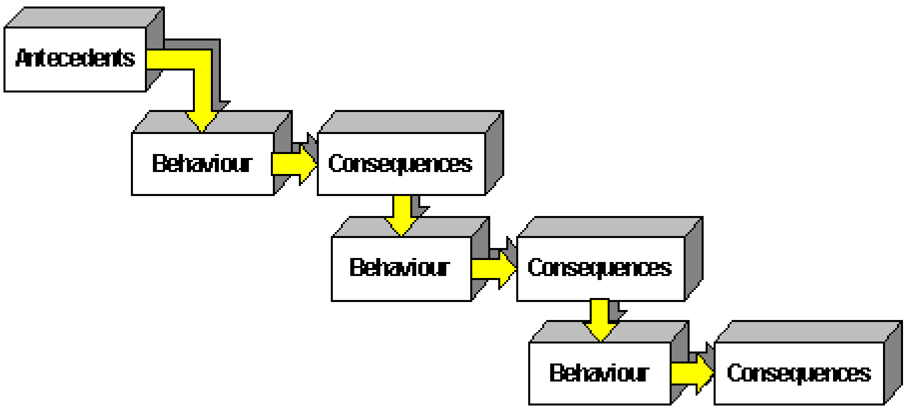


Figure 1 Antecedents include the consequences of previous behaviour [Atherton 99]

Learning is measured by estimating the probabilty of that stimulus causing the correct response by the frequency with which it does effect the correct response. The behaviouristic approach to learning produces isolated and inert information which does not make use of higher order thinking skills or result in the development of conceptual models [Laurillard 95].

Cognitivism

There is an external reality and an internal representation of that reality.

Cognitivism does recognise the existence of mental phenomena but still views the enviroment as an objective reality about which we may have knowledge. Learning attempts to make that representation of knowledge as accurate as possible. It began with Gestalt theories which attempted to theorise human perception and the order imposed on the world by the mind [Atherton 99].

This well known Gestalt image demonstrates the problem that the mind has with ambiguity. The mind can see the image as a vase or two faces but not both. This indicates that the mind insists on finding patterns in things.

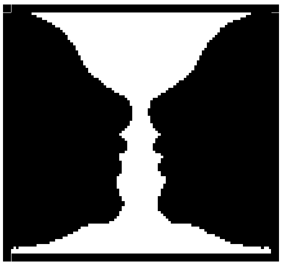


Figure 2 Ambiguity and the mind [Huffman et al 97]

Cognitivism was greatly influenced by the work of the developmental psychologist J. Piaget whose work is based on the observation of children completing exercises and tasks. Cognitivism attempts to describe how people understand material, their aptitude and learning styles. Piaget describes the assimilation of information which results in an accomodation in the mind to create a framework for the new information. These processes allow adaptation of the representation of reality in the mind when new information is encountered. Organisation refers to the structuring of the adapted information [Atherton 99]. Behaviour is caused not by the enviroment but by mental processes.

The mental processes of the mind were modelled on computer processing. Knowledge in memory is ordered internally and new knowledge is integrated into this ordered schema as it is acquired. Information is selectively received by 'Attention'. This information is then integrated into the inherent order of memory via a process of 'Encoding'.

Information becomes knowledge when it is integrated into the existing cognitive structure [Tuckey 92]. The knowledge can then be remembered in the process of 'Retrieval'. New information is syndissertationed in the context of what knowledge is already in memory so prior learning becomes significant.

The emphasis on teaching and learning strategies shifts to techniques to complement the attention, encoding and retrieval of knowledge [Newby 96]. This can be achieved by the careful organisation of content, and the use of analogies and mnemonics. Although cognitivism recognises the importance of the learner making sense of new information, the teacher maintains her central role as a transmitter of objectively correct information which the student should absorb. The function of learning in cognitivism is to create an accurate representation of the external world in our minds.

Constructivism

There is no reality. Knowledge is constructed and relative. Meaning is negotiable.

Theories and proponents of Constructivism vary in the radicalism of their rejection of reality, however all constructivist theories propose that knowledge is subjective and cannot be separated from the knower. There is no singular, accurate representation of the world conceivable by the human mind, just an interpretation of experience. Constructivism takes much from the study of cognitivism and subsumes the theories of internal cognitive processes into a new theory which rejects the idea of a knowable reality.

The individual will actively construct meaning in the context of the knowledge she already has and in the context of the belief system and culture of which she is a part. In this way, meaning is subjective, socially negotiated and must be adaptive. Mental models in the constructivist schema are fluid and adaptive and are modified through collaboration or reflection [Tuckey 92]. Knowledge is a collection of concepts which fit with the experience of the individual [Atherton 99]. Learning becomes a change in meaning constructed from experience.

If we accept that knowledge is an individual interpretation of experience then learning is the construction of a new or refined interpretation [Tuckey 92]. The influential constructivist developmental psychologist, L. Vygotsky noted the increased effectiveness of a child working on a task in collaboration with an adult and proposed that the engagement with another person encouraged refinement of the child’s thinking which increased performance [Atherton 99]. He proposed that a zone of proximal development exists which is the distance between the level of actual development and the more advanced level of potential development that comes into existence in communication between unequal partners [Holcomb et al 98].

J. Bruner emphasised the importance of cognitive structures or mental models which must be in place to provide a structure in which the integration of new information can occur. In other words, the meaning ascribed to new information is dependent on how it connects with the knowledge already present in the individual's mind [Tuckey 92].

New information is important for different reasons to the recipients of the information. Similarly, some students cannot grasp what is being taught because they do not possess the required conceptual framework within which to contextualise the information. The information remains isolated and is forgotten. Bruner focused on the predisposition or readiness of the student to learn, promoting a spiral organisation to content – building on what is already known. He also advised the extrapolation of a subject by going beyond the information given and transferring the knowledge gleaned across subjects and disciplines. This extrapolation extends the generality and scope of the knowledge being acquired.

The Implications of Constructivism

Constructivism is compatible with contemporary neurophysiological models of how the brain learns [Atherton 99]. The teacher is removed from their central role of transmitting information to that of a guide to encourage students to construct their own understanding and negotiate meaning in a social context. As there is no definitive solution to a problem, the guiding must be tentative [Atherton 99] and encourage the student to use their critical faculties to work out the general principles themselves. Collaboration, interaction, discussion and reflection all facilitate this mediation of knowledge. This learner-centric approach has significant implications for the design and implementation of educational systems.

Learner as an individual

The learner is an individual constructing her own knowledge independently of the teacher. The control is in the learner's hands and the responsiblity for learning is her's alone. This new role requires that the learner is ready to take on this responsiblity. She must be able to employ effective learning strategies which are appropriate to the learning task at hand. An educated person is someone who has learnt to learn [Drucker 94].

She must have a sufficient knowledge framework within which to make sense of the new information. This framework should be built upon as progress is made [Atherton 99]. Theoretical and analytical knowledge requires critical thinking skills. The degree of student participation is crucial.

Learner in an environment

The information being presented should be personally relevant to the learner. A concrete context should be established to demonstrate the authenticity of the knowledge so that the immediacy of the link between the knowledge and the promise of real-world competence is increased. The information should be transferred to other problem domains to illustrate the significance of the knowledge and the generality of the concepts being applied. The information should be presented in a way that is easy to assimilate [Wade et al 97].

Learner as a social being

There must be guidance in the students examination of his ideas and concepts so he can focus and refine his conceptual development. The student must be given the opportunity to negotiate meaning through discussion, collaboration and/or reflection. These strategies promote refinement of conceptual development through the student evaluating her own progress by externalising her own ideas.

Computer as an amplifier

The computer amplifies but does not discriminate on the quality of its amplification [Oppenheimer 97] , [Greenberger 64]. When computers are used as a new medium for teaching, the improvement in the performance of the students is only 'modest' [Reeves 98]. The full benefit of technology will not be achieved without making transformative changes in the teaching and learning strategies of students and teachers or in other words ' what's wrong with education cannot be fixed with technology' [Oppenheimer 97].

Traditional teacher-centric approaches to learning do not transfer successfully to technology and must be revolutionised [Wedekind 98]. The philosophy must change from computers as teaching machines to computers as tools to empower learners and teachers [Oppenheimer 97]. The constructivist approach implies that students will learn more with a teacher than from a teacher [Newby 96] similarly, students will learn more with a computer than from a computer [Reeves 98].

Mindful learning is where flexible learning strategies are employed by the student in a deliberate way to tackle specific problems [Katz 96]. The use of a computer can encourage mindful and challenging learning [Reeves 98].

The application of theoretical and analytical knowledge is becoming a requirement in the majority of new jobs [Drucker 94] and this level of knowledge requires deep, reflective thinking [Reeves 98]. While traditional teaching has required the learner to initiate cognitive tools [Katz 96], the computer is a cognitive tool. As such, the computer can empower individuals to design their own representations of meaning during thinking, problem solving and learning and encourage them to actively reflect on what is being learned, and actively integrate new and existing knowledge [Reeves 98] , [Tuckey 92].

Constructivist techniques

The technologies of the World Wide Web provide a diverse range of techniques that can be used to satisfy these implications of Constructivism.

Learner as an individual

Customisation

- Modules can be combined dynamically to enable a different focus or to suit alternate user features such as learning styles.

- Learners should be allowed to be independent but should be able to avail of guidance when it is required.

- Tools should be provided which allow the learner to organise and present their knowledge thereby making the learner's understanding explicit. (timelines, graphs, maps, narrative sequence, concept maps, bookmarks, annotation)

- Individualized instruction should be optimised while preventing isolation of the student by providing feedback and opportunities for collaboration.

- Learners should be motivated toward an explicit overall objective and sub-objectives of which the learner is made aware.

- The subject material should not be over-simplified or over-organised

- The interactivity of the learner and content should be maximised.

- Analysis should be encouraged rather than the memorising of facts, for example, students should be encouraged to discover principles by themselves [Atherton 99] .

- The course sequence may require a response from the learner before continuing – the interactivity is enforced as the learner must provide some answer, for example, asking the learner to construct their own analysis or criticism before they are given hints or allowed to access the expert analysis.

- Students cannot progress to the next module until they have sufficient comprehension - this can be implemented with competence tests.

- Learners should be given the opportunity to practise, apply, test, and evaluate their knowledge [Wade et al 97]. For example, in an SQL course a user should be able to formulate their own queries against a database as part of the course.

Learner as a social being

- The student should be given the facility of interaction with instructors and other students. This may take the form of :

  - Correspondence (e-mail lists) where learners can make their ideas explicit by the written word.

  - Tele-conferencing where learners can make their ideas explicit verbally.

- A shared whiteboard where learners can make their ideas explicit in various media including words, graphics, video and sound.

- The technique of modelling, coaching and fading may be used whereby the teacher or expert demonstrates a task (modelling), assists the learner in completing the task (coaching) and then retracts their assistance as the learner masters the task at hand (fading).

- Project based learning where students are assigned a project to be completed as part of a team.

- Learner in an environment

- Case Studies can be used to introduce issues in the context of real world problems.

- The instructor should set problems whose resolution requires adaptation of new knowledge in a different context.

- The learner's knowledge should be generalised by applying what is being learned across a number of domains.

- Problems should be convincing.

- The instructor should attempt to provide a variety of stimuli and a variety of contexts [Hunt 97] , [Wade et al 97].

- Subject matter should be contextualised to demonstrate its applicability in real world scenarios and make it seem worth knowing.

- Learners should be given control of what information they access to stimulate innovation and discovery [Wade et al 97].

Summary

This chapter presented the development of educational theory with an emphasis on the current principles of Constructivism. The implications of Constructivism were discussed and the chapter concluded with a review of the constructivist compliant techniques that can be used in web based learning.

Chapter 3

ADAPTIVE HYPERMEDIA

Introduction

This chapter examines the nature of Hypermedia and its application in the educational domain. The function and methodology of adaptivity in Hypermedia is considered followed by an analysis of the pertinent characteristics of users required to construct a user model. The chapter concludes with a survey of current Adaptive Hypermedia systems.

Hypermedia

A Hypermedia system consists of Hyperdocuments. Hyperdocuments are nodes which have links through which the user can traverse to other nodes [Brusilovsky 96] , [De Bra 98]. This allows the information in the Hypermedia System to be accessed in a non-linear manner. The user can 'jump' along a link from one page to another regardless of the physical location in Hyperspace of the linked page [De Bra 98]. The content and link structure should be designed so that any possible path through the links is logically valid for the user.

The sufficient support of the variety of possible reading orders in a Hypermedia system is not a trivial task [De Bra 98]. The user may suffer from cognitive overload as she attempts to comprehend the expanse of unstructured information provided. Similarly she may become disoriented within the system and not know where the current node is positioned in relation to the node which contains the information she requires [Eklund 95]. This is a phenomonen known as 'Lost in Hyperspace' which is a frequent occurence in users who are not experienced with Hypermedia navigation or with large Hyperspaces such as the World Wide Web. Even experienced users become spatially confused in systems whose information structure is random and incoherent.

Users may also be inefficient in their navigation through Hyperspace, following redundant or sub-optimal references which extend the time it takes to reach their objective [Kayama, Okamoto 97]. Hypermedia experience is one example of the many differences that may exist between users of the same Hypermedia system.

Some Hypermedia systems are used by widely varying user populations with potential differences in many areas including language, hardware, software, backgrounds, aims, interests and abilities [Fink et al 96] , [De La Passardiere, Dufresne 92].

The first proponent of linking information together in this way was Vannevar Bush in his description of a Memex machine in 1945 which would aid a human by allowing him to create links to information which he may not be able to store in long term memory. These associative links could be referred to when required and thus provided a useful cognitive tool for the user [Vannevar 45]. Indeed the structure of Hypermedia systems and the non-linear linking of associated concepts and ideas has been compared to the structure imposed on information by the human brain.

Hypermedia in Education

There are a number of reasons why Hypermedia has been implemented in the educational domain. The analogy between the structure of Hypermedia systems and the cognitive structure of the human brain implies that the brain should be able to assimilate information more readily and the associative structure of the material may assist the development of cognitive flexibility which is a requirement of analytical thinking [Wedekind et al 98]. Humans naturally communicate multimodally and Hypermedia systems support the use of various media to transmit information. Users are given the freedom to browse through the educational content at their own pace and in order to meet their individual objectives. They may choose contexts, subject-matter and media to suit their own preferences and requirements.

The overall objective in educational systems is learning. This is a more abstract goal than that of finding information. The difficulty of comprehension and navigational confusion which are a feature of standard Hypermedia systems become more critical in an educational context. The effort and motivation required by a student to learn needs to be sustained over a longer period than in standard Hypermedia systems.

Cognitive overload and navigational confusion are frustrating for users and occur most often to inexperienced learners whose motivation may be the most tenuous. The freedom offered by Hypermedia systems is a double edged sword. Experienced users can refine their higher order thinking skills by engaging constructivist learning strategies and browsing the course content with their own conciously formulated agenda. The disorientation and confusion sustained by novice users can lead them to become disinterested and disengage from the learning process suffering from a sense of fatalistic helplessness [Atherton 99]. Users must have the experience and confidence to be able to take control and responsibility for their own learning.

Adaptive Hypermedia

Adaptive Hypermedia attempts to alleviate some of the difficulties encountered in Hypermedia systems by adapting the system individually for each user. The system collates information about each user into a user profile and this model is used to make assumptions about how best to change the system to benefit an individual user. The system may infer user objectives and help the user to discover the scope of information available or delineate a relevant path to get to the information required [De La Passardiere, Dufresne 92].

Adaptive systems infer the requirements of a user and modify the system accordingly. This introduces the problem of balancing control between the user and the system and the issue of the extent to which a user should be made aware of system made changes i.e. the transparency of the adaptivity.
The correctness of assumptions made by the system cannot be guaranteed. This argument implies that users should be able to control the system adaptivity. Adaptable changes are those which originate from and are controlled by the user. Adaptive changes originate from and are controlled by the system [Fink et al 96].

The system adaptivity may be hidden entirely from the user so that the user is unaware of changes made by the system on her behalf. Alternatively the adaptivity may be negotiated with the user, allowing the user to accept or reject modifications suggested by the system. The modifications may be visible to the user but the user may not be able to change them. For example a link which is visible as a link, but dimmed and inaccessible to the user.

Users should have some control over the adaptivity but should not have to control it continuously [Espinoza, Hook 95]. System designers must attempt to strike a balance between the control allowed to the user and the ease of use of the system. It is imperative that users should not be surprised, disoriented or displeased by the changes made by the system [De La Passardiere, Dufresne 92]. When the usability of the interface is in opposition to the potential effectiveness of the system, the designer must attempt to provide adequate balance.

So what features of the system are modifiable? The system may customise the link structure or format which is offered - this is known as adaptive navigation. Similarly the system may vary the content displayed - this is known as adaptive presentation. The system may adapt the modality of the content or the prominence of links or content. Orientation aids and search facilities may be included, omitted or highlighted depending on the information contained in the user model and the rules used to apply the changes [Kay, Kummerfeld 95] , [Fink et al 96].

Adaptive Hypermedia in Education

The problems encountered by users in conventional Hypermedia systems are shared by users in the educational domain. Adaptive Hypermedia is used to alleviate cognitive overload and orientation diffculties as in other domains but it is also implemented in an attempt to solve more pressing problems specific to the educational domain. As with any group of Hypermedia users, learners may vary in their Hypermedia experience and objectives. However there are particularly significant features of learners which influence the effectiveness of the use of Hypermedia in the context of education. These include preknowledge, cognitive style, learning style, maturity, general ability, confidence and motivation. These features influence the ability of students to accept effectively the additional mental load caused by the need to monitor and self-evaluate as well as learn [Specht 98].

Although increasing learner control is thought to increase the learner's motivation and engagment, results in performance using adaptively controlled environments have been superior to systems within which the user is left to their own devices [Specht 98]. Studies have shown that users of educational Adaptive Hypermedia systems are faster, more goal-oriented and take fewer steps to complete the course. Adaptive Hypermedia learners are less likely to repeat the study of content they have already covered [Eklund, Brusilovsky 98].

Representation of the Learner

A user model contains explicitly modeled assumptions which represent those characteristics of an individual which are pertinent to the system [Fink et al 96].

The system can consult the user model it has constructed to adapt the performance of the system to that user's characteristics [UM 99]. User modelling allows the computer to create a personalised dialogue with the user and to share a context which should promote effective communication [Lai 98][Eklund, Sawers 98].

The Stereotype Model

One of the simplest ways to model a user is to create fixed stereotypes of the expected users of the system. New users are categorised and the system will customise its performance according to the category which has been set for that user. A common example of a stereotype would be a novice user where the other stereotypes are intermediate and expert users. The advantage of this approach is its simplicity, the disadvantage is that it constitutes a limited and inflexible representation of the user.

The Overlay Model

The overlay model is widely used in adaptive hypermedia systems in the educational domain. A model of the user's knowledge is constructed on a concept by concept basis and updated as the user progresses through the course. This allows for an independent and flexible model of the user's knowledge for each topic [Brusilovsky et al 96a]. The estimation of whether a user knows a concept may be inferred from information acquired directly from the user or judged on the basis of whether the user has read that section or completed a test on that concept.

The knowledge domain must be modularised into specific topics or concepts. The complexity of the model depends on the granularity of the structure of this domain knowledge and the granularity of the estimation of the user's knowledge. The granularity of the estimation may vary from a binary value being stored for each concept - known/not known, to a quantative estimation - good, average, poor, to a qualitative estimation - there is a 70% probability that the user knows this concept.

The model may be made visible to the user so that the user can see the learning path and the learning achievement in parallel [Lee, Jong Wang 97]. This visibility may also serve as a navigational aid to the student so that they can see what they have covered and to what extent. The model may be made adaptable by allowing the user not just to see the model but to change it. In this scenario, the system must trust that the user has the expertise to leverage the model effectively.

The Combination Model

The Stereotype and Overlay techniques of user modelling are often combined in educational adaptive Hypermedia systems. The user may be categorised by stereotype initially and then this model is gradually modified as the overlay model is built from information acquired from the user's interaction with the system.

Building the User Model

There are a number of sources for the information which will be used for the construction of the user model. The system acquires data about the user and infers user characteristics from this data. The validity of the asssumptions depends on the technique used to acquire the information. Automatic modelling by the system may be unreliable. Any inferences made by the system about user characteristics are ultimately a guess [Espinoza, Hook 95]. For this reason collaborative or cooperative modelling is frequently implemented. The user describes pertinent characteristics to the system directly. While the construction of the user model must be as effective as possible, it is also desirable that a balance be accorded between the extent of the involvment required of the user and the extent of processing required of the system.

Users can provide feedback to the system directly by filling out questionnaires or forms. Indirect feedback is acquired from the results of exercises or problem solving tasks set by the system. The system may record the mouse clicks and keyboard strokes of the user to track their navigation path through the system. Requests for help from the system may be recorded.

Similarly the system may record the time spent at each node or page of the course. Changes made by the user to adapt the system may also be incorporated into the user model.

User Properties

The designer of the educational system must decide what features of the user are pertinent to potential customisation by the system. As with the Overlay technique of user modelling, these characteristics may be described in a binary, qualitative or quantitative model. These characteristics are inputted by the user and/or inferred by the system. Some examples of user features which have been used in educational systems are Novice/Expert, Adults/Children, Hurry/Browsing, First Time/Revision, Language, Technology, Disability. A selection of user characteristics are presented which may influence the way in which a learner uses an educational system, these are the user's objective, preknowledge, cognitive style, learning style, maturity, general ability, confidence, motivation, preferences and background.

OBJECTIVE

The objective or goal of the user is a description of what the user is trying to achieve. This may be inferred by the context of the content. The system infers that the solution of a problem or the completion of a task that has been set is a local objective for the user. The system may infer that when a user follows a specific path it is because the immediate objective is to learn the piece of knowledge to which the path leads [Eklund 95]. The global objective of a user will be to learn the contents of the course. One individual may be a novice wishing to aspire to intermediate level, another may be an expert who is revising.

The system can tailor the presentation to meet the differing objectives of both users. The user may have to specify their own objectives explicitly and are therefore made more aware of their own learning objectives. Users of adaptive Hypermedia systems have been shown to be more goal-oriented.

PREKNOWLEDGE

The rate and manner in which a learner assimilates knowledge is dependent on the learner's previous knowledge of the subject matter.

Adaptive systems need to gauge the level of prior knowledge of the student and then monitor the user's mastery of concepts and build upon the knowledge acquired by the student as they progress through the course. Direct feedback or test results may be used to infer the knowledge of the user at the start of the course. The system should then recognise changes in the user's knowledge as they progress and update the user model accordingly. Support can be gradually phased out as the student's knowledge increases. [Paolucci 98]

The system should ensure that prerequisite concepts are known by a user before she advances to a new topic. Technical terms should be avoided or explained until the learner is familiar with their use. Hypermedia experienced learners are known to be more likely to navigate in a non-linear way. Similarly learners who are familiar with the subject matter are more likely to navigate non-linearly and therefore reap the benefits of Hypermedia learning [Eklund 95].

COGNITIVE STYLE

The computer can be used as a cognitive tool to develop higher order thinking skills. Learners who learn by associating and linking different ideas and information will be more effective at learning in a Hypermedia based system. Such learners think, perceive and solve problems in an active, exploratory manner. They exercise strategic analysis of the meaning of the subject matter [Dao, Parent 98].

Some users are more comfortable with the use of words and equations – this is known as a verbal cognitive style.
A user who prefers the use of symbols and images exhibits a spatial cognitive style. Learners may think about problems in an abstract or concrete manner with corresponding emphasis on analysis or intuition [Galitz 97].

Active learners who are confident in their learning strategies regardless of the subject matter are called field independent learners.

Cognitive styles which are critical and field independent are more conducive to the effective use of educational hypermedia [Paolucci 98]. Educational hypermedia systems need to allow for different cognitive styles and attempt to nurture an analytic cognitive style in users of the system [Dao, Parent 98].

LEARNING STYLE

The learning style of a particular user changes depending on the time and context and mood of the learner. The factors which may affect learning style include the user's physiological and psychological state, the prevalent cognitive style of the user and their prior experience of Hypermedia in general and the course content in particular [Paolucci 98]. Field independent learners are better hypermedia learners. Learners may have a holistic learning style and wish to understand the context of the material they are learning. A learner with a serial learning style may tackle a new subject with a step by step approach [Dao, Parent 98]. These learning styles correlate with the specific and generic mental models which are constructed in the mind of learners. It has been claimed that learners may be able to process information more effectively if it is presented in a manner that is closely matched by their learning style [Paolucci 98].

MATURITY

Hypermedia systems do not teach but provide the student with the means and opportunity to learn of their own accord [Eklund 95]. The student needs to have acquired the skills to apply appropriate learning strategies to each learning task [Laurillard 95]. The student must be able to take responsibility for their own learning and organise the learning environment in a manner that suits their own learning style [Paolucci 98].

Adaptive Hypermedia systems need to take account of the sophistication of the learning skills of the user.

GENERAL ABILITY

The intelligence of the user is a factor in their ability to assimilate information. An adaptive system can take the general ability of a user as sourced from Grade Point Average or Psychometric testing and use this as a measure of how much guidance and support the user may require [Paolucci 98]. The aim should be to give the user as much freedom and independence as they are capable of managing effectively.

CONFIDENCE

Users who are inexperienced in a Hyperspace environment or who are new to the subject matter of a course are more likely to exhibit sequential behaviour i.e navigating forward and back only [Eklund, Brusilovsky 98]. These users do not digress from a linear path because they are afraid of becoming disoriented and confused. Adaptive systems should provide security for these users to embark on non-linear paths and be confident that they will be able to cope and find their way to the information they require.

A common feature of learners who are navigating in a non-linear way is the lack of a feeling of closure when a task or course is completed. Because there are so many possible paths, the user is not confident that they have covered the entire information space. Adaptive Hypermedia systems can display the progress of individual users so that they know what has been covered and where. Users can then have the confidence to use greater initiative in their navigation [Eklund 95].

MOTIVATION

Sources of motivation are as varied as the users themselves. There are extrinsic motivational factors such as exam results, degrees or certificates and improved career options. Intrinsic motivation is crucial to the Constructivist learning process and can be improved by maximising a learner's control and independence. However novices may lose motivation if they are given too much freedom and their confidence deteriorates. An adaptive system should maximise motivation by emphasizing interactivity and providing feedback to the learner. It should also provide guidance and support when required. For example intelligent online help systems which sense if the user is having difficulty.

PREFERENCES

Users preferences should also be incorporated into the user model. These preferences are an adaptable rather than adaptive change to the user model and may include preferred colours, preferred technology or media.

BACKGROUND

The user's background may include their profession, work experience, beliefs and hyperspace experience. The user would ordinarily supply this information directly and the system may operate a stereotype model for targetted users.

Methods of Adaptivity

There are several approaches to implementing adaptivity in Hypermedia Systems. Adaptive methods whose application is usefully transferred to the educational domain are detailed in the next section. The methods described include methods of adaptive navigation, structural and historical adaptation, adaptive presentation and customised help.

Adaptive Navigation

Adaptive Navigation attempts to guide the the user through the system by customising the link structure or format according to the user model. The adaptive navigation employed will direct the level of guidance given to the learner and the corresponding measure of freedom that is granted [Specht 98]. Adaptive navigation should control ineffective navigation by novices without restricting expert users [Eklund 95]. A number of user properties will influence the navigation freedom allowed including familiarity of the subject matter, the Hypermedia experience of the user and the maturity of the learner. Similarly the complexity of the material and the usability of the interface must be considered [Specht 98].

RELEVANCE

Link adaptivity may require the system to decide on the relevance of certain sections of the course content to a particular user at a particular time. This decision is reached based on the information in the user model. As an example the decision may be based on the current objective or goal the system has inferred for the user. If a link connects to information which is not required to meet the current goal, the link may be marked as irrelevant. Similary, the concept to which a link is connected may require knowledge of concepts that the user has not yet covered. These links may be marked as irrelevant at this time.

DIRECT GUIDANCE

Direct guidance is provided by the system deciding where the user should link to next and presenting the user with this option. This is also called curriculum sequencing as the system enforces a path through the course. This path is customsied for that particular user but the advantages of Hypermedia are lost when the user cannot organise their own learning through the Hyperspace. When link annotation and direct guidance are offered together, users who are not confident of their ability to work through the course independently are more likely to click on the next button and accept direct guidance [Eklund, Brusilovsky 98].

LINK ORDERING

Link ordering is when the system sorts a list of links according to their relevance to the user. The system filters the links on the basis of the user model and presents the list with the most relevant links displayed at the top. This type of link adaptivity is often used for indexes or table of contents. A user who is inexpereienced with the content of the course or with Hyperspace generally can be disoriented by a link order which is unstable.

LINK HIDING

Link hiding restricts the navigational choice offered to a user. The system decides what links are not relevant to the user and changes the format to that of regular text so that the link is not recognisable to the user as a hyperlink. The link may be removed completely so that the user cannot access it even accidentally [De Bra, Calvi 98]. Link hiding can reduce the cognitive load on the user and conceal the complexity of a course while supporting the stable ordering of links. However, the usability of link hiding is questionable for a number of reasons. Users do not like to be restricted. There is a danger that the user will form an incorrect mental map of the Hyperspace. A sense of completion of the course will be difficult to attain when the user cannot be confident that all the links have been displayed.

ADAPTIVE ANNOTATION

Annotation refers to adding information to a link so that the user has more of an idea of where the link will lead and whether it complies with the current objective of the user. Link annotation allows the user to be advised as to the degree of relevance the system applies to a link on the basis of the user model, and the user may then choose her own path. A link can have a number of different states the values of which may be displayed to the user by colour, icons, or font formats.

World Wide Web browsers currently use link states with two values - visited links (the default for which is purple) and unvisited links (the default for which is blue). Adaptive hypermedia systems can extend this idea to show links with three states to signify concepts that are learned, well-learned or unknown. [Eklund, Sawers 98] propose links with four states: visited, unvisited, current and suggested. Links with up to six states have been implemented in Adaptive Hypermedia systems. The link may be changed to a light colour to suggest that the link is dimmed - this gives the effect of hiding without restricting the user. Annotation gives the user a degree of freedom and supports stable ordering and the formation of correct mental maps [Eklund et al 97].

Structural Adaptation

Structural Adpatation [Eklund 95] attempts to give a spatial context [De La Passardiere, Dufresne 92] to the user by adapting visual representations of the structure of the Hyperspace environment on the basis of the user model. The system attempts to cater for differences in spatial cognition by aiding the user in her comprehension of the organisation of the hyperspace and her physical position within it [Espinoza, Hook 95]. Overview maps, local maps, fisheyes, filters and indexes are all structural aids which the system may adapt for the user.

Historical Adaptation

Historical adaptation attempts to give a time context to the user by adapting representations of the user's path through the system [Eklund 95]

History trails, footprints which are marks made by the system, landmarks which are marks made by the user and progression cues [De La Passardiere, Dufresne 92] may be customised by the system for each user.

Adaptive Presentation

Adaptive Presentation is the customisation of the content of the course to match characteristics specified in the user model. The granularity of the content adaptation may vary from words in the text to the loading of different pages or the application of different media. Explanations may be customised to contain additional information, pre-requisite information, or comparative explanations. These variations of explanations for concepts exist to suit differing levels of knowledge or insulated stereotypes.

The adaptivity may be implemented with conditional text where text is fragmented into words, phrases or paragraphs which constitute a discrete unit of information about a concept. The text unit is displayed if the user model conforms to the required conditions for the display of that text unit. For example, if a prerequisite concept has not been covered by the user, the text explaining that concept may be added to the page when it is displayed. 'Stretchtext' allows additional text to be displayed on the page if the system deems it necessary. This extra text can be collapsed if the user does not wish to see the extra information or expanded if the system decided to provide the page without the display of additional information.

Different pages or fragments of text may be displayed for different users. An example would be a technical term with which the user is unfamiliar. If the user has not come across the technical term, a non-technical substitue phrase will be used until the user has discovered the node which contains a definition of that term. Thereafter the technical term will be assumed to be understood by the user and will be included in normal text. Similarly an acronym may be fully described the first three times the user comes across it and thereafter only as an acronym.

If the courseware is constructed dynamically each user may potentially see an individually tailored course that is different to the course displayed for all other users. Frame-based techniques adapt at a finer level of detail, varying the words and phrases used in the text. Natural Language Processing is required to ensure that the resulting text is coherent. These techniques may also be combined for example conditional text which is also stretchtext.

Customised Help

The help provided to the user can also be adapted. The extent of detail which is provided may depend on the user's previous knowledge or expertise or on the time that has been spent at a task. Learners should be helped to solve the problem not just given the answer. Adaptive Hypermedia attempts to provide only enough information for an individual to allow them to work it out for themselves. The help could give hints at a level of simplicity based on the user's profile.

Problems that users encounter may be caused by not understanding or not covering background information or prerequisite concepts. Adaptive help can list the links to concepts on which the current node depends so that the user can work out what needs to be understood before the goal or task with which she is having difficulty can be accomplised.

Survey of Adaptive Hypermedia Systems

InterBook

The purpose of the InterBook design is to simplify the process of creating adaptive electronic textbooks on the WWW. The precursor to InterBook was a WWW based LISP textbook called ELM ART.

InterBook uses concept-based indexing and an overlay user model to provide adaptivity. A domain concept is an independent unit of knowledge in a given domain. Concept based indexing is used to provide additional information about the content of each hypermedia page by indexing it with related domain concepts.

The glossary, which also functions as an index, is a central part of an InterBook electronic book. Each concept is an outcome concept or a prerequisite concept. An outcome concept represents the piece of text or other media that explains that concept. A prerequisite concept is a concept on which the current node is dependent.

Each concept has a glossary page and a text unit. So the glossary exhibits the same structure as the domain knowledge. Links connect each unit of text to the glossary for each concept. Links from the glossary to the unit of text which explains it are generated dynamically depending on the user model. The table of contents is also comprised of clickable links. Prerequisite-based Help gives support to the user when in difficulty by listing the prerequisite concepts required for that task or module.

Adaptive Navigation is implemented by link annotation using checkmarks and coloured balls.

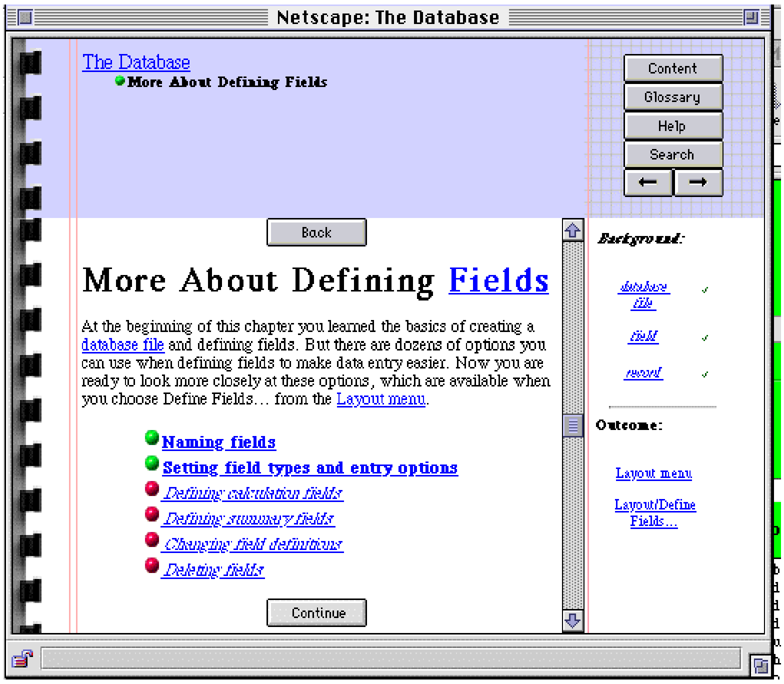


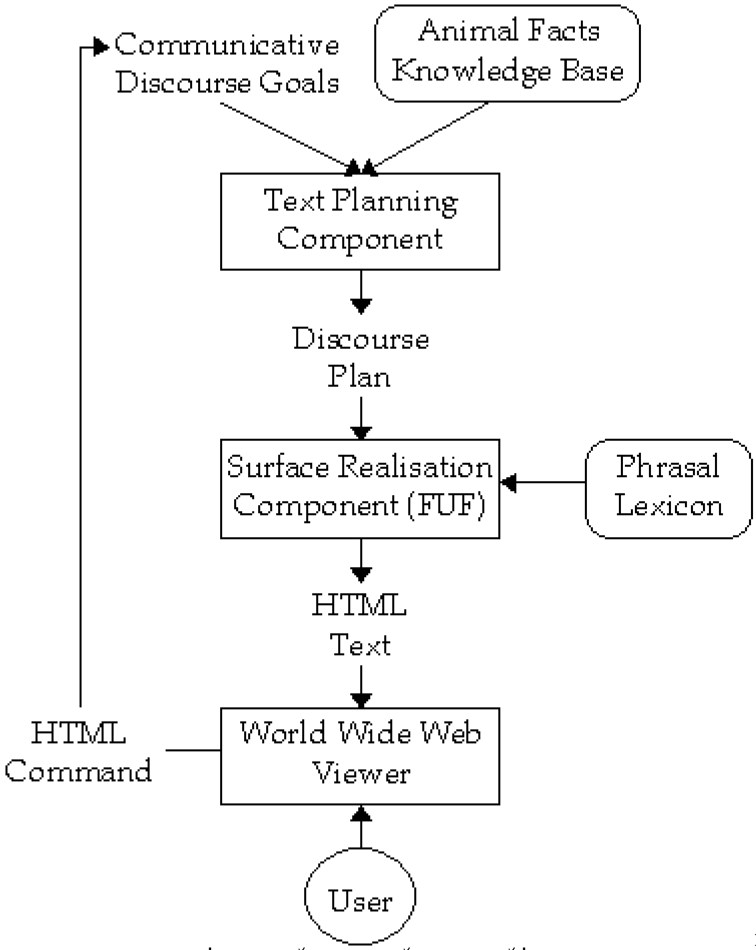
Figure 3 Screenshot from the InterBook system

A green ball and bold text designates the state of the link as ready and recommended which implies that all prerequisites are well-learned or at least learned. A red ball and italic text designates a link that connects to content that is not ready to be learned which suggests that some prerequisites are not yet learned. A white ball is used when the link connects to information that is not new implying that all outcome concepts are learned or well-learned. Checkmarks by a link show that a link has already been visited. If the link is small then the content behind the link is known implying that learning has started. A medium checkmark designates content that is learned, while a large checkmark indicates content that is well-learned.

The technology used includes HTML, LISP and the Common LISP Hypermedia Server (CL-HTTP). The adaptivity is implemented in a relatively cheap and useful way [Brusilovsky et al 96b] [Brusilovsky, Schwarz 97] [Eklund, Brusilovsky 98]. Annotating links as undesirable or forbidden may be an unattractive option for confident users who wish to learn by exploring and do not wish to be guided. Link annotation should be intuitive and unobtrusive, these conditions are not fully achieved in the InterBook system.

PEBA-II

PEBA-II is a text generation system that dynamically generates description of animals in an interactive hypertext environment.



The system uses [illegible] ndividual user. The fragments into which the data is divided are called slots which are assembled into a node. The system decides what slots to present and the order into which the slots should be organised on the page. Natural Language Generation aims to produce coherent language text from an underlying representation of knowledge. Natural Language techniques are used in PEBA-II to ensure that the resulting page of text sounds natural to a human reader.

Comparison is used in the PEBA-II system to distinguish or liken new concepts to known concepts thus building on the knowledge that the user already has and building on the knowledge that the user is creating by following the course. The systems identifies, compares and contrasts animals with other animals with which the user is familiar.

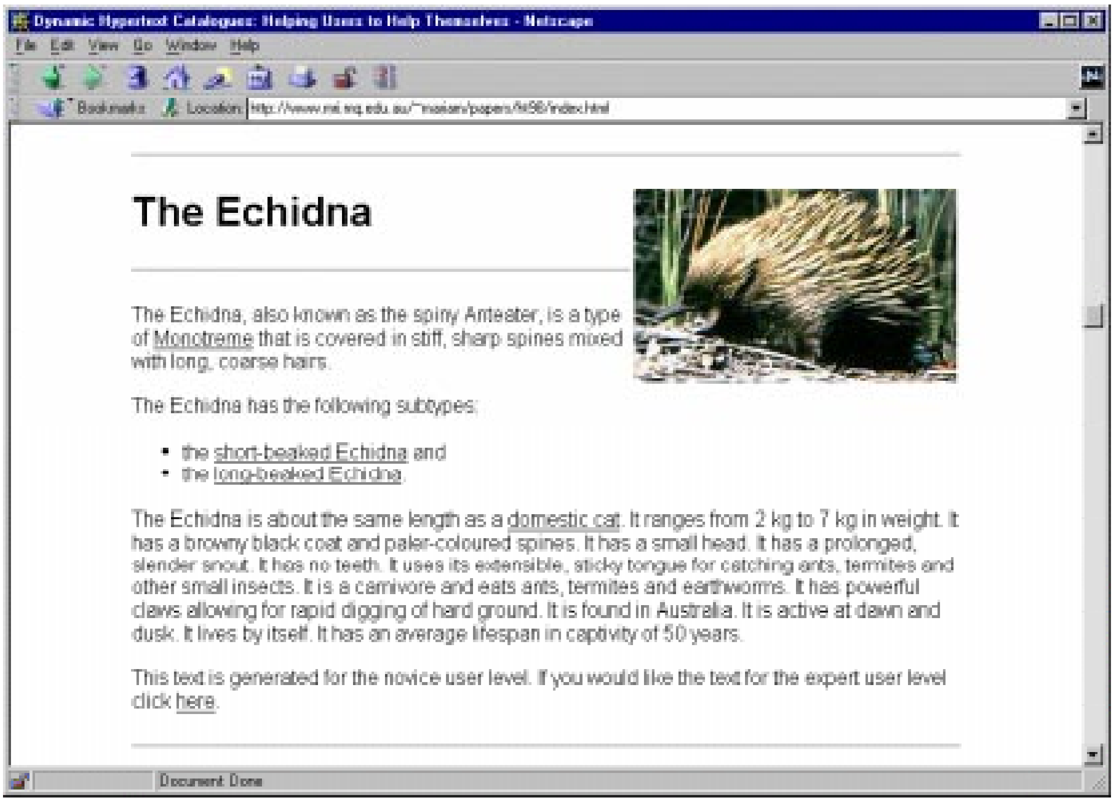


Figure 5 Example of dynamic content in the PEBA-II system

Technical words are not included for naive users. The user model is initialised by a stereotype of expert or naive user and an overlay model specifies what animals the user already knows from those indicated as known by the user and the animals and concepts covered in the course i.e. the discourse history. The system only marks as known those animals or concepts that were recently encountered. The fact that a user may forget is therefore taken into account.   PEBA-II uses a phrasal lexicon so that words to short phrases to long phrases can be decomposed as required [Milosavljevic 97][Milosavljevic et al 97][Milosavljevic, Oberlander 98].

The use of comparison is interesting as shown above where the length of the Echidna is compared to the length of the domestic cat for a naive user. Expert users would be given the length in centimetres. The implementation of Natural Language techniques gives greater flexibility in the granularity of text fragments. The cost of this however, is complexity.

AHA Adaptive Hypermedia Architecture

In AHA, the user model is constructed of user preferences indicated by the user directly, user knowledge initialised by stereotype and an overlay model which consists of boolean variables - true if a concept is known and false if the the concept is not known. The system also allows a simple text export and import of the user model.

Adaptive Navigation is implemented by links with three possible states - desired, undesired and uninteresting. The standards colours of WWW browsers are used - blue links are desired, purple links are uninteresting which implies the information has been visited and does not represent new information to be learned and dark grey links indicate undesired information for which prerequisites have not been covered. This is similar to dimming the link and is a user friendly alternative to disabling. Links may also be hidden in the text, disabled or removed entirely.

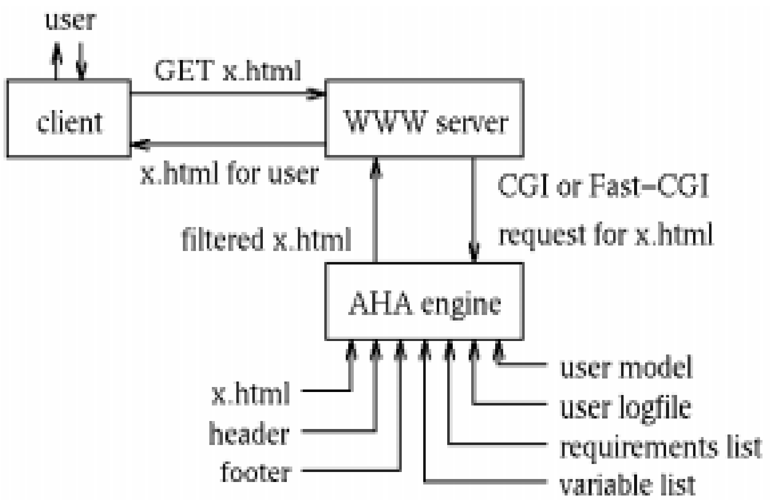


Figure 6 AHA Architecture

Adaptive Presentation is implemented through conditional inclusion of fragments depending on the user model. Alternative presentation and hiding of text is also used depending on the inferred knowledge of the user.

The system will switch to verbose mode for a novice user who has just started to use the system. A concept explanation is included in the current page if it is a prerequisite concept for that page which has not been covered.

Similarly if a technical term has not been covered a substitute will be used. For example the word ‘page’ is used instead of the term ‘node’ until the description of a node is encountered by the user and marked as encountered in the user model.

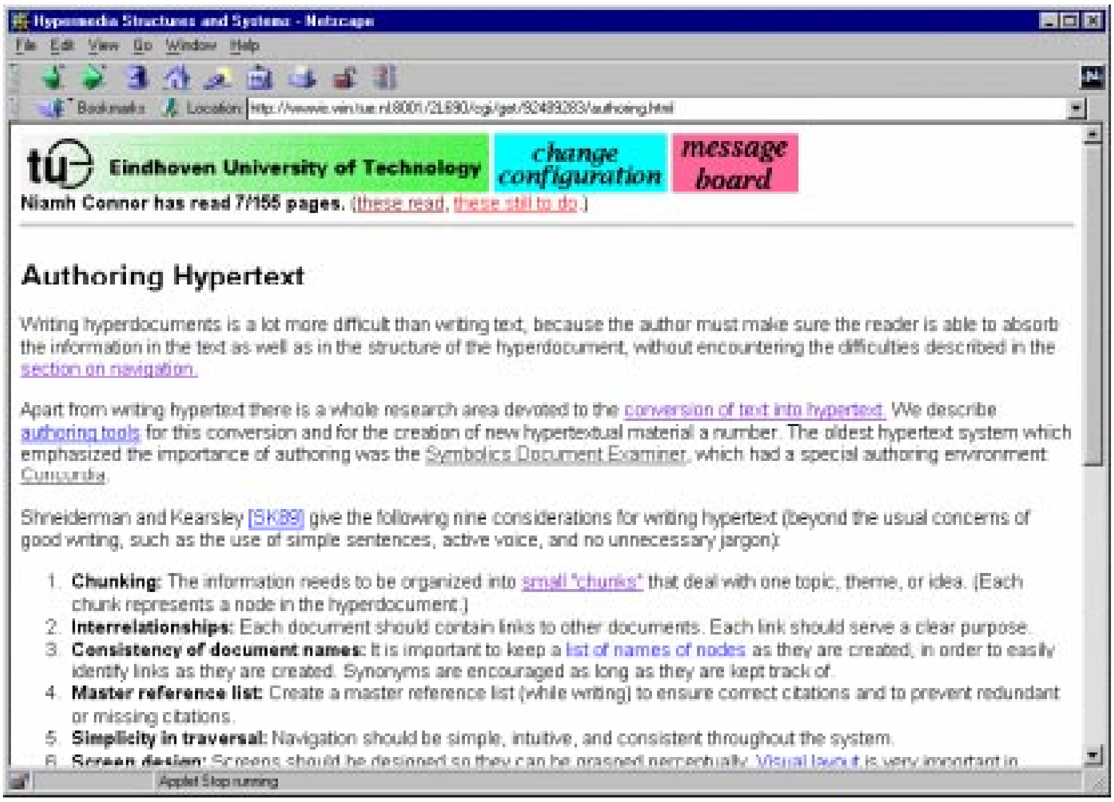


Figure 7 Screenshot of an application based on AHA

The technology employed is conciously limited to standard Web languages and tools - structured HTML comments and the CLASS attribute to links supports conditional fragments. The AHA system is still under development - CGI and Fast-CGI scripts have been implemented while Java-Servlets are being investigated [De Bra, Calvi 98] , [De Bra, Calvi 97].

The usefulness of the adaptivity in AHA must be measured in the context of the complexity of creating and maintaining the system. AHA offers substantial configuration options which allow users to register their preferences (for example the colour of link states) and even change the overlay model directly. This requires a considerable degree of trust in the competence of the user to monitor their progress and direct their own adaptivity. The configuration process also introduces a learning curve in the initial use of the system.

Summary

This chapter examined the use of Hypermedia and discussed the value of adaptivity in the educational domain. The requirements of user modeling in adaptive systems were reviewed and a number of Adaptive Hypermedia systems were described.

Chapter 4

METADATA STANDARDS AND APPROACHES

Introduction

This chapter describes the current effort to produce standards for metadata in the educational domain. The organisations involved and their individual contributions are detailed. The chapter concludes with a review of the approaches and technologies which support metadata.

Standards and Specifications

CEN

CEN (Committee for European Normalisation) is one of three European standards bodies along with CENELEC (European Committee for Electrotechnical Standardization) and ETSI (European Telecommunications Standards Institute) which make up the European standardisation system. These standards bodies produce voluntary technical standardization at the request of industry or the European commission [CEN 99].

IEEE

IEEE (Institute of Electrical and Electronic Engineers) provides a forum for harmonizing proposals from different working groups on the same problems. Proposals that have been approved by IEEE will be carried forward to the more rigorous process of the international ANSI or ISO standards.

W3C

W3C is an international consortium whose purpose is to develop common protocols that promote the evolution and interoperability of the World Wide Web. W3C establishes specifications which constitute industry standards.

Metatdata for Educational Systems

Metadata

Metadata is machine understandable information for the web [W3CM]. A metadata element describes a resource or is associated with an object that describes that object [NORD]. A resource constitutes an entity that can be either used or referenced on the web i.e. ‘anything that has identity’ [RFC 2396]. The volume of data on the web has become too vast for human management. The use of metadata to describe the meaning and structure of information to machines should enable the automation of the management of web resources [W3CR]. The use of metadata in education should ease the location, use and management of educational resources.

Metadata Standards

The transfer of learning technology from LAN or CD-ROM to the web represents a transition from a closed environment to an open environment. Proprietary solutions are suddenly a distinct limitation in terms of interoperability and extensibility, key requirements for open environment applications [Richards 98].

Metadata specifications provide guidelines in the implementation of metadata to promote the interoperability and extensibility of metadata applications. Providing the descriptive information provided by metadata in a standard way will allow ease of use, location and management of resources especially in large systems such as the world wide web.

A ‘specification’ differs from a ‘standard’ in that a standard originates from an accredited standards body and is produced by following a specific, open process [Richards 98]. Specifications tend to be drawn up with a minimum of bureaucracy within industry. Specifications are used to draw up actual standards.

Dublin Core

Dublin Core is a generic metadata specification for all domains originally proposed by the Dublin Core Metadata community in December 1996. It has been continually updated through a series workshops attended by experts invited from various domains involved in metadata related activity [Weibel 99].

The Dublin Core model consists of 15 metadata elements which are interoperable, easy to create and simple to index [NORD]. Version 1.1 has recently been announced as a proposed recommendation of the Dublin Core Metadata Initiative which indicates that it is a stable specification likely to be adopted by the Dublin

Core community. The metadata elements are Title, Creator, Subject, Description, Publisher, Contributor, Date, Type, Format, Identifier, Source, Language, Relation, Coverage, Rights [Dublin Core 99]. More and more information services on the Web are using Dublin Core as the basis for their descriptions of Web resources [MMI-DC 99].

Metadata Standards in Education

IMS Instructional Management System

The IMS specification describes metadata as 'descriptive labels used to index resources for use' [IMS 99]. The term use includes management, discovery and delivery.

The primary objective of the IMS metadata model is 'to promote and facilitate the discovery and retrieval of appropriate learning resources' [IMS 99]. The standard attempts to ensure interoperability by describing a metadata model and to enable extensibility by supporting the evolution of the metadata structures it proposes. The set of metadata and the rules of syntax and semantics which are to be applied to a learning object is known as a Schema.

Educational resources have properties which can be described by metadata elements which in the IMS specification are called tokens. The IMS specification fixes the name and organisation of these tokens to allow the interoperability of educational systems. This should also allow educational resources to be found, managed and reused efficiently. Some of the metadata elements described which are relevant in the context of user modelling and adaptivity are Level, Difficulty, Discipline, Educational Objective, Prerequisite, Presentation, Semantic Density, Language, Technical Format and OS requirements [IMS 99].

CEN/ISSS Learning Technologies Workshop

A Learning Technologies Workshop set up by ISSS (Information Society Standardization System) in March 1999 was initiated by the European Commission to support the development of 'Europe's Learning Society' [CEN/ISSS 99].

Some of the proposed objectives of the workshop are to promote the adoption of appropriate standards in the area of technology assisted learning within Europe, to encourage participation in the creation of specifications which may develop to become standards, to ensure that global standards meet specific European requirements and to propose recommendations for localised requirements within Europe when global standards do not adequately address European issues [CEN/ISS 99].

CEdMA

CEdMA (Computer Education Management Association) is a professional association of education managers in hardware and software manufacturers.

A forum is provided for managers and directors of computer education to discuss the computer training issues. They are striving to collaborate in the promotion of 'excellence in Education, Training & Learning' [CEdMA 99]. An international organisation like CEdMa can play an influential part in the adoption of learning technology standards.

MMI Workshop

The MMI (Metadata for Multimedia Information) Workshop provides an open
forum for collaborative discussion on the subject of metadata for multimedia information [Richards 98]. Their work has included researching the requirements of metadata for multimedia information, recommending a metadata model on the basis of this research, providing information, guidelines and advice to the public and commercial sectors and cooperating and sharing information with other groups involved in producing similar or complementary recommendations. The workshop completed its work in June 1999 but a new Workshop has been organised to deal with the Dublin Core Metadata set. A draft business plan has been formulated outlining the objectives of the 'MMI-DC' workshop.

The scope of this new workshop includes an investigation into the requirements of implementing Dublin Core Version 1.1 in a European context including multilingual issues, and promoting awareness of the Dublin Core specification in Europe [MMI 99].

ADL

ADL (Advanced Distributed Learning) is an American Department of Defence initiative to provide guidelines on the creastion of effective largescale learning environments. The specification concentrates on four areas: vision, products, technical process and collaborative forum. 'Vision' involves the research of potential learning and job environments of the future. The area of 'Products' will investigate the content and tools of learning that are available now, what is required now and what should be developed to provide a distributed learning environment in the future. Technical process refers to the 'enablers that make advanced distributed learning possible' [ADL 99]. 'Collaborative Forum' will allow discussion and interaction on the goals of the initiative. ADL uses IMS specifications [IMS 99].

The ADL initiative expects to 'stimulate and take full advantage of progress being made in knowledge management systems and technologies that enhance learning and performance -- anywhere, anytime' [ADL 99]. This should improve the application of education, not only within the Department of Defence but in American schools, universities, corporations and in the public domain.

ARIADNE

ARIADNE (Alliance of Remote Instructional Authoring and Distribution Networks for Europe) is a European Union research and deveopment project which promotes the sharing and reuse of educational resources. Its purpose is to harmonise education and training policies throughout Europe by creating an infrastructure for communication and collaboration.
It has developed a metadata structure by extending the Dublin Core element set to 70 elements grouped in 9 different categories. The infrastructure for this metadata takes the form of an interconnected system of 'Knowledge Pools'. The Knowledge Pool System is a distributed database of educational documents and their descriptions [Duval 98].

Ariadne proposes tools for the creation of courseware from scratch and the construction of courseware from components already available in the Knowledge Pool System. Ariadne has been collaborating with IMS within the forum of the IEEE LTSC project. It is also working with the CEN workshops initiated by the European Commission.

AICC

AICC (The Aviation Industry CBT Committee) is an international association of technology-based training professionals. The AICC provides guidelines for the development, delivery, and evaluation of CBT (Computer Based Training)
The scope of the committee includes the promotion of effective computer based training strategies in the Aviation industry, the elaboration of guidelines to enable interoperability, and the provision of an open forum for the discussion of training technologies [AICC 99]. The emphasis of AICC specifications is on the Aviation industry. IMS and AICC are working together under a memorandum of understanding to create consistency in their approaches [Newman 99].

LOM

LOM is the Learning Object Metadata Working Group of the IEEE Learning Technology Standards Committee. The LTSC describes metadata as 'information about an object , be it physical or digital' [LTSC 99]. The LTSC working group recognises that while the need for learning is expanding the lack of metadata for learning resources has created a 'critical and fundamental constraint on our ability to discover, manage and use these objects' [LTSC 99].

A Learning Object is defined as any entity which can be used or referenced during technology supported learning. The metadata set of a learning object is the minimal set of properties required to ease their management, location and evaluation. These properties may be obligatory, optional, conditional or not allowed. The type of properties ascribed to objects include author, owner, format, object type and terms of distribution. Pedagogical properties, such as prerequisites and teaching style, may also be included if appropriate.

The Learning Object Model metadata maps directly to the Dublin Core metadata element set and has incorporated information from both the ARIADNE and IMS projects. The model comprises a set of 'tokens' which are used to describe the properties of learning objects in the context of security, privacy, commerce and evaluation.

PAPI

PAPI is the IEEE Public and Private Information Specification which is a standard format for the representation and communication of student profiles. The purpose of the specification is to allow the creation of student records which can be communicated between educational systems over the lifetime of a learner.

The profile information for a learner is divided into four areas.

1. Personal Information which is for private consumption such as the student's name, address and Social Security Number.

2. Preference information which may be for public consumption, such as the technology available to the student, the learning style of the student, physical limitations or disabilites. This information is collected with the cooperation of the student, i.e. it is negotiated.
3. Performance information which is for consumption by technology. This consists of the observable behaviour of the student and may include grades, reports, logs.

4. Portfolio information which is for consumption by humans, such as the student's accomplishments and works.

The PAPI specification also incorporates the Dublin Core metadata element set.

The information used to construct the user profile is inferred by the system, directly input by the user or is constructed by the user and system in collaboration. PAPI will also discuss the privacy and security issues involved in the storage and communication of user profile information. It is as yet incomplete.

GESTALT

GESTALT (Getting Educational Systems Talking across Leading Edge Technologies) which is funded by the ACTS programme, is an educational environment for online learning which extends the IMS metadata definitions. A user profile is constructed from information acquired from the user by asking the user to complete forms displayed by a wizard. The profile information stored includes the user's educational history and technology available to the student.

Some of the profile details acquired include personal details, contact details, qualification details, skill details, learning preferences and mode of delivery. The learning preference is stored as a boolean value of 'Yes' or 'No'. The user profile is created as an XML document that will be stored physically on the user's machine. Profiles.dtd is the Document Type Definition which defines the structure and format of the user profile. The IBM XML parser is used to parse the XML files and Internet Explorer 4.0 is used as the client browser as it supports version 1.0 of the XML specification.

Metadata Technologies

Markup

Markup refers to anything added to a document which adds special meaning or provides extra information, for example the date of publication or author of the document. To be useful the meaning of the markup must be known to the user of the document. A human reader looking a description of a document in a foreign language would need to understand the foreign language equivalent of the word 'Title' to make sense of the document description.

Similarly a machine needs to be able to make sense of document markup that it encounters. A set of rules must be defined which represents the meaning of the markup unambiguously. These rules must declare what constitutes markup and what the markup means. Markup may provide stylistic information for the purpose of presentation, structural information to represent the document's structure or semantic information to describe the meaning of the content.

Tags delimit the content of an element so for the example <*author*>Naomi Wolf<*/author*> , <*author*> is the opening tag, <*/author*> is the closing tag and 'Naomi Wolf' is the content. An element of markup comprises the tags and content all together.

A tag may have a property called an attribute. A font may have a colour, for example <*FONT COLOR*=Red>. *COLOR* is the name of the attribute while 'Red' is the value of the attribute. SGML is a markup language that is used to create other markup languages. HTML is an example of a language written in SGML i.e HTML is an SGML application [Boumphrey et al 98].

SGML

SGML is the Standardized Generalized Markup Language which became the ISO889 standard for defining markup languages in 1986. SGML enables an author to provide formal definitions for each of the elements and attributes in the language so that authors can create their own tags which relate to their specific content. SGML is powerful but complex, the specification is longer than 500 pages. SGML documents require a DTD (Document Type Definition), which defines the Markup rules, to be included or sent with a document so that it can be interpreted.

HTML

HTML was an SGML application created by Tim Berners-Lee in 1991 to markup technical papers so that they could be shared across heterogenous platforms. The DTD for HTML is now stored in a browser so the DTD location does not need to be included in HTML files anymore. The infamous Browser Wars were precipitated by browser manufacturers such as Netscape and Microsoft creating their own tags supported only by the DTD included in their own browsers.

The application of HTML on the Web is limited for a number of reasons. The set of tags that can be used is fixed so authors cannot create new tags to meet their individual requirements. HTML is primarily concerned with the presentation of information so software is concerned only with how to render the document and has limited knowledge of what the data within it means or how it should be structured. If a user wishes to retrieve a subset of information from a web page, they must download the entire HTML document. As the demand for network bandwidth continues to increase, this comprises a considerable disadvantage.

XML

XML, the eXtensible Markup Language, is a data format for structured information interchange on the Web [W3CX][GMDX]. XML is a subset of SGML which has been developed under the auspices of the World Wide Web Consortium (W3C). XML is a metalanguage which supports the creation of a markup language with customised tags and specialized document types. This allows the structure in data to be represented in a text-based way so that it can be authored and interpreted unambiguosuly [Oracle 99]. The layout of documents is controlled by the use of stylesheets.

The XML 1.0 standard was recommended by the W3C in February 1998. XML frees web development from dependence on the inflexibility of HTML on the one hand and the complexity of SGML on the other. It allows platform and application independent exhange of information.

XML supports powerful linking functionality which allows bidirectional and multiway links as well as links to a span of text rather than a single point. It allows you to store or link to metadata. There are no predefined elements in XML so the format used to link to metadata is the prerogative of the developer.

DTD

DTD is a Document Type Definition which contains a formal definition of a particular type of document [W3CX]. It sets out the rules for the naming and organisation of element types. An XML parser can use this information to identify the element types and relationships within a user defined document type.

XLink

A link is an explicit relationship between two or more data objects or portions of data objects. In hypertext a link provides a connection to a resource. A resource is anything that can be accessed across the network. XLink previously known as the Extensible Link Language and XML-Link is one part of the Linking system used by XML.

The current working draft of XLink is derived from three other linking specifications - HTML, HyTyme used within the SGML specification and Text Encoding Initiative Guidelines also known as TEIP3. There is a separate specification to describe the linking mechanism which is XLink and another specification to describe the referencing of sections of documents which is called XPointer, the second component of XML's linking architecture. Both XLink and XPointer are described using XML syntax.

XLink provides many advanced linking features including bidirectional links, links that point to multiple locations and links that exhibit different types of behaviour. Any XML element defined by the author can be used as a link. An inline link is a link that serves as one of its own resources i.e the content of the linking element acts as a resource. Most links are therefore inline. A multidirectional link is a link that can be traversed from more than one of its resources. The user can go in any direction through the resources of the link. This is possible because XLink allows links to have one to many and many to one relationships with the resources to which they point.

An out-of-line link is a link whose content does not serve as one of the original link's resources. For example a document may contain a list of links to other documents for the purpose of managing links rather than for the links to be displayed for the user. Application processing software is the intended user of out-of-line links. A simple link uses the HREF attribute as in HTML to point to only one resource. An Extended link is an XML link that can point to several resources and doesn't have to be an inline link.

The user will be offered a choice in the form of a pop up menu and they can then choose to link to whichever resource they require or cycle through all of the resources in the list in any order. Extended link groups provide groups of links that can be unidirectional, out of line or inline links. Extended Link Groups can be used to store a list of links to other documents i.e. out-of-line links.

Actions by links include opening new windows, replacing views or cycling through windows all of which would previously have required Javascript. The *ACTUATE* attribute of a link is used to specify when a link should be activated. The values of *ACTUATE* are 'AUTO' in which case the link traversal is automatic and 'USER' in which case the link is not traversed until traversal is requested by the user for example by clicking on the link. The *SHOW* attribute of a link is used to specify the behaviour of a link. The values for the *SHOW* attribute are *EMBED* in which case the resource will be embedded into the current document, *REPLACE* in which case the resource will replace the content of the current window or *NEW* in which case a new window will be opened.

XLink provides the ability to dynamically assemble documents by embedding linked resources. The *ROLE* attribute of a link is used to tell the user agent the function of the link. This will allow the processing application to give the user more information about the link before it is traversed. This type of link annotation allows users to get more information about links before they traverse the link. The *BEHAVIOR* attribute of a link is used to tell the user agent what action to take when the link is traversed [Pitts-Moultis, Kirk 99] , [Boumphrey et al 98] , [Chang, Harkey 98].

XPointer

The XPointer specification describes the addressing of the internal structure of XML documents. XPointers allow the locators in XML links to point to specific places in resources. They can reference elements, attributes and other specific parts of an XML document.
XPointers will allow requests for fragments of documents rather than the whole document thus saving valuable bandwidth. The format of an XPointer is a list of location terms that begins with a keyword which specifies an absolute location.

There are a number of absolute locations available in a document. The first is the root element of the document, if there is no absolute location specified in the XPointer the root is assumed as keyword. The origin refers to the original destination of a link traversal - this is only useful for linking within the same document. As in HTML any element with an ID or a NAME attribute is also available as an absolute location. The XPointer is read from left to right starting with the keyword and then with each following location term described in relative terms to the last. The relative terms are described in relationships such as child, ancestor and preceding or following siblings.

The general format of an XPointer is

 # [absolute location] . relative location ([integer] , [Node Type]).

An example given in the working draft of the XPointer specification is

child(3,DIV1).child(4,DIV2).child(29,P) which designates the 29th paragraph of the 4th subdivision of the 3rd major division of the location source [Boumphrey et al 98][Pitts-Moultis, Kirk 99] [Chang, Harkey 98].

DOM

Browsers which allow scripting implement a generic abstraction of a document which provides references for scripts and programs to access and manipulate those parts of the docment that are scriptable. In other words the object model exposes objects to scripts.

The window of a document, forms within a document and the URL of a document are examples of document objects described in the object model which can be manipulated by scripting. Browser support for object scripting has varied widely depending on the browser vendor, version and operating system [Goodman 98]. This prompted the W3C to develop a DOM specification to provide 'a platform- and language-neutral program interface that will allow programs and scripts to access every element in a document and update the content and structure of documents in a standard way' [W3C DOM]. The W3C DOM is expected to be implemented in all systems that require document manipulation including HTML and XML.

The core DOM specification describes a set of object definitions to represent a document and the objects that occur within a document, this includes the document itself, document nodes, elements and attributes of those elements. The XML DOM extends the core DOM specification to include additional objects specific to XML documents such as DTDs, entities and CDATA. The Java DOM core API and Java DOM XML API provide generic methods for manipulating documents and XML documents [Chang, Harkey 98].

RDF

Resource Description Format provides a framework for defining and using metadata to support interoperability between applications that exchange metadata on the web. Without RDF, applications would have to agree in each instance on the format prior to communicating metadata.

Providing a standard model of how to apply and communicate metadata allows metadata definitions to be reused and inherited. In an object oriented model a resource corresponds to an object and an object's property corresponds to an instance variable. Statements can be combined to create a description. RDF uses XML for its interchange syntax and namespace [W3CR].

Summary

This chapter has described the current push towards standardisation of educational metadata. This included a survey of the organisations involved and their contributions to the combined effort to produce an educational metadata standard based on consensus. The chapter concluded with a review of emerging metadata technologies.

Chapter 5

SYSTEM ARCHITECTURE

Introduction

This chapter proposes an architecture to support metadata driven adaptive educational courseware. The issues which must be addressed include the appropriate storage and retrieval of the data and the access of the metadata. The description of the design will include the elements of adaptivity that will be implemented, the proposed methods of adaptivity and finally the means of presentation to the user.

Storing the Data

Persistence Engine

A persistence engine is used to store data and may take the form of a relational database, object database or a file system. A database will be used to ensure data integrity and an appropriate level of performance.

The Relational Data Model

Relational database systems are the most dominant database systems in use today [Chang, Harkey 98]. The tabular format of the relational data model is simple to use and understand and provides a high degree of data independence. Stuctured Query Language which is the database language for data definition, manipulation and control within a relational database, is widely supported across vendors. Object oriented databases are an emerging technology which add persistence to objects so that database objects appear as programming language objects. As the demonstration of this dissertation is data focused, a relational database is appropriate.

Course Structure

The course is comprised of topics. Each topic is comprised of concepts which constitute a unit. Features of a concept include the definition of a concept, a description of the concept, an example or illustration to demonstrate the concept, a description of the example or illustration.

Accessing the Data

Web Data Access

The web consists of a universal naming service, network protocol and interface; Uniform Resource Locator (URL), HTTP (Hyper Text Transfer Protocol) and HTML (HyperText Markup Language). These three components provide a universal infrastructure for developing client/server data access applications. The limitations of this infrastructure are the limited graphical user interface, lack of support for common data types and the fact that it is static and stateless. The web does not provide a direct mechanism for accessing data. There are a number of ways to circumvent this limitation.

CGI

The Common Gateway Interface (CGI) allows methods to be invoked in three ways: a URL in the address field of a web browser, a HTML anchor or an action in an HTML form. The web server will then invoke the CGI program which can return results in HTML encoded format which the Web Server will output as a standard HTML page. CGI programs can be slow and are written specificallly for each required application. Web server vendors do provide APIs (Application Programming Interfaces) which provide improved performance but such APIs are limited to the web server suppplied by that vendor.

Java

Java provides an ideal solution for developing web based applications.

Java can overcome the limitations of the web infrastructure without compromising platform independence [Chang, Harkey 98]. The Java Virtual machine is included in all current web browser and web servers so no plugins are required on the client.

APPLETS

Java applets enable data access to take place at the client. They can offer a more sophisticated GUI by deploying the Java AWT or Swing components in browsers which support them. Java applets may have performance problems as the applet has to be downloaded to the client. Processing on the client using data queries and retrieval necessitates an increased use of bandwidth

SERVLETS

Java servlets allow data access to take place at the web server. This precludes the need for any extra software on the client other than the web browser. The Java Servlet API is a standard Java extension so server side programs can be fast without compromising extensibility. All communication between the client and server is characterised as a Servlet Request or Servlet Response object. This encapsulation of data improves the scalability of systems which depend on servlets for client/server communication. Servlets also enable services to be provided to users behind firewalls.

RMI

Java RMI is remote method invocation between peers or between client and server when programs at both ends of the invocation are written in Java. For complicated applications it may be useful to use applets for the user interface and server applications in Java for data access with RMI for communication between the two.

RMI can accept requests from the client and invoke methods in the server application to access the database and return the results to the client. Java object serialisation can incur a large overhead on the server. Implementing an RMI application also requires considerable human resources in terms of programming hours and expertise [Siple 98] , [Chang, Harkey 98].

CORBA

The Common Object Request Broker Architecture (CORBA) is similar to RMI except that it enables the integration of heterogenous, distributed systems. While RMI suffers from the limitation of using Java at both ends of the method invocation, CORBA can interoperate because it is language neutral. The price it pays for this independence is complexity. CORBA is useful for the interaction of applications written in different languages, as such its use for the purposes of this demonstration would be excessive [Siple 98].

JDBC

There are a number of ways to retrieve data from a relational database. Embedded SQL is embedding SQL statements in a programming language. There are a number of disadvantages to this method including the requirement for a pre-compiler and the production of non-portable code. Call level interfaces such as ODBC can be ported in binary to any database system with the appropriate drivers. However call-level interfaces are limited to dynamic SQL statements i.e. statements processed at program run-time. ODBC uses handles to represent database concepts such as the database connection and ODBC driver. Java's object oriented approach allows these objects to be be accessed directly with call-level interfaces using JDBC or with embedded SQL using SQLJ.

The JDBC (Java Database Connectivity) API specifies the conditions for the communication between Java and a database. This includes how the Java application opens a connection, communicates, executes SQL (Structured Query Language) statements and retrieves the results of queries. Database vendors can use the JDBC API to build drivers for accessing their databases.

These JDBC drivers provide an application with consistent and valid access to a database by translating Java requests from the application to the native language of the database.

There are four types of JDBC driver as listed in the table below.

TYPE	NAME	FUNCTION
I	ODBC-JDBC Bridge	translates Java call into an ODBC call
II	Native API-Java	translates Java call into a native db call on the client
II	JDBC Network-All Java	translates Java calls to ‘network’ protocol, which calls native methods on server.
IV	Native Protocol-All Java	accesses database directly

Figure 8 JDBC Driver Types

Type IV Native-Protocol-All-Java driver

Unlike the other drivers listed in the table above, Type IV drivers access the database directly. This is possible only when the native database protocols are rewritten. This greatly reduces the overhead required in maintaining native library code, opening and closing connections, and translating Java to database-specific calls. Type IV drivers also allow applets to communicate directly with the database. However, because applets can only open connections back to the server from which it was downloaded, the web server and database server would have to be run from the same machine [Siple 98].

This introduces a security issue and a scaling issue. Using Type IV drivers in this design offers more flexibility as the system can access the database using applets and/or servlets.

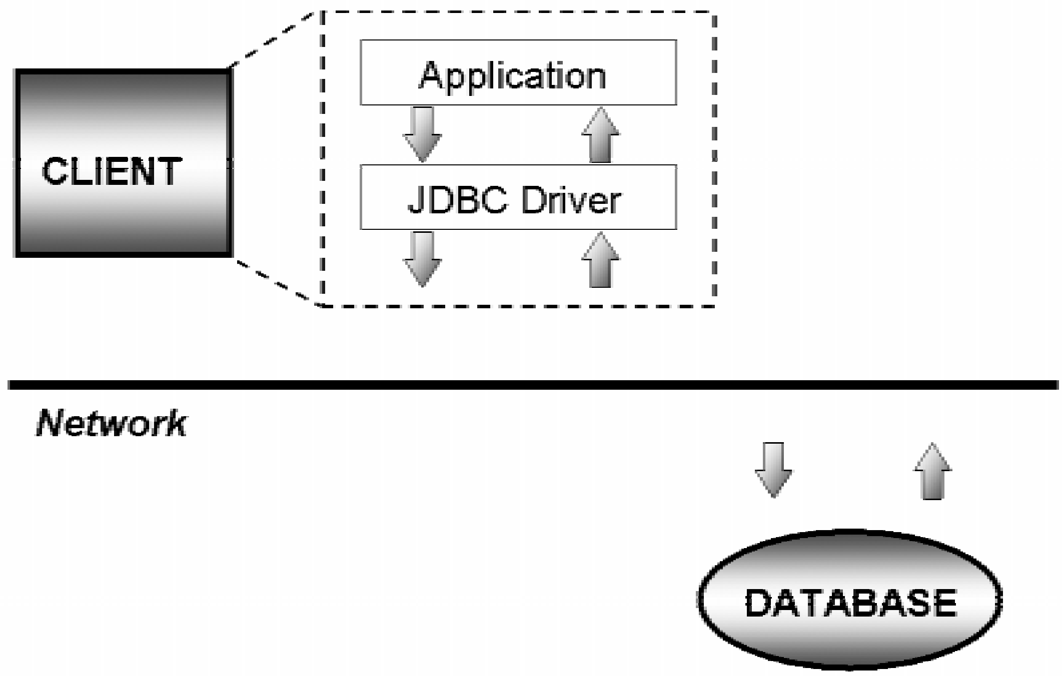


Figure 9 Type IV JDBC driver

Storing the Metadata

Storing the metadata in the database ensures that the data is current and extensible. The metadata for the course data may be stored separately or as an indivisible part of the data. For this demonstration the metadata is stored with the data in XML format. Each set of data and metadata therefore constitutes a meaningful unit in itself.

The meaning and structure of the data becomes an inseparable part of the data and can be appropriately used by an application whether for display of the data or to communicate the data to another application.

Storing the data in XML also provides flexibility in design and implementation choices by relying on the lowest common denominator service [Idris 99].

Accessing the Metadata

The metadata elements of the data must be processed to be useful. An XML processor is required to process XML markup and provide a set of APIs to access all the parts of the XML data. There are a number of publicly available XML parsers including Lark, Larval, XP, MSXML and IBM's XML4J [XSW].

XML may be processed using a tree-based API or an event-based API. With an event-based API actions are executed depending on parsing events which are reported to the event handlers of an application. The appearance of any structure, i.e. start and end tags, in the document constitutes a parsing event. The advantage of an event-based API is that it does not create a conceptual tree of the document in memory so it requires less memory. Event-based APIs are useful for machine generated data. SAX, Simple API for XML, is an event-based API.

A tree-based API compiles an XML document into an internal tree structure and allows an application to navigate that tree. The Document Object Model (DOM) was one of the metadata standards discussed in Chapter 3. DOM is a tree based API. Tree-based APIs are appropriate for data which is human readable. Tree based APIs allow the nodes of the tree to be restructured providing much more advanced manipulation of the XML document than is possible with an event-based API.

DOM constructs a default object model which is a tree of Java Object nodes.

This object model is the in-memory representation of the data that came from the XML document. This transmutes the data and the metadata of the XML documents to a form which Java programs can access independently of the way in which the XML was directly processed. The set of Java classes that retrieves the XML data from the database and populates this object model acts as an XML source for the rest of the system [Idris 99][Chang, Harkey 98][NATA][Johnson 99].

Adaptivity

What?

This section discusses the adaptivity that will be implemented in the demonstration.

User Features

The types of adaptivity that can be implemented have been discussed in Chapter 3. There are four aspects of the user which should optimally be taken into account when providing an adaptive solution. These are the user's objective, experience, performance and preferences. The user's objective is what the user wants to achieve by taking the course. The user's experience should include their hypermedia experience, the user's knowledge of the subject matter and the user's maturity as a person and as a learner. The user's performance is how the user is getting on during the course. This requires feedback from the user. Indirect feedback would include the recording of time spent at each node or the tracking of the user's navigational paths. Direct feedback would include answers to questions or test results. The preferences of a user are concrete aspects of the course which are more suitable for that user or which are more favourable to the user as a matter of personal choice. This would include the technology being used for example the speed of the modem or the browser version. It may also include the size of the font for the visually impaired or the colours used as a personal choice. The course content may also be adapted according to preferences for example instructions for setting up the JDK differ depending on the Operating System.

Adaptive Link Annotation

Annotation of links allows the user to have an idea of what the content of the link is and where it is before they decide to traverse the link. Useful adaptivity for annotations would include the relevance of the link for that user at that time - this information should be conveyed using different colours for the link. This would require the application knowing whether the user has covered the content to which the link connects or whether the information is relevant to the user's current objective and knowledge. Information that may help the user before they click on the link is whether or not the link will embed the content, open a new window or replace the current window. The java program will access the user model and determine the action for the link - this information should be conveyed to the user by including an icon as part of the link.

This icon will denote that the link will replace the current window.

This icon will denote that the link will open a new window.

This icon will denote that the link will embed the content in the current page.

The Table of Contents is constant and stable for each user to minimise disorientation and ensure a sense of closure.

Navigation

The presentation of links should be adaptive. When a topic has prerequisite concepts a link should be presented to the user at the start of the new topic. If the user has not covered this concept the link should embed the definition in the text. A further link should be presented to open a new window with further information about the concept should the user desire to follow it. This further information would include the concept's descriptions and example or illustration. If the concept has been covered by the user or is assumed to be known from the direct information given about the user's previous experience, a link should be presented but not embedded.

Content

The content should be appropriate for the user's maturity and hypertext experience. The content developed for the demonstration of this dissertation is a course in the Java programming language.

As such the target user is assumed to be an adult with experience in Hypertext systems as the subject matter is an advanced Internet subject. The content is adapted depending on the user model.

The learner is assumed to have an overall objective which is the outcome the learner wishes to achieve by taking the course. If the granularity is increased, the learner's objective may deviate from the overall objective. This deviation is assumed to be temporary. A learner may wish to achieve a comprehensive understanding of the Java programming lanuguage but feel that networking is too advanced for the programming tasks that he or she had in mind after completion of the course. The learner can then deviate from his overall objective for that portion of the course and follow the content appropriate for the overview objective.

There are four possible objectives for this course - **overview, comprehensive, practical and revision**. Concept definitions are displayed for all objectives. The content thereafter is dependent on the current objective which is assumed to be the overall objective unless the objective has been changed by the learner to indicate a changed objective for that topic. The meaning of the objectives are communicated to the user by questions which the learner is requested to answer during the registration process. The overview objective is targeted to users who wish to know about the Java programming language but do not expect to be able to program by the end of the course. It therefore represents an easy, short path through the content concentrating on basic theory and general information. The comprehensive objective provides an in-depth, detailed view of the content and represents a long, step by step path through the content including theory and examples. The practical objective is to help learners to begin programming as quickly as possible and to learn by example. The focus is on writing programs rather than learning the theory and design of the language. The revision objective provides a concise summary of the topic for learners for whom the content should be familiar. This represents a shorter, more difficult path through the content.

In this demonstration the user is able to change the objective at any time to display the content available for alternative objectives. This increases the

learner's control and allows a sense of closure without overloading the learner with irrelevant material. This feature is adaptable rather than adaptive because the initiator of the change is the learner him or herself. The user could be interrogated before the start of the course or the start of each topic in which case the change to a new objective would be initiated by the application on the basis of the knowledge procured directly from the user. This approach was not taken for a number of reasons. The user may not be able to gauge the level of their knowledge on a particular topic until they see the content. Obtaining such detailed information from a learner may also prove to be an irritation. We have established that the target users of this demonstration are adults likely to have hypermedia experience. The applicability of this design depends on the premise that the potential users of this course have the capacity to effectively determine the most suitable objective for their own learning.

The knowledge of the learner is modelled as an overlay of the concepts of the course. These are assumed to be known by the user because they have been covered by the learner in the course or because the user's previous knowledge obtained from the registration process would indicate that the user should know these concepts. The knowledge of these concepts determines whether a link to a prerequisite concept is automatic or actuated by the user. An automatic link embeds the content in the text when the page loads. This will offer a definition of the concept and a link which may be followed if desired to view further information in a new window.

If the concept is assumed to be known, the link to the prerequisite concept will not be automatically embedded but the concept name will be displayed as a link with this icon ▼ to indicate that the link may be activated to embed the concept definition in the current page. The embedded concept definition will include a link for further information. This embedding of content produces the effect of a collapsing or expanding link in the text which is automatically expanded if the user is assumed not to know the concept and collapsed if the user is thought to know the concept.

The application makes assumptions about the knowledge of the learner on the basis of the user model which if inaccurate may be rectified by the learner.

Contrast/Comparison

Additional information will be displayed for the user when the content may be enhanced by a contrast or comparison. The text will be enclosed by a box to draw attention to the information and an appropriate icon will be used to designate whether the information is emphasising a similarity or difference.

How?

This section discusses how the adaptivity discussed may be implemented.

XLink & XPointer

XLink and XPointer are discussed in chapter 4. A collapsible link can be implemented in XLink by switching the value of the ACTUATE attribute from 'AUTO' to 'USER'. The SHOW attribute should be set to 'EMBED'. This can be changed to 'REPLACE' or 'NEW' as the context demands. Setting the ACTUATE attribute to 'AUTO' in addition to the SHOW attribute as 'EMBED' allows page content to be created dynamically from fragments of data.
The granularity of these fragments can be as fine as the exactitude of the XPointer specification allows. XLink also allows applications to provide link annotation through the use of the ROLE attribute.

Although Microsoft's Internet Explorer fully supports the XML verion 1.0 standard, unfortunately the XLink and XPointer specifications are at a draft stage and as such there are no publicly available browsers which support linking in XML. For this reason an alternative must be investigated for the purposes of this demonstration.

DHTML

Dynamic HTML (DHTML) is not a standard or a particular technology, it is a concept with a close relationship with the XML family of technologies. It is a language neutral interface that allows programs and scripts to dynamically access and update the content, structure and style of web documents by exploiting and enhancing a combination of current technologies: HTML, stylesheets and scripts. The Dynamic HTML object model is based on the Level 0 recommendation for the XML Document Object Model. CSS (Cascading Style Sheets) is similar to but not as comprehensive as XSL (the eXtensible Stylesheet Language). The significance of DHTML is that it can change and control the web document in repsonse to the user's behaviour without contacting the server thus saving time and bandwidth, improving response time, speed and performance [Chang, Harkey 98][Goodman 98].

HTML 4.0

HTML 4.0 recognises the advantages of separating the content and style of web pages and allows the author to explicitly structure web documents using the *CLASS*, *ID*, and *DIV* attributes.

CSS

A style sheet is a set of rules to be applied to the display of text and graphics on a web page. Cascading Style Sheets (CSS) Level 1 is a W3C stylesheet standard for assigning style rules to HTML elements which are applied when the page loads.

These style rules can be changed after the page has been loaded using ECMAscript. CSS style sheets can be inline, embedded or linked. An inline style is a style definition placed within a given tag which applies only to the text within that tag. *<P STYLE=* "font: 12pt palatino*">* will format the text from this tag until the *</P>* corresponding to the end of this particular paragraph. An embedded style sheet is placed on the web page between *<STYLE>* and *</STYLE>* tags. A linked style sheet is a separate file with style sheet definitions that may be applied to a number of pages. The cascading of multiple rules depends on a hierarchy of priority. For example a style rule which is defined in the same document as the element to which the rule applies takes precedence over a style rule defined in a separate style sheet for the same element i.e. when more than one style rule is applied to an element an inline style overrides an embedded style which overrides a linked style.

The HTML element within which an inline style is defined is the target of the rule. The target of embedded and linked style rules are specified by selectors. CSS provides Type, Attribute and Contextual selectors.

Type selectors apply a style based on the selection of one or more HTML elements.

Attribue selectors apply rules to elements depending on their attributes. The value of the CLASS attribute is extensively used for applying CSS style rules.

Contextual selectors apply a style based on the relative position of an element such as when it is preceded by a particular tag i.e. the style is applied to the element only when the parent tag meets the rule defined.
For example the following statement

```
P  OL  EM { font-color: red }
```

will change the font to red only when emphasized text denoted by *EM* occurs within an ordered list (*OL*) which occurs within a paragraph (*P*) [BUSC][Goodman 98][Chang, Harkey 98].

ECMAscript

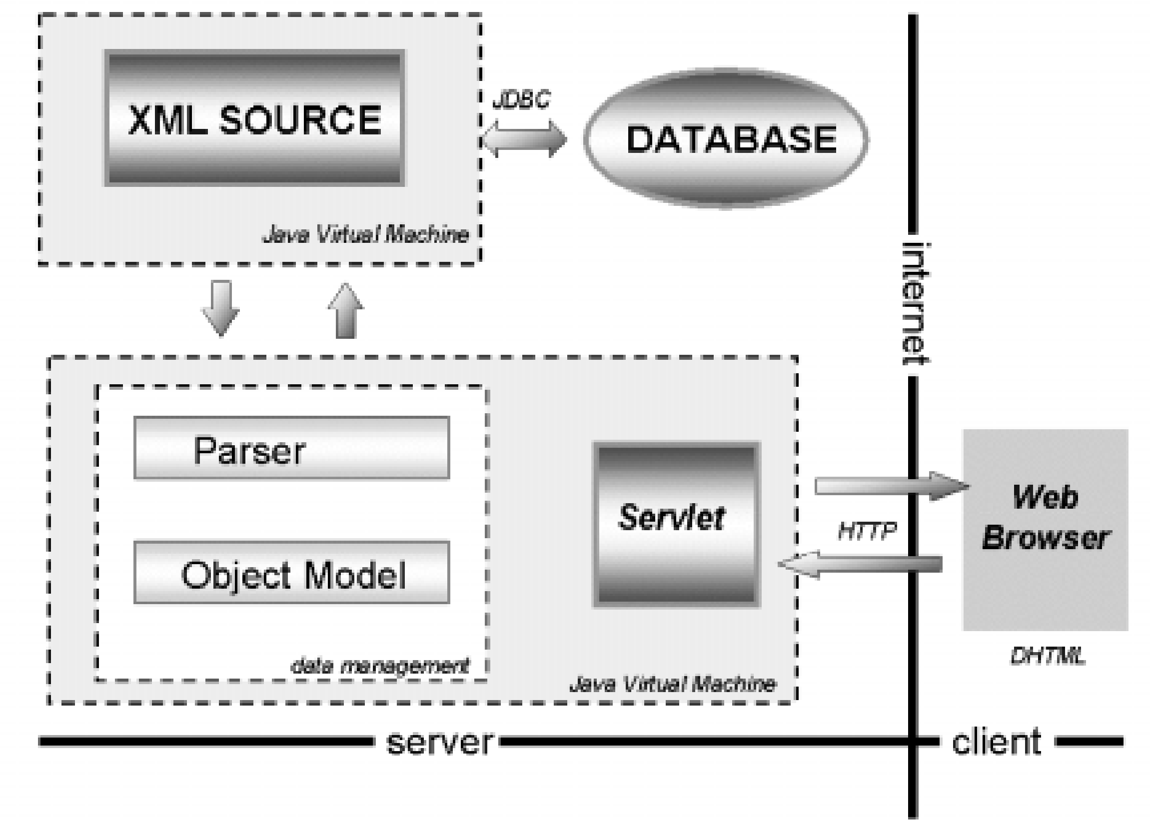
ECMAscript is a client-side scripting language built to work with any object model. It is an ECMA published standard based primarily on JavaScript. The term ECMAscript has become synonymous with the term JavaScript. ECMAscript enables access and control of the objects in a web document.

DOM level 0

When a web page is loaded, browsers create a hierarchical tree in memory of the elements in a web document which can be influenced by scripts. The document objects include the browser window which contains another object document which contains an object form which contains the objects form elements. These document objects may be controlled by scripting in script aware browsers. The object model implemented complies with level 0 of the W3C DOM specification.

Presentation of data to the user

The data which is presented to the user will depend on the values stored in the user model and the adaptivity consonant with those values.



adapted from DeveloperLife.com [Idris 99]

Figure 10 System Architecture

Summary

This chapter proposed the following metadata driven architecture for educational Adaptive Hypermedia. The XML data will be retrieved from the database using a JDBC Type IV driver and SQL. The XML data will be processed by Java and transformed from the default DOM object model to a custom Java object model. Servlets will use this custom object model to build a presentation layer in DHTML whose content and linking depends on the current user model.

Chapter 6

IMPLEMENTATION

Introduction

This chapter describes the implementation that was developed to demonstrate the feasibility of the architecture proposed in Chapter 5. The implementation process is outlined with a description of the methods and technology used.

Course subject

The demonstration will use a course in the Java programming language to demonstrate the implementation of adaptive hypermedia using metadata.

Hypermedia Experience

Java is an advanced Internet topic and as such target learners would typically have extensive hypermedia experience which implies that they would be comfortable navigating in a non-linear environment.

Prerequisites

The design of programming languages is greatly influenced by the successes and failures of previous technologies. The experience of the learner in other technologies may exempt them from areas of theory in the Java course.

An example of this is the concept of object oriented programming which constitutes a substantial learning curve for learners who do not have experience of object oriented technologies. Similarly the basics of the world wide web and HTML is required to fully understand the use of Java Applets and to embed Applets in HTML pages.

Objective

Global Objective

For the purposes of this demonstration, the potential objective of target learners of programming courses is broken down into four specific areas. During registration new course participants are asked to check the statement which most closely resembles their own objective in taking the course. A word is highlighted in each statement and this word is used to label the radio buttons in the objective box displayed at the top of the screen during the course. The user chooses the objective or goal which corresponds to what they wish to achieve. This encourages the user to reflect about what they wish to achieve by studying the course. As discussed in Chapter 3, an advantage of adaptive hypermedia systems is that users tend to be more goal oriented. This promotion of intrinsic motivation is compatible with the constructivist theories of learning discussed in Chapter 2.

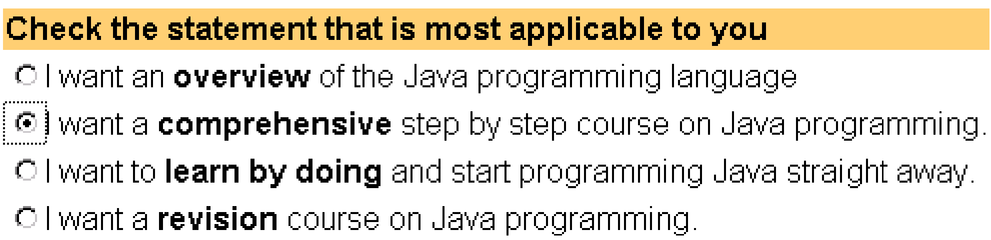


Figure 11 Specifying the global objective

OVERVIEW

Students may wish to know about Java but do not aspire to either mastery of the theoretical details of the language or to the development of Java programs. The objective of these learners is to gain an overview of the language.
An overview of Java would focus on a general approach to the language which is accessible to learners regardless of their experience in other programming technologies.

COMPREHENSIVE

Students who wish to learn about the language theory and its application in creating software applications may take a detailed, step by step approach to the content as is provided by the content set as comprehensive.

PRACTICAL

Students who wish to understand the theory and practical use of the language are considered to have a practical objective. Such students may have a direct, short term requirement to implement java programs as quickly as possible. This goal corresponds to a surface learning approach by the learner and is appropriate for learners who prefer to learn by example.

REVISION

Target users who wish to revise are given content with a higher level of brevity and difficulty. Such users would be expected to have experience of the Java language but wish to recap what they have already covered. This is also useful for students who are uncertain of their comparative experience. They can follow a concise course to see what aspects they already know and what they need to revise by switching, for example, to a comprehensive objective for certain topics or concepts.

LOCAL OBJECTIVE

The content displayed for a particular user is dependent on the objective that the user specified at registration. Before the content of each topic is displayed, the value of each user's objective is retrieved from the database. This objective is the user's global objective for the course. The content for the current topic which corresponds to that objective is displayed for the user. When this content is displayed the user has the option of changing the objective by clicking on the radio buttons of the objective box shown below which is displayed at the top of the screen.

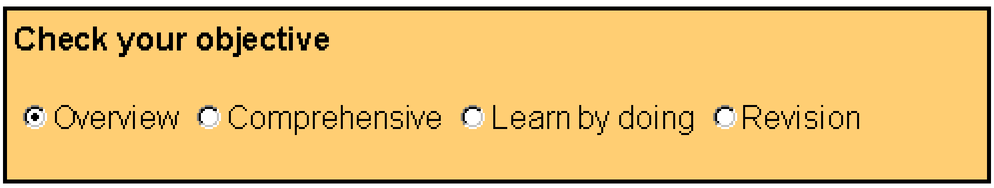


Figure 12 Modifying the learner's local objective

The content of each XML element is placed inside a DIV element of the same class. The ID attribute of the DIV element is set to one of the four objective values, 'OVERVIEW', 'COMPREHENSIVE', 'PRACTICAL' or 'REVISION'. The class by default is hidden by setting the value of 'display' to 'NONE' in the cascading style sheet linked to that page. One empty DIV element is also defined. If the ID of a DIV element matches the objective value retrieved from the database, the content of that DIV element is displayed on the page using the insertAdjacentHTML( ) method which takes the parameters (element reference, text). . The insertAdjacentHTML() method is available for all scriptable elements in Internet Explorer 5 and inserts HTML and text adjacent to the element specified. The empty DIV element ID is passed as the element reference so all of the course content is inserted adjoining this empty DIV element on the page.

The content which corresponds to the user's global objective is automatically displayed for the user when they link to a topic. If the user changes the objective in the box displayed at the top of the page, alternative text is shown corresponding to the checked objective. This adaptability is performed by the Dynamic HTML without any communication with the web server.

CONTRASTS AND COMPARISONS

The syntax of Java has been influenced by the development of other programming languages most notably C and C++. It is useful to provide contrasts and comparisons to C and C++ for users who have previous experience of these languages. Users indicate their knowledge of these languages by marking a checkbox during the registration process as illustrated below.

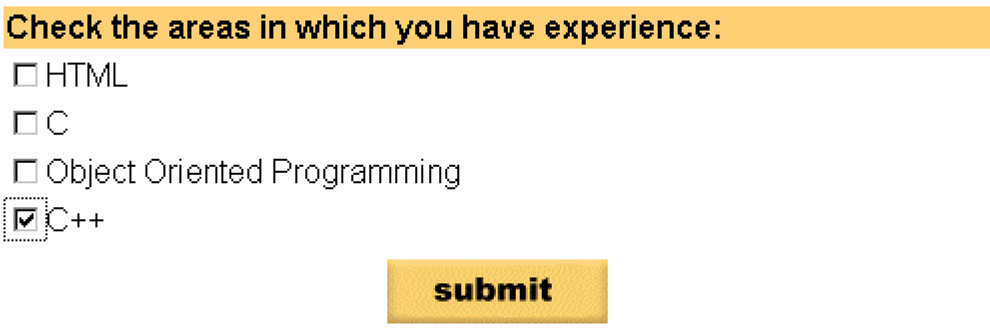


Figure 13 Registering the learner's background

When a contrast or comparison is relevant to a particular concept, it will be higlighted for the user by displaying the text in a separate box.

A comparison is designated with a check icon to indicate similiarity.

The Java keyword public is similar in usage to public in C++

Figure 14 Comparison

Similarly a contrast is designated with a cross icon to indicate difference, in the case below when a user who has experience in the C programming language is learning about the import statement.

The Java import command is not at all similar to the #include command in C although they accomplish similar functions. The Java import command behaves more like a linker - it tells the Java compiler and interpreter where to find classes, variables, method names and method definitions rather than actually including the code as the C preprocessor does.

Figure 15 Contrast

Course Usability

It is important that the course should be intuitive and user friendly. To assist in these aims the navigation icons and colour scheme used is consistent throughout the course.   The navigation icons as shown below are displayed in a separate frame at the bottom of the screen so that they are visible and accessible at all times to the user.

Figure 16 Navigation icons

The use of frames is often criticised because they can load inside each other and create unwanted scrollbars causing a cluttered screen which is confusing for the user. For this demonstration horizontal scrollbars are hidden and an ECMAscript function is used to prevent frames loading within another frame by ensuring the document is loaded in the topmost position using the object reference ‘top’.

During registration new course participants are asked to check the statement which most closely resembles their own objective in taking the course. A word is highlighted in each statement and this word is used to label the radio buttons in the objective box displayed at the top of the screen.

Database

Oracle

The Oracle Relational Database Management System (RDBMS) is an industry leading database system designed for mission critical data storage and retrieval. The RDBMS is responsible for accurately storing data and efficiently retrieving that data in response to user queries [Holowcza 99]. The development of this demonstration using Oracle required access to the Oracle 8.0 ugrad database running on allen.cs.tcd.ie.

Database Organisation

The Oracle supported data types used in this presentation are NUMBER, VARCHAR2.

VARCHAR2 is the character data type which can contain letters, numbers and punctuation. The maximum size of a VARCHAR2 column in Oracle 8 is 4,000 bytes.

NUMBER is the numeric data type which can contain integer or floating point numbers.

LONG datatype can hold character data of virtually any size. The LONG data type is similar to VARCHAR2 but no maximum length is specified.

COURSE CONTENT

TOPIC TABLE

The organisation of the TOPIC table is described in the table below. The TOPIC_NAME column stores the name of the topic for the human reader. The TOPIC_NAME column is a primary key and so is unique. This prevents duplication of topics as new topics are added to the database at a later date. The CONCEPT_NUM column lists the number of concepts which make up the topic. The CONCEPT_LIST column is a list of the concepts of which the topic is comprised. The list consists of concept IDs separated by the + sign which is parsed by a Java program before retrieving the content of each concept in turn.

FIELD NAME	**DATATYPE**	**SIZE**	**CONSTRAINTS**	**PRIMARY KEY**	**DESCRIPTION**
TOPIC_NAME	VARCHAR2	20	NOT NULL	✔	name of the topic
CONCEPT_NUM	NUMBER	10			number of concepts in a topic
CONCEPT_LIST	VARCHAR2	100			list of concept ids

Figure 17 Topic table

CONCEPT TABLE

The organisation of the CONCEPT table is described in the below. CONCEPT_NAME is the name of the concept. CONCEPT_ID is a unique identifier for each concept. The CONCEPT_CONTENT column contains the XML content for that concept.

CONCEPT	**DATATYPE**	**SIZE**	**CONSTRAINTS**	**PRIMAR Y KEY**	**DESCRIPTION**
CONCEPT_NAME	VARCHAR2	3			concept name

CONCEPT_ID	VARCHAR2	10	NOT NULL	✔	concept ID
CONCEPT_ CONTENT	LONG	n/a			content of the concept

Figure 18 Concept table

USER MODEL

LEARNER_DETAILS TABLE

The LEARNER_DETAILS table models the learner to enable the application to implement adaptivity appropriately. The information is acquired directly from the user during the registration process and indirectly as the content is displayed during the course.

This is the registration form which the user is requested to submit as a new course participant.

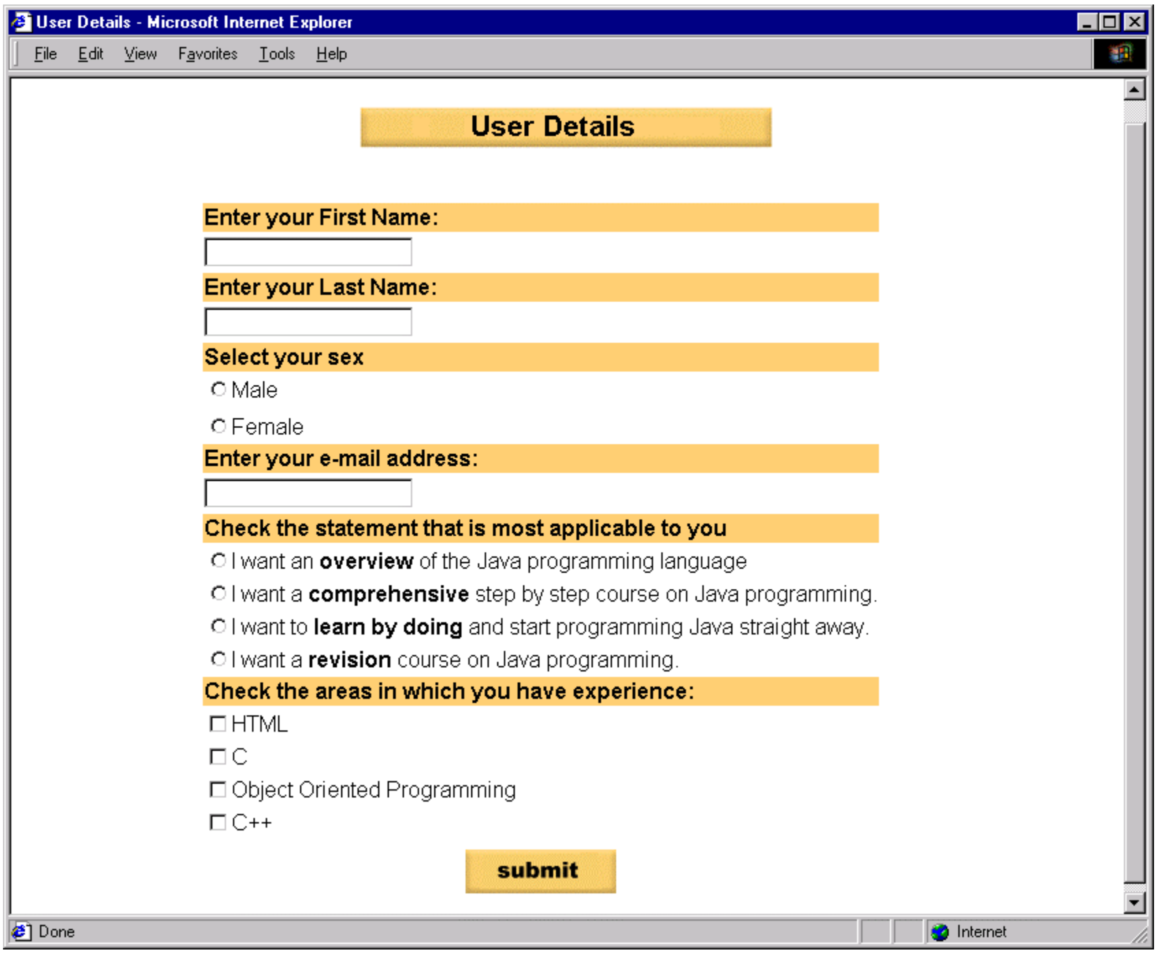


Figure 19 Registration process

The LEARNER_DETAILS table has been separated into three areas for the purposes of this discussion. All of the fields in the three tables below are part of the same table in the Oracle database.

The fields in the figure 20 are required for adminstrative and statistical purposes.

LEARNER_DETAILS	DATATYPE	SIZE	CONSTRAINTS	PRIMARY KEY	DESCRIPTION
LEARNER_ID	NUMBER	10	NOT NULL	✔	
OBJECTIVE	VARCHAR2	15			global objective
FIRST_NAME	VARCHAR2	30			First name

LAST_NAME	VARCHAR2	30			Surname
EMAIL	VARCHAR2	50			e-mail address
SEX	VARCHAR2	10			sex

Figure 20 Learner details table A

The value stored in the objective field is used to adapt the content to match the user's global objective as discussed above.

LEARNER_DETAILS	DATATYPE	SIZE	CONSTRAINTS	PRIMARY KEY	DESCRIPTION
OBJECTIVE	VARCHAR2	15			Learner's global objective

Figure 21 Learner details table B

The table below describes a sample of the concepts of the course which may be covered by the user and stored in the database as known or unknown.

LEARNER_DETAILS	DATATYPE	SIZE	DESCRIPTION
OBJECTIVE	VARCHAR2	15	global objective
FIRST_NAME	VARCHAR2	30	First name

EMAIL	VARCHAR2	50	e-mail
SEX	VARCHAR2	10	sex
OOP	VARCHAR2	10	OOP concept
CPLUSPLUS	VARCHAR2	10	C++ concept
CEE	VARCHAR2	10	C concept
HTML	VARCHAR2	10	HTML concept
IMPORT	VARCHAR2	10	import concept
APPLET	VARCHAR2	10	Applet concept
APPLICATION	VARCHAR2	10	Application concept
CLASS	VARCHAR2	10	Class concept
CODEBASE	VARCHAR2	10	Codebase concept
HSPACE	VARCHAR2	10	Hspace concept
VSPACE	VARCHAR2	10	Vspace concept
INHERITANCE	VARCHAR2	10	inheritance concept
INIT METHOD	VARCHAR2	10	init method concept
SUBCLASS	VARCHAR2	10	subclass concept
SUPERCLASS	VARCHAR2	10	superclass concept
THREAD	VARCHAR2	10	thread concept
NULL	VARCHAR2	10	null concept
PUBLIC	VARCHAR2	10	public concept
STRING	VARCHAR2	10	string concept
OBJECT	VARCHAR2	10	object concept
JAVA	VARCHAR2	10	java concept
INTERPRETER	VARCHAR2	10	interpreter concept
COMPILER	VARCHAR2	10	compiler concept

Figure 22Learner details table C

XML FORMAT

Parser

IBM’s XML parser for Java was one of the highest rated Java XML parsers in Java Report's February 1999 review of XML parsers [Anez 1999] and the best performer in performance testing by Developerlife.com a resource for XML and Java developers run by the Bean Factory, a software development company who build custom XML applicaton servers [Idris 99]. The IBM XML4J parser is used in this implementation because it is a high performing parser with thorough compliance with the DOM level 1 standard.

CONCEPT.DTD

The following extract is the data type definition created for the Adaptive Java course. The DTD is designed to describe a generic concept which can be transferred to any educational course. A further extract following the DTD shows an example of the text of the Adaptive Java course marked up in XML with alternative descriptions for users with different objectives.

```
<?xml encoding="UTF-8"?>
<!ELEMENT
CONCEPT (DEFINITION|DESCRIPTION+|EXAMPLE|CODE+|A+|CONTRAST+|COMPARISON+)*>
<!ATTLIST CONCEPT
NAME CDATA #REQUIRED>
<!ELEMENT DEFINITION (#PCDATA)>
<!ELEMENT DESCRIPTION (#PCDATA|CODE)*>
<!ATTLIST DESCRIPTION
LEVEL  CDATA #IMPLIED>
<!ELEMENT EXAMPLE (#PCDATA|DESCRIPTION|CODE)*>
<!ELEMENT CODE (#PCDATA|DESCRIPTION)*>
<!ELEMENT A (#PCDATA|DESCRIPTION)*>
<!ATTLIST A
HREF CDATA #IMPLIED>
<!ELEMENT CONTRAST (#PCDATA)>
<!ATTLIST CONTRAST
TARGET CDATA #IMPLIED>
<!ELEMENT COMPARISON (#PCDATA)>
<!ATTLIST COMPARISON
TARGET CDATA #IMPLIED>
```

Applet Concept

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE CONCEPT SYSTEM "http://wilde.cs.tcd.ie:9999/XML/concept.dtd">
<CONCEPT NAME="Applet">
<DEFINITION>
Applets are programs that are downloaded from the World Wide Web by a Web browser and run inside an HTML Web page.
</DEFINITION>
<DESCRIPTION LEVEL="COMPREHENSIVE">
Before Java, World Wide Web pages were a combination of text, images and forms that used gateway programs running on the computer that hosted the pages.  These gateway programs required special access to the Web page server machine, so most Web users did not have the ability to use them.  Writing them required even more expertise.  In contrast, programmers of all skill levels can write Java applets.  You can test them with any Web browser that handles Java programs, and put one on a web page without requiring any special access from a Web provider.
</DESCRIPTION>
<DESCRIPTION LEVEL="PRACTICAL">
You will need a Java-enabled browser such as Netscape Navigator or Microsoft Internet Explorer to run applets.  Java has many built-in security features so running Java applets on your machine is still a relatively safe thing to do.
</DESCRIPTION>
<DESCRIPTION LEVEL="REVISION">
A Java Applet is a Java program that executes inside a browser.
Essentially, a small portion of the browser window functions as a virtual
computer, and the Java program executes on this computer. If there are
several Applets executing in a browser window, they can execute
independently. Applets run under severe network restrictions. An applet
cannot access URLs of any computer other than the one it is running from and cannot read or write to the local hard disk.
</DESCRIPTION>
<DESCRIPTION LEVEL="OVERVIEW">
Java applets are small segments of Java code that are downloaded onto a
visitors own computer. Once there the Java applet code runs on the host cpu. Applets are specifically designed to make sure that the applet cannot harm the client environment.  The applet is restricted from gaining direct access to the client system. Java applets allow web pages to show animation, interactivity and other features not present in static web pages.
</DESCRIPTION>
</CONCEPT>
```

Web Server

The web server used in this demonstration was the the JavaWebServer version 1.1.3 . The Java Servlet API is supported by a number of web server applications using the Java Servlet Development kit. However, Sun's Java Web Server provides a user friendly GUI for the adminstration of the server which includes the configuration of servlets.

Browser

The elements of the document object model which are scriptable differ for Netscape Navigator 4 and Internet Explorer 4 (IE4) and 5 (IE5). IE4 and IE5 reflow the page automatically when an element is changed [Goodman 98]. For the purposes of this demonstration IE5 will be used. An equivalent application can be written for Netscape Navigator 4 but the page must be reloaded before changes to document objects take effect.

Summary

This chapter has described the implementation of the Adaptive Java course using the architecture proposed in Chapter 5. The implementation process was outlined with a description of the methods and technology used.

Chapter 7

EVALUATION AND CONCLUSION

Introduction

The objective of this chapter is to evaluate this dissertation and present the conclusions which have been reached. It is useful to restate the objectives of the dissertation as a part of the evaluation. The primary objective of the dissertation was to investigate the use of metadata to support web-based adaptive, educational courseware.

Evaluation

The achievement of the aim of investigating the usefulness of metadata in the area of Educational Adaptive Hypermedia required investigation of current educational theories for web based education. This was undertaken in Chapter 2 and included proposals for learning approaches that comply with the tenets of constructivist pedagogy. The methodology of adaptive systems was addressed in Chapter 3 with particular emphasis on the properties of a learner which can be to leverage the efficacy of adaptive changes. The proliferation of educational metadata standards was discussed in Chapter 4.

An architecture to support metadata driven adaptive educational courseware was proposed in Chapter 5. [Wade et al 97] advise that the design of a technology based educational course requires two phases of design. The first is the choice of appropriate technology to support the desired set of educational modes. The choice of technology for the demonstration was discussed in Chapter 5. The second is the design of the actual course material and the environmnent from which it is accessed which was covered in the discussion of the implementation in Chapter 6.

Pedagogy, Usability and Adaptivity

[Wade, Lyng 99] ascribe the most common failures of educational systems to a lack of pedagogical support and / or usability problems experienced by learners. The Adaptive Java course demonstration attempted to create a course compatible with constructivist principles of teaching. Constructivism promotes customisation of course material to meet the requirements of the individual learner. In view of this ideal, a user model is constructed to customise the content based on the user's background and the user's global and local objectives. While taking the Adaptive Java course, the user can also alter the sub-objectives for a topic which displays content corresponding to those objectives thus increasing interactivity and learner control.

The adaptivity based on the user's objective ensures that a user is required to know only the information needed at that stage of the learning process as recommended by [Galitz 97]. This approach takes account of the fact that a user's learning style or cognitive style may change at different times and within different areas of the course. For example a user whose cognitive style is verbal may follow the subject matter corresponding to the comprehensive objective. For a particular topic they may choose to change to the 'Practical' or 'Learn by example' objective which correlates more closely to a user with a spatial cognitive style who prefers to learn by illustration. This choice may be initiated by the learner's mood, confidence or previous experience of that topic area.

The introduction of adaptivity should not reduce the usability of a system. The usability of educational systems is particularly significant simply because the goal of the system is learning and to achieve this goal the system must sustain the user's motivation. The implementation of adaptivity in the Adaptive Java course enhances the usability of the system.
The objectives are displayed in an unobtrusive manner at the top right hand corner of the screen. The content changes dynamically as the user checks the radio buttons labelled with the objective to which the button corresponds. This provides a natural and intuitive response to user actions. The system tolerates different levels of user familiarity and performance and so is compliant with the HCI (Human Computer Interaction) principle of flexibilty [Wade, Lyng 99].

[Galitz 97] describes design consistency as the key to forming mental models. The Adaptive Java course interface exhibits a uniform style. [Wade et al 97] promotes the use of a uniform style throughout the course along with the use of a small set of colours. Colours are used in the Adaptive Java course system only when required and the use of colour is consistent as recommended in [Wade, Lyng 99]. The default background is white while an orange background is used for illustrations and examples. Orange is also used to highlight information including navigation icons, dividing lines and additional contrasts and comparisons which are designated by orange icons and surrounded by an orange border. The screen is also separated into a Navigation area and a Presentation area which are both displayed in a consistent and predictable relative position on the screen as advocated by [Wade, Lyng 99]. The usability of adaptive systems must observe the tendency of users to ignore instructions. [Galitz 97] attributes this dislike of direction to the 'trial and error' approach of human problem solving. The usefulness of Adaptive systems are greatly reduced if the user must endure a learning curve before they can begin to learn the actual material of the course.

Authoring and Reusability

Concepts can be added, modified or removed from the Adaptive Java course with no change to the actual application. The content of the Adaptive Java course consists of Topics which are comprised of concepts. The concepts are marked up in XML so that the Adaptive Java application can appropriately apply the presentation and adaptivity required for the course.
New concepts can be added to the database at any time, the only requirement for its use in the course is that it complies with the concept DTD (data type definition). If the concept is listed as belonging to a Topic, the concept will be displayed in the browser as part of that Topic and the adaptivity will be implemented depending on the concept's XML tags. The concept field in the database contains units of XML that contain independent meaning. These units can be re-used by any XML processing application that can retrieve the XML concepts to present the data to a user or communicate the data to another application. For example the Adaptive Java application could choose to display only the content between the definition tags of a concept to serve as a glossary. This would require trivial changes to the application code.

The adaptivity of the demonstration is independent of the actual course material as it is implemented using out of line stylesheets and javascript files. The adaptivity can be changed simply by modifying or replacing these files. For example, if a course director wished to confine users to the course material corresponding to only one objective for the purpose of complying with curricular regulations, only the stylesheet of the Adaptive Java application needs to be modified or replaced. This constitues repurposing of the content.

System improvements

The user model in the Adaptive Java course comprises a relational database table with a field for each property of the user. Saving the user model in XML format in the database would allow the communication of the student's profile information to other XML compliant applications.

Classes receiving data from the XML parser should rely on generic code for parsing so that the code is not dependent on the parser of any one vendor in this case IBM™.

Currently adding a new topic requires trivial code changes to the Java servlets that display the information. An extra database table which lists the topics of the course would allow topics to be added and modified without requiring changes to the application code. The information in this table and the concept lists in the

Topic table could be implemented in XML so that the entire course structure aswell as the content is reusable and extensible. The presentation would then be entirely independent of the content.

The implementation of the overlay model of the user's knowledge should be updated according to the results of competence testing during the course. This would allow a quantitative measurement of the level of knowledge the student has about a concept or topic rather than the boolean value of true or false as is currently implemented. This could not be implemented in the time frame available.

The design of the Adaptive Java application included adaptive collapsible linking and adaptive link annotation. Time constraints prevented the completion of the implementation of this functionality.

The original design of the demonstration was envisaged as using XLink and XPointer for their linking capabilites and the direct display of XML in an XML compliant browser. This would significantly reduce the programming effort required to implement adaptivity through Dynamic HTML. The XLink and XPointer specification however is incomplete.

The user should be given the option to change their global objective permanently. In the current implementation the global objective is stored in the database and the change to a local objective changes the display only until the user moves to a new topic. This change would be trivial.

The examples used in the course are the same for all users. It would be useful to have alternative examples depending on the user model. Interactive examples that allow the user to test their knowledge [Wade et al 97] would be of benefit to the learner.

The course could be implemented to use either Microsoft Explorer™ or Netscape Navigator™. The Netscape implementation would require reloading of the page when users modified their local objective.

Conclusions

Education and Technology

Technology can provide many benefits. The benefits are most powerful when technology is implemented in a transformative manner. The repercussions of the emerging global computer culture will involve a transformation in the structure of society itself. The pervasive influence of technology will increase as capacity expands and cost continues to decline. The benefits reaped come hand in hand with dependence.

Education is already becoming unmanageable without technology. This dependence will increase as educational institutions are restructured to maximise technology benefits. In practice traditional methods of education were rarely ideal and often excluding. As our understanding of learning is refined, it must inform the implementation of technology in education and its effectiveness should be continually reviewed and evaluated. The customisation and learner centric approach which technology can support is a positive contribution to the delivery of education.

Adaptive Hypermedia in Education

There is an element of trust inherent in educational Hypermedia systems. The developer of the system trusts that the user of the system has an adequate level of maturity and / or experience to use the system effectively. Authors of Adaptive Hypermedia systems transfer this trust in varying degrees to the system and allows the system to infer the requirements or objectives of the user and make decisions on their behalf. If the adaptivity is visible, the user must also explicitly trust the system. If the system is adaptable, the user can choose to trust the system or not. The user may be involuntarily dependent on the judgement of the system. The user may be aware of this dependence or be oblivious to it. The proliferation of intelligent user agents on the web constitute an interesting approach to sanitizing the chaos of the web for the individual user. The transferral of this approach to the educational domain is inevitable and will surely appear in the guise of adaptive hypermedia. The level of trust which is granted to computer systems will increase in parallel with the advances in technology in general and in Articificial Intelligence techniques in particular.

The value of adaptivity in hypermedia systems needs more evaluation. The evaluation of educational hypermedia needs more evaluation. Hypermedia is acclaimed for its role in the development of cognitive flexibility and the construction of conceptual mental models. Commendable as is this objective, cognitive flexibilty and conceptual models are not easy to measure. The success of a hypermedia educational course will be judged on the evaluation of students knowledge. The use of qualitative methods such as psychometric testing, though controversial, is increasing. It is likely that computer systems will perform these evaluations and their results will inform the design of future educational systems.

The effectiveness of Adaptive Hypermedia is dependent on the appropriateness of the representation of the knowledge and the accuracy of the representation of the learner. A prerequisite of both of these requirements is a comprehensive understanding of human behaviour and human learning and the effective transfer of this understanding to a computer. Neither of these requirements can be guaranteed at this time. For this reason encouraging the learner to collaborate with the computer is a more reliable option.

Metadata standards

Standards and specifications are being formulated by various organisations and interested parties in the area of educational technology. These de facto standards are technical specifications that are stable and widely used.

These organisations are collaborating in an effort to produce a de jure standard with an emphasis on consensus, convergence and interoperability. As an accredited standards body, the IEEE Learning Technology Standards Comittee is the primary focus and beneficiary of this effort.

Meanwhile the W3C is continuing in its effort to produce recommendations for technologies with metadata support. The XML version 1.0 specification has been endorsed as a W3C Recommendation; deployment will soon become widespread in industry. The XLink and XPointer specifications are as yet incomplete. The Dublin Core metadata initiative which has become an industry standard recently announced Version 1.1 as a proposed Dublin Core Community recommendation.

Current metadata models such as the LTSC LOM are useful for the searching, retrieval and management of courses. Metadata models and data type definitions should become publicly available at a granularity which can be used within a course by adaptive applications.

The use of metadata in adaptive Hypermedia

XML is set to become a widely accepted data format. The separation of structure and presentation will enable data to be used for display to the user, transferral to another application for further processing or alternative presentation on a multitude of devices from palm tops to television screens. The ability of an application to 'understand' the meaning of the data it is processing is a considerable advantage in the area of adaptive hypermedia where the presentation of the expert knowledge is dependent on the representation of the user model. The overhead incurred in the authoring of adaptive systems will be reduced by the use of a common format to describe the meaning of all of the component parts of a course and the meaning of the represented properties of a user. This meaning can be used by applications to implement adaptivity for their own purposes.

The construction and intercommunication of user models over the lifetime of an individual will allow systems to more accurately chart user characteristics over time. This approach though increasing the effectiveness of computer systems introduces implications for the privacy of the individual. Users should be made aware of the way in which the system tracks information about them and the way in which their knowledge is evaluated. The use of such personal information should be voluntary and transparent.

[Chang, Harkey 98] describe the web as a universal infrastructure with the Java programming language as a universal programming language and XML as a universal data representation. These technologies comprise a powerful strategy for hypermedia applications which will encourage the design of systems which are inherently intelligent but allow interoperability with XML as the lowest common denominator service. The meaning inherent in XML data will enable the design of extensible systems which enable reuse, repurposing and restructuring of data.

System Architecture

The benefits of Adaptive Systems must justify the complexity of authoring that is often incurred in the delivery of individualised customisation. The Architecture described in Chapter 5 provides an extensible framework by creating an XML DTD which describes the required properties of a course concept. This structuring of course content increases the design flexibility of the application, simplifies the repurposing and reuse of the course material and constitutes a substantial increase in the value of the initial implementation. The separation of adaptivity and content adds further value by the ease with which the adaptivity can be modified, redesigned or hidden entirely.

Future work

Future work in this area may also include the description of a metadata model which describes independent units of courseware at the finest viable level of granularity and the development of a corresponding data type defintion for use in adaptive systems. Comprehensive evaluation of the usefulness of adaptive Hypermedia systems should be undertaken. Similarly the privacy issues introduced by communicable student records should be researched. The development of domain independent tools for the construction of knowledge representations and user models would ease the authoring of adaptive systems.

Summary

The completion of this dissertation necessitated a substantial amount of reading in the areas of educational theory and adaptivity in Hypermedia Systems and the mastery of a number of technologies including XML, DOM, JDBC, SQL, Oracle, CSS, Javascript, which represented a substantial learning curve. This final chapter has evaluated the dissertation and presented its conclusions.

Bibliography

[AICC 99]

Aviation Industry CBT Committee. http://www.aicc.org/

[Anez 1999]

Java XML Parsers. A Comparative Evaluation of 7 free tools. Juancario Anez. http://archive.javareport.com/9902/html/products/prod_rev.shtml. February 1999

[Appleyard 99]

The Battle of Wits – pg 44 The Sunday Times Magazine 1999

[Ariadne 99]

http://ariadne.unil.ch

[Atherton 1999]

James Atherton. http://websites.ntl.com/~james.atherton/learning/CONTENTS.HTM. January 1999

[Barker et al 1998]

Mental Models and their role in Teaching and Learning. Philip Barker, Paul van Schaik, Spencer Hudson, Check Meng Tan. Mental Model Research Initiative Human-Computer Interaction Laboratory University of Teesside, Borough Road, Middlesbrough, TS1 3BA United Kingdom. Proceedings of Edmedia pg 105. 1998

[Boumphrey et al 98]

XML Applications. Frank Boumphrey, Olivia Direnzo, Jon Duckett, Joe Graf, Paul Houle, Dave Hollander, Trevor Jenkins, Peter Jones, Adrian Kinglsey-Hughes, Kathy Kingsley-Hughes, Craig McQueen and Stephen Mohr. Wrox. ISBN 1-861001-52-5. 1998

[Brusilovsky 96]

Methods and Techniques of Adaptive Hypermedia. Peter Brusilovsky. User Modeling and User-Adapted Interaction 6: pg. 87-129 Kluwer Academic Publishers. The Netherlands. 1996

[Brusilovsky et al 96 - 1]

ITS on the World Wide Web. Peter Brusilovsky, Elmar Schwarz, Gerhard Weber. Intelligent Tutoring Systems. Proceedings of Third International Conference, ITS '96. Montreal, Canada June 1996 Claude Frasson, Gilles Gauthier, Alan Lesgold (Eds.) 1996

[Brusilovsky et al 96 - 2]

A Tool for Developing Adaptive Electronic Textbooks on WWW. Peter Brusilovsky, Elmar Schwarz, Gerhard Weber. Proceedings of AACE WebNet. 1996

[Brusilovsky, Schwarz 97]

Concept-based Navigation in Educational Hypermedia and its implementation on the WWW. Peter Brusilovsky and Elmar Schwarz. Proceedings of Edmedia. 1997

[Busch, Olsen]

Cascading Style Sheets Complete. David D. Busch & J.W. Olsen. Mc Graw-Hill. ISBN 0 - 07 - 913703 – 2

[Cairncross 95]

The Death of Distance. Frances Cairncross. The Economist Newspaper. September 30th 1995

[Cairncross 97]

A connected world. Frances Cairncross. The Economist Newspaper. September 13th 1997

[CEdma 99]

http://www.cedma.org/

[CEN 99]

The Commitee for European Normalisation. http://www.cenorm.be/

[Chang, Harkey 98]

Client/Server Data Access with Java and XML. Dan Chang, Dan Harkey. Wiley. ISBN - 471 - 24577 – 1. 1998

[Dao, Parent]

Adaptive Learning Model for Workplace Environmental Training. Kim Dao, Dr. Lise Parent. Proceedings of Edmedia pg 1611

[De Bra 98]

Definition of hypertext and hypermedia. Hypertext Course. Paul de Bra. Eindhoven University of Technology. 1998. http://wwwis.win.tue.nl:8001/2L690/

[De Bra, Calvi 97]
AHA: a Generic Adaptive Hypermedia System. Paul De Bra, Licia Calvi.
http://wwwis.win.tue.nl/ah/

[De Bra, Calvi 98]
AHA! An open Adaptive Hypermedia Architecture. Paul De Bra, Licia Calvi. http://wwwis.win.tue.nl/ah/

[De La Passardiere, Dufresne 92]
Adaptive Navigational Tools for Educational Hypermedia. Brigitte de La Passardiere, Aude Dufresne. Computer Assisted Learning, I. Tomek (Ed.) (pg. 555 - 567) Berlin, New York: Springer Verlag. 1992

[Drucker 94]
The Age of Social Transformation. Peter F. Drucker. The Atlantic Monthly. November 1994

[Dublin Core 99]
http://purl.oclc.org/dc/

[Duval 98]
Extension of Dublin Core for Educational material. Erik Duval. Univ. of Leuven Report of the Metadata Workshop held in Luxembourg. 26 June 1998

[Eklund 95]
Cognitive models for structuring Hypermedia and implications for learning from the world wide web. John Eklund. Proceedings of AusWeb. 1995

[Eklund et al 97]
A Study of Adaptive Linik Annotation in Educational Hypermedia. John Eklund, Peter Brusilovsky, Elmar Schwarz.
http://wwwis.win.tue.nl/ah/ 1997

[Eklund, Brusilovsky 98]
The Value of Adaptivity in Hypermedia Learning Environments: A Short Review of Empirical Evidence. John Eklund, Peter Brusilovsky http://wwwis.win.tue.nl/ah/ 1998

[Eklund, Sawers 98]
Customizing Web-Based Course Delivery in WEST with Adaptive Navigation Support. John Eklund, James Sawers. http://wwwis.win.tue.nl/ah/ 1998

[Espinoza, Hook 95]

An Interactive WWW Interface to an Adaptive Information System. Fredrik Espinoza, Kristina Hook. Proceedings of the User Modelling for Information Filtering on the World Wide Web, a mini-workshop at the Fifth International Conference on User Modelling. http://wwwis.win.tue.nl/ah/ 1995

[Fink et al 96]

User-Oriented Adaptivity and Adaptability in the AVANTI project. Josef Fink, Alfred Kobsa, Andreas Nill. http://wwwis.win.tue.nl/ah/ 1996

[Galitz 97]

The essential guide to user interface design : an introduction to GUI design principles and techniques Wilbert O. Galitz. New York. - Chichester : Wiley, ISBN 0471157554. 1997

[GESTALT 99]

GESTALT Getting Educational Systems Talking Across Leading Edge Technologies. Work Package 3, D0301 Design and Specification of the RDS (Resource Discovery Service) Reference A367/TCD/WP03/DS/L/003/b1 26-Mar-99

[Goodman 98]

Dynamic HTML, The Definitive Reference. Danny Goodman. O'Reilly. ISBN 1 - 56592 - 494 – 0 1998

[Greenberger 64]

The Computers of Tomorrow. Martin Greenberger. The Atlantic Monthly. May 1964

[Hearst 99]

Modern Information Retrieval. Chapter 10: User Interfaces and Visualization
Marti Hearst. This chapter appears in Baeza-Yates and Ribeiro-Neto: Modern Information Retrieval published and copyright Addison Wesley Longman Limited 1999 and is reproduced by arrangement with the publishers at http://www.sims.berkeley.edu/~hearst/papers/irbook/newtoc.html 1999

[Holcombe et al 98]

Computer Social Interaction and Individual Cognitive Growth: Vygotsky's Role of Language - Both Social and Private Speech. Dr. Terry L. Holcomb, Dr. Jon I. Young, Dr. Supranee Kuhataparuks. Proceedings of Edmedia pg 167. 1998

[Holowcza 99]

Oracle SQL *Plus: An introduction and Tutorial. http://cimic.rutgers.edu/~holowcza/oracle/sqlplus/

[Huffman et al 97]
Psychology in Action Fourth Edition. Karen R. Huffman, Mark Vernoy, Judy Vernoy
John Wiley and Sons, Inc., 1997

[Hunt 97]
Designing Instruction for the Web: Incorporating New Conceptions of the Learning Process.
Nancy P. Hunt. Proceedings of Edmedia. 1997

[Idris 99]
XML, Java, databases and the Web. Nazmul Idris. http://www.developerlife.com. 1999

[IMS 99]
Instructional Management System. IMS Meta-Data Specification: Draft Document Meta Data Version 1.02 3/2/99
http://www.imsproject.org. 1999

[Johnson 99]
XML for the absolute beginner. A guided tour from HTML to processing XML with Java.
Mark Johnson. http://www.javaworld.com/jw-04-1999/jw-04-xml_p.html

[Katz 96]
The Relationship Between Learner's Goal Orientation and Their Cognitive Tool Use and Achievement in an Interactive Hypermedia Environment. Heather A. Katz. Proceedings of Edmedia pg 1697

[Kay, Kummerfeld 95]
User model based filtering and customisation of web pages. J. Kay, R.J. Kummerfeld.
http://wwwis.win.tue.nl/ah/

[Kayama, Okamoto 97]
A mechanism for knowledge navigation in Hyperspace with Neural Networks to Support Exploring activities. Mizue Kayama, Toshio Okamoto. Proceedings of Edmedia.

[Lai 98]
User-Modelling in Artificial Intelligence and Human-Computer Interaction. Jim W. Lai.
http://www.io.com/~jwtlai/usermodel.html. 1998

[Laurillard 95]
Multimedia and the changing experience of the learner. Diana Laurillard. The British Journal of Educational Technology, Vol 26, No3. 1995

[Lee, Jong Wang 97]
Intelligent Hypermedia Learning System on the Distributed Environment. Sei-Hoon Lee, ChangJong Wang. Proceedings of Edmedia. 1997

[LTSC 99]
http://grouper.ieee.org/groups/ltsc/. 1999

[Mc Naught]
The impact of Information Technology on Practices and Policy in Higher Education. Carmel McNaught. Proceedings of Edmedia. pg 940.

[Milosavljevic 97]
Augmenting the User's Knowledge via Comparison. Maria Milosavljevic.
http://wwwis.win.tue.nl/ah/

[Milosavljevic et al 97]
Text Generation in a Dynamic Hypertext Environment. Maria Milosavljevic, Adrian Tulloch and Robert Dale. http://wwwis.win.tue.nl/ah/

[Milosavljevic, Oberlander 98]
Dynamic Hypertext Catalogues: Helping Users to Help Themselves. Maria Milosavljevic, Jon Oberlander Proceedings of the 9th ACM Conference on Hypertext and Hypermedia. June 1998

[MMI Workshop 99]
http://www.cenorm.be/isss/Workshop/MMI/Default.htm

[Newby 96]
hapters 1 – 3. Instructional technology for teaching and learning : designing instruction, integrating computers and using media. Timothy Newby. Prentice-Hall International (UK), 1996. ISBN 0023866950. 1996

[Newman 99]
The Process of Standardizing: The IMS Project's Place in the Standards Landscape. Denis Newman Director of Market Development, IMS Project. http://www.imsproject.org/standards.html

[Oppenheimer 97]
The Computer Delusion. Todd Oppenheimer. The Atlantic Monthly. July 1997

[Oracle 99]
http://www.oracle.com/xml/

[Paolucci 98]
Hypermedia and Learning: The Relationship of Cognitive Style and Knowledge Structure. Rocco Paolucci.
roceedings of Edmedia. 1998

[PAPI 98]
PAPI Specification Learning Technology: Public and Private Information. Public and Private Information (PAPI) Date: 1998-11-26 Version 5.00 incomplete. edutool.com A division of Farance Inc. http://edutool.com/papi. 1998-11-26

[Pitts-Moultis, Kirk 99]
XML Black Book. Natanya Pitts-Moultis, Cheryl Kirk. Coriolis Technology Press.
ISBN 1 - 57610 - 284 - X

[Reeves 98]
Answering critics of Media and Technology in Education. Thomas C. Reeves.
Proceeding of Edmedia. 1998

[RFC 2396]
Uniform Resource Identifiers (URI): Generic Syntax, Internet RFC 2396.
http://info.internet.isi.edu/in-notes/rfc/files/rfc2396.txt

[Richards 98]
The Emergence of Open Standards for Learning Technology. Tyde Richards. Macromedia Interactive Learning Division. http://www.learnativity.com/body_standards.html. September 2, 1998

[Russell, Norvig 95]

Artificial Intelligence, A Modern Approach. Stuart Russell, Peter Norvig. Prentice Hall International Editions New Jersey Section 19 pg 561 – 597. 1995

[Shneiderman 99]

Designing information-abundant web sites: issues and recommendations.
Ben Shneiderman. http://ijhcs.open.ac.uk/shneiderman/shneiderman.html

[Sherry Chang 98]

Learning Theory and Advertising. Min-Yu Sherry Chang.
http://www.utexas.edu/coc/admedium/Spring98_Theory/learning.html

[Siple 98]

The Complete Guide to Java Database Programming. JDBC, ODBC and SQL.
Matthew Siple. MCGrawHill ISBN 0-07-913286-3

[Specht 98]

Empirical Evaluation of Adaptive Annotation in Hypermedia. Marcus Specht.
Proceedings of Edmedia. 1998

[Tuckey 92]

Uses of New Technology in Higher Education - Guiding Principles. Catherine Tuckey. ICBL Reports, 001/92 Institute for Computer Based Learning, Heriot-Watt University, Edinburgh. 1992

[UM 99]

7th International conference on User Modellling. home page
http://www.cs.usask.ca/UM99/. 1999

[Vannevar 45]

As We May Think. Vannevar Bush. The Atlantic Monthly. July 1945

[W3C RDF]

http://www.w3.org/RDF

[W3C XML]

Frequently Asked Questions about the Extensible Markup Language Version 1.41 (6 October 1998)
W3C Special Interest Group http://www.ucc.ie/xml/faq.html

[W3C]

http://www.w3.org/TR/REC-xml

[Wade et al 97]

Design and Delivery of Tele-educational Courses. Vincent P. Wade, Mark Riordan, Conor Power. Journal of Distance Education. Vol. XII, No 1 / 2 Simon Fraser University Publications. 1997

[Wade, Lyng 99]

Experience Improving WWW based Courseware through Usability studies. Vincent P. Wade, Mary Lyng. 1999

[Wedekind et al 98]

Teaching and Learning with Flexible Hypermedia Learning Environments. Dr. Joachim Wedekind, Martin Lechner, Dr. Sigmar-Olaf Tergan. Proceedings of Edmedia. 1998

[Weibel 99]

The State of the Dublin Core Metadata Initiative. April 1999. Stuart Weibel D-Lib Magazine Volume 5 Number 4. ISSN 1082-9873 http://www.dlib.org/dlib/april99/04weibel.html. April 1999

[xmlsoftware.com 99]

XML software. http://www.xmlsoftware.com/parsers/