INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI[®]

NOTE TO USERS

Page(s) not included in the original manuscript and are unavailable from the author or university. The manuscript was microfilmed as received.

36 - 46

This reproduction is the best copy available.

UMI[®]

Metadata Management for Multimedia Interactive TeleLearning System

by

Lei Yuan

A Thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfillment of
the requirements for the degree of

Master of Computer Science

Ottawa-Carleton Institute for Computer Science

School of Computer Science

Carleton University

Ottawa, Ontario

April 4, 2001

© copyright

2001, Lei Yuan

National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-61013-6

Canada

The undersigned recommend to the Faculty of Graduate Studies
and Research acceptance of the thesis

“Metadata Management for
Multimedia Interactive TeleLearning System”

submitted by Lei Yuan
in partial fulfillment of the requirements for
the degree of Master of Computer Science

A handwritten signature in black ink, appearing to read "A. Karmore", is written over a horizontal line.

Thesis Supervisor

A handwritten signature in black ink, appearing to read "Frank Dehn", is written over a horizontal line.

Director, School of Computer Science

Carleton University

April 4, 2001

Abstract

The growth of the Internet and our needs for learning has produced massive amounts of telelearning resources. However inefficient metadata management has created a critical constraint on our ability to discover, manage and use these resources.

This thesis aims at improving information retrieval and knowledge discovery in telelearning by effective metadata management. The Metadata Management System implemented in this thesis will facilitate the creation, storage and utilization of learning object metadata, and help distant learners to achieve the optimum learning effect.

We adopted some existing metadata specification and added multimedia metadata as a special class. The system was designed and implemented as a database driven three-tier Web application. Metadata was collected through author input or generated by system. XML was used as the data transfer format between different tiers to facilitate information interchange. Once completed, this system architecture will be incorporated into the Multimedia Interactive TeleLearning System (MITS) developed at the Multimedia and Mobile Agent Research Laboratory of the University of Ottawa.

Acknowledgements

There are many people I would like to acknowledge for completing this thesis. They either made this work possible or helped me maintain a semblance of sanity long enough to get through it all.

My work with Professor Ahmed Karmouch, my thesis supervisor was by far the most rewarding experience I have had in Ottawa-Carleton Institute for Computer Science. I had learned so much about scientific research and have had a great time as well.

My sincere appreciation goes to the Multimedia Information & Mobile Agent Research Laboratory group members: Zhongyao Zhang, Ali Al-Shammari, Beilei Huang, Thong Pham, Hicham Ouahid, Yaoping Wang, Ben Fulchuk and Zheng Cui. They have provided the wonderful academic atmosphere for me to work in.

My wife, Qing Zhu, for all the support she provided. Without her effort, I would not have finished my thesis with full-time working. And our daughter, Linda Yuan, who was born in the new millennium, gave me so much pleasure of being a father.

I would also like to thank my parents and my brother for their unfailing love and support which helps me to keep improving myself.

TABLE OF CONTENTS

1	INTRODUCTION.....	1
1.1	BACKGROUND.....	1
1.2	WEB-BASED DISTANCE LEARNING (WBDL).....	2
1.2.1	Advantages of WBDL.....	3
1.2.2	Disadvantages of WBDL	4
1.2.3	Criteria in evaluating WBDL	5
1.2.4	WBDL Types	6
1.3	OBJECTIVES.....	8
1.4	OUTLINES.....	8
2	RELATED WORKS	10
2.1	MATADADA MANAGEMENT OF DISTANCE LEARNING RESOURCES	10
2.1.1	Introduction to Metadata	10
2.1.2	Metadata Characteristics	11
2.1.3	Metadata in multimedia systems	12
2.1.4	Research Works in Metadata Management.....	14
2.2	ACTIVE SERVER PAGES (ASP)	19
2.2.1	Overview of Active Server Pages	19
2.2.2	How does it work	20
2.2.3	Script in an ASP page.....	21
2.2.4	ASP Objects	21

2.3	EXTENSIBLE MARKUP LANGUAGE (XML)	23
2.3.1	What is XML.....	23
2.3.2	Compare XML with HTML and SGML	23
2.3.3	The advantages of XML.....	24
2.4	COMPONENT OBJECT MODEL (COM)	25
2.5	TRANSACTION SERVER.....	27
3	MULTIMEDIA INTERACTIVE TELELEARNING SYSTEM (MITS).....	29
3.1	DISTANCE EDUCATION MODELS	29
3.1.1	Broadcasting Model	30
3.1.2	CD-ROM Courseware Model	30
3.1.3	Distributed Computer-Based Model	32
3.2	DISTANCE EDUCATION PARTICIPANTS	34
3.2.1	Distance Education Designers.....	34
3.2.2	Teacher	35
3.2.3	On-line Facilitator	35
3.2.4	Courseware Author	36
3.2.5	Distant Learners	36
3.3	MULTIMEDIA INTERACTIVE TELELEARNING SYSTEM (MITS)	36
3.3.1	Objectives and Requirements of MITS	37
3.3.2	Generic System Architecture	38
3.4	THE METADATA MODEL OF MITS	41
3.4.1	Actors in MITS.....	41
3.4.2	Actions for metadata	42

3.4.3	Metadata Structures.....	43
3.4.4	Metadata Creating	45
4	DESIGN OF THE METADATA MANAGEMENT SYSTEM (MMS) FOR MULTIMEDIA INTERACTIVE TELELEARNING SYSTEM (MITS).....	46
4.1	GENERAL SPECIFICATION.....	46
4.2	SYSTEM ARCHITECTURE	47
4.3	USER SERVICE TIER	50
4.4	BUSINESS LOGIC TIER	51
4.4.1	Courseware Rendering Manager.....	51
4.4.2	Courseware Presentation Manager.....	52
4.4.3	Courseware Evaluation System.....	53
4.4.4	Client Access Controller	54
4.4.5	Profile handler	54
4.4.6	Web Server.....	54
4.5	DATA SERVICE TIER	54
4.5.1	Courseware Database	55
4.5.2	Media Content Server.....	58
4.6	ONLINE ASSESSMENT AND TESTING SYSTEM (OATS)	59
4.6.1	Information Model of OATS.....	59
4.6.2	Q/A Block	60
4.6.3	Evaluation Unit	62
4.6.4	Components of OATS	64
4.7	TASKS OF THE OATS	67

4.7.1	Identify an Author	68
4.7.2	Add Media Object	69
4.7.3	Author Create Question.....	69
4.7.4	Organize Exam.....	70
4.7.5	Identify a student.....	71
4.7.6	Select an exam.....	72
4.7.7	Take an Exam.....	73
4.7.8	System Store Exam Record.....	74
4.7.9	Feedback.....	74
5	SYSTEM IMPLEMENTATION	75
5.1	INTRODUCTION.....	75
5.2	THE IMPLEMENTATION OF DATA SERVICE.....	76
5.2.1	Overview	76
5.2.2	Courseware Database	77
5.2.3	Profile Database	77
5.2.4	Database Access Control.....	77
5.2.5	Media Object Repository	79
5.3	THE IMPLEMENTATION OF BUSINESS SERVICES	80
5.3.1	Data Access objects.....	80
5.3.2	The Business Objects	83
5.4	IMPLEMENTATION OF USER SERVICE	84
5.4.1	Web Site Information Structure	85
5.4.2	LearnerLogin.asp.....	86

5.4.3	AuthorLogin.asp.....	89
5.4.4	AddImage.asp.....	90
5.4.5	Upload.asp.....	91
5.4.6	AddVideoClip.asp.....	93
5.4.7	AddQuestion.asp.....	95
5.4.8	MakeExam.asp.....	100
5.4.9	SearchExam.asp.....	105
5.4.10	ExamGate.asp.....	107
5.4.11	TakeExam.asp.....	108
5.4.12	ExamResult.asp.....	111
6	CONCLUSIONS AND FUTURE WORK	112
7	REFERENCES	115

List of Tables

Table 3-1	Data Fields for each class of Metadata.....	45
Table 4-1	Media Specific Information	66
Table 5-1	ASP source related with student registration	87

List of Figures

Figure 3-1	MITS' Major Components	40
Figure 4-1	System Architecture	48
Figure 4-2	System Data Flow	49
Figure 4-3	Courseware Data Object Model	56
Figure 4-4	OATS System Structure	65
Figure 5-1	The data model of Courseware Database	78
Figure 5-2	The data model of Profile Database	79
Figure 5-3	Web Site Information Structure	86
Figure 5-4	Author Logon Dialog Box.....	89
Figure 5-5	Author authenticated Logon dialog box.....	90
Figure 5-6	Upload images dialog box.....	91
Figure 5-7	File-choosing dialog box.....	92
Figure 5-8	File name input dialog box.....	92
Figure 5-9	Image information form I dialog box	93
Figure 5-10	Image information form II dialog box.....	94
Figure 5-11	Welcome dialog box.....	96
Figure 5-12	Type of questions dialog box I.....	96
Figure 5-13	Question content and answer type dialog box.....	97
Figure 5-14	Multiple correct answer dialog box.....	97
Figure 5-15	Free-response question dialog box	98
Figure 5-16	Single correct answer dialog box	98
Figure 5-17	The explanation for the questions dialog box	99

Figure 5-18 Confirmation of sending data to the server dialog box	99
Figure 5-19 Creating new exam dialog box I	100
Figure 5-20 Creating new exam dialog box II	101
Figure 5-21 Questions selection dialog box.....	102
Figure 5-22 Exam organizing dialog box I	103
Figure 5-23 Exam organizing dialog box II.....	104
Figure 5-24 Exam organizing dialog box III	105
Figure 5-25 Exam searching dialog box I.....	106
Figure 5-26 Exam searching dialog box II.....	107
Figure 5-27 Question presenting dialog box I	108
Figure 5-28 Question presenting dialog box II	109
Figure 5-29 Question presenting dialog box with answers III.....	109
Figure 5-30 Question presenting dialog box IV.....	110
Figure 5-31 Question presenting dialog box V.....	111
Figure 5-32 Exam result dialog box.....	111

1 Introduction

1.1 Background

Based on the backgrounds and knowledge, the terms “TeleLearning” and “Distance Education” have been used interchangeably by researchers [26].

Distance learning is a process and a system of connecting learners with distributed learning resources. It makes education and training more accessible, convenient, effective, and cost-effective for both the education provider and learners. TeleLearning is an integrated system for the planning, delivery and management of training programs through the use of advanced telecommunication services. TeleLearning allows organizations to deliver training from an instructor-led location to students at remote sites located anywhere in the world. The students can interact with the instructors and each other as if they were together in the same classroom.

In the late 1950's and early 1960's, distance education came into being when broadcasting Radio classes were transmitted to customers in Europe. No interaction or bi-directional communications between the teacher and the learners was provided in this broadcasting model. Newly designed and produced TV series introduced students to new subject areas that could not be learnt in classroom in the late 1970's, so they were considered to be an important complement to the classroom curriculum [26]. Meanwhile, the development of communication technologies, such as email, affected the trend of distance education by providing some kind of interaction between teacher and students.

With the development of technology and economy there is a growing need for more training and education and a similar need for more efficient and cost-effective delivery of this training. Universities are facing shrinking institutional budgets, while 75% of the existing workforce will need retraining in the next 10 years. To meet these demands, efficient training solutions must be developed rapidly. Whether for formal education, continuing professional development, or for employee training or retraining, distance learning solutions need to deliver the right kind of educational content to learners - anytime, anywhere. From students to doctors, from consumers to business people, voice, data and multimedia services along with collaborative learning can be available for access from the home, hospital, school, government office or anywhere that people require interaction with other people and information. With this network a community can begin to realize a solution for "lifelong learners".

1.2 Web-based Distance Learning (WBDL)

WBDL is the distance education that is delivered over the Internet or over a company's Intranet. Web browser, such as Netscape Communicator/Navigator or Microsoft Internet Explorer is used for accessing the training program. Other types of Internet training refer to any program that can be delivered from a remote source, even email correspondence courses, or the transfer of files of course materials. Training over the World Wide Web, and training using an Intranet's Web, specifically refers to the readily available, interactive, multimedia nature of Web browsers and associated plug-ins. WBDL can be delivered to any computer, anywhere, that can access the Internet or company Intranet [11].

1.2.1 Advantages of WBDL

WBDL has a lot of advantages compared with other computer-based training [12]

- *The technology is cross platform*

Unlike other forms of computer-based training, WBDL can be accessed by Windows, UNIX (including Linux) or Macintosh users, usually without requiring additional software, as far as the system is equipped with a Web browser. Usually the courseware is authored once, and delivered to any machine over the Internet or the Intranet.

- *Accessibility, convenience, flexibility*

Users can access the training at the time of their need, and only as much as they need, known as “Just in time and just enough”. A training program could also be proceeded at the users’ own pace and at their own place.

- *Wide availability*

These days more and more people have access to a Web browser, and are connected to a company’s Intranet or the Internet, which makes accessing WBDL really easy.

- *Inexpensive worldwide distribution*

WBDL can be accessed from any computer anywhere in the world, greatly reducing the distribution costs associated with training in other media.

- *Cost savings and time savings*

Because the Internet can be accessed from any location, there are no travel costs for bringing learners to a centralized school or workshop. The actual time required for training by computer average about 50 percent that of instructor-led training, additional cost savings are realized.

- *Ease of update*

If changes need to be made in the program after the original implementation, they can be made on the server where the programs are stored and all learners can access the update. Courses can be designed to access designated current information, from any other server worldwide for on-the-fly update whenever anyone accesses the program.

1.2.2 Disadvantages of WBDL

- *Limited Bandwidth*

Limited bandwidth means slower data transmission and bad performance for audio, video, and even extensive graphics. These restrictions can cause long delays and can adversely affect the learning process. The problem can be serious over the public Internet where more traffic jams occur, but is less severe on a company's Intranet, which usually has higher bandwidth.

- *Lack of human contact*

Actually it is a big concern that glowing screens are replacing friendly faces. Decreasing instructor-led training makes some trainees uneasy. Today's WBDL programs are sometimes so static and the level of interactivity is often very limited. This is gradually improving, and as it does the impact of the training on performance also improves.

- *WBDL takes more time and more money to develop than expected*

Like any first-time challenge, learning about and implementing new technology takes more resources than expected. However, people can make it easier by starting with a simple program and building on success.

1.2.3 Criteria in evaluating WBDL

Here are ten recommended criteria for evaluating WBDL by the semiannual Multimedia and Internet Training Awards [11] [21]:

1. Content: Does the program include the right amount and quality of information?
2. Instructional Design: Is the course designed in such a way that users will actually learn?
3. Interactivity: Are users engaged through the opportunity of their input?
4. Navigation: Can users determine their own way through the problems? Is there an exit option available? Is there a course map accessible? Is there an appropriate use of icons and/or clear labels so that users don't have to read excessively to determine program options?
5. Motivational Components: Does the program engage users novelty, humor, game elements, testing, adventure, unique content, surprise elements, and so on?
6. Use of Media: Does the program appropriately and effectively employ graphics, animation, music, sound, and video? Is the gratuitous use of these media avoided?
7. Evaluation: Is there some type of evaluation, such as completion of a simulation? Is mastery of each section's content required before proceeding to later sections? Are section quizzes used? Is there a final exam?
8. Aesthetics: Is the program attractive and appealing to the eye and ear?
9. Record Keeping: Is student performance data recorded, such as time to complete, question analyses, and final scores? Is the data forwarded to the course manager automatically?

10. Tone: Is the program designed for its audience? Does it avoid being condescending, trite, pedantic, and so on?

1.2.4 WBDL Types

In terms of interactivity and amount of multimedia contents, WBDL can be sorted into three types [12]:

- **Type 1: Text and Graphics WBDL Programs.**

Text and graphics programs are sometimes simply paper-based course materials placed on the Web so that student can access them in e-format. Sometimes these courses are assigned the most pejorative of appellations: “page turners”, which is very negative for computer-delivered training, since no one enjoys using a program only to read the screen and click the “Next” button. On the other hand, some text and graphics programs designed from the ground up for Web delivery can be an appropriate solution for a given training need. Many contain hyperlinks to other materials for further study, or charts and graphs that further illustrate the learning point. These pages usually provide a lower level of interactivity, although not necessarily a lower level of information. One of the advantages of text and graphics sites is that no special authoring tools are necessary.

The text and graphics model is the place many people start with development of WBDL, because it is the easiest to create and is accessible to the most people. However, in many cases it can only minimally be considered training. To gain greater effectiveness from this medium, one needs to move to the next level and add more interactivity.

- **Type2 – Interactive WBDL Programs**

To engage the learner with stimulating interactivity is the promise and the future of computer-delivered instruction, whether delivered on CD-ROM or the Web. Interactivity

and instructional design is to WBDL is as a good story line is to a movie or game play is to a video game. Interactivity at its best is a simulation of the work situation. At a minimum, it can include application exercises, drag-and-drop, column matching, testing, text entry, and even programming code entry. This goes beyond simple text and graphics presentation and brings the learner into the program to engage with the content and practice the skills.

- **Type 3 – Interactive Multimedia WBDL Programs**

Truly interactive multimedia is the Holy Grail of WBDL. Most programs that fall into this category allow the user to manipulate graphic objects in real-time, sometimes taking on the quality of a game-playing exercise. The simulations are realistic and the situations are often difficult. Appropriate use of audio and/or video helps from an instructional point of view and from the human side as well.

The promise of interactive multimedia training was first realized with CD-ROM technology. It allowed the large audio or video files to be stored on a portable disk and presented nicely on the computer screen, without the wait times of the Internet. The Web is following close behind with improved transfer speeds, and is an improvement on the CD-ROM in terms of storage space and ease of update.

The interactive multimedia type of WBT programs are usually authored with a software tool that allows for the programming necessary to add the multimedia and manage the high levels of interactivity and record keeping. Java applets can be built to manage this, but most authors prefer the traditional full-scale authoring tools, such as Authorware or ToolBook II, which have plug-ins for delivery over the Internet.

An interactive multimedia course allows the student to enter into a world that attempts to mimic a part of real life, providing immediate, real-life responses to user input. If something goes wrong, such as the student taking the wrong action in response to hazardous situation, the program lets him know by simulating a meltdown.

An excellent example of interactive multimedia WBT is the Web Interactive Training program at NASA's Kennedy Space Center [20].

Web-based interactive multimedia is an effective way to provide training in a risk-free environment. Users can see the result of their actions immediately. The use of graphics, audio, and video provides the advantage of making the environment more realistic and the training more effective and more enjoyable.

1.3 Objectives

The main objective of this thesis is to develop a Metadata management system to facilitate the creation, storage and utilization of learning object metadata, and help distant learners to achieve the optimum learning effect.

The system is database driven and based on XML and COM. Specifically, emphasis will be placed on the design of the system architecture. Once completed, this architecture will be used as part of the Multimedia Interactive TeleLearning System (MITS) developed at the Multimedia and Mobile Agent Research Laboratory of the University of Ottawa.

1.4 Outlines

The rest of the thesis is structured as follows:

Chapter 2 addresses the related works, which include the Metadata Management issues, and the introduction of XML, ASP and COM.

Chapter 3 introduces the Multimedia Interactive TeleLearning System (MITS) developed in our lab. The generic system architecture of MITS is described and the common requirements and objectives of MITS are identified.

Chapter 4 discusses the design of the Metadata Management System (MMS) for Multimedia Interactive TeleLearning System (MITS). In this chapter, the design issues, system architecture and major components of MMS are described. The system takes advantage of HTML/JavaScript, XML, COM, and metadata management theory to provide an innovative distance learning system over the Internet. It employs a three-tier client/server system architecture, which includes User Service Tier, Business Service Tier and Data Service Tier.

Chapter 5 gives the implementation details of MMS.

Chapter 6 concludes this thesis, and also provides suggestions for future research works.

2 Related Works

2.1 Matadada Management of Distance Learning Resources

2.1.1 Introduction to Metadata

As our need for learning expand dramatically, the number of distance learning resources grows exponentially over the Internet. However the potential users can not efficiently locate the educational material on the web [13] [29]. For example someone wants to take an online course about Perl programming, and the course should be multimedia-enabled and interactive. To do this search, usually he/she will go to a search engine web site like Altavista or Excite and type in keywords “multimedia, interactive, Perl, course”. It will not be a surprise that such a search will give over 4 million web pages that are related. Next, these links will be tried one after another.

This “try and error” method normally is inefficient, time consuming, and really frustrating. The potential of the Web for providing rich materials and experiences is often abandoned as teachers and learners find themselves relying solely on strategies of hit and miss. The result is an over-reliance on serendipity, coupled with persistent filtering. So a common methodology and language for describing online learning resources that will facilitate purposeful and effective searching is required.

The bibliographic information about a document required for efficient retrieval is known as “metadata”. Metadata is also defined as “data about data”. It is descriptive

information about an object. The learning resources are the data; the descriptive information about these resources is the metadata. Metadata records contain information about a resource in a structured format as values of labeled attribute [6]. Metadata from a variety of sites can be gathered, indexed, and then searched according to criteria that refers to specific attributes or identifies values that are of interest. The search should only return the materials whose records are appropriate matches to the query. Metadata allows the user to locate, evaluate, access, and manage available online learning resources in a more efficient way. Examples of metadata for learning materials are the title, the author, the targeted learning level, and the educational objectives of the material.

Metadata is distinct from, but intimately related to, its contents. Without changing the content, the author of the learning materials might change the title. The user's initial perception of the material may be altered, but the data itself keeps the same. Conversely, the author may change the content, which may increase the appropriate learning level of the material. The metadata must then be changed appropriately. In a word, metadata is a complement to its contents and reflects the content's attributes to interested users [19].

2.1.2 Metadata Characteristics

Metadata describes different aspects of data, including:

- Identification -- Who developed the data set? What geographic area does it cover?
What is the name of the data set? What themes of information does it include?
How current are the data? Are there any restrictions on accessing or using the data?

- Data Quality -- Is information available that allows a user to decide if the data are suitable for his or her purpose? How good are the data? What is the positional and attribute accuracy? Are the data complete? Is the consistency of the data verified?
- Distribution -- What is the price of the data? From whom can I obtain the data? What format is available? What media are available? Are the data available online?
- Spatial Reference and Organization Information -- How is the data organized (data models, topology)? How is the data referenced to the real world (coordinate systems, datum)?
- Entity and Attribute Information -- How is the information encoded? What geographic information (roads, houses, elevation, temperature, etc.) is included? Were codes used? What do the codes mean?

2.1.3 Metadata in multimedia systems

Development of TeleLearning systems faces the serious problem of how to handle large amounts of variety of multimedia. Therefore, there is a growing demand for database and information systems support in the area of management, modeling, and processing multimedia. There is a need to explicitly capture fair amount content-information as well as application-specific semantics by means of a variety of metadata, e.g., attribute-based annotations, multimedia indexes, and intentional descriptions, to allow appropriate access to, selection of, and processing of multimedia.

The development of metadata, including the generation, management, and proper utilization of data or information about data is one of the key problems to be solved. Metadata plays a far more important role in managing multimedia data than does the

management of traditional information retrieval techniques applied to text-only data.

Some of the reasons for this are:

- **Different query paradigm**

The exact-match paradigm for querying is no longer suitable for querying or retrieving various types of multimedia. Metadata of multimedia have the additional advantages of being more amenable to traditional data retrieval and manipulation techniques than the raw multimedia. For example, text annotations of images or other media can be managed by a relational database very easily. To identify and specify correlation at the metadata level rather than between raw data items is more useful and convenient.

- **Inefficiency**

Very large raw data volume is involved in various multimedia data. This has consequences on effective management and retrieval of multimedia. Although a content-based search is possible, it cannot be used very frequently for performance reasons.

- **Inadequate processing techniques:** Content-based processing techniques are hard to apply to BLOB data.

- **Semantics of multimedia**

Semantics and context are very important in dealing with multimedia. Through analyzing and processing, semantics can be made explicit to some extent on different abstraction levels, from feature values to knowledge-based concepts. However, metadata describing this semantics may still not be sufficient for exact-match

querying. A different querying paradigm is needed to allow a proper mapping of the user's ideas to the semantics and the raw data.

As metadata will possess much less volume than the multimedia themselves, processing and indirect retrieval using abstract information or metadata of the multimedia seem to be a promising approach to enhance querying and processing and to improve response time. However, in many situations metadata is not sufficient for managing multimedia.

2.1.4 Research Works in Metadata Management

2.1.4.1 Dublin Core

The original goal of the Dublin Core was to achieve consensus among librarians, scholars, and Internet standards makers on "a list of metadata elements that would yield simple descriptions of data in a wide range of subject areas" [29]. Principally all Dublin Core elements are optional and repeatable, though they may be otherwise constrained by individual implementers. In 1997 the set of 15 Core elements were stabilized [4] and the first document was submitted [30]. These elements can be divided into three groups:

- Elements related mainly to resource content: Title, Subject, Description, Source, Language, Relation, Coverage
- Elements related to copyright issues: Creator, Publisher, Contributor, Rights
- Elements related mainly to the instantiation of the resource: Date, Type, Format, Identifier

The combination of these elements with a capacity for extension through use of qualifiers or collection-specific elements has motivated implementation of Dublin Core

metadata in a great variety of disciplines. Implementers need to determine for themselves how to balance the need for richer bibliographic description and the need for efficient cataloging.

The Dublin Core has emerged as the dominant architecture for classification of web materials. The major education metadata projects described below either were defined as an extension of Dublin Core or adapted to the Dublin Core model during development.

2.1.4.2 Education Network of Australia (EdNA)

EdNA [5] is collaboration between Australian states and territories and all sectors of education and training.

The EdNA metadata record is based on the Dublin Core but does not use the Relation, Source, and Contributor elements. The following are proprietary to EdNA:

1. Most additional elements are related to managing the records themselves (meta-metadata), these elements include:
 - Entered: date of creation of metadata record
 - Approve: approver of item for inclusion
 - Suggestor: people who suggested item
 - Reassessment: months until resource should be reassessed
 - Categories: classification system in directory of resources
2. There are two EdNA elements which recognize the hyperlink of web resources and specify constraints on gathering documents to index:
 - IndexLevel: number of levels of links to follow
 - IndexSites: number of servers to access when following links

3. Two elements were added to the Dublin Core set which apply specifically to the educational characteristics of the resources. They are:
 - Review: third-party review of the resource
 - UserLevel: controlled vocabulary of user and school level. Limited to “students” and “teachers” only.

2.1.4.3 ARIADNE

ARIADNE [23] is part of the education and training program of the European Union Telematics Application Program. Its goal is to facilitate sharing and reusing of electronic pedagogical material by universities and corporations.

ARIADNE metadata elements are extensions of the Dublin Core. They can be grouped into several categories:

1. General information about the resource: Identifier, Title, Creator, Date, Language, Publisher, Source.
2. Semantics of the resource: here the Subject element of Dublin Core is divided into four sub-elements: the academic discipline, the main concept, synonyms, and other possible subject terms.
3. Pedagogical attributes: metadata elements specific to the educational characteristics of the resource.
 - user type: either “learner” or “author”
 - document type: includes “expositive” (learning from instruction or study) and “active” (learning by doing)
 - document format: value depends on document type

- usage (optional): text comments on how to use the resource

Specifically, if the value of user is “learner”, the following elements apply:

- didactical context: values from a list of learning styles
 - course level: country and educational level specified in that country (e.g., “US, K-12”)
 - difficulty level: 'low', 'medium', or 'high'
 - interaction quality and semantic density: 'low', 'medium', or 'high'
 - pedagogical duration: minutes needed by an average learner to use the resource
4. Technical characteristics: locator of documents for resource retrieval, format, file size, and installation information.
 5. Conditions for use: “free” or “not free”, and price for non-free resources.
 6. Meta-metadata: creator of the metadata, date, language, and version information about the metadata record itself.

2.1.4.4 Gateway to Educational Materials (GEM)

GEM [9] is sponsored by US Department of Education. This project has developed a metadata structure and cataloging program for distributed cataloging of K-12 resources. The project targeted lesson plans and curricular materials for cataloging.

GEM uses all the Dublin Core elements and extends Dublin Core with a variety of elements for K-12 curricular resources, including:

1. Audience: includes the immediate user of the resource and the student population to be served

2. Pedagogy: it has three sub-elements: teaching (instructional method), grouping (of students in classrooms) and assessment.
4. Quality and Standard: they are available to represent evaluations by outside agencies.
5. Duration: corresponds to pedagogical duration in ARIADNE

2.1.4.5 Information Management System

IMS is a cataloging system sponsored by software corporations, publishers, and institutions of that develop education resources on the Internet [16]. The system is designed to be extensible for a wide variety of resources.

IMS metadata defined distinct sets of metadata elements to characterize different kinds of resources [17]. This is the most important feature in IMS design. A small base set is mandatory on all resources while other metadata elements are optional. “Container” concept is used in IMS to define the set of metadata for a particular object. IMS also has the following features:

1. The Base Set of metadata contains the minimal elements to describe all resources cataloged in IMS. These elements are also defined in Dublin Core, which includes title, date, publisher, subject, description, format, and identifier. Meta-metadata is also included in the base set. This element includes the following items: author (of the metadata), creation date, last modification date, validator, and container type.
2. The container types include Item, Module, and Tool. The Item container is used for a unitary resource such as a text or image. The Module container is for a learning resource with a specific educational value or purpose, such as a course,

topic, assessment or assignment. The Tool container is for a learning resource that provides a function for the user, such as a word processor, calculator, or statistical analysis package.

3. The Module container uses the full set of IMS metadata elements; the Item and Tool containers use subsets of this set. The Item set adds only “author”, “price code” and “rights” to the Base set, providing minimal elements for the description of education resources that are not curricular materials. The Tool set includes the elements “user support” and “platform”. The Module container includes a particularly detailed set of elements for educational methods and objectives, including objectives mandated by government agencies.

IMS introduction of containers with distinct sets of metadata elements for different types of educational resources provides the convenience in cataloging by excluding unnecessary elements from a record structure.

2.2 Active Server Pages (ASP)

2.2.1 Overview of Active Server Pages

Active Server Pages is the latest server-based web technology from Microsoft, designed to create interactive HTML pages for an Internet web site, or corporate intranet site. It is just part of an all encompassing concept called the Active Platform which has been developed along with Microsoft Windows NT Server and Microsoft Internet Information Server (IIS) [8].

Simply speaking, an Active Server Page is a file with a .asp suffix that contains a combination of script logic and HTML statements. When IIS4 receives an HTTP request for an ASP file, the final HTML response is generated dynamically from the static HTML statements plus the insertion of any HTML generated by the scripting.

The concept of ASP is similar to Netscape's server-side JavaScript and Allaire's Cold Fusion. It is designed as a convenient alternative to old standards like CGI scripting using Perl scripts. But ASP supports VB script - a language anyone can learn in a day - is extensible with custom components and is free with every Microsoft NT Server license. ASP is also part of Microsoft's Dynamic Internet Application Architecture (DNA) which structures an Internet application into multiple layers or tiers and that is the point where the Web application Server fits into this strategy. Web application server is the middle-tier of an Internet application and communicates with the presentation layer (ASP) to offer the business logic of an application.

2.2.2 How does it work

When an ASP application is created, an HTML page with additional ASP script commands is created too. Normally according to the request URL, the HTTP server sends the file directly to the client's Web browser as a response. With an ASP page, IIS recognizes that this file has additional script commands and processes the script before sending the page to the client. The recognition depends on the file extension. While HTML pages get a .htm or .html extension, ASP applications get .asp as a file extension. Within an ASP script, standard HTML code is sent to the browser, while script commands are executed locally on the Web server. Script commands can create additional HTML code, which goes direct to the client [8].

2.2.3 Script in an ASP page

An ASP application can contain client-side and server-side script, even in different script languages. To distinguish every group of script commands, the script starts with the script tag `<SCRIPT>` and ends of course with the tag `</SCRIPT>`. This tag supports two attributes: `LANGUAGE` and `RUNAT`. `LANGUAGE` is for the script language chosen (VBScript or JavaScript) and `RUNAT` is for the machine where the code executes. The following segment contains server-side scripts:

```
<SCRIPT Language="VBScript" RUNAT="SERVER">  
...  
</SCRIPT>
```

The server identifies everything between these tags as server-side script and uses the specified language to interpret the script commands.

The following segment contains client-side scripts:

```
<SCRIPT Language="JavaScript">  
...  
</SCRIPT>
```

2.2.4 ASP Objects

ASP enables the scripting logic to interface to a number of ASP intrinsic objects, which automatically handle many of the menial tasks, and so simplifies the amount of development required [8]. The five predefined objects that come with ASP are **Application**, **Session**, **Request**, **Response**, and **Server**.

- **Application:** It is the object for the global application, independent of users accessing the application. The object stores all information that is valid for the whole application. The object remains active for the life of the application and doesn't go away until the application is unloaded or until the Web server stops.
- **Session:** The Session object is created and dedicated for each session. It can be helpful as a container to store any temporary value for a session that should not become a cookie.
- **Request:** The Request object is used to access information associated with the web environment or an HTTP request.
- **Response:** The Response object is for the output of data. This includes creating the HTML stream, cookies, and also environment information like the timeout for caching the page.
- **Server:** The most important function of the Server object is to create instances of ActiveX components. You use the CreateObject method to instantiate a new ActiveX component.

In addition, the real power of ASP comes from its ability to interface with external COM compliant software components, including those supplied with ASP, those provided by Windows NT, those supplied by software vendors, and those created by developers. For example, ASP is supplied with ActiveX Data Objects (ADO) that provides a high performance interface to databases that are ODBC or OLE DB compliant.

2.3 Extensible Markup Language (XML)

In just a few years, the World Wide Web and HTML have taken the world by storm. But the limitations of HTML and the ever-increasing demand for more flexibility in Internet systems has driven developers to go for XML, the Extensible Markup Language, a powerful tool for data representation, storage, modeling, and interoperation [2].

Hereunder is an introduction to XML: where it came from, why it's necessary, how to use it in designs, and its interoperability with existing Internet technology.

2.3.1 What is XML

XML (eXtensible Markup Language) is a markup language for documents containing structured information. Structured information contains both content (words, pictures, etc.) and some indication of what role that content plays. A markup language is a mechanism to identify structures in a document. The XML specification defines a standard way to add markup to documents.

2.3.2 Compare XML with HTML and SGML

XML is more versatile than HTML to deliver structured document over the web. The major purpose of HTML is to present information to Web browsers. In HTML, both the tag semantics and the tag set are fixed. The W3C, in conjunction with browser vendors and the WWW community, is constantly working to extend the definition of HTML to allow new tags to keep pace with changing technology and to bring variations in presentation (stylesheets) to the Web. However, these changes are always rigidly confined by what the browser vendors have implemented and by the fact that backward

compatibility is paramount. And for people who want to disseminate information widely, features supported by only the latest releases of Netscape and Internet Explorer are not useful.

Furthermore, XML is better suited to serving documents over the web than SGML, although it is a restricted form of SGML. SGML is the Standard Generalized Markup Language defined by ISO 8879. SGML has been the standard, vendor-independent way to maintain repositories of structured documentation for more than a decade, but it is too complex and difficult to implement just for a web browser. Full SGML systems solve large, complex problems that justify their expense. Viewing structured documents sent over the web rarely carries such justification.

2.3.3 The advantages of XML

Some benefits of representing information in XML is listed below [2]:

- **XML is straightforward to use over the Internet.** Even without robust and widely available XML browsers, users can view XML documents easily by using editors.
- **Tags have nothing to do with how the document is displayed.** The markup indicates what the information *means*, not how to display it. The formatting information for an XML file (if there is any need for formatting) is usually written in a style language and stored separately from the XML. Separation of content and presentation is a key concept inherited from SGML.
- **XML documents are easy to create.** Although there are sophisticated editors to create and edit XML content, a good alternative is to create XML documents directly in a text editor, or with simple scripts, etc.

- **Relatively easy to process from programming point of view.** High quality XML parsers from different vendors are available for free and a lot of programming languages are supported: Java, C, C++, Visual Basic, Perl, etc. If you write a DTD and use a validating parser, the parser does most of the error checking and input validation. Moreover, if your system needs to interoperate with other systems, you can choose a standard DTD (like XML/EDI, for example), so that other systems will automatically understand your system's vocabulary, and vice versa.
- **XML support a wide variety of applications.** XML is beneficial to a wide variety of diverse applications: authoring, browsing, content analysis, etc.

2.4 Component Object Model (COM)

COM is a binary standard for object interaction. The fundamental goal of COM is to enable people to create applications that are assembled from pre-built parts — that is, components. For example, an order entry application might use a data entry grid component to make entering the ordered items easier. Another component could look up a customer's city and state based on the zip code entered. Yet another component could compute sales tax on the order. To make this type of component-based application a reality, several technical challenges must be met [31].

The first challenge is to provide a standard way to locate components and create objects using those components. Otherwise, the overhead of learning how to use every component and coding the logic to locate the components and create objects presents a barrier to reusing pre-built code. In conjunction with the standard mechanisms, there

must be a way to uniquely identify components that are created independently by different vendors, to prevent scenarios in which an application developer asks for one type of object and gets a completely different type. Component developers must be able to generate unique identifiers for their components and provide those components to others without going through any centralized authority.

The second challenge is to provide a standard way of interacting with objects. Again, if no standards exist, the overhead of learning how to use the objects creates a barrier to reuse. Ideally, this standard mechanism for object interaction would not care where an object was located — in the application's process, in another process on the application's machine, or on another machine altogether. Usually, interprocess and remote communications require a great deal of complex coding in the application. However, if there is a standard way to interact with objects, it should be possible to hide all this complexity from the application developer — and hopefully from the component developer as well.

Third, application developers must not have to worry about what programming language or tool was used to create the components they want to use. Otherwise, the component market becomes fragmented and the cost of identifying and purchasing components that will work in a certain environment is too high. True language independence is a complex challenge for an object model. Everything about the object — memory allocation, method names, parameter types, calling conventions, and so on — must be defined in such a way that the object can be created by code written in one language and used by code written in another.

Finally, it must be possible to create new versions of applications and components. This requirement sounds trivial until you remember that the applications and components may be built at different times and that multiple applications running on a given machine might be using the same component. It must be possible to upgrade a component in such a way that it will continue to work with existing applications. This upgrade might involve fixing problems in existing features of the component or adding new functionality. In either case, existing applications must not break because a component was updated.

The binary standard for object interaction helps COM meet all of these requirements. COM is part specification, part implementation. The specification defines how to call objects in a language-independent and location-independent manner as well as how to locate and identify components and create objects. The implementation provides system services that do the actual work of locating components and loading them into memory, performing interprocess and remote communications, and so on.

2.5 Transaction Server

Microsoft Transaction Server (MTS) can be viewed as an object broker and uses COM to map client applications to services resident in software components (objects). It is designed to reduce the time and complexity in the development of multi-tier applications by supplying much of the infrastructure to provide a robust, scalable, high performance and distributed architecture.

MTS insulates the developer from many of the traditional complexities by automatically handling all threading issues, the sharing of resources and the handling of transaction context across objects. Writing an application with COM compliant software

components and installing them in the MTS environment eliminates the need to develop your transaction processing logic. MTS monitors the transactions for any failure and if such a problem occurs, it will automatically handle the rollback.

MTS provides “just in time” object activation and “as soon as possible” deactivation to improve performance and scalability. MTS also provides database connection pooling: it recycles ODBC database connection in a similar fashion to the way it recycles objects. The connection is deactivated rather than closed after a timeout. When a component requires an ODBC database connection, MTS will first try to assign it out of its pool of deactivated connections. A new connection is only established when the pool is empty.

By using MTS, a developer just needs to concentrate on developing the ‘business logic’ as the underlying plumbing and infrastructure is handled for them.

3 Multimedia Interactive TeleLearning System (MITS)

3.1 Distance Education models

Since the 1900's, the education model "lecture-and-book" was very popular. According to the students' ages and/or experiences, they are grouped and gathered together in a classroom to learn a specific topic from the teacher at a specific time. This education model has been efficient for a long period of time [27]. This traditional model has the following constraints that degrade the learning process and limited the learning efficiency for each individual learner:

- Time: Students have to join a class at a specific time.
- Space: Students are supposed to attend a lecture at a specific classroom. When students are geographically scattered and require knowledge transmission this model would fail.
- The high ratio of students to the teachers.

It is known that non-linear knowledge structure is superior to the traditional linear text-based structure in terms of knowledge diffusion [10]. Therefore, new models are required for distance education. Based on the utilized infrastructure, provided services and level of scalability, these models are classified into Broadcasting model, CD-ROM Courseware model and TeleLearning model. Each one of these models has its own advantages and disadvantages.

3.1.1 Broadcasting Model

Broadcasting model is the first form of distance education systems [26].

Advantages:

This model has a low requirement of infrastructure of hardware and software. The broadcasting lectures could be easily attended by watching a TV at home or in the office. Video conferencing technology is used in some broadcasting systems. But there are some limitations.

Disadvantages:

- The “Time” constraint in the traditional “lecture-and-book” model still exists in this model. The course time schedule has to be followed, and the students could not take the lecture at their most convenient time [28].
- Lack of interaction. It is a one-way learning process from the teacher to the students. Although by adding the telephone service for real-time communication has been tried to solve this problem, only a very low level of interaction could be acquired.

3.1.2 CD-ROM Courseware Model

CD-ROM Courseware model improved broadcasting model in terms of quality and quantity. Lectures could be stored and delivered by Compact Disks (CDs) and finally presented on computers. Courseware materials may comprise a lot of images, graphics, animation, voice and even audio/video clips in order to enrich the courseware presentation.

Advantages:

- A courseware on a CD-ROM provides some kind of interaction between the learner and courseware presentation agents “GUIs” to facilitate the learning process.
- Courseware material being made available for access by a learner at any time. Thus, the learner controls the learning speed.
- Courseware material reusability. The students could repeatedly use the CD-ROM and try to understand the course deeply, which is not provided in the traditional model and the broadcasting model.

Disadvantages:

- The students-to-teacher or students-to-students interaction is not provided.
- The quantity of the lectures to be transmitted is limited to the CD-ROM’s capacity.
- Mostly the lecture delivered by a CD-ROM is static [27] [10], because most CDs are of the type “read only”. Now although the re-writable CDs are much more popular than before, the most suitable way to update the CD’s content is to throw away the old one, and order a new one.

Therefore, a new Distributed Computer-based Model “TeleLearning Systems” is proposed and designed to overcome all these constraints and offer an enhanced distance education environment. TeleLearning systems must be designed in a way that supports the learning services, facilitates the education process, and achieves the education goals.

3.1.3 Distributed Computer-Based Model

The distance education models addressed previously are affected by several constraints that would create many obstacles for distant learners. Although the Broadcasting systems and CD-ROM Courseware systems are very popular, they do not support two-way interactivity, scalability and full accessibility. With the emerging infrastructure and enhanced multimedia information processing technologies, innovative TeleLearning systems can be designed and implemented to deliver courseware content locally or remotely to distant learners. Courseware is an educational software entity that contains different knowledge components, yet it resembles the objectives of a traditional course. A seamless education environment built by TeleLearning systems has the following features:

First, sophisticated TeleLearning systems involve interactivity between teacher and students, between students and the learning environment, and among students themselves, thus resembling active learning in the classroom [24][26]. For example, they provide the ability to for students from different social, cultural, economic and experiential backgrounds to discuss topics of similar interests.

Second, the education process should be more accessible and reachable by large groups of learners. This provides more opportunities for those who could hardly access education resources through the traditional learning environment [28].

Third, due to the achievement within multimedia computing and networking infrastructure field, computer-based learning becomes more powerful by integrating

multiple media of information into the system and providing more effective and expressive ways to represent knowledge.

Furthermore, they try to present knowledge structure in a form closer to the reality, by using sophisticated multimedia presentation environments.

Finally, students are offered a real-time, interactive and reusable information interchange through different platforms. For instance, a student can be easily connected to a teacher or an on-line facilitator to obtain help on a specific topic using e-mail, chat, telephone connections and video conferencing.

From this model students gain not only new knowledge but also new social skills, including the ability to interact, communicate and collaborate with widely dispersed colleagues whom they may have never met before [26]. Without such connectivity, distance education degenerates into the old correspondence course model of independent study, where students become isolated and eventually drop out. But in comparison with the face-to-face traditional classes, there is a considerable lack of dialogue in TeleLearning systems [26].

TeleLearning systems are restricted by networking capabilities and information coding scheme [27]. Now the World Wide Web (WWW) is very attractive to an educational institute. In spite of the benefits of WWW, it is still a slow communication medium.

Currently, several practical and experimental forms of TeleLearning systems are being developed and placed on the WWW. For example, United Kingdom's Open University, Vancouver's Open Learning Agency, K-12 programs, Project BIO [18],

Norway's NKS Distance Education System, etc [26], as well as our TeleLearning System "MITS" (Multimedia Interactive TeleLearning Systems).

3.2 Distance Education Participants

In the traditional education model, there is direct "face-to-face" interaction between teacher and students. In distance education environment, lots of participants are involved. For example, distance education system designers, courseware authors, teachers, on-line facilitators, producers, media specialists, database administrators, service providers, technicians and, of course, learners. Different participant has their specific roles to play in order to integrate the distance education environment. The most important participants will be addressed below, along with their new roles.

3.2.1 Distance Education Designers

Distance education designers have to consider a variety of issues that could affect the distance education process, such as the learner's characteristics and needs, the influence of media upon the education process, strategies to increase interactivity as well as active learning, accessibility, and the new roles of teacher, on-line facilitator and student.

Any successful distance education system must focus on the instructional needs of the students, rather than on the technology itself [26]. It is also essential to consider students' interests and experience, educational levels, instructional problems and familiarity with TeleLearning methods and courseware delivery. Thus, designers must build distance education systems that stand on solid ground. Intelligently designed

TeleLearning system can make a significant and positive difference in the way students are learning.

3.2.2 Teacher

The teachers' role in the traditional education environment is different from that in distance education situation. In a traditional model, teachers interact directly with their students. It is face-to-face or one-on-one communication. Teachers prepare their support materials, lecture notes, assignments and tests. However, in distance education, teachers have to be allowed to choose, willing to make choices and qualified to implement their choices effectively [26]. In other words, flexibility should be granted to teachers to develop their personal teaching approach utilizing the variety of options offered by technology. There is no direct contact between the distant teachers and students. Communication is provided through e-mail, videoconferencing, or chat. It is mediated by a host of team partners, which may include editors, designers, producers, technicians, media specialists, site facilitators and service providers [26]. Therefore, it is essential to prepare a well-set plan and coordinate staff's activities to construct a courseware as well as provide enhanced interaction mechanisms in order to deliver courseware content to distant learners.

3.2.3 On-line Facilitator

The on-line facilitator monitors and tutors students by chat, telephone, e-mail or audio/video conferencing equipment. Although he/she doesn't need to be a teacher, it is an extension of the site teacher.

NOTE TO USERS

Page(s) not included in the original manuscript and are unavailable from the author or university. The manuscript was microfilmed as received.

36 - 46

This reproduction is the best copy available.

UMI[®]

3. Easy maintenance and updating: The system will provide an interface to permit learners, authors and service providers to submit updates, which will then be monitored, reviewed and accepted or rejected by data management engine.
4. Easy evolvement: System and data requirements will continue to evolve. A key goal in designing the metadata system is to accommodate changing types of information requirements without changes to the database structure or user interface. More functionality should be easily added in the future without big changes to the existing system.
5. Flexible: In order not to prevent the integration of MMS from other systems, also not to limit its use in the long run, MMS should not be too specific or too complex. It should be part of an integrated system, so it can be combined with the existing or future Internet application.

4.2 System Architecture

This section describes the design of the Metadata Management System (MMS) for the Multimedia Interactive TeleLearning System (MITS). The system takes advantage of HTML/JavaScript, XML, COM, and metadata management theory to provide an innovative distance learning system over the Internet. It employs a three-tier client/server system architecture, which includes User Service Tier, Business Service Tier and Data Service Tier. The details of each tier are described in the next section.

Figure 4-1 System Architecture

As illustrated in Figure 4.2, in general a full cycle for taking an online lecture is described below.

Figure 4-2 System Data Flow

Step 1: Learner logs on to the system.

Step 2: The Presentation Manager negotiates the Learning Style with the Learner.

Step 3: The Presentation Manager receives Preference, Progress and Performance information from the Profile Database to support the decision-making process.

Step 4: Based on the Progress and Performance information, the Presentation Manager sends Query to the Courseware Database to search for appropriate learning materials. The Courseware Database searches learning object metadata and returns Content Indexes, i.e., a list of Locator Indexes that match the search criteria.

Step 5: The appropriate locator indexes (e.g., a lesson plan) are sent to the Rendering Manager to delivery to Learner.

Step 6: Evaluation System receives learner response, queries Courseware Database and notices Presentation Manager of Learner Performance. Go to Step 4.

4.3 User Service Tier

As the name implies, User Service Tier provides front-end services to the system users, which include:

- Receives user input and formats it into XML before sending to the Business Logic Tier for processing.
- Takes information from the Business Service Tier and formats it for presentation to the user.

Major components in this tier include:

- **Web browser:** Netscape Communicator or Internet Explorer, must be version 4 or higher. JavaScript is essential in presenting information and gathering user input. In most conventional web application, “thin client” Web browser only takes completed HTML document from the server and relies on the server to process and format data. This solution doesn’t require powerful client machine but induces frequent round-trips and database operations, and results in overloaded server, crowded bandwidth, and compromised performance. With the computing power increasing and the cost decreasing, now most client machines have the ability to process complex data. By using JavaScript, the computing power of the client machine is utilized and the overall system performance is improved.
- **Web server extension:** ASP scripts to parse and format XML, generate HTML and interface with Business Logic Tier.

4.4 Business Logic Tier

Business Logic Tier acts as an agent to fetch data from various sources according to user request and consolidate them into a single source. It is made up of six components: Profile Handler, Client Access Controller, Courseware Evaluation System, Courseware Presentation Manager, and Courseware Rendering Manager. All of them work together as Web-server extensions to serve user requests.

4.4.1 Courseware Rendering Manager

Courseware Rendering Manager [32] is responsible for rendering course contents in the format of HTML pages. Courseware Rendering Manager is composed of four modules: Content Loader, XML Converter, Media Object Retriever and Media Object Rendering Tools.

Content Loader queries Courseware Database to get the courseware logical structure, then asks **Media Object Retriever** to retrieve necessary static media contents (texts, images, etc.) from the media content database; Content Loader then queries media content archive database to get the locator of media objects. Texts are used to construct the XML Data Island. For other media types like images and audio-video-clips, locator index received from their metadata are applied instead of the media itself. These metadata for multimedia objects along with textual data are organized into XML format and will be converted to HTML documents by **XML Converter**. As a result, a request to Multimedia Content Server is embedded in the <object> or <embed> tags of HTML page

dynamically generated, and media contents will be retrieved and streamed directly from the media contents server to the learner's terminal upon request, at presentation time.

Media Object Rendering Tools are a group of web browser plug-ins to help rendering media objects such as streamed audio, video and SMIL documents.

4.4.2 Courseware Presentation Manager

Presentation Manager is the kernel of the Business Logic Tier. Its functionality include:

- Negotiate the Learning Style with the Learner.
- Generate treatment to learner performance.
- Utilize Performance Information from Evaluation Module and Profile Database to make courseware adaptable to learner requirements.
- Combine the Preference information, the Learner's Performance information (past history, future objectives), and the available learning materials (searched and retrieved via Index) to determine the appropriate Learning Content (e.g., current lesson, and schedule of lessons).

Presentation Manager consist of Progress Monitor, Style Controller and Courseware Constructor:

- Progress Monitor simply records the Learner's learning progress information and stores it to Profile Database.
- Style Controller is responsible for preference negotiating with learners. When learner logs on the system for the first time, it accepts learner selection for preference information and stores into Profile Database. Later in learning process, it will first query the metadata of learning objects to determine if these preferences can be

satisfied. If not, it will start the negotiating process again until conflict is eliminated; otherwise it will let Courseware Constructor to decide learning contents.

- Courseware Constructor determines the Learning Content and format for courseware presentation. It requests and receives Performance and Progress information from the Profile Database. Typically, historical information is retrieved, but current information may be available as well from Evaluation System. Combining these with preference information and context information from Style Controller, Courseware Constructor determines the appropriate learning content to present. Constructed learning content structure is sent to Rendering Manager to be delivered to learner in multimedia format.

4.4.3 Courseware Evaluation System

The existence of Evaluation Module makes the TeleLearning system different from other online multimedia entertainment. It is the major component that constructs the feed back loop in learning process. The Evaluation process uses the metadata of learning contents to provide context to the learner's response and lead to appropriate Evaluation. For example, a learner is expected to answer a multiple choice question and the correct answer is "B", but the Evaluation process has no context that "B" is the correct answer. In this case the metadata of learning content provides the context ("question #1234, the correct answer is B") to correlate the correct response (e.g., the selection of "B"). The Evaluation Module receives the learner's response to the questions, then query the courseware database for metadata of the Exam and Q/A blocks in order to assess learner performance. Performance information is then sent to the Presentation Manager. At the same time, the Evaluation System stores Performance information in the Profile Database. The detail of

this system will be addressed in detail in OATS (Online Assessment and Testing System) section.

4.4.4 Client Access Controller

Client Access Controller is responsible for user authenticity check and user session management. It checks the user login request against Profile. Once the login request is accepted, a separate session is created for each authenticated user.

4.4.5 Profile handler

Profile handler is a data access component that works like a bridge between Profile Database and other Business Logic Tier components. Other components can not access Profile Database directly, instead, Profile Handler handles the requests.

4.4.6 Web Server

The Web server we chose for the system is Internet Information Server (IIS) 4.0 from Microsoft. IIS has been designed for ease of use, security, scalability, portability, and extensibility, and is recognized by many to be superior to a lot of the alternatives.

4.5 Data Service tier

Data Service Tier hosts data repository for the system. There are three database servers involved:

- Profile Database
- Courseware Database

- Media Content Database

4.5.1 Courseware Database

The design of an appropriate data model for a multimedia information system will ensure smooth navigation and fast access between real-world application entities and multimedia information objects.

In MITS, we have adopted the metadata model mentioned in Chapter 3 to design our Courseware Data Model (CDM). The Courseware Data Model describes and represents courseware content by specifying the logical structure and presentation information, in order to facilitate courseware authoring and ensure its delivery to distributed educational groups. In other words, CDM aims to simplify the creation of courses and facilitate their manipulation into database systems. In addition, the Courseware Data Model incorporates courseware with teaching architecture, and guide students through the learning process by specifying the presentation information of the courseware content.

The Courseware Data Object Model (CDOM) is the logical structure of courseware. It describes how a courseware is organized into major components and sub-components, such as Courses and Modules, respectively. As illustrated in Figure 4.3, three levels of data components are maintained: base components, learning units, and pure container. All data components have the nine classes of metadata but each of them has a different collection of metadata field extensions in addition to the core set.

Since the CDOM is designed based on object-oriented paradigm, it satisfies different object-oriented features. For instance, it supports the reuse of logical structures and media objects. The courseware logical structure can be reused to create a new courseware by using different learning objects.

Figure 4-3 Courseware Data Object Model

Base Components are a group of granular learning objects. Each of them can reference one or more media objects stored in Multimedia Content Database. In another words, they contain learning specific metadata of media objects, which differentiate them from ordinary multimedia objects. Base Components are highly reusable: each of them can be referenced by one or more learning units at the same time. Authors can create new Learning Units with existing Base Components, or after some modification to suit different real-life semantics. MITS Base components include:

- Audio clip
- Video clip
- Image
- Graph
- Text block

- Animation
- Question/Answer block
- Downloadable Files

What deserves more description is Question/Answer block. In early approach of MITS this type is not included. This type of data object provides a template so authors can create some questions with answers for each learning module to enhance the interactivity. Currently only multiple-choice questions are available.

Learning Unit is a kind of container type. Each Learning Unit references one or more Base Components, and some spatial, temporal or behavioral relationships are defined. Learning Units can be very well mapped into real-life entities. This group includes:

- Notes: Mainly composed of static media like texts, images and graphs. No interaction provided. Notes are easy to author but notes-only courses won't bring learners with impressive learning experience because of the lack of interaction and its linear nature.
- Online Instruction: Well-designed animation with necessary audio and video support. The authoring of Online Instruction involves more work than other objects in this group. Authors tend to make a great deal of reuse of this type of object.
- Presentation: Composed of audio clips, video clips, text blocks and images. Spatial and temporal relationships must be specified for final layout. Presentation can be specified in SMIL format. User interaction includes VCR-like control: play, fast forward, rewind, and stop.
- Assignment: consists of static media like text block and image.

- Exam: a group of Question/Answer blocks related to the corresponding learning module. Authors can reuse existing Q/A blocks to construct new exams, or create new Q/A blocks themselves.

The highest level is Pure Container. This group includes Course, Module, and Section. Courses are organized into multiple Modules, which are in turn composed of Sections, although sometimes a Module may have no Section at all. This level of objects is more context-related. Generally speaking, Pure Containers are not as reusable as Learning Units and Base Components. Authors often design and create Pure Containers based on a large quantity of Base Components and Learning Units.

4.5.2 Media Content Server

The design of such information systems starts with a media content production phase where new contents are created from scratch, or by importing and editing existing ones, and stored on servers to allow their sharing and reuse by a maximum of users. In our system, we distinguish archiving servers (for media contents long-term storage) from media content servers (for contents' real-time delivery to users). We also distinguish between the courseware database server, which provides database management system services to the TeleLearning applications, from the application server that implements more specific and end-user-oriented functionality.

The consideration of synchronous media contents (video, audio and animation) and their interactive presentation in real-time puts new requirements on content storage and delivery server's capacity and performance. Media content servers store the subset of

contents which are frequently or recently accessed. They must be equipped with storage devices of very fast access time and must have a very high throughput capacity. Contents which are not frequently accessed or not accessed recently are stored on the archive servers which must be equipped with very high capacity storage devices but not necessarily of rapid access time.

Media content servers maintain media content indexing information (contents' semantic and external descriptions) of archive database. Media contents referenced by courseware documents and requested by a user, at presentation time, are searched locally and if they are not found, they are loaded from the archiving servers. Media content servers deliver contents to the user through distribution network, using the RTP protocol [27] and control its presentation by adapting their throughput according to media type and the user's interactions (fast forward, pause, etc.).

4.6 Online Assessment and Testing System (OATS)

Online Assessment and Testing System (OATS) is a crucial subsystem in MITS.

4.6.1 Information Model of OATS

The Information Model describes the data structures that are used to provide online assessment and testing. The key data structures involved are:

- Question/Answer Block (Q/A block): the basic unit for testing is composed of question and answers;

- Evaluation Unit: a group of Q/A blocks;
- Evaluation Result: performance of one learner or a group of learners in evaluation.

By using these data structures users can easily import and export their question and test materials. These data structures also support exchanging multiple Q/A blocks, Evaluation Units, Evaluation Result or any of their combinations within a single data structure. Details are presented next.

4.6.2 Q/A Block

A Q/A block is defined as the fundamental block that contains one question with one or more responses. There are various Q/A blocks in terms of response type. The *Response-type* is the type of response required from the user, e.g. a selection for a multiple-choice question or a string for a fill-in-blank question.

4.6.2.1 Response-type category

The adopted response-types in OATS of MITS are all single response, which means a response can not act as a container for other response.

Timing constraint: There are two types of Q/A blocks in terms of timing constraint.

- Time Dependent means the time taken to respond is important and must be taken into consideration in assessment process.
- Time Independent means the time taken to respond is not significant. By default Q/A blocks are time independent unless specified by authors.

Question Type: The colloquial question types used in Q/A blocks of OATS include:

- True/false – multiple choice questions with either a true or false response identified by the response identity. This type of question is easy to create and easy to evaluate.
- Single choice – multiple selection question with only one of the available selections correct.
- Multiple choice – multiple selection question with one or more of the choices correct.
- Fill-in-blank (FIB) – question with formatted entry place for text or number input.
- Free Response – no selection provided and learner can freely input textual answer.

Media Type: media type refers to the type of digital media used in a Q/A block.

- Text only: only textual data is used.
- Multimedia: non-textual digital media is used. To facilitate management and authoring, OATS enforces the following rules in using digital media:
One or more images used in question body;
One video or audio clip used in question body;
For single choice or multiple choice questions, each selection can have at most one image.

4.6.2.2 Test purpose category

Knowledge based: test learner's knowledge. Question is straightforward and no analysis is required to answer this type of question. Normally the correct answer can be found inside learning material. This type of question is easy to create and should be easy to answer.

Skill based: test learner's problem solving skill. Answer is not available directly in learning material so learner must analyze the problem using acquired knowledge and

come up with the answer. Generally this type of question is more difficult than knowledge based, also it takes author more effort to create.

Difficulty Level: difficulty levels for Q/A blocks are provided by author, and range from 1 to 5 in OATS where 1 is the easiest and 5 is the toughest.

Access Level: according to learner's learning context, stage and authority, some access control is enforced on all Q/A blocks. Two levels of access control is applied:

- All access: available to all learners at any time in any learning context;
- Limited access: only learners at certain stage in certain learning context can access.

4.6.2.3 XML binding for OATS Q/A block

The data model for OATS Q/A block can be defined as hierarchy with many elements and sub-elements, which is easy to be represented in XML format.

4.6.3 Evaluation Unit

An evaluation unit (EU) is one or more Q/A blocks grouped together to serve some assessment purpose. In terms of assessment purpose, there are 2 kinds of evaluation unit in OATS: quiz and exam.

4.6.3.1 Quiz

Quiz consists of 3-10 Q/A blocks and is for self-evaluation purpose only. Quiz is time independent, which means no time dependent Q/A blocks should be included; however most quiz should take 5-10 minutes to finish. To guarantee performance and timing without scaring learner, only textual and knowledge based Q/A blocks whose difficulty

level is 1 or 2 can be included in quiz. Learner performance is recorded for statistic purpose only and is rated as “pass” or “fail”. There are three kinds of quiz in OATS.

- Post-quiz: quiz designed to evaluate the learning result after a learner finished certain learning module. It is a mandatory part in the learning context but is optional for learners to take. Learner will be warned to review the learning module if the result is unsatisfactory.
- Pre-quiz: quiz taken before a learner start on certain learning module to check if preliminary knowledge is obtained. Providing this type of quiz to a learning module is optional. If learner fails pre-quiz, system will warn learner to review some related learning modules.
- Review-quiz: review-quiz is either organized by author or generated randomly by system on learner’s request. When learner wants to review certain learning unit, system will pick up a certain number (specified by learner) of Q/A blocks from all quizzes in the learning unit or its sub-unit.

To enhance the effect of quizzes, learner has the option of review the solution and explanation after finishing each Q/A block, or review at the end.

4.6.3.2 Exam

Exam is composed of at least 10 Q/A blocks. Unlike quiz, exams are normally time-dependent, and learner performance in exams will be recorded for assessment. All kinds of questions can be included in exams. The only constraint is that multimedia Q/A blocks can not exceed 30% to accommodate learners using low-speed connection. Exam

contents cover a certain or a range of learning modules. In terms of coverage there are 3 kinds of Exams:

- Section Exam: evaluates learning result of a section; contains 10—20 Q/A blocks.
- Module Exam: evaluates learning result of a module; covers all sections in the module; contains 20—40 Q/A blocks;
- Course Exam: evaluates learning result of a course; covers all modules in the course; contains 40—100 Q/A blocks.

In the progress of an exam, learners won't be able to check the solution. However, they can do so after the exam is finished and the evaluation is done.

4.6.4 Components of OATS

As we can see from Figure 4.4, the Online Assessment and Testing System consists of two principal components:

- Authoring System
- Assessment Engine

Both components has several functional modules which work together to provide specific service to client. Details come next.

4.6.4.1 Authoring System

The Authoring System is composed of Media Object Metadata Extractor, Q/A Block Constructor and Evaluation Unit Organizer.

Media Object Metadata Extractor (MOME) is responsible for extract technical metadata for multimedia objects like image, audio and video. To avoid reliance on too

Figure 4-4 OATS System Structure

much type-in information by the author, MOME analyze the digital media and generate metadata, mainly technical information, including:

- Locator of the digital media. If it is a URL supplied by the author, MOME will verify the URL;
- Size in bytes of the digital media;
- Format: MIME type, like “image/jpeg”;
- Media-specific information. As follows:

Media type	Information
Image	Dimensions in pixels (width, height)
Video	Duration in seconds.
Audio	Duration in seconds.

Table 4-1 Media Specific Information

After generating, this information will be stored for use by other components.

Q/A Block Constructor (QABC) is in charge of creating Q/A Blocks. It makes the process easy by guiding authors step by step through Q/A Block creating, and makes it efficient by enforcing constraints on all necessary data and metadata. The following procedures are followed in doing so.

1. Authors specify the type of Q/A block they want to create.
2. QABC provides corresponding templates so authors can then provide all necessary information.
3. QABC validates author input and creates the Q/A Block.
4. Created Q/A Block is stored in Courseware Database.

Evaluation Unit Organizer (EUO) helps authors to organize exams and quizzes.

The following functionality are provided:

- Powerful search capability. Authors provide search criteria and EUO queries the Courseware Database to find matching Q/A Blocks.
- Fast and convenient EU organizing. Authors have two options in EU organizing. They may browse through search result and select Q/A blocks themselves; or they

may provide criteria like EU type, covered learning module, number of Q/A blocks needed, average difficult level, percentage of Multimedia Q/A Blocks, and EUO will generate the EU automatically.

Assessment Engine consists of Marking Manager and Treatment Manager. It checks learner response to EU and gives out evaluation result with appropriate treatment. Two major components involved are Marking Manager and Treatment Manager.

- **Marking Manager:** checks learner's response against question/exam marking schemes and gives out the mark. For Exams, the mark is stored in Profile Database through Profile Handler. For Quizzes, mark is generated but no result is stored in the database.
- **Treatment Manager:** analyzes learner performance in an EU and performs a certain treatment, i.e., suggest learner to review some learning material, encouragement, notify author of the result, and so on.

OATS is not an independent subsystem in MITS. It works together with other system components to provide evaluation functionality. In next chapter, a minimal implementation centered on OATS is described.

4.7 Tasks of the OATS

A task specifies the behavior of a system or a part of a system and is a description of a set of sequences of actions that a system performs to yield an observable result of value to an actor. The simple high-level statement of "create an exam" or "take an exam" can be broken down into a set of tasks that are required to be implemented to support the online

assessment. As we implement the system, these tasks serve to help validate system architecture and verify the system as it evolves during development. A subtask can be thought of as an individual step that the user takes to accomplish the primary task. We will find that these high-level subtasks also become the main parts of the application.

For author

- Register author
- Identifies author
- Author adds media object to system
- Author creates question
- Author organizes exam

For learner:

- Identify learner
- Learner selects an exam
- Learner takes an exam
- System records exam results
- System gives learner feedback

These nine subtasks encompass all of the high-level steps necessary to take an exam electronically. These tasks are decomposed into subtasks.

4.7.1 Identify an Author

Authors have access to sensitive data so they must be carefully identified before utilizing the system. Online registration of author is not considered to be an option because of the

difficulty in ID verification. Therefore, system administrator takes the responsibility of registering authors to the system through a desktop application.

This identification task is very simple. It is composed of the following subtasks:

- Request author logon credentials: username / password
- If valid: display author information. Allow user to make changes.
- If invalid: if logon exists, offer the option of email password.

Once validated to the system, the author may go ahead with 3 options: add media object, create question, or organize exam.

4.7.2 Add Media Object

Both data and metadata of media object is required by the system. The author provides some metadata, and the system generates the rest.

This task includes the following steps:

- Author provides information about the media object. The location of the media object can be either an URL or a local file.
- If the location is an URL system will check its existence.
- If the location is a local file, the author will be asked to send it to server through FTP or HTTP.
- System checks further information about the media object.

4.7.3 Author Create Question

Author can create questions by following a predefined procedure, which results in standardized question format and efficient evaluation process. Reference to multimedia objects can be added to question content.

The steps to create question include:

- Author select question type
- Author input textual contents of the question
- If media object is used, author will give information about it. The precondition is, metadata of the media object should already exist in the database.
- Author make changes to the ongoing question until satisfied
- Question is sent to server and stored in the database

4.7.4 Organize Exam

Exam can be created only after a certain amount of questions is hold by the database.

Each exam is served for an evaluation purpose. No duplicate questions are allowed in the same exam. The following subtasks are involved in this process:

- Gather metadata of this exam
- Select appropriate questions for this exam. This selection is based on a group of standards, which include:
 1. Field of content
 2. Question type and answer type
 3. Date created and last modified
 4. Difficulty level
 5. Availability of multimedia contents
- Specify mark weight for each question selected

After exams are organized, students can take them to evaluate learning results.

4.7.5 Identify a student

Assumption here is that the Evaluation-Feedback subsystem can work as a stand-alone module, therefore it is necessary to be able to identify individual students. Student profiles are also kept by the system.

The logic of identifying a student is different from that for an author. If the student has used the system before, then there should be a way to retrieve all of that student's information from the database so that this user can make any necessary change. If the student is new to the online evaluation system, the system should request the required information from the user. Once this information has been gathered and verified, the user will select an ID and password that can be used for future logon.

In breaking down the subtasks of the identification task, we have:

- Request student ID
- If it's an existing user:
 - Display information for validation
 - Allow student to make changes
- If it's a new user:
 - Ask for input of required information
 - Display gathered information for validation
- Establish a new session for user

After logging on, student may select and take an exam.

4.7.6 Select an exam

Normally a quiz or exam is provided at the end of an online course to consolidate learning effect. No search is necessary in this case. However, student also has the option of picking up an exam to test the command of a specific portion of knowledge. To do so, the system must provide multiple methods of selecting an exam. While there are countless standards that can be used in searching for a particular exam, the system will support those methods that would be commonly used by the target user group.

- **Search by exam code:** The user would be prompted to enter the code of the exam. Exam Code is the exact unique identifier of exams. By assigning a unique, ten-character string to each created exam, the system provides that exam with its own, unduplicated identity. Because of the nature of exam code, the user would be required to enter the exact string for the exam they were interested in.
- **Search by learning module:** This is the most commonly used method. The student would be prompted to enter the name of the learning module, or just select the name from a list, and the system would display all exams related to that module. The learning module can be found in a number of different ways. The user could enter the complete path of the module, or retrieve it step by step: select course title from a list, and then select the title of the learning unit. Each of these methods was considered as common search criteria for the system's target user group. Difficulty level and number of question are also provided as supplemental search criteria.
- **Search within user profile:** Sometimes students want to retake an exam. Now that all exam records are kept in user profile, they can just select from the "exam taken" list

to retake the exam. Date and mark are provided for convenience as search criteria to look for exams within student profile.

- Predefined searches: The service provider could create a series of predefined searches that would be displayed for the user. These searches would be displayed in the form of a list of exams that the service provider wished to group together. For example, the service provider wished to create a list of “new exams”, “Top 10 exams with the most examinees”.

4.7.7 Take an Exam

Student could take the exam right after selection. At this point, the system would advice the user how many questions there were in the exam and the time limit. Depends on exam type, the process to take the exam is different.

- Once an exam started, system would display questions one after another. Students type in their answer or select answer from a list. System would display the next question only after the previous one was answered.
- Student could go back to any answered question before finishing the exam.
- For quiz and module test, users were provided with the option of checking the correctness of their answer before proceeding to the next question. Explanation would be displayed if the answer were wrong.
- For comprehensive test, users wouldn't know if their answer was correct or not before finalizing the exam. After a comprehensive test was finished, system would display incorrectly answered questions along with corresponding explanation.
- After the last question was answered, student could choose to finish the exam.

4.7.8 System Store Exam Record

Once student finished an exam, the record was stored by the system. Students may take the same exam multiple times and a new record was kept every time.

4.7.9 Feedback

System provided feedback to students in the following formats:

- Marks achieved and grade in scale of Excellent/Good/Pass/Fail.
- Explanation of incorrectly answered questions.
- Analysis of student performance.
- Suggestions.

5 System Implementation

5.1 Introduction

Using the system architecture and metadata management concepts presented in Chapter 4, an Online Assessment and Testing System (OATS) was implemented in the University of Ottawa's Multimedia and Mobile Agent Research Lab (MMARL) as part of MITS (Multimedia Interactive TeleLearning System) project. The OATS system successfully demonstrated a number of key concepts described in the thesis, such as metadata creating, metadata utilization, XML-based information interchange, ease of use, interactivity, learner-centered learning environment, etc. The successful accomplishment of this system laid a solid foundation for future development of the MITS project.

The user services are implemented successfully with HTML/JavaScript and supports Internet Explorer 4.0 or higher. JavaScript must be enabled on client browser. This thin-client scheme makes the application available on virtually any platforms that the browser is supported, and keeps the requirement on client to a minimum.

Business services are implemented on Windows NT 4.0 (with service pack 4) environment using COM and ASP technologies. The programming language chosen for creating COM objects is Visual Basic. Server-side script language -- VB Script and JavaScript are used in coding Active Server Pages. Data are interchanged between user services and business services in XML format, and XSL is applied to format XML document into HTML. Microsoft Transaction Server (MTS) is used to provide a development framework for data objects.

Courseware database and profile database are implemented using relational database model. The database server chosen is Microsoft SQL Server 6.5. A lot of stored procedures are coded to handle data operations.

The remainder of this chapter was organized as follows: the implementation of data model is described in the next section, followed by the implementation of Business Service tier and the User Service tier. Finally, system integration procedure is described.

5.2 The implementation of data service

5.2.1 Overview

The source of data for an application resides in the data service. How the data is stored and organized should be transparent to the user of the data service. In this mean XML is an ideal format for data service since it acts as a common format for different data stores. For a small amount of data, textual files can be used to contain these XML documents. However, a DBMS has to be used in case of big amount of data. Unfortunately, there is very few DBMS to manage the efficient storage and query of XML document, and relational database is still the most popular choice for storing business data. In this system, most data are stored in a relational database (Microsoft SQL Server 6.5).

5.2.2 Courseware Database

This section describes the implementation of the data model of Courseware Database with emphasis on Evaluation Unit and Q/A Block. The data model of Courseware Database is shown in Figure 5-1.

5.2.3 Profile Database

Profile database stores both author and learn profile information. Learning profiles include learner's performance, preference, progress and private information. The data model of Profile Database is shown in Figure 5-2.

5.2.4 Database Access Control

Access control on database server side is enforced with the following methods:

Group level control

System users are divided into 3 groups: learners, authors, and administrators. Each group of user has different authority on different tables. No learner has login entry in the master database of SQL Server, which means learners can not bypass the application and login to the database directly. On the contrary, two logins are created for use by data objects in Business Service tier to perform database operations on behalf of learners: one for read access and the other for write access. Login and password for authors and administrators are stored in the Master database, which means that they are not managed at the application level.

Figure 5-1 The data model of Courseware Database

Figure 5-2 The data model of Profile Database

Stored Procedure level control

Most database queries and transactions are performed through stored procedure call. Access for each group of database logins is explicitly defined when stored procedures are created. Using stored procedure can hide database structure from middle tier applications.

Table level control

When database tables are created, access level for each group of database logins is explicitly defined. Users won't be able to query tables if they don't have access to those tables.

5.2.5 Media Object Repository

For this minimal implementation, all media objects are stored as operating system files. Advanced Media storage and processing techniques will be applied in the future.

5.3 The implementation of Business services

Mainly there are two types of objects in the business service tier of the system: **Business Objects** and **Data Access Objects**.

Business objects are used to encapsulate the business logic of the system and will be called directly by the ASP source code. They will be responsible for manipulating the data in the system to provide the user with the proper information.

Data Access Objects are used to encapsulate the database access to the system. Business Objects will call these objects. In some instances, they can be directly called by ASP source code. These objects will directly manipulate that information in the database as well as managing the connections of the database.

5.3.1 Data Access objects

There are two packages and a total of ten data objects in this system.

The DpUsers package contains one interface and two class modules that handle data operation of profile database.

The DpExam package includes four objects to handle all database operations related to Evaluation Unit and media object metadata. They are DMultimedia, DModule, DQuestion, DExam and DExamRecord.

5.3.1.1 DpUsers package

5.3.1.1.1 IdUsers Interface

IdUsers defines no properties but four methods. The use of Variant type allows maximum flexibility in the selection of data type for arguments and return values. All the function names are self-explaining.

5.3.1.1.2 The DStudent Data Object

The DStudent data object is responsible for managing the data connection between the BStudents business object and the database. This object supports the retrieval of student information, the addition of new students, the modification of student information. This object interacts with different related tables in the database. The DStudent object implements both the IdUsers interface and the ObjectControl interface, which is provided in Microsoft Transaction Server Type Library. Therefore, it must implement all the methods defined in IdUsers and ObjectControl.

5.3.1.1.3 The DAuteurs Object

The DAuteurs object is responsible for managing the data connection between the BAuteurs application object and the database. This object supports the retrieval of author information, adding new authors, the modification of existent author information.

The DAuteurs class also implements the IdUsers interface and the ObjectControl interface. Besides, it adds one more method suspend_author to temporarily disable an author from login to the system.

5.3.1.2 DpExam package

DpExam package contains ten objects to handle all Evaluation Unit related data operations.

5.3.1.2.1 DMultimedia Object

The DMultimedia object is responsible for managing the data connection between the BMultimedia business object and the database. This object supports the addition of multimedia object metadata, retrieval of media object information based on supplied criteria. The object returns the search result to the calling object (the BMedia object). DMultimedia implements the ObjectControl interface.

5.3.1.2.2 DModule Object

The DModule object is responsible for managing the data connection between the BModule application object and the database. This object supports the retrieval of information about learning module based on some supplied criteria. Actually in this minimal implementation neither addition nor deletion of learning modules is necessary, therefore this object only performs some read-only queries. DModule object also implements the ObjectControl interfaces.

5.3.1.2.3 DQuestions Object

The DQuestions object is the biggest and most complex data object in the system. It manages the data connection between the BQuestions application object and the database. This object supports the retrieval of question metadata and question data, creating and adding new questions, associating multimedia objects with questions or answers, the modification and correction of existing questions. The database tables supporting this object are QUESTIONS, ASK, ANSWERS, QUESTION_ANSWER, QUESTION_ASK, QUESTION_EXPLAIN, ANSWER_MM, ASK_MM, and EXPLAIN_MM.

5.3.1.2.4 The DExams Object

The DExams object is responsible for managing the data connection between the BExams application object and the database. This object supports the creation of new exams and modification of existing exams, retrieval of exam data and metadata according to some supplied search criteria. The object mainly interacts with the MULTIMEDIA table, MIMETYPE and MM_COPYRIGHT table.

5.3.1.2.5 DExamRecord Object

The DExamRecord object is responsible for dealing with exam records. It also implements the ObjectControl interface.

5.3.2 The Business Objects

BpExams package includes two application objects: BQuestions and BExams.

5.3.2.1 BQuestions Object

The BQuestions object implements the ObjectControl interface. There are 6 methods defined in this business object.

1. The addQuestionMeta method is used to add a new question to the system. It only adds the question metadata. An XML document containing the question metadata is passed as an argument to the method. After parsing, corresponding data fields are extracted and used to construct arguments to be passed on to data objects.
2. The addWholeQuestion method is responsible for adding the whole question, which includes question content, all answers, explanations and references to media objects. As with addQuestionMeta method, the XML document passed to this method as an argument is parsed, then methods of data objects are called to finish the operation.

3. The `modifyQuestion` method is used to modify any part of a question, including question metadata and data. All methods of data objects may be called upon request.
4. The `selectQuestions` method is responsible for searching and retrieving questions according to some criteria.

5.3.2.2 BExams Object

The BExams object implements the `ObjectControl` interface. There are six methods defined in this business object.

The `makeExam` method is responsible for creating entire exams. After parsing the XML string passed as an argument, some data object methods will be called. During the process of making an entire exam, `DExams.addExam` is called first to create an entry for the exam in the database. Then comes `DExams.setExamAuthor`, which links the exam to its authors. `DExams.addQuestion` is called afterwards to add questions to the exam with mark weights and time limit. The last two methods can be called multiple times if necessary.

Here the `selectExams` method is provided merely as an XML creator and parser. It parses the XML argument, call the `DExams.selectExams` method, and put query results in an XML document to return.

5.4 Implementation of User Service

Microsoft Internet Information Server 4 is the web server used. VBScript and JavaScript are used for scripting the ASP sources. HTTP is the transport used for moving the XML document between the business service and its users. JavaScript is also used to construct requests in XML format on client side. Normally, the user fills out the HTML form and

presses the OK/Submit button, which will trigger some JavaScript function to construct the value of the form variable in XML format. The value was then sent to the server through an HTTP POST request. On the server side, the Active Server Page instantiated a COM component that has the business service functionality. The received form variable is passed to the COM component and the business rules are executed, then an XML document with the results is returned from the COM component.

5.4.1 Web Site Information Structure

The user services are fulfilled by a group of ASP source files and static HTML pages. The structure is shown on figure 5.3.

Index.html is the welcome page and provides two user groups with links to login pages. AuthorLogin.asp deals with author login request. LearnerLogin.asp handles learner login, register and personal information update.

After logon, an author may choose to supply the system with media object metadata, create questions, or organize exams. AddImage.asp and addVideoClip.asp are responsible for gathering media object metadata input from authors, while addQuestion.asp and makeExam.asp handle the input of questions and authoring of exams.

Figure 5-3 Web Site Information Structure

Learners have the options of searching for exams or retake an exam. SearchExam.asp enables learners to search for an exam with various criteria, or choose from the last ten exams taken. The request of taking an exam was sent to ExamGate.asp, and was processed by TakeExam.asp. This ASP file is responsible for presenting the exam contents to learners. Once an exam is finished, ExamResult.asp processes user input and displays the result of the exam (i.e. passed, failed) with analysis.

5.4.2 LearnerLogin.asp

The LearnerLogin.asp file was used to handle learner registration, logon, and information update. Here is a description of this file with sample code:

Global Stuff

```
<%@ LANGUAGE="VBSCRIPT" %>
```

<% Option Explicit %>

<% Response.Expires = 0 %>

<% Dim **state** %>

The **Expires** property of the ASP **Response** is set to 0 so that this page is generated each time the browser loads it. The state global variable helps the ASP source file to handle multiple functions. Based on the value of this variable, the page will perform different functions. Generally it is a good idea to put the entire ASP source related with student registration in one place.

Table 5.1 explains the relationship between the state variable and page function flow.

The value of "state"	Function
1	Display logon forms to get user input for username and password. Authenticate user.
2	User not authenticated. Display information gathering form and gather user information.
3	Display user information to authenticated user and wait for update input.
4	User authenticated: display available links.

Table 5-1 ASP source related with student registration

Step 1 – Obtain username and password

When state is set to 1, the user will be queried for username and password. To do this, the user will be prompted to fill in a form.

The XML assembling work was done by the JavaScript function. Once the “Submit” button is clicked, this function was executed and an XML document will be assembled based on the value of the input fields.

Step 2: Validate user logon

Since user logon information are sent from client browser in XML format, it can be easily passed to the authenticateUser method of BStudent business object as an argument. Here is some sample code:

```
<% sLogon=Request.Form("userlogon") %>
...
<% set oUserData = Server.CreateObject("BpUsers.BStudent") %>
<% sLogonResult = oUserData.authenticateUser(sLogon) %>
<% set oParser=Server.CreateObject("Microsoft.XMLDOM") %>
<% oParser.loadXML(sLogonResult) %>
```

sLogon is the received query string in XML format. After creating a reference to the BStudent application object, sLogon is passed to the authenticateUser method as an argument. The logon result is also an XML document, and an XML parser will parse it before the result can be displayed in HTML. oParser is a reference to Microsoft XML Parser, a COM object.

If the logon attempt fails, learner will be prompted to try again, or register for a new account.

Step 3: Update user information

Valid user may update their information. To update user information, all of the user information will be transferred from BStudent object in XML format. After applying XSL to the XML document, HTML is displayed to the user.

When user is done with personal information update and the “Submit” button is clicked, all user information (no matter changed or not) will be assembled into an XML string and sent back to web server side. In turn, the string is passed to the updateUser method of BStudent object.

5.4.3 AuthorLogin.asp

This ASP file handles author login, and author information update. The page logic is similar to that of the LearnerLogin.asp file. Authors must supply a username and a password to login to the system, as shown in figure 5-4.

The image is a screenshot of a web browser window titled "MIT Home" and "Microsoft Internet Explorer". The address bar shows "http://192.168.1.1/mts2/". The main content area is titled "MIT System Author Logon". It contains a sidebar on the left with links: "FOR STUDENTS", "FOR AUTHORS", "DOWNLOADS", "FORUMS", "MISCELLANEOUS", and "RESEARCH WORK". The main text area says: "To use the MITS System as an author you must Login. You are identified by your Author Number and a Password." Below this, there are input fields for "Author Number" (containing "254963") and "Password". A "Submit" button is located below the password field. The status bar at the bottom shows "Done" and "Internet".

Figure 5-4 Author Logon Dialog Box

User input is assembled into an XML string and sent to the ASP file through POST request. After retrieving logon request at the server side, a reference to BAuthors

business object is created and the request string in XML format is passed to BAuthors.logonUser() method. As shown in figure 5-5, once authenticated, the server creates a new session for the author and provides the following options:

1. Create metadata for media objects:

- Input metadata for an image.
- Input metadata for a videoclip

2. Create questions

Organize exams.

Figure 5-5 Author authenticated Logon dialog box

5.4.4 AddImage.asp

Authors provide metadata of images through this page.

5.4.5 Upload.asp

Usually it is not necessary to upload the media file to the server side. However, authors may upload images (.gif or .jpeg) in case of necessity as shown in figure 5-6, and an ASP module is provided to handle the file uploading and technical metadata retrieval.

Once the “Browse...” button is pressed, a file-choosing dialog will show up. (Figure 5-7). The file name shows up in the input box shown in figure 5-8 after the “Open” button is pressed.

Figure 5-6 Upload images dialog box

File uploading starts when “Submit Query” button is clicked. After the file is uploaded successfully, it will display some technical properties of the file and asks for input of other classes of metadata (file name, size in bytes, MIME type, image dimension). Information generated by system includes:

Figure 5-7 File-choosing dialog box

Figure 5-8 File name input dialog box

- File name
- Size in bytes
- MIME type
- Image dimension (width, height)

Need manual input from author:

- Title
- Description

Figure 5-9 Image information form I dialog box

5.4.6 AddVideoClip.asp

Authors provide metadata of video clips through this page. File uploading is not provided for video clips. Authors must enter the following information by type-in or select from a list:

- Name
- MIME type

- Size in bytes
- Dimension for display
- Color depth
- Time span in seconds
- Quality in fps
- Minimum bandwidth requirement for real-time playback
- URL
- Title and description

Http://174.0.0.5/index2/ - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Stop Refresh Home Search Favorites History Mail Print

Address http://imgurl.2/ Go Links

Please give some information about this VideoClip (all fields required)

FOR STUDENTS

FOR AUTHORS

Input an easy-to-remember name for this Video clip:

What is the MIMETYPE of this Video clip?

How big is the file size (bytes)?

What is the **width** do you want this video clip to be displayed?

What is the **height** do you want this video clip to be displayed?

How long is the video clip (in seconds)?

What is the **Color depth** of this video clip?

What is the quality of this video in terms of frames per second? fps

What is the **minimum bandwidth requirement** for real-time playback of this video clip?

The URL of this video clip is:

Download Downloads

Forums

Miscellaneous

Research Work

Show Video Clip

Done

Figure 5-10 Image information form II dialog box

5.4.7 AddQuestion.asp

This ASP source file is different from previous mentioned files in that it provides different page logic to different client browser types. For IE4 or higher, it took advantage of DHTML features. For other browser types, the user interfaces look similar but there are more round-trips between the client browser and the web server.

Over years users have been accustomed to dialog box based ‘wizards’ that guide them through information collecting or complex tasks. With IE4 and IE5, it is easy to create these rich user interfaces. In this ASP source file, some rich web-based dialog boxes are created through a combination of DHTML and a few IE4-specific tags. The other advantage of using a ‘wizard’ is to simplify user input, which makes it efficient to validate and convenient to assemble those input data into XML document. With the help of JavaScript, user browser does most validation work so client-server roundtrips are minimized. AddQuestion.asp provides both the dialog box user interface and server-side processing. The user interfaces and descriptions are listed below.

When the author first connects to addQuestion.asp, it determines whether or not the user has already provided the data to be processed. If not, then the page assumes that the user needs to input information about the question and it loads the wizard dialog box.

The user then clicks the “Next” button and navigates through the various panes of the wizard. Each of these panes is within the same HTML page generated and is displayed using DHTML.

The user selects answers from the drop-down lists to provide metadata about the question to create, then goes on to supply the question text and selects answer type.

Figure 5-11 Welcome dialog box

Figure 5-12 Type of questions dialog box I

Question Content and Answer Type

Select a subfield for this question.

Server installation and upgrade

Input text content of the question.

setting values you should enter in the SQL Enterprise Server Manager Configurations window.

What type of answer does this question need?

Multiple choice with multiple correct answer

Multiple choice with multiple correct answer

Multiple choice with single correct answer

Free Response

Next > Cancel

http://vega/Eshop/Question/addquestion.asp Local intranet zone

Figure 5-13 Question content and answer type dialog box

Next, the user will be asked to input all answers for the question according to the “type of answer” selection. If it’s “multiple choice with multiple correct answer”, the dialog box will display input fields with check boxes.

Please provide 4 answers to the question and mark at least one correct answer.

Multiple Choice -- Multiple Correct Answer

☐ A 4

☒ B 16

☒ C 64

☐ D 256

< Back Next > Cancel

http://vega/Eshop/Question/addquestion.asp Local intranet zone

Figure 5-14 Multiple correct answer dialog box

For “free response” option, only input field is provided.

Question Authering Wizard of MITS -- Web Page Dialog

Please provide the answer for this free-response question

Free-response Answer

16

< Back Next > Cancel

http://vega/Eshop/Question/addquestion.asp Local intranet zone

Figure 5-15 Free-response question dialog box

“Multiple choice with single correct answer” results in radio buttons with input fields.

Question Authering Wizard of MITS -- Web Page Dialog

Please provide 4 answers to the question and mark the correct one.

Multiple Choice -- Single Correct Answer

☐ A 16

☐ B 32

☐ C 64

☐ D 256

< Back Next > Cancel

http://vega/Eshop/Question/addquestion.asp Local intranet zone

Figure 5-16 Single correct answer dialog box

Optionally, authors can provide explanation for questions, which are gathered by the next dialog box.

Figure 5-17 The explanation for the questions dialog box
Then comes the last step.

Figure 5-18 Confirmation of sending data to the server dialog box
When the “Send” or “Finish” button is clicked, a JavaScript function is called to assemble all data input into an XML document, which is then sent to the Web server through POST request.

Server-side processing of addQuestion.asp includes the following steps.

1. An object reference is created to BQuestions business object.
2. The XML document is passed to the BQuestions.addWholeQuestion function as an argument. This method returns an XML string.
3. The returned XML string and its associated XSL are passed to TransformXMLStr function of the ASP source file. Resulted HTML is then returned to client browser.

5.4.8 MakeExam.asp

This ASP source file handles operation of exam organized by authors. Again a 'wizard' like dialog box based user interface is provided to facilitate the process. First, authors need to name the exam, select the relevant learning module from a list and specify exam purpose and usage. The user interfaces and descriptions are listed below.

Create New Exam - Microsoft Internet Explorer

Name of Exam:

Which learning module is this exam designed for?

What purpose is this exam designed for?

Usage of this Exam is:

Figure 5-19 Creating new exam dialog box I

Like in addQuestion.asp, once “Next” button is clicked (figure 5-20), a JavaScript function is called to assemble an XML document, then send it to server side for processing. On the server side, a reference to BExams business object is created and the received string is passed to BExams.addExam function, which returns an XML string. Again TransformXMLStr function of the ASP source transforms the XML string to HTML using the corresponding XSL file. Resulted HTML is displayed as part of the ‘wizard’.

Afterwards, another click of the “Next” button brings the window where authors can specify criteria to select questions for the exam, as shown on figure 5-21.

Figure 5-20 Creating new exam dialog box II

Question Selection - Microsoft Internet Explorer

How many questions are required? I will select on my own. ▾

Field: MCSE: SQL Server 6.5 Admin ▾

What Type of Question do you need? Knowledge Test ▾

Answer Type: Multiple Choice – Multiple Correct Answer ▾

Difficulty: from 2 ▾ to 5 ▾

Multimedia Contents: I don't care ▾

Created/Modified since: Month 03 ▾ Year 1999 ▾

Reset Next >>

Figure 5-21 Questions selection dialog box

In figure 5-21, authors may specify the number of questions needed (optional), the learning module related to the exam, type of questions and type of answers. Difficult level, inclusion of multimedia object or not, and date modified are also among selection criteria. This time when “Next” button is clicked, an XML document is constructed by JavaScript function and sent to server side through a POST request. Upon receiving this request, a reference to BQuestions business object is created by makeExam.asp file and BExams.selectQuestions function is called to parse the XML document and select questions according to the criteria. The method returns an XML document containing the result set of question metadata. Then TransformXMLStr processes this document

together with its XSL style sheet to generate HTML. Next dialog box is shown as follows:

MITS Exam Organize - Microsoft Internet Explorer

Exam: SQLEexam02

Available Questions: 100/100 Selected Questions:

12 Mark 5 OK
13 >> Cancel
14 << Up
15 Reset Down
16
17

Module: MCSE-SQLA Field: Server installation and upgrade
AnswerType: SNGL Test Type: P
Difficulty: 2 Modified: 2/10/00
Show question

Figure 5-22 Exam organizing dialog box I

The author can then click question number shown on the “Available Questions” list to check properties of the question. At any time, a click on the “Show question” button will pop another browser window to display the whole content of the question now selected from the list. If satisfied, they may type in the mark weight of the question and click the “>>” button to move it from the left list to the “Selected Questions” list. Also they may select a question number from the right list and click the “<<” button to remove an unwanted question. Like shown below(in figure 5-23):

MIT S Exam Organize - Microsoft Internet Explorer

Exam: SQLExam02

Available Questions:	20/100	Selected Questions:	OK
12	Mark 5	6 (5)	Cancel
13	>>	7 (15)	Up
14	<<	9 (20)	Down
15	Reset	10 (25)	
16		11 (15)	
17			

Module: MCSE-SQLA **Field:** Server installation and upgrade
AnswerType: SNGL **Test Type:** P
Difficulty: 2 **Modified:** 2/10/00

Figure 5-23 Exam organizing dialog box II

Clicking on the “Up” or “Down” button can change the sequence that questions appear in the exam. At last, a click on the “OK” button will result in a constructed XML document to be sent to the ASP on the server. Again on such request a reference to BExams business object is created from the ASP source, and the received XML document is passed to the BExams.addQuestionsToExam function as an argument. The returned XML document by this method is processed with its XSL and the result page is shown in figure 5-24.

Figure 5-24 Exam organizing dialog box III

5.4.9 SearchExam.asp

This ASP source is responsible for providing a user interface for learners to search for exams they would like to take. The page is shown in figure 5-25.

Information shown in this frame includes three parts:

1. A form to let learners to search for exams. In this implementation only two criteria are provided: name of learning module, and number of questions. For convenience, two drop-down lists are provided.
2. A table lists up to ten exams the learner took recently. Learner may click any of the hyperlinks to retake one of them.

Figure 5-25 Exam searching dialog box I

- On receiving a search request, the ASP source will first assemble the request in XML format, then create a reference to the BExams business object. The assembled XML document is then passed to the BExams.selectExams function, which returns the search result also as an XML document. ASP source files process this returned XML document against its XSL style sheet, generates the HTML segment and places it in the beginning of the page, so the page looks like in the figure 5-26.

Figure 5-26 Exam searching dialog box II

A table containing hyperlinks to those found exams are added before the “Exam Selection” block. Like the “Retake an Exam” table, the hyperlink here will send a GET request to ExamGate.asp, which will in turn present the exam to learners.

5.4.10 ExamGate.asp

This ASP source file acts as a gateway to the TakeExam.asp. It does the following steps of work:

1. Check if the learner is authenticated. If not, display a warning message asking the learner to register.
2. Pass on the exam request to TakeExam.asp. Pop up a floating dialog box in which TakeExam.asp will present the exams.

3. Set return path of the TakeExam.asp to SearchExam.asp using JavaScript so that once the exam is finished or learner exits in the middle, the search exam page will be displayed in the current browser window.

5.4.11 TakeExam.asp

TakeExam.asp handles the following tasks:

1. Check if the learner is authenticated. Asking the learner to register if not.
2. Present the exam questions to learner on request.
3. Collect and evaluate the correctness of learner answers; display correct answer on request (if applicable).
4. Assemble learner answers into an XML document and send to ExamResult.asp for marking result.

The dialog box to presenting questions and answers is shown as follows:

Figure 5-27 Question presenting dialog box I

Question 1 (Mark: 25/100)

The CD-ROM on which SQL Server is distributed contains several directories. Which of the following does NOT refer to a directory from which you can install SQL Server?

☐ A. ALPHA

☐ B. MIPS

☐ C. i386

☐ D. X86_64

Home Web Page Dialog

Question 1 (Mark: 25/100)

The CD-ROM on which SQL Server is distributed contains several directories. Which of the following does NOT refer to a directory from which you can install SQL Server?

☐ A. ALPHA
☐ B. MIPS
☐ C. V386
☐ D. INTEL

Choice **d** is correct. The SQL Server Setup program for Intel-based computers is run from the V386 directory on the SQL Server CD-ROM. The setup program for Alpha-AXP-based computers is located in the ALPHA directory and the setup program for MIPS-based computers is located in the MIPS directory. There is no directory on the SQL Server CD-ROM named INTEL. SET p 29. Installing the Server Software.

109

Otherwise the correct answer with explanation will appear in the lower part of the dialog box. However, any answer change after viewing the correct answer will be ignored by the JavaScript that assembles the exam result XML document.

At any time, a click on the “End Exam” will bring out a warning message asking the learner to confirm exit. This can avoid some accidental exit of the exam.

Figure 5-30 Question presenting dialog box IV

When the last question is done, the learner will have the option to review previous answered questions, or just click “Finish” button to finalize the exam, as shown on figure 5-31. Once the “Finish” button is clicked, the assembled exam results in XML format is sent to ExamResult.asp through a POST request for evaluation, and the current dialog box is closed.

Figure 5-31 Question presenting dialog box V

5.4.12 ExamResult.asp

This ASP source receives the XML document containing user response to exams, then pass on the document as an argument to the takeExam function of the BExams business object. The result (pass, fail, etc.) is then displayed to the learner.

Figure 5-32 Exam result dialog box

6 Conclusions and Future Work

In summary, metadata management is crucial in distance learning system, especially for large-scale development and implementation of efficient and effective distributed learning. As a part of the Multimedia Interactive Telelearning System (MITS), the Metadata Management System (MMS) implemented in this thesis helped to achieve the goal of building a seamless distance education environment.

By embracing some widely accepted technical standard of metadata for the content of digital libraries (Dublin Core) and some emerging metadata specification for distributed learning (IMS), a metadata schema composed of ten classes of learning object metadata was established and a functional subsystem including authoring and evaluating modules is created on top of it. The system allows courseware authors to provide metadata for learning objects and utilize these metadata to create evaluation modules; it also provide learners with an efficient tool to find evaluation modules that suit their needs and evaluate learning effects by using the metadata of the corresponding learning modules. Using XML as the data format allows the maximum flexibility in presentation, data manipulation and data interchange. The Metadata Management System in MITS laid a solid ground for future development work.

There are some issues need to be improved in the current system:

- Multimedia data processing:

The system is very limited in the ability to process media objects like audio, video and images. To automate the metadata creating process, some more advanced techniques are needed, especially for metadata in multimedia format.

- Database server selection

Due to the resource and knowledge limitation, Microsoft SQL Server 6.5 was chosen as the database server of the system. It doesn't have the ability to generate, store and process XML data so these works must be done in the middle tier. With the fast advance of XML research, some database with built-in functionality of storing, processing and querying XML data are available, like MS SQL Server 2000 and eXcelon. Taking advantage of these mature XML-capable DBMS will great reduce developing cycle and improve system performance and reliability.

- Implementation

It is a minimal implementation. In terms of metadata schema, only the data classes in the base schema are implemented. To make it more robust and compatible with future metadata specification, it will be necessary to include and make use of other metadata information.

- XML research

New techniques in XML can be applied to metadata management.

The next generation of this system should be able to develop a multimedia, interactive, reusable, agent-based seamless education platform to satisfy the following requirements:

- Integrative: Effectively integration of third parties products into the existing platform.
- Scalable: To support increasing number of users and amounts of content.
- Extendable: Easy to maintain and add new data and functions to meet the growing and evolving needs of the market.

- Flexible: To provide easy information update and retrieval to meet the needs of multiple groups of users.
- Reliable: To provide consistent & constant service.
- Customizable: To meet each user's needs and support individual participant's style.

7 References

- [1] A. Al-Shammari, A. Karmouch (1998) Designing and Modeling a Multimedia TeleLearning Database, *Proceedings of IEEE Canadian Conference on Electrical and Computer Engineering '98*, Waterloo, Canada.
- [2] F. Boumphrey (1998). XML Applications.
- [3] C. Chen, D. Meliksetian, M. Chang, and L. Liu (1996) Design of a Multimedia object- Oriented DBMS", *Multimedia Systems, Press ACM*, Volume 3, Number 5-6, pp.217-227
- [4] L. Dempsey and S. L. Weibel (1996) The Warwick Metadata Workshop: A Framework for the Deployment of Resource Description. D-Lib Magazine. <http://www.dlib.org/dlib/july96/07weibel.html>
- [5] Education Network Australia project page. <http://www.edna.edu.au/EdNA>
- [6] Eisenhower National Clearinghouse Resource Finder
<http://www.enc.org/rf/index.htm>
- [7] J. Emery, A. Karmouch (1996) A Playback Schedule Model for Multimedia Documents, *IEEE Multimedia*, Volume 3, Number 1, pp50-61.
- [8] A. Fedorov (1998) Professional Active Server Pages 2.0
- [9] GEM Project Site. <http://gem.syr.edu/Workbench/index.html>
- [10] A. Grace (1995) Can Multimedia Help People Learn Faster, *Proceedings of the IEEE International Conference on Multimedia Computing and Systems*
- [11] B. Hall. FAQs About Web-based Training, 1998. <http://www.brandon-hall.com/brandon-hall/faqs2.html>

- [12] B. Hall. Web-Based Training CookBook, , 1997
- [13] M. A. Hearst. "Interfaces for Searching the Web," Scientific American, March 1997, 68-72.
- [14] N. Hirzalla, O. Megzari, and A. Karmouch (1995) An Object-Oriented Data Model and A Query Language for Multimedia Database, *IEEE ICECS*, Amman, Jordan
- [15] B.Huang, A.Karmouch (1998) Enhanced Interactivity in a Multimedia System over the Internet, *Proceedings of IEEE Canadian Conference on Electrical and Computer Engineering '98*, Waterloo, Canada
- [16] IMS Global Learning Consortium. <http://www.imsproject.org>
- [17] IMS Metadata Specification version 1.0, 1999.
<http://www.imsproject.org/metadata>
- [18] T. Ingebritsen, G. Brown, J. Pleasants (1997) *Teaching Biology on the Internet*, *Proceedings of WebNet'97*, Toronto, Canada
- [19] "Introduction to Metadata", <http://www.imsproject.org/metadata/Mdintro.html>
- [20] NASA's Web Interactive Training, 1998. <http://wit.ksc.nasa.gov/>
- [21] C. Perey. Determining Whether Web-based Training Is Right for You, 1998.
http://www.perey.com/On-line_learning9.htm
- [22] N. Poon, and A. Karmouch (1997) 3 Dimentional Multimedia Interactive Courseware Authoring in Telelearning System, *Proceedings of CCECE'97*, St. John, Canada
- [23] Public Information Site of the ARIADNE Project. <<http://ariadne.unil.ch>>

- [24] R.C. Schank (1994) Active Learning through Multimedia, *IEEE Multimedia*, Volume 1. No. 1, Spring, pp.69-78
- [25] A. Seffah, and R. Bouchard (1997) "The Intranet as a Cognitive Architecture for Training and Education: Basic Assumptions and Development Issues", *Proceedings of WebNet'97*, Toronto, Canada.
- [26] L. Sherry (1996) Issues in Distance Learning, *International Journal of Distance Education*, Volume 1, No. 4, September, pp.337-365
- [27] R. Wang (1996) A Broadband Multimedia Telelearning System, Master Thesis, Ottawa, Canada
- [28] R. Wang and A. Karmouch (1996) A Broadband Multimedia Telelearning System, *Proceedings of 5th Int. IEEE Symposium on HPDCMCE*, Syracuse, USA.
- [29] S. L. Weibel, J. Godby, E. Miller, R. Daniel. OCLC/NCSA Metadata Workshop Report, 1995.

http://www.oclc.org:5047/oclc/research/conferences/metadata/dublin_core_report
- [30] S. L. Weibel, L. Stuart, J. A. Kunze, and C. Lagoze. Dublin Core Metadata for Simple Resource Discovery, 1998. <ftp://ietf.org/internet-drafts/draft-kunze-dc-02.txt>
- [31] S. Williams , C. Kindel . The Component Object Model: A Technical Overview, October 1994, http://msdn.microsoft.com/library/techart/msdn_comppr.htm
- [32] Z. Zhang, and A. Karmouch (1998) "Multimedia Internet Platform for Distance Learning Applications", *Proceedings of 19th Biennial Symposium on Communications*, Kingston, Canada