E-learning framework for online laboratories

Duan, Bing

2005

Duan, B. (2005). E-learning framework for online laboratories. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/4226>

<https://doi.org/10.32657/10356/4226>

Nanyang Technological University

Downloaded on 26 Apr 2023 18:30:59 SGT

e-Learning Framework for Online Laboratories

Duan Bing

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Master of Engineering

2005

TA
416
D812
2005

Acknowledgments

I would like to take this opportunity to express my earnest gratitude to my supervisor Dr. Habib Mir M. Hosseini for his invaluable guidance, assistance and encouragement. It is my great pleasure to pursue my Master of Engineering under Dr. Habib so that I could enjoy the study.

My special thanks to Dr. Ling Keck Voon, who has been with me throughout my study to provide me valuable ideas, advices and support that helped me solve difficulties I encountered. Dr. Ling has spent a great amount of time helping me with my research direction and future plan. His fruitful discussion and kindly guidance are deeply appreciated.

My sincere thank to Prof. Robert Gay Kheng Leng and his Ph.D student Mr. Teo Chao Boon from Information Communication Institute of Singapore (ICIS). I could always get fresh ideas from them through the regular e-Learning meeting.

I'm grateful to all research scholars and graduate students in the Instrumentation & System Lab for their help. My work would not have been able to proceed smoothly without the assistance from the technicians: Mr. PC Tan, Ms. Joanne and Ms. Noriki from Instrumentation & System Lab, and Ms. Huey Pin and Ms. Guek Har from Automation Lab.

Finally, I dedicate this thesis to my beloved wife Liu Min and my son Yu Xin. They have always been the biggest support, physically and mentally, to my study that enabled me to complete the thesis successfully.

Summary

Recently, great attention has been paid to apply e-Learning strategy for the future education. However, seldom research can be found in the literature for the e-Learning subjects that involve hardware (i.e., Lab-based experiment courses). Although, "Learning Design Best Practice and Implementation Guide" introduced by IMS has presented some use cases for a virtual laboratory, they have been chosen to validate the conceptual model of a Lab-based learning system.

Learning Objects (LO) are discrete units of learning resources based on agreed e-Learning standards. They promote great reuse of resources within new instructional systems development. However, there are some difficulties to apply Learning Object (LO) concept to the Lab-based course due to the different nature of their content. For the Lab-based course, it uses the real hardware that is actually not considered in the existing e-Learning standards. In this research, we have proposed a framework for Online Laboratories (OnlineLab) to facilitate the design and deployment of Lab-based courses. The framework is an extension of Sharable Content Object Reference Model (SCORM), which handles and processes the lab apparatus in a standard and uniform e-Learning environment. We named it as Lab-based Learning Management system (LLM). The Apparatus Run-Time Environment, the key module of the framework, is highlighted in the thesis to provide the online lab services, such as response the lab session request, experiment timetabling, data collection and report generation.

We have also presented a Lab-based Learning Object (LLO) reference model for the hardware related Learning Object. The LLO separates the hardware and its control module so that we could make use of existing lab equipments to design the LLO for any e-Learning system. In order to identify and define the LLO, an Apparatus Markup

Language (AML) has been introduced for Lab-based Learning Object (LLO). The AML is used to describe the LLO, such as the name, location of the apparatus, control command and experiment data. It provides an effective means to solve the reusability and interoperability issues for the lab apparatus when it is used in different LLMs. To reuse the lab apparatus, we could modify some portion of the AML to adapt to the new subjects. In addition, lab apparatus from multiple sources can work equally well with different e-Learning systems since they all ‘understand’ the AML. The initiative of AML in the research is to define and design Lab-based Learning Object. But, it also presents an ideal solution for remote monitoring and control instrument.

Table of Content

Acknowledgments	ii
Summary	iii
Table of Content	v
Abbreviations	vii
List of Figures	viii
List of Tables	x
Chapter 1 Introduction	1
1.1 Motivations	3
1.2 Objectives.....	4
1.3 Major contribution of this Thesis	5
1.4 Organization of the Thesis	6
Chapter 2 Background of e-Learning Systems	8
2.1 Current Situation.....	8
2.2 Learning Content Management System	9
2.3 Key Organizations.....	11
2.4 Trends and Future of e-Learning.....	13
Chapter 3 A Framework for Online Laboratories	16
3.1 Related Research	16
3.2 Problems Faced	18
3.3 Use Cases Analysis	19
3.4 A Framework for Online Laboratory	21
3.4.1 Apparatus Virtual User Interface.....	24
3.4.2 Apparatus Run-Time Environment	25
Chapter 4 A Model for Lab-based Learning Objects	36
4.1 Reusability	36

4.2	Interoperability.....	38
4.3	Lab-based Learning Object Model.....	38
4.3.1	Existing Instrument Communication Standards.....	42
4.3.2	User Interface	43
4.3.3	Communication Interface	48
4.4	Apparatus Markup Language	50
4.5	XML Parser	52
Chapter 5	Experimental Results	55
5.1	Coupled Tank	55
5.2	Reuse Coupled Tank	59
5.3	Ohm’s Law Board	62
5.4	Conducting Online Experiment.....	64
Chapter 6	Conclusions and Future Directions	67
6.1	Achievements	68
6.2	Future Directions.....	69
Appendix		72
References		83
Author’s Publications		89

Abbreviations

Abbreviations	Meanings
LO	Learning Object
LOM	Learning Object Metadata
LLO	Lab-based Learning Object
RLO	Reusable Learning Object
SCO	Shareable Content Object
LMS	Learning Management System
CMS	Content Management System
LLM	Lab-based Learning Management
API	Application Programming Interface
OOP	Object Oriented Programming
AML	Apparatus Markup Language
XML	extensible Markup Language
IML	Instrumentation Markup Language
LCMS	Learning Content Management System
AIML	Astronomical Instrumentation Markup Language
VIML	Virtual Instrumentation Markup Language
SCORM	Sharable Content Object Reference Model

List of Figures

Fig 3.1: Proposed e-Learning framework for Online Laboratory 22

Fig 3.2: Sequence diagram of a lab session 26

Fig 3.3: SCORM conceptual Run-Time Environment 27

Fig 3.4: Proposed Apparatus Run-Time Environment 29

Fig 3.5: SCORM Interface 30

Fig 4.1: Reusable Learning Object for different learning content 36

Fig 4.2: Proposed Lab-based Learning Object 39

Fig 4.3: Module for controller design 40

Fig 4.4: LLO for lab experiment 41

Fig 4.5: The user interface of indicator meter 45

Fig 4.6: The user interface of control meter (with a editable text box below) 46

Fig 4.7: The user interface of control knob 47

Fig 4.8: The user interface of graph chart 48

Fig 5.1: Coupled Tank 55

Fig 5.2: Controller for Coupled Tank 56

Fig 5.3: State diagram of the VUI for Coupled Tank 58

Fig 5.4: Apparatus VUI for Coupled Tank (Client Interface) 58

Fig 5.5: System identification on Coupled Tank 61

Fig 5.6: Ohm's Law circuit board 62

Fig 5.7: Apparatus VUI for Ohm's Law (Client Interface) 63

Fig 5.8: User conducts Online Experiment on OnlineLab 64

Fig 5.9: Sample Web page of OnlineLab 66

Fig 6.1: Case one: controlling & monitoring apparatus via centralized server 70

Fig 6.2: Case two: controlling & monitoring apparatus directly 71

List of Tables

<i>Table 3.1 : User cases in success scenarios.....</i>	<i>20</i>
<i>Table 3.2: User cases in exception.....</i>	<i>21</i>
<i>Table 3.3: Apparatus APIs definition.....</i>	<i>33</i>
<i>Table 4 .1: Dejnition of an indicator meter.....</i>	<i>45</i>
<i>Table 4.2: Definitiona control meter.....</i>	<i>46</i>
<i>Table 4.3: Definition of a control knob</i>	<i>46</i>
<i>Table 4.4: Definition of an indicatorfor graph chart</i>	<i>47</i>
<i>Table 6.1: AML for Coupled Tank Apparatus.....</i>	<i>78</i>
<i>Table 6.2: AML for Ohm’s Law Apparatus</i>	<i>82</i>

Chapter 1 Introduction

The term ‘e-Learning’ refers to a mixture of different preferred learning methods, which delivers to the learner through the use of information technology and is supported with instructional design and engaging content. The trend of using e-Learning as an educational platform is increasing in corporations, universities and industries. The dream of teaching and learning from anywhere, at anytime becomes reality with the construction of e-Learning infrastructure. The e-Learning has become commonplace in the past few years around the world.

The e-Learning is changing more and more person’s life and work around the world in many positive ways. One major change is the way educational materials are designed, developed, and delivered to users. To facilitate the sharing and interoperating of the e-Learning materials created by different corporations and organizations, many consortia and organizations [1-5] have been working on different, but closely related, aspects of e-Learning. They intend to develop and promote the specifications and standards for e-Learning technologies.

To the best of our knowledge, the most significant achievement of e-Learning is that the e-Learning framework for Learning Content Management System (LCMS) and Learning Object (LO) have been proposed. The learning content, defined as Learning Object, is one of the most valuable concepts since it provides a way to reuse the learning material. This is an effort to allow as many people as possible, whether commercially or academically, to contribute and develop LO that could be used or re-used by different LCMS. According to the released e-Learning standard: Learning Object Metadata (LOM) [6] by IEEE Learning Technology Standards Committee (LTSC), the LO could be any digital or non-digital entity. However, most LOs, for the current e-Learning

system, mainly refer to digital material, such as HTML pages, text document, image files, flash, applets, ActiveX and video clips that could be delivered over the Internet. Non-digital LOs, such as Lab-based courses, are seldom discussed in the literature. In addition, researches on the design and implementation of Lab-based courses for e-Learning are rare.

An e-Learning system consists of at least one Learning Content Management System (LCMS) and several learning content. The LCMS manages students and learning events and collates data on student's progress. The content of e-Learning system is called Learning Object (LO), which is **“any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning”** (IEEE LTSC, LOM standard 2002) [6]. The LO is commonly viewed as the smallest element of stand-alone information required for an individual to achieve an enabling performance objective or outcome. Using the object-oriented programming (OOP) paradigm, instructional designers can build small (relative to the size of an entire course) instructional LOs that can be reused a number of times in different learning contexts. In such cases, LO becomes Reusable Learning Object (RLO). This is a major factor in addressing the reusability aspect of the Advanced Distributed Learning (ADL)'s Sharable Content Object Reference Model (SCORM) [7].

From the e-Learning point of view, the concept is also applicable to Lab-based teaching and learning practice. Hence, the hardware apparatus in this thesis proposed is treated as Lab-based Learning Object, or called LLO for short, which is used to design a typical lab experiment. The proposed LLO has the potential to serve as a reference model for Lab-based courses. The advantages of the proposed LLO are:

- Lab apparatus can be used and re-used between different lab experiments;

-
- The same apparatus can be applied for different lab experiments or teaching purpose;
 - The LLO provides a reference model for manufacturers to develop apparatus that supports Lab-based e-Learning;

With the LLO, lab technician can maintain the e-Learning enabled Lab experiments conveniently via the e-Learning infrastructure. More importantly, the student can take the advantages of e-Learning to carry out experiments using real instrument and apparatus from anywhere and at anytime.

1.1 Motivations

Laboratories, which can be found in all engineering and science programs, are an essential part of the education experience. Not only do laboratories demonstrate course concepts and ideas, but also they bring the course theory alive so that students can see how unexpected events and natural phenomena affect real-world measurements and control algorithms. However, lab resources, e.g. equipment and timetabling, are often the constraints that limit institutions in providing students with meaningful and relevant experimental experiences. In general, equipping a laboratory is a major expense and its maintenance can be difficult. For the purpose of developing, packaging and delivering high-quality Lab-based courses for e-Learning whenever and wherever students are needed, we introduce the RLO to Lab-based course in this research.

In recent years, research on Internet-based laboratory is attracting more and more interest. The use of Internet for supporting and integrating the activities of Lab-based courses is growing in many universities [10-13]. In this thesis, the Internet-based laboratory is termed Online Laboratory to highlight that real apparatus are used for Internet enabled e-Learning. To the best of our knowledge, almost all proposed systems

for remote, virtual or Online Laboratory [10-31], simply, provide remote access to the experiment for controlling or monitoring purposes. From the e-Learning point of view, these systems do not include Learning Management System and Content Management System (CMS) support. In other words, no universal platform has been so far developed in a way to provide rich learning environment, nor any standard development for designing and deploying lab experiment

In this thesis, the issue of applying e-Learning strategy to Lab-based course (i.e., lab equipment or apparatus for online experiment) has been investigated. An e-Learning framework and some reusable modules for Online Laboratory have been proposed in our research. We aim to combine the research in Online Laboratory system and e-Learning infrastructure to provide a rich learning environment. One of consideration of the proposed framework is that it should be interoperable with any standard compliant e-Learning system. If some standardization work can be done for the proposed framework, anyone from commercial or academic institutions could contribute and develop Lab-based Learning Objects (LLO) in a cost and labor saving manner. Meanwhile, the LLO could be used or re-used by LCMS systems if the Lab-based Learning Management system (LLM) is included.

1.2 Objectives

Reusability and interoperability of the lab resources are the two major issues that we aimed to solve in our research. Reusability means that Lab-based learning content developed in one context can be transferred to another context, and interoperability means that Lab-based learning content from multiple sources can work equally well with different e-Learning systems. The main objectives of this research include:

-
- 1). **Prompting the Lab-based Learning Management system (LLM):** The current e-Learning system is not able to manage lab apparatus due to the different nature of the existing LO and hardware equipment involved LO. As an additional module to the e-Learning system, the LLM will be prompted to handle the hardware management requests. Then, a prototype LLM will be developed to demonstrate the design procedure and strategy in this research. The goal is to enable many kinds of educational materials to be created, using a consistent notation, which can be implemented uniformly in multiple learning environments.
 - 2). **Defining Lab-based Learning Object (LLO) for LLM:** The LLM will be useless without LLO support. The LLM is the container for the LOs (in such case, it's the real hardware laboratory experiment). In this research, we will give the LLO concept model analysis. It is used to define the model of LLO. Based on the model, the design of LLO will be presented to illustrate how LLO could be integrated with the LLM. Meanwhile, we will also introduce how the LLO could be reused in a time saving manner.
 - 3). **Proposing solution to integrate LLM with e-Learning system:** All functions or modules provided by LLM are reusable and shared by all LLOs. Once the LLM is implemented, the integration of LLM and LMS will be investigated so as to extend and enhance the current e-Learning system for Lab-based teaching and learning.

1.3 Major contribution of this Thesis

The research activities related to e-Learning have been reviewed from different aspects, such as past technologies, current situation and future predication in this thesis.

As the e-Learning is becoming increasingly prevalent and having more significant and positive impact on the education area, we have presented an e-Learning framework to facilitate the design of Lab-based Learning Object (LLO) for Online Laboratory. The purpose of the framework is to provide a useful and meaningful learning environment with real hardware Learning Object to users so that they could conduct the lab experiment on this setup, and also enjoy the Online Lab session. Two concepts: Lab-based Learning Management system (LLM) and Lab-based Learning Object (LLO) have been discussed, which have the possibility to become the learning standard when more elaborate work could be done.

The e-Learning framework, which is an extension to SCORM with additional reusable modules, such as Apparatus Run-Time Environment, Apparatus APIs and Apparatus Time-Tabling, has been discussed in detail. One feature of the proposed framework is that it can easily be integrated with any existing e-Learning system that follows the SCORM specification.

Another contribution is that we have presented a reference model of the Lab-based Learning Object. To define and identify LLO, we have also proposed an XML-based Apparatus Markup Language (AML) so that people could contribute the lab apparatus for e-Learning system in an easy way. Due to the extensibility and flexibility of the markup language, the AML can be easily written and modified when used in different situations. One potential use of the AML is remote monitoring and control instrument.

1.4 Organization of the Thesis

This thesis focuses on the e-Learning framework for Lab-based courses, which consists of six chapters and is organized as follows:

In Chapter 1, a brief introduction of the research background and motivation are stated. The contributions and objectives of our research are also presented in this chapter. In Chapter 2, a review of the e-Learning is given which includes the current situation and our future prediction for e-Learning related research and industry activities. In Chapter 3, we present some related research on Online Laboratory (OnlineLab for short) system. To better utilize previous research outcome and overcome the existing disadvantages, an e-Learning framework is proposed in this chapter. The framework is called Lab-based Learning Management system, or LLM for short. The details of the e-Learning framework for OnlineLab are also discussed. It is one of our major contributions since the LLM provides a standard development environment to use and re-used lab apparatus in any e-Learning system. In Chapter 4, we present a Lab-based Learning Object reference module (LLO) for the proposed LLM. Two issues of lab apparatus's reusability and interoperability have been discussed following by the details of an XML-based Apparatus Markup Language (AML) for LLO. Two experimental works are given to illustrate the LLO concept in Chapter 5.

In the last Chapter, our conclusions and future directions are presented.

Chapter 2 Background of e-Learning Systems

The Internet offers the perfect technology and environment for e-Learning where users can be uniquely identified, content can be specifically presented, and progress can be individually monitored, supported, and assessed. Technologically, researchers are making rapid progress towards personalized e-Learning on the Web using object architecture and adaptive technology [1-5].

Many of issues already discussed in e-Learning are central to learning management system design and learning content delivery. Some specifications and standards have been published by e-Learning organizations [6-8]. They described learning activities in a rich contextual way, not only could materials and systems be better integrated with those activities – but also more usable for practitioners and learners. Learning activities could be selected and delivered by the learning management system itself to meet the requirement of individual learners.

2.1 Current Situation

A number of organizations have been working on different, but closely related, aspects of e-learning technology. These organizations have made great strides in their separate domains, but they have not been well connected to one another. For example, ADL [1] has built a common “reference model” for the foundation of Web-based learning. The IMS [3] specifications, broadly defined as “distributed learning”, includes both online and off-line settings, taking place synchronously (real-time) or asynchronously. Some of these specifications are general, anticipating a wide variety of implementations by various user communities, while others are rooted in earlier practices

and require adaptation to newly emerging approaches. Following is the current released e-Learning standard and specifications, which are mainly from IMS Project [3] and LTSC of IEEE [4]:

- *IEEE Data Model For Content Object Communication;*
- *IEEE ECMA Script Application Programming Interface (API) for Content to Runtime Services Communication;*
- *IEEE Learning Object Metadata (LOM);*
- *IEEE extensible Markup Language (XWL) Schema Binding for Learning Object;*
- *IEEE Metadata Data Model;*
- *IMS Content Packaging;*
- *IMS Simple Sequencing;*

Besides the standards listed, Question & Test Interoperability (QTI), Learning Information Package, Vocabulary Definition Exchange and so on, have also attracted considerable interest.

2.2 Learning Content Management System

A typical representation of an e-Learning system is described by the following equation [9] :

$$\text{LCMS} = \text{LMS} + \text{CMS} [\text{RLOs}]$$

As shown in the above equation, an e-Learning system consists of:

- **A Learning Content Management System (LCMS):** it is "a multi-user environment where learning developers create, store, reuse, manage, personalize, and deliver digital learning content from a central object repository" [9].

-
- **Learning Management System (LMS)**: the main purpose is to manage students and learning events and to collate data on learner progress.
 - **Content Management System (CMS)**: it is used to simplify the creation and administration of online content (articles, reports, pictures, etc.) used in publications.

With the LCMS, learners not only receive the desired instructions (just-in-time learning), but also receive the desired portion of the instruction (granular learning, or just-enough learning).

In a CMS, complete learning courses are assembled from several self-contained chunks called "content components". These content components, when used in the learning domain, are called Learning Objects. One important advantage of the LO approach is reusability. Learning Objects could be combined to form a hierarchy of lesson, module, course, or curriculum that can provide a rich learning environment and to reduce the time, instructor skill or cost associated development. The benefits of using LO are:

- Increased value of content: the value of content is increased every time it is reused. This is reflected in time saving by avoiding re-design efforts.
- Improved flexibility: when content is captured in an object format, it can be reused much more easier than it is rewritten for each new context or application.
- Improved updating, searching and management: metadata tags describing various attributes of a Learning Object help organize, identify and locate relevant content. This improves searching, facilitates management and maintenance, and helps filter and select the relevant content for a given purpose.

-
- Content Customization: the Learning Object approach enables a just-in-time approach to customization by allowing designers to select, assemble, and rearrange content according to stakeholder needs.

To facilitate the widespread adoption of the LO, the LTSC of the Institute of IEEE was formed in 1996 to develop and promote instructional technology standards. A similar project called the Alliance of Remote Instructional Authoring and Distribution Networks for Europe (ARIADNE) [4] had already started with the financial support from the European Union Commission. At the same time, another project called Instructional Management Systems (IMS) [3] Project started in the United States, with funding from Educom. The IMS Project is a consortium of educational institutions, software companies and publishers. The objective is to promote the widespread adoption of specifications that will allow distributed learning environments and content from multiple authors to work together.

2.3 Key Organizations

Many organizations are working in one or more phases of the process to develop e-Learning related standards that ensure interoperability of learning solutions. Some are presented below:

- **Advanced Distributed Learning (ADL)** [1]: is an initiative launched in 1997 by the US Department of Defense and the White House Office of Science and Technology Policy. Their recently released Shareable Courseware Object Reference Model (SCORM) provides one of the best and most recent examples of the application and integration of these learning standards. SCORM divides the world of learning technology into functional components.

-
- **Aviation Industry CBT (Computer Based Training) Committee (AICC)** [2]:
was formed out of a need for hardware standardization of CBT delivery platforms in 1988. It had published a variety of recommendations, including hardware and software configuration. Their computer-managed instruction (CMI) guidelines had the greatest impact.
 - **IMS Global Learning Consortium (IMS)** [3]: Headquartered in Burlington Massachusetts, it aims to develop and promote open specifications for facilitating online distributed learning activities. These specifications describe the key characteristics of courses, lessons, assessments, learners and groups. IMS is a consortium formed by almost 200 commercial, governmental and other entities. Currently, the IMS has some 80 contributing members, a significant number of which are American and British commercial entities, and also including universities and federal governmental agencies. The IMS is the only standards development organization highlighted here that has significant, direct representation from those involved in the school or K-12 sector. This representation includes governmental representation from education ministries, as well as SIF (Schools Interoperability Framework), and USOeC (US Open e-learning Committee). Moreover, the nationalities and sectors represented by this membership have recently been expanding significantly.
 - **IEEE Learning Technology Standards Committee (LTSC)** [4]: formed in 1996, is an accredited standards development organization. Within the IEEE, the LTSC focuses on standards development specifically in the area of e-learning technologies - producing "accredited technical standards, recommended practices and guides" (LTSC, 2002). The LTSC also "coordinates formally and informally with other organizations that produce specifications and standards for similar

purposes" (LTSC, 2002). These other organizations include the IMS and the e-learning standards development body in the ISO/IEC. The LTSC's active membership includes individuals from small and large private sector organizations, the US military and affiliated organizations, as well as governmental organizations and universities of various nationalities. Standards are seen as benefiting significantly from the approval that can only be conferred by organizations such as the ISO and IEC - those with official, delegated international representation. The most widely acknowledged specification is the Learning Object Metadata (LOM) specification, which defines element groups and elements that describe learning resources. The IMS and ADL both use the LOM elements and structures in their specifications.

There are, of course, other standard organizations (many of them have national standards bodies) that make significant contributions to international e-learning standards development, such as PROMETEUS sponsored by European Commission, ARIADNE European Union research project [50]. They are also working on e-Learning to provide benefits to e-Learning providers, education institutions, corporate end-users, individual learners and e-Learning industry in general.

2.4 Trends and Future of e-Learning

It is clear that the e-Learning is changing rapidly as the technology advances in network, computer software and hardware. From a delivery standpoint, more and more people will continue to recognize e-learning's ability to quickly and effectively build knowledge and develop certain skills while dramatically reducing training-related time and costs. This is true not only for academic teaching purposes but for self-learning as

well. As a result, e-Learning is becoming more acceptable to corporations, society, and academia.

In the last decade, the e-learning industry was increasingly converging with other management tools, providing managers with a unified view of all financial, user and vender considerations. However, from another point of view, the e-Learning is still at very early stage in the development of technologies and strategies. There are still many issues to be solved before a full-fledge e-Learning system can become reality. It is not easy to state the foreseeable future of e-Learning, but it is possible to paint some scenarios of what the world could be like:

- **Anywhere Learning:** when we talk about e-Learning, it mainly refers to use computer to access the e-Learning system. In the near future, handheld/mobile devices could be used for learning and people could access the course materials from anywhere since different versions of Learning Object are already provided. In addition, people could do offline or online learning to take advantage of collaboration with more people.
- **Adaptive Learning:** The e-Learning system can dynamically recognize the role and profile of each user, and respond accordingly. For example, when a user logs on, the user management system can immediately identify that person and "understand" whether they are a normal user, an instructor or an administrator, and deliver content accordingly. Once successfully logged in, the user could continue learning progress or restart his/her learning since the e-Learning system keeps track of all learning paths for every user.
- **Prescriptive Learning:** A particular user always has unique characters to distinguish him/her with others. It is true when different users study the same subject and may absorb knowledge differently. In such case, the e-Learning

system utilizes prescriptive learning capabilities to identify knowledge gaps and prescribe the correct path for filling those gaps, so users do not waste time studying material they already know.

- **Reusable Content:** Learning content is reusable across any e-Learning system, for any purpose. It allows anyone in an organization or other place to contribute learning material. Thus, content comes from subject-matter experts, rather than instructors, resulting in more practical and effective courseware. This radically extends the value of the e-Learning infrastructure to the point where it is truly a flexible and scalable learning environment.
- **Rich Learning Object:** The Learning Object could be any form of digital or non-digital material. So, it is easy to provide hardware related e-Learning courses, such as Lab-based experiment subject. In this case, anyone could login to e-Learning and interact with hardware to do more practices (this is actually our research goal). User could personalize his/her learning path on the same/different lab equipment. In addition, the pre-defined lab instruction could be sent to the e-Learning server to coordinate the lab session. User could download and analysis the experiment data at anytime.

The first deployments of true e-Learning system may become true very soon. Meanwhile, the e-Learning will transform and grow at a fantastic pace with other technologies as well. It should be admitted what the e-Learning has become — an easy to use, flexible, powerful and efficient future's educational platform.

Chapter 3 A Framework for Online Laboratories

Currently, the e-Learning system is widely used in many places hosting as regular or continuous education program. Most of them, such as Blackboard (Blackboard Inc) [32], Swift Author (from Gemini Inc.) [33], Macromedia Authorware (from Macromedia Inc.) [34], TopClass Publisher (from WBT Systems) [35], provide a very good platform to handle and process traditional digital LOs, which can be delivered by network. To provide more useful learning experience, research on Internet-based laboratory, which has been conducting in many universities, is attracting more and more interests. They have been used to supplement the traditional engineering courses with remote experiments [10-31]. As discussed earlier, the Lab-based course has not been considered by current e-Learning system due to the different nature of the learning content. In this chapter, an e-Learning framework for Online Laboratory will be presented to include lab equipment or apparatus. In addition, to facility the design and development of new LO, the Lab-based Learning Object is also introduced. It should be noted that Online Laboratory is only one part of the e-Learning. So, the solution we proposed in this research thesis is to extend the e-Learning to Lab-based courses rather than prompting a complete new e-Learning system.

3.1 Related Research

Starting on 1992 at Stanford, Lambertus Hesselink and his colleagues had designed an Internet accessible laboratory — Cyberlab [10] on Senvid [37]. It provides controlling laboratory equipment and instrumentation on an optical processor over the Internet.

Students are given the option of observing the Fourier transform computationally through Cyberlab.

Benetazzo, L (1999) described the specification and design of a geographically distributed system based on arbitrary waveform generator, oscilloscope, digital multi-meter and signal analyzer [11]. In his design, students were only allowed to use the equipments, but tutors, Ph.D. students, and some selected students in advanced measurement courses were allowed to contribute in the creation of the lab components.

In June 2000, the i-Lab, which was funded by I-Campus, a MIT-Microsoft alliance, was launched [42]. It aimed to create and demonstrate technologies that can produce revolutionary IT-enabled teaching models and improved educational tools. These projects included: Chemical Reactor, Heat Exchanger, Mechanical Structures and Microelectronics Weblabs. Among these projects, one is already opened for remote access, which allows student to take current-voltage measurements on transistors and other microelectronic devices through a semiconductor parameter analyzer in real time.

F. J. González-Castaño, et al. (2000) designed an Internet access laboratory using the Java/CORBA paradigm [12]. The system provides remote access to real equipment on SBC68K (a single board computer based on a Motorola MC68000 microprocessor), used in a Computer Architecture laboratory. It uses CORBA technology to manage real equipment so that anyone could invoke it remotely as another set of CORBA objects.

A low-cost internet-based tele-robotic system was applied in remote robotic education in 2000 [13]. It is based on China Internet. Song You, et al. has addressed some issues involving Internet time-delays when data was transmitted between client and server.

Others, such as the AIM-Lab at Rensselaer Polytechnic Institute [38], LAB-on-WEB at University Graduate Centre [39], and I-Lab at Chalmers University of Technology

[40], have all been used in courses on semiconductor devices and circuits with experiments performed on microelectronic test chips and commercial devices. VLab at National University of Singapore allows students to conduct several remote experiments on coupled-tank, oscilloscope and helicopter via the Internet [41]. Similarly, the REAL (Remotely Accessible Laboratory), a virtual laboratory accessible through the Internet, was implemented as a telematic service to allow remote control on mobile robots [14].

3.2 Problems Faced

Hitherto, most work reported above concerned the setting up of remotely accessible experiment system for research or stand-alone teaching purpose. From the e-Learning point of view, these systems lack a Learning Management System (LMS) and Content Management System (CMS) support. In other words, no universal platform has been so far developed in a way to provide rich learning environment, and no any standard development for designing and deploying lab experiment has been proposed.

Moreover, existing LCMS does not provide a suitable framework to incorporate Lab-based courses. Framework for the design, development and deployment of Internet-based lab experiment, which could inter-operate with existing LMS, was seldom discussed in the literature. A notable exception is perhaps [15], which proposed an XML-based framework for the development of new paradigms on Internet-based laboratories. The tele-operation of academic control system was regarded as the simulation problems or real-time plants control system. The main idea of this research was to define an abstract entity called RLAB system by an XML DTD and publish it in a remote lab system. But, the XML DTD for Internet-based laboratory was still at the very beginning. Many issues, such as the format and structure of the DTD, were not discussed.

Most internet-based labs, presented above, are standalone and difficult to share or reuse. As a consequence, many functionally similar, but independent systems have been developed again and again in many places. As discussed, the current research on Online Laboratory has some shortcoming and disadvantage to extend it to e-Learning (i.e., no LMS, CMS and LO support). However, they provide a good ground for design and implementation of OnlineLab system from many aspects, such as system architecture, real-time issue, and Internet time-delay. They have also presented a great deal of experimental setup for remote controllable equipment, which is already test-run by some universities. These researches can be better utilized to provide a more attractive and useful e-Learning environment if some form of integration or combination can be done.

3.3 Use Cases Analysis

Developing a model for a system prior to its construction or renovation is as essential as having a blueprint for large building. The use case, borrowed from Unified Modeling Language (UML) [36], is adopted in this thesis. Use case modeling is one of the most widely used analysis techniques for Object-Oriented development. The use cases are goals that are made up of scenarios, which in turn consist of a sequence of steps to achieve the goal. Each step in a scenario is a sub-, or mini-goal or use case. Each sub-goal requires another use case (subordinate use cases) or an autonomous action that is at the lowest level desired by the use case description. It is useful to write the most important and influential use cases in the expanded format, but the less important ones can be deferred until the construction cycles in which they are being tackled.

Similar to the real laboratory, Lab-based courses for e-Learning must provide the same services for the student. Based on the IMS's "Learning Design Best Practice and

Implementation Guide” for a Virtual Lab [8], the use cases of Lab-based courses in e-Learning domain are presented in the Table 3.1.

Type	Description
Success Scenario	<p><u>For sys-admin:</u></p> <ul style="list-style-type: none">a). Adding new experiment,b). Removing experiment,c). Failure detecting/recovering,d). System Logging. <p><u>For lab-manager:</u></p> <ul style="list-style-type: none">a). Providing experiment courseware,b). Assisting student to conduct experiment,c). Maintaining the experiment,d). Developing new Lab experiments,e). Managing students account. <p><u>For students:</u></p> <ul style="list-style-type: none">a). Selecting laboratory,b). System opens lab environment,c). New student joining in a lab session and observing the experiment phenomenon,d). Passing the control to the partner to conduct the experiment,e). Completing lab activities,f). Closing the lab session,g). Completing Learning Objective.

Table 3.1: User cases in success scenarios

As the Lab-based courses involve use of real hardware equipments, it is possible that unpredictable result or exception may occur during a lab session. In general, Table 3.2 summarizes some of the exception use cases.

Type	Description
Exceptions	<p><u>For sys-admin:</u></p> <p>System does not have experiment resources available. In this case the system must:</p> <ul style="list-style-type: none">▪ Provide an alternative route to another similar equipment or,▪ Display some relevant document. <p><u>For lab-manager:</u></p> <p>The student meets problem and leaves lab session in incomplete state. The lab-manager must:</p> <ul style="list-style-type: none">▪ Provide the synchronize assistance by instant chat message or online phone calling,▪ Replay the students by asynchronies method, i.e., Email. <p><u>For students:</u></p> <p>Students lost the connection or the experiment equipment is in improper running status. Then:</p> <ul style="list-style-type: none">▪ Student closes lab session,▪ System preserves lab environment for the student to continue the experiment,▪ Student continues with other activities within the learning path.

Table 3.2: User cases in exception

The proposed e-Learning framework design for Lab-based courses is based on the use cases listed in Table 3.1 & 3.2.

3.4 A Framework for Online Laboratory

Based on the use cases analyzed above, it is found that the e-Learning could offer a perfect solution to solve most of them. Although there are small-unenclosed details, it

will not affect the design and implementation of such Online Laboratory system for the new e-Learning. In this section, we introduce the proposed framework for Online Laboratory, which is named Lab-based Learning Management system, or LLM for short. Fig 3.1 shows an overall architecture of the proposed e-Learning framework.

Fig 3.1: Proposed e-Learning framework for Online Laboratory

The framework is an extension to the SCORM [7] specification with proposed additional modules, such as Apparatus Virtual User Interface, Apparatus Run-Time Environment, etc. These modules mainly deal with functionalities of lab apparatus and e-Learning server communication. The SCORM specification published by ADL is a reference model that defines the interrelationship of course component, data models and protocols so that Learning Objects can be shared across systems that conform to the same model. It contains a collection of specifications adapted from global specification bodies and consortia to provide a comprehensive suite of e-learning capabilities.

The SCORM deals with the launching, communicating, and tracking of content between the learning resources and the learning management system. It provides means

by which learning resources can be reusable and interoperable across multiple Learning Content Management Systems. It consists of three components, which are:

- **Learning; Resources**: represent learning content (Web page, JavaScript, XML document, Flash object, picture.. .), or called LO (a collection of one or more).
- **LMSAPIs**: the communication mechanism between LMS and LO. These are used for collecting and logging of learning related data or learner's interaction with content objects
- **SCORM Run-Time Environment**: learning management system run-time environment, which manages students and learning events to collate data on learner progress.

To extend the SCORM to the Lab-based courses, we propose the framework with addition modules as following:

- **Apparatus Virtual User Interface**: or App-VUI for short, the remote control panel for the student to control the real apparatus and observe the experiment status during the lab session.
- **Apparatus Run-Time Environment**: or App Run-Time for short, the standard and uniform apparatus's run-time environment, which provides the Online Laboratory experiment services for the students.
- **Lab-based Learning; Object**: or LLO for short, the real hardware equipment, which consists of both the hardware and software that controls the hardware.

The most important element of the framework is the Apparatus Run-Time Environment. It provides many standard functions including lab session management, load balancer and report generator sub-system. These functions are reusable and shared by all Lab-based Learning Objects, which are the experiments in the Lab-based course. As an additional module, the App Run-Time is compulsory for hardware involved LO

such as LLO e-Learning, but optional for traditional LO. This is based on the consideration of the compatibility of e-Learning system when it is used in different learning content.

3.4.1 Apparatus Virtual User Interface

To carry out an experiment from Internet, students need to work in an interactive environment. The Apparatus Virtual User Interface (App-WI) provides multiple students to observe and conduct the online lab experiments. The App-VUI allows the student to issue control commands through the user interface. Meanwhile, the experiment feedbacks, such as the apparatus response and status, are all shown in the App-WI display area.

To design a reliable App-VUI, the Internet time delay needs to be considered. The data transmitted over Internet will be delayed depending on the communication bandwidth available, adopted protocol and other facts, such as the amount of data, transmission distance [10]. The Internet time delay is a critical issue and it is not able to overcome in most situations. In this research, we adopt a protocol to improve the reliability from two aspects: one is the delayed transmission and another is the lost data:

- For the first scenario: Sever will send back an acknowledgement (ACK) message to the client to confirm that the control command has been received. A timer is used to check the ACK. If no ACK received after time out occurs, the same command will be resent.
- For the second scenario: If there is still no ACK after twice time-out, the App-VUI declares that the connection is lost. User will be informed to stop the lab session.

In addition, we proposed an optimization module in the server site. It could direct the user to the most efficient connection to the lab apparatus with the shortest distance and least time delay according to some pre-defined rules. However, one must be highlighted here is that it only works in the situation of a large system with redundant apparatus. Then, the optimization module could achieve high efficiency. Due to the time limit, there are still many issues existed in the optimization module and more researches need to be done in the future.

For the Internet time delay, the proposed solution above may not be perfect. But, it provides a mechanism to reduce its effects as much as possible in this thesis. One of recommended future directions will focus on the time delay issue.

3.4.2 Apparatus Run-Time Environment

In most cases, for a real lab session, one apparatus is only for one user at a time. Multi-user's operation may confuse or damage the hardware if no prevention is taken. In this thesis, two basic principles are defined:

- 1). For any online lab session, only one user is permitted to conduct a particular experiment at a time;**
- 2). Any user can conduct a particular experiment in a specified time period.**

Of course, other users will be allowed to watch the experiment (for watching or monitoring purpose). Based on the principles, the user has to wait for the apparatus to become available if earlier student was using the apparatus. To prevent a user hogs the lab apparatus, a timer is used to reset the lab session by release the connection. For any lab session, there is one daemon provided by the Apparatus Run-Time to collect the user command and experiment data.

A sequence diagram of an online lab session running in the Apparatus Run-Time Environment is denoted in Figure 3.2.

Fig 3.2: Sequence diagram of a lab session

Before we discuss the details of the Apparatus Run-Time Environment, the SCORM's Run-Time Environment (as shown in Fig 3.3) will be presented to provide a basic ground to design Apparatus Run-Time.

Fig 3.3: SCORM conceptual Run-Time Environment

As can be seen in Fig 3.3, the SCORM Run-Time Environment consists of a Server site and one or more Client site, which mainly deals with: how to delivery and launch Learning Object to the learner's Web browser (Client), how a Learning Object communicates with the LMS (Server), and what information is tracked for a Learning Object and how the LMS manages that information.

In the SCORM Run-Time, the launch process defines a common way for LMS to start Web-based Learning Object and defines procedures for the establishment of communication between the launched object and the LMS. The communication

mechanism is standardized with a common API using ECMAScript [51] (ECMAScript is an object-oriented programming language for performing computations and manipulating computational objects within a host environment as defined by ISO/IEC). The API is the communication mechanism for informing the LMS of the conceptual communication state between a Learning Object and an LMS (e.g., initialized, terminated and/or in an error condition), and is used for retrieving and storing data (e.g., score, time limits, etc.) between the LMS and the LO.

There are several key concepts that are introduced in the SCORM Run-Time Environment. They cover the essential LMS responsibilities for sequencing LOs during run-time and allowing LOs to indicate navigation requests. Subjects discussed in SCORM Run-Time Environment include:

- **Run-Time Environment Management:** Launching of learning content– LOs, Management of communications with a LO, Run-time environment data model management;
- **Application Programming Interface (API):** LMS API requirements, SCORM communication requirements, communication error conditions;
- **Run-Time Environment Data Model:** Data model management and behavior requirements, Data type requirements;

Since the SCORM is a standard e-Learning reference model, which is already accepted by most of organizations around the world, in our research, we have proposed an architecture which extends it to a new type of Learning Object: lab apparatus.

Corresponding to SCORM Run-Time, we have introduced an Apparatus Run-Time Environment to solve the communication between LCMS and lab apparatus. The Apparatus Run-Time Environment is the key to achieve the e-Learning goal in Lab-based courses. In general, the App Run-Time provides the run-time environment to

manage a variety of lab apparatus and coordination with the SCORM Run-Time. Fig. 3.4 depicts the proposed model.

Fig 3.4: Proposed Apparatus Run-Time Environment

As shown in Fig. 3.4, there are three types of interfaces for the Apparatus Run-Time to communicate with other modules:

- **Message processor:** responsible for interacting with the client. Its functions include receiving the control commands and feedback the experiment data and status to the student.

- **Apparatus API**: provides communication channel for App Run-Time Environment and Apparatus to exchange data. These APIs provide all the necessary functions for initiating a connection to the actual apparatus and delivering data and commands.
- **SCORM Interface**: used to exchange the information about the student and learning progress with SCORM Run-Time. The LCMS contains all users information and status data for historical purposes or for other purposes such as reporting, auditing, diagnostic or statistical. The SCORM Interface is a set of APIs provided by Apparatus Run-Time to allow LLM to communication with the LCMS in a consistent manner. The following figure (see Fig 3.5) illustrates the communication relationship between an Apparatus Run-Time and SCORM Run-Time. As shown in the figure, four APIs is currently considered in the proposed system, which are Initialize(), Getvalue(), Setvalue() and Commit(). The “Initialize()” API must be called to execute “GetValue()” or “SetValue()” to read or change LCMS’s data. The “Commit()” API is used to enable the “SetValue()” function and terminate the last “Initialize()” call.

Fig 3.5: SCORM Interface

To provide the Online Lab services, the App Run-Time includes several core modules, such as Time-Tabling, Session Management and Load Balancing, etc. These modules of Apparatus Run-Time Environment will be presented in detail as following.

A). Session Management

In most cases, the lab-technician is responsible for guiding the student to conduct the experiment and collect the experiment results. In the App Run-Time Environment, the session management is acting the same role. Sometimes, it provides synchronous assistance (by instant chat message or phone calling) for the student when the important experiment is in progress and the student meets problem. On the other hand, the asynchronous assistance (by Email) is also included in some unimportant situation. The session management module is responsible for monitoring the lab sessions in order to:

- Terminate the experiment once time-out occurred.
- Reset the apparatus for a new user.
- Inform the lab manager in case of any hardware failure.

Considering the management of real laboratory, all students who enrolled the course share the lab resources by a scheduled timetable. For system identification, a unique UID corresponding to the user will be generated if the user wants to conduct a particular experiment. The design can guarantee that only one user has the chance to access the apparatus at a time. Meanwhile, unauthorized user can't control the apparatus due to the absence of UID.

B). Load Balancing

In the real world, lab resources, such as lab schedule or opening hours, equipment, lab assistants are always limited. Although lab resources can now be accessed 24 hours a day, 7 days a week through e-Learning infrastructure, lab equipment is still limited. Hence some form of load balancing is necessary. The purpose of load balancing is to increase utilization and enhance apparatus availability. High availability can be defined

as redundancy. In this case, a few lab apparatuses, which have the same functions and I/O, need to be developed and made available in the App Run-Time Environment. So, when several requests come to a same experiment, the load-balancing module can perform automatic balancing to re-direct the request to point to different apparatus according to a pre-scheduled rule. Also, the whole e-Learning system can offer different services for special users, such as researchers and LO testers. It may become a useful and special feature for some commercial e-Learning provider.

C). Report Generator

Once the student completes the experiment, he/she may need to download the experiment results and control command data for further analysis. In the proposed Apparatus Run-Time Environment, there is one report generator module, which performs the data collection during a lab session. All data, such as the experiment feedback, status or user command, will be collected automatically. To enhance the readability for human being and machine, the proposed report generator could process these data according to the different XML Schema. So, it could easily generate a personalized report for different user using the same experiment data. In some cases, the student report may be simpler than lecture's.

D). Apparatus API

To allow a user to conduct Online Lab experiment, there must be a common way to start, stop and control the apparatus. In the proposed App Run-Time environment, a set of Application Programming Interfaces (APIs) have be defined to provide a consistent manner to communicate with apparatus. Table 3.3 defines the App Run-Time environment's communication APIs.

API	Execution
LOInitComm	<p>Description: LMS initializes the communication with the Apparatus. It must be executed before calling other APIs.</p> <p>Parameter: The Apparatus's system ID that are used to establish the connection with the apparatus. The ID is unique in such case.</p> <p>Return Value: Long integer indicates the connection handler if success. Otherwise, return 0.</p>
LOCloseComm	<p>Description: LMS closes the connection handler.</p> <p>Parameter: The return value of LOInitComm.</p> <p>Return Value: Boolean. "True" indicates the close action was successful. "False" means some error occurred when close the connection.</p>
LOSendCmd	<p>Description: LMS sends the command to the apparatus.</p> <p>Parameter: The return value of LOInitComm, Command String (defined in next section).</p> <p>Return Value: Boolean. "True" indicates the send action was successful. "False" means some error occurred when sending the command.</p>
LOGetStatus	<p>Description: LMS gets the status of apparatus, i.e., running status, apparatus's indicator status.</p> <p>Parameter: The return value of LOInitComm, Status Code (defined by the different apparatus).</p> <p>Return Value: String. The status of the apparatus.</p>
LOLastError	<p>Description: This function returns an error code resulting from the previous API call. The code can be retrieved many times and will be kept unchanged until the new API call is made.</p> <p>Parameter: The return value of LOInitComm.</p> <p>Return Value: A pre-defined integer number, Error description.</p>

Table 3.3: Apparatus APIs definition

These APIs provide a standard way to initialize a connection, send command and receive data from an established socket. So, different LCMSs could provide the same Online Lab experiment services by invoking the same APIs.

E). Advanced Services

In addition to the modules introduced above, more useful modules can be developed to enhance system usability. Live video, discussion forum, Q&A whiteboard and e-notification (by email) can be embedded into the App Run-Time. For the convenience of users, a new feature, SMS (short message service) Reminder is also introduced. SMS reminder provides an easy way to remind the students who is in the system and when the apparatus is idle. Additionally, the administrator can get the information of the laboratory by SMS in the area without Internet access. Normally, two ways can be employed to achieve SMS function via either SMS gateway (Internet) or GSM (Global System for Mobile Communication) modem. The second way is adopted in our scheme for the future system upgrade and migration.

In general, the LCMS provides the Web-interface for the student to select the online experiment. Then, the App Run-Time allocates lab resource (i.e., Time-Tabling) for selected apparatus. If no student is using the equipment, the App Run-Time will setup the connection with apparatus controller via Apparatus APIs. Otherwise, the App Run-Time lets the student join the queue. Once the App Run-Time is connected to the apparatus, then the student can issue the control commands and receive the experiment data via the established communication connection. Apparatus APIs provides a standardized method for a LCMS to communicate with real lab experiments. So, the App Run-Time Environment can be considered as a container for the Lab-based Learning

Objects (LLO). Every LLO, which is integrated into the environment, can share the resources provided by the system.

Chapter4 A Model for Lab-based Learning Objects

Laboratory resource’s Reusability and Interoperability are two major issues that need to be addressed in this research. Before the lab experiment is used or re-used, the experiment must be located. It is hard to locate and use it if there is no proper data structure and format to describe the equipment because the Online Laboratory involves the real hardware.

4.1 Reusability

In the e-Learning domain, the solution for reusability is to store not only Learning Object but also descriptions of the Learning Object. Thinking of the LO as data, the descriptions are data about the data, or metadata. IEEE LTSC released Learning Object Metadata (LOM) standard [6],which potentially includes information about the title, author, version number, creation date, technical requirements and educational context and intent. Metadata can be written for an entire course, a section, a page, or even a single picture or movie clip. Fig. 4.1 illustrates the idea of reusable LO for different learning content.

Fig 4.1: Reusable Learning Object for different learning content

The benefit of metadata is that it can enable powerful search capabilities across content repositories. In this research, we consider the lab experiment as the Learning Object for Online Laboratory system. However, due to the different nature of the lab experiment and LO, the LOM standard is not able to handle Lab-based Learning Object. Reasons are as following:

- The LOM is designed for digital material, i.e., document, image file, executable program.
- The metadata information of LOM is insufficient to describe a real hardware, i.e., the communication between the equipment and LCMS.

In addition, the Lab-based equipment has some special characteristics so that additional vocabularies must be defined to describe the hardware Learning Object.

There has been a widely used technology called extensible Markup Language (XML) that offers a solution for this problem. We applied XML to Lab-based Learning Objects. By using XML, we define the LLO's interface and mechanism for moving commands and data between the control program and equipment. LLO developers will no longer need to create software, specialized APIs, and/or distributed objects to communicate with the hardware. Instead, they will only need to provide an XML description of the interface and how the communication takes place. Tools that can parse and interpret the XML will be able to automatically generate the required user interface to control the instrument, construct messages to the instrument and display the data that is returned. The XML to describe the LLO is called Apparatus Markup Language, or AML for short. The detail of AML will be discussed later.

4.2 Interoperability

To make interoperability possible, there must be a common way to launch Learning Object, a communication channel to send/receive data, as well as some predefined data elements that are exchanged between LMS and object during its execution. ADL proposed SCORM Run-Time Environment [7] to provide a mean to solve problem regardless of the tools used to create the learning content. The Run-Time completes the launch process by establishing a ‘handshake’ between the LO and the LMS that launches it, and breaking that handshake when the LO is no longer needed. In addition, It allows SCORM content to ‘set’ and ‘get’ data on the LMS, such as assessment results, and to check and address any errors that occur during these processes.

In the SCORM specification, only one communication scenario has been proposed, which is between Learner and LMS. However, the Lab-based Learning Object needs one extra type of communication between LMS and Apparatus. It should be full-duplex connections, as data needs to be transmitted back and forth between user and apparatus.

To realize the interoperability for Lab-based experiment across multiples LCMSs, there must be a common way to initialize, shutdown and control the apparatus. A series of Apparatus APIs (AppAPI for short) has been defined as discussed in a previous section. They allow a LCMS to communicate with real lab experiments in a consistent manner.

4.3 Lab-based Learning Object Model

In the proposed LLM, the LLO is different from the LO in terms of communication. For LO, only one communication link between user and LCMS is considered. For LLO, full-duplex communication between LCMS and lab apparatus must be considered.

To model a lab experiment as a LO in the LLM, one has to take into account the fact that LLO involves real instrument or apparatus. A LLO consists of both the hardware and the controller software. To allow user to access and control the hardware, a communication mechanism must be provided for data exchange between the LLM and the controller software. A Lab-based Learning Object Model is presented in Fig. 4.2.

Fig 4.2: Proposed Lab-based Learning Object

In this thesis, a LLO refers to a real hardware with a controller (i.e., electrical board, medical equipment and optical device) that can be connected to the network via Transmission Control Protocol/InternetProtocol (TCP/IP) protocol. The TCP connection is used for the important data (i.e.,user command and experiment status) whereas User Datagram Protocol (UDP) connection is introduced for streaming data (i.e.,experiment video). The controller can be any form of embedded or PC based devices with a communication module that enables the connection to the network. For the software part, it executes the local control of the hardware and collects the experiment’s data. As a whole unit, the controller must have the physical interface, such as DA/AD converter,

data acquisition card, to link with the actual equipment. Besides the controller and the Lab apparatus, an XML-based Apparatus Markup Language description file is used to describe the apparatus as shown in Fig 4.2.

The controller is developed in a modular manner. A modularized controller is shown in Fig 4.3. Modularizing is the designing of software applications before coding. Modular design makes the control program more readable and maintainable. Every module shown in Fig 4.3 is one function, which performs a specific task.

Fig 4.3: Module for controller design

The details about the AML will be discussed below. To define and describe a LLO, the AML must contain the following information:

- 1). **The Control User Interface of the LLO:** it is the graphical User Interface for user to interact with the hardware. In our case, the GUI will be automatically generated upon successful registration of the LLO.
- 2). **The Communication Interface of the LLO:** the communication interface defines the data exchanged between LCMS and LLO in machine-understandable manner. In this research, the AML also defines the command format and how the communication occurs between LCMS and LLO.

Obviously, if the AML for LLO can be standardized, the interoperability of LLO could be easily achieved. In addition, different Lab experiments could also reuse the LLO in a consistent manner. Fig. 4.4 denotes the use of LLO for Lab experiment.

Fig 4.4: LLO for lab experiment

The communication between LLO and LCMS generally could be treated as computer to instrument communication. Instrument communication refers to the mechanism for machine-to-machine's data and information exchange. Basically, the instrument sends or receives information to/from another end of equipment, machine or computer. Information from the instrument may inform the other end about its processing status, performance, and errors. If the interface is implemented using a defined standard, then machine could use the same interface to 'talk' with others as human beings say a same language. So, they could understand each other. Sometimes the communication interfaces are used not only to monitor a machine, but also to control it. This includes downloading settings, configuring various software controls, starting, and stopping the machine processing.

4.3.1 Existing Instrument Communication Standards

In early 1980s, information and data were exchanged between instrument and computer in almost any format including from ASCII to device dependent compacted binary code. To allow a computer to communicate with instruments in a consistent manner, the IEEE 488.2 [53] was proposed and approved in 1987. It provides a level of standardization in message and data formats, message exchange protocols and common commands shared among a wide range of instrument devices. It defines a number of requirements for the communication interface, such as pulsing the interface clear line for 100 μ s, setting and detecting End of Identify (EOI), setting/asserting the Remote Enable (REN) line, sensing the state and transition of the Service Request (SRQ) line, sensing the state of Data Valid (DAV), and timing out on any I/O transaction.

In the semiconductor industry, an equipment communication standard – the SEMI E4 SECS-I protocol [54] has been used since 1980, which is based on RS-232 technology. Due to the advancement of network technology and its usage, the High Speed Message Task force (HMTF) [55] was formed to prepare for balloting a standard to address network communications with higher speed and throughput. The objective of the HMTF was to develop standards that could operate on a wide choice of platforms, thereby promoting wider adoption of the standard. So TCP/IP has been selected as the protocol for high-speed messages.

The latest equipment communication is IPC (short name for Institute of Interconnecting and Packaging Electronic Circuits) [56], which is defining new communication standards for retrieving information from the equipment based on Extensible Markup Language (XML), a common and popular communication language. The new standards promise facilitating access to process information in order to increase equipment efficiency and reduce costs. IPC 2501 (Definition for Web Based Exchange

of XML Data) defines the protocol, message headers, and rules of message exchange. This proposed standard (soon to be published) uses several Internet standards wherever possible including TCP/IP, Hypertext Transfer Protocol (HTTP), Simple Object Access Protocol (SOAP) and XML.

4.3.2 User Interface

The User Interface for LLO refers to App-VUI in this thesis. It is a special LO in term of the way it was delivered and presented to the remote user. To carry out an experiment from Internet, students need to work in an interactive environment. The App-VUI allows the student to issue control commands through the user interface. From e-Learning point of view, the LLO consists of the Lab apparatus and its controller. So, the User Interface is treated as the metadata. Since the LOM is insufficient to describe a LLO, a new specification is needed.

Practically, User interfaces are the most visible and frequently changing part of an application. The development of UI is one of the most labor-intensive parts of application or system development and maintenance. At present, XML is a new player in the UI technology. It is easy to write, easy to change, and is perfectly suited to the changing nature of UI development. Following are several related activities of XML for User Interface:

- 1). Alternate Abstract Interface Markup Language (AAIML) [43]: It aims to develop an Alternate User Interface Access standard. The design includes an “XML-based language that would be used to communicate an abstract user interface definition for a service or device to a user’s personal device which could act as a Universal Remote Console.”

-
- 2). Abstract User Interface Markup Language (AUIML) [44]: It is an XML vocabulary that has been designed to allow the intent of an interaction with a user to be defined. This clearly contrasts with the conventional approach to user interface design, which focuses on appearance. All the interaction information can be encoded once and subsequently rendered using 'device dependent rendering' so that users can actually interact with the system.
 - 3). Extensible Interface Markup Language (XIML) [45]: It is an XML-based "interface representation language for universal support of functionality across the entire lifecycle of a user interface: design, development, operation, management, organization, and evaluation." It aims to provide a common specification and development infrastructure for user interface professionals of all types from interaction designers, to software engineers, to usability experts.
 - 4). Extensible User Interface Language (XUL) [46]: "XUL" is glossed alternately as "XML-based User Interface Language," "XML User Interface Language," and "Extensible User Interface Language." The version 1.0 draft attests "XML User Interface Language." XUL is a "standards-based interface definition language" associated with the Mozilla XPToolkit Project.
 - 5). User Interface Markup Language (UIML) [47]: It is an XML-compliant language. It is designed to serve as a single language, which permits creation of user interfaces for any device. It is independent of any user interface metaphor, such as graphical user interfaces or voice-response. UIML describes the appearance of a UI, the user interaction with the UI, and how the UI is connected to the application logic.

In this research, we have proposed a much simpler Apparatus Markup Language (AML) to describe and present a Lab-based Learning Object.

We have defined two categories of component for the LLO. One is the Control Variable and another is Indicator Variable. The former refers to those control components whose value could be changed. The Indicator Variable is used to show the status of the LLO and user is not allowed to control it (or the value of the Indicator can not be changed by user). Followings are four examples to define an 'Indicator Meter', 'Control Meter', 'Control Knob' and 'Indicator Graph' using AML (as shown in Fig 4.5-4.8). The first and fourth are components to show the changes or status of the LLO. The second and third are control components to allow the user to change the value (the value could be sent to the apparatus via the established connection).

1	<!--
2	Control Variable: meter
3	Property: name, location, size, max & min value, step, precision
4	Remark: if the indicator is "on", the meter value will be shown at the
5	bottom. To specify it as a control variable, add one more property
6	"type=indicator".
7	-->
8	
9	<meter name="V1"
10	location="160, 30"
11	size="150"
12	min="0"
13	max="10"
14	increment="2"
15	step="1"
16	precision="2"
17	indicator="on"
18	type="indicator" />

Table 4.1: Definition of an indicator meter

Fig 4.5: The user interface of indicator meter

1	<!--
2	Indicator Variable: meter
3	Property: name, location, size, max & min value, step, precision
4	Remark: if the indicator is "on", the meter value will be shown at the
5	bottom.
6	-->
7	
8	<meter name="V2"
9	location="160, 30"
10	size="150"
11	min="0"
12	max="10"
13	increment="2"
14	step="1"
15	precision="2"
16	indicator="on"/>

Table 4.2: Definition a control meter

Fig 4.6: The user interface of control meter (with a editable text box below)

1	<!--
2	Control Variable: knob
3	Remark: if the indicator is "on", the meter value will be shown at the
4	bottom.
5	-->
6	
7	<knob name="Vin"
8	location="60, 300"
9	size="150"
10	min="0"
11	max="10"
12	increment="1"
13	precision="2"
14	step="0.5"
15	indicator="on" />

Table 4.3: Definition of a control knob

Fig 4.7: The user interface of control knob

Data can be displayed in many different forms. For a instrument, it's usual to show the running data continuously in a way for easily watching. The graph chart is a numeric indicator that displays data continuously so that people could find the trends and the changes of the data. An example to show the graph chart is defined in Table 5.4.

1	<!--
2	Indicator Variable: graph
3	Property: name, location, size, label, graphs, axisX & axisY, background,
4	grid, text, border, grips, child
5	Remark: child is the name of the chart need to be shown.
6	-->
8	
9	<graph name="Water Level"
10	location="0, 0"
11	size="600, 100"
12	label="Water Level"
13	graphs="3"
14	axisX="0, 10, 200, 1"
15	axisY="0, 0.5, 1, 0.05"
16	background="230,230,230"
17	graphground="black"
18	grid="0, 150, 0"
19	text="black"
20	border="black"
21	grids="on"
22	borders="on" >
23	<child id="SetPoint" color="red" label="[SetPoint]" />
24	<child id="Tank 1" color="green" label="[Tank 1]" />
25	<child id="Tank 2" color="blue" label="[Tank 2]" />
26	</graph>

Table 4.4: Definition of an indicator for graph chart

Fig 4.8: The user interface of graph chart

The UI provides the control panel for user to interact with the lab apparatus. But, what are the command to be sent and the data to be received? The communication details are presented next.

4.3.3 Communication Interface

We have proposed a much simpler communication protocol for LLO. In this protocol, the TCP/IP is selected to transfer data between LLO and LCMS server. The communication interface of the LLO mainly refers to the command and data format. We have defined two categories of component:

- a). **Control Variable**: an object allows user to change its value or properties.

In general, an event will be triggered if the value or properties have been changed.

- b). **Indicator Variable**: an object used to show the status or experiment data.

The user could not change its value.

Besides these components discussed above, there are some User Interface accessories, such as line, circle and timer. In most cases, they are used to make the UI more user-friendly but not critical for the system.

During a lab session, the user could change the value of the Control Variable. Any change will trigger an event that is pre-defined by the XML. The event then packages a

command to send out through the established connection. For the time being, all command have been designed to have the same format as:

$$\text{Control-Name} = \text{Control-Value}$$

For example, when the user changes the value of a “**Spinner**”, a command string: “**pump1=2.3**” will be packaged and sent out. The same idea (if the LLO has any data needs to be sent back to the user) a command will be packaged like:

$$\text{Indicator_Name} = \text{Indicator-Value}$$

After receiving the feedback, the UI knows how to parse the command string and display the value in the correct place. The constraints of the command string are:

- The value of Control-Value and Indicator-Value can be any form of integer, fraction or string;
- More command could be appended to the command string to form a longer command. In this case, a delimiter (special character, such as ' - ', '#') is use to separate them;
- The max length of a command is limited to 1000 characters to avoid communication buffer overflow.

The proposed communication message format is simple to understand and easy to implement. In most cases, it offers very high efficiency in terms of the message process and transmission time. The benefits of the proposed message format are:

- Easily understood for both machine and humankind;
- Easily packaged and parsed by machine;
- Extensible for different situation.

In addition, it could be adapt the proposed communication message format to wide use, such as equipment and instrument remote monitoring and control system.

4.4 Apparatus Markup Language

R. Pastor [15] presented an XML-based framework, Remote Laboratory Extended (RELATED), for the development of Web-based laboratories. The idea is to define an abstract entity called RLAB system by an XML DTD so that lab experiments can be described by RLAB. Then, the RLAB schema is published on the server for general use.

A group at NASA's Goddard Space Flight Center is using a dialect of XML called the "Instrument Markup Language" (IML) [48], which is used to describe graphical user interfaces to control and monitor the instrument, command sets and command formats, data streams, communication mechanisms, visualizations, and more. The Astronomical Instrument Markup Language (AIML) is a domain-specific implementation of the more generalized Instrument Markup Language (IML). Both AIML and IML vocabularies are based on the XML, which is instrument description that encompasses instrument characteristics, control commands, data stream descriptions (including image and housekeeping data), message formats, communication mechanisms, and pipeline algorithm descriptions.

A similar project, Virtual Instruments Markup Language (VIML) [49], has been submitted to XML.ORG [52] (an industry web portal formed and introduced in June 1999 by OASIS) and is under development. The VIML can be used to describe location, protocol and device information for a network of virtual instrumentation devices and/or systems. The main objective is to publish control panel of virtual instrumentation to Web.

To the best of our knowledge, the AML for LLO represents a first attempt to provide the real hardware as Reusable Learning Object. The goals are:

- Provide a platform independent of XML format and schema to describe the lab apparatus.

-
- Simplify the development, modification, maintenance and extendibility of lab apparatus.
 - Facilitate multiple iterations of the lab apparatus description during design and separate implementation (representation) from description.

In short, AML is a specification for using XML to aid in the exchange of data and commands with apparatus. The AML is used to describe the control user interface, commands and data that can be transferred. A tool reads the AML, then automatically creates an appropriate user interface for issuing commands, and finally sends the commands in response to user input. When responses come back, the same tool can use the AML specification to interpret the incoming data and automatically present it with an appropriate user interface.

We have proposed an AML standard as following:

- An AML must start from `<AML update="auto" port="2002" >`. 'update' is an attribute of the root tag which is used to define whether the client should regularly send the data to the server or it should do it manually after a button is pressed. The tag is optional and the values are 'auto' and 'manual'. If not, provided client will update automatically by default. 'port' is the other attribute of the root tag. It defines the port address to be used to communicate with the server. The default value is 8000.
- Any object must be defined as a tag name. e.g.: `<textbox ...../>` to define a text box to allow user to type in something.
- Attributes are used to store the values of the object, like the size and the location of an object. e.g.: `<textbox size="100, 100" />` to define the size of the text box with width=100 and height=100 pixel.

- The elements that an object has should be added as child tags only if the object can have one or more of these elements, like charts in a graph object. e.g.:

```
<!--
To define a graph chart to show the changes of pump voltage
-->
<graph name="Pump Voltage"
  location="0, 150"
  size="600, 100"
  axisX="0, 10, 200, 600"
  axisY="0, 0.5, 1, 5">
  <chart id="FlowRate1" color="green" label="Pump 1" />
  <chart id="FlowRate2" color="yellow" label="Pump 2" />
</graph>
```

- The attributes and child elements must be optional as long as it is possible.

Certain objects are set to either indicator or control variable (by default, it is control variable) and can be changed by the 'type' attribute. The standard can be used by any application written in any programming language as long as it is capable of displaying the items. The AML files can easily be modified to suit the needs of the situation.

The AML file can be written according to the definitions format provided using any common text editor. It can also be produced by the IDE that may be available in a later time. Two complete sets of AML for LLO are presented in Appendix.

4.5 XML Parser

Once the AML has been created based on the LLO, we can now import the AML to LCMS server. An XML Parser provided by the LLM could interpret the AML and automatically generate the User Interface. The UI packages the command, which will be send to the lab apparatus and also display the feedback. In this research project, the Java is selected as the programming tool and Java Applet is introduced to display the UI because:

-
- Cross platform: Java is currently available on all major Operating Systems, allowing users to create secure and interactive applications in a cross platform environment.
 - Thin: the Java Applet is a lightweight package of java-based library. It contains the lightweight controls, event handling routines and its own paint.
 - Secure: the Applet has no access to the users resources or other potentially sensitive and personal documents.
 - Easy to use and develop: XML-based environment enables instructor to quickly build instrumentation UI and run the AML files directly in Web browser.
 - Easy to deploy and maintain: It is easy to deploy from a central server and easy to upgrade and maintain. The LLO is managed centrally, which allows them to reflect all needed changes from the single entry point from the LCMS site.

The XML Parser has some built-in components, such as “Button”, “Label”, “Spinner” and “Line”. In addition, a guide to describe the use of an individual component is provided. Sometimes for different lab apparatus, the instructor may require special component (i.e., a Knob with different scale and range). It could be easily done to add new component to the XML Parser.

When a user starts a lab session, client computer downloads and installs the XML Parser only once. Every time the UI is displayed, only the AML file is downloaded. Thus, especially in slow connections using the parser would consume much less resources than making a UI directly using Java. The parser uses a simple layout manager, which only repositions and resizes objects. Firstly, it goes through all the child elements of the root node. It then compares the node names with the known object names in the definitions class. When a match is found, it calls the matching class and passes the node to the class. Next, it adds the returned object to its pane. Once UI is launched, the parser will use the

port defined in the root tag to make a connection to the LCMS server. For the security reason, the UI can only connect to the server from which it is downloaded.

Due to the high extensibility and flexibility of the AML, any organization or individual person can easily add his or her own objects to the XML Parser. By extending the parser class, it is quite simple to add or remove their own objects. If needed, the tag and attribute names in the XML file can be changed easily by editing the definitions class.

Chapter 5 Experimental Results

In this chapter, two lab apparatus, which have been developed based on the proposed LLO concept, are presented to illustrate the implementation of e-Learning framework for Lab-based courses.

5.1 Coupled Tank

The Coupled Tank apparatus, as shown in Fig 5.1, consists of two small perspex tower-type tanks mounted above a reservoir, which functions as storage for the water. Two independent pumps pump water into the top of each tank. The head of water in each may be visually read on the attached scale at the front of the tanks. Each tank is fitted with an outlet at the side near the base, and a plastic hose connects this outlet with a return through the reservoir lid. The amount of water, which returns to the reservoir, is approximately proportional to the head of water in the tank since the plastic water-return tube at the base of the tank functions

The level of water in each tank is monitored by a capacitive-type probe, which, in conjunction with electronic circuits in the box at the rear of the unit, provides an output

signal proportional to the water level. For normal operation, this DC signal is a voltage in the range from 0 to +5 volts. The zero level represents the rest point of the water when the tank is empty (approximately 20 mm). And at the +5v state, the water level begins to overflow down the rear plastic standpipes, which occurs at approximately 300 mm.

An embedded device with a CPU, D/A and A/D modules functions as the controller for the Coupled Tank apparatus. The objective of this experiment is to maintain the water levels in the tanks at the specified heights. From the control point of view, it can be treated as a Multi-Input-Multi-Out (MIMO) system (in this case, it is two-inputs (pumps) and two outputs (sensors) plant). The apparatus is designed for teaching elementary feedback control principles. By applying Modular Programming strategy, a control program is designed as shown in Fig 5.2.

Fig 5.2: Controller for Coupled Tank

As shown in Fig 5.2, the Controller for the Coupled Tanks consists of following modules:

- **Controller.VI:** Control Module, which executes the duty of controller and activates the various control methods (actions), such as the Manual mode, On/OffControl and PID with their specified parameter values.
- **Read Cmd.VI:** Communication Module, which reads control commands that are sent from a TCP connection.
- **Write Dat.VI:** Communication Module, which sends data collected to LCMS server via TCP for on-line trending and analysis at a remote location.
- **Process Cmd.VI:** Other Module, which parses the control commands received into the action part and the parameters part.
- **Log Dat.VI:** Other Module, which collects the experimental data

Corresponding to the Controller, a Web-based user interface needs to be created to allow student interact with the Coupled Tank via Internet. As discussed in previous section, the VUI has two parts: one is the control panel and another is the display area. For the Coupled Tank, the display area is created to plot the pump voltages and water level in real-time manner. So, the user could watch the change of the control signal and output of the apparatus. Before we start to design the VUI, a state diagram of Coupled Tank VUI has been given (see Fig 5.3). State diagrams are used to help the developer better understand any complex/unusual functionalities or business flows of specialized areas of a system. In short, State diagrams depict the dynamic behavior of the entire system, or a sub-system, or even a single object in a system. In our case, the state diagram is used to describe the behavior of the VUI, which defines all possible states of the VUI when events occur.

Fig 5.3: State diagram of the VUIfor Coupled Tank

The VUI (see Fig 5.4) of Coupled Tank will be generated automatically once the user opens the experiment session.

Fig 5.4: Apparatus VUI for Coupled Tank (Client Interface)

An XML-based Apparatus Markup Language is used to describe the VUI of the Controller, which can be found in the Appendix. Since an XML parser module used to predict the Apparatus Markup Language has been developed in App Run-Time Environment, the lab apparatus can be easily recognized and identified by the OnlineLab system. The changing, modifying or upgrading of the lab apparatus can be easily done by editing the AML definition file.

This VUI (shown in Fig. 5.4) allows the student to conduct experiment with different controller, such as Manual, ON/OFF, PID (Proportion-Integral-Derivative) control, and to study the effect of adjusting the various parameter setting available in the controller. As this is a relatively slow process, the water levels and the control signals to the pump can be charted on the VUI when the experiment progresses. The student can also download all data collected by the LLM for further analysis when the experiment has been completed.

The LLO described in this section is called Controller-LLO for our reference.

5.2 Reuse Coupled Tank

As the Coupled Tank have been designed based on the LLO model to allow student to test run their controller parameters. In this section, the same apparatus will be reused to form a new LLO, namely system identification LLO or SysID-LLO for short.

To achieve high accurate when a controller is designed, the student is required to conduct system identification on the Coupled Tank. System identification covers a very wide range of techniques for obtaining a system model from inputs and outputs to the apparatus. In above Controller-LLO, only manual control is provided so that the student has to manually change the input signal and collect the output data. By using the same apparatus, we have added a new system identification module – pseudo random binary

sequence (PRBS) module on the controller part. Accordingly, the AML for this module is provided as following.

```
<!--
System identification on Coupled Tank. The pseudo random binary sequence
(PRBS) is used to run the sys identification on this apparatus. To reuse the
Controller-LLO, the AML is changed as following.

Note: only the "panel" section is replaced compared with the Controller-LLO's
AML.
-->
<panel name="PRBS"
    label="PRBS">

    <label name="Multi"
        location="10, 20"
        label="Multiplier"/>

    <spinner name="Multi"
        size="40, 20"
        location="80, 20"
        min="0"
        max="50"
        step="0.1"/>

    <label name="Bits"
        location="10, 60"
        label="Bits"/>

    <spinner name="Bits"
        size="40, 20"
        location="30, 60"
        min="0"
        max="16"
        step="1"/>

    <label name="Amp"
        location="80, 60"
        label=""/>

    <spinner name="Amp"
        size="40, 20"
        location="100, 60"
        min="0"
```

```
max="50"
step="0.1"/>

</panel>

<!--
The AML of Controller-LLO is followed to reuse its module.
-->
```

The SysID-LLO is shown as Fig 5.5.

Fig 5.5: System identification on Coupled Tank

In above LLO, most functions have been reused, such as communication module, data process module and plot chart, etc. Student could run the SysID-LLO to collect the system ID data that could be used to design their own controller. After that, they can conduct the experiment on the Controller-LLO to test run the parameters obtained from SysID-LLO.

5.3 Ohm's Law Board

The objective of this experiment is to teach student the basic principle of Ohm's Law. During this lab session, students are allowed to interact with a circuit board to conduct various experiments. Student could change the power supply, turn on or off the switch and measure the voltage and current so that he/she could find the relationship of Resistance (R), Amperage (I) and Voltage (V). This relationship is well-known as Ohm's Law. The experiment setup uses a real electric circuit so that student could play with the real measurement to determine the relationship R , I and V . As shown in Fig 5.6, the experiment setup is made up of four parts:

Fig 5.6: Ohm's Law circuit board

- 1). **A Power Supply:** A power supply has a positive and negative terminal that creates a potential difference. This is frequently, but erroneously, taken as being equivalent to potential energy. They are not identical. Potential difference is proportional to potential energy and the two can be related via the work done per unit charge. This measure of energy per unit charge measured is called voltage (V) and the unit of measurement is called volts.
- 2). **Four Resistances (R):** Resistance inhibits the amount of current running along the circuit. The greater the resistance, the lower the current. Resistance is a bit

like inertia in mechanics. For a given force, the greater the mass (inertia) results in the lower the acceleration. Here, for a given voltage, material with high resistive will inhibit the flow of electrons (e.g., current) through it.

- 3). **Four LEDs:** to show the status of the switch. The LED will turn on if the switch is in ‘ON’ status, vice versa. They could be seen through web-camera.
- 4). **Four Switches:** they are used to turn on or off the circuit.

The board is connected to a computer through a Data Acquire Card (DAQ). So, it is controllable locally or remotely (with additional communication module). The designed VUI(The AML for Ohm’s Law is available at the Appendix) of Ohm’s Law circuit is shown in Fig 5.7.

Fig 5.7: Apparatus VUI for Ohm’s Law (Client Interface)

The VUI allows user to change the Power supply through a Knob. Once the Knob has been changed, user could press the “Measure” button to read the current and voltage

shown on the meters. In addition, the different connection can be achieved through turning on or off the different switch. It can be found that user could find the relationship R , I and V easily based on the data recorded. Since the circuit is not software simulation, users may found the Ohm's Law a slight different from the one that they have learned from the class. In fact, it is one of the foundations of the OnlineLab to teach student how to find the problem and solve it. The difference may be caused by temperature, humidity or the accuracy of the A/D or D/A of the DAQ.

5.4 Conducting Online Experiment

In this section, mechanism for conducting the online experiment on OnlineLab will be presented. A typical user operation flow is shown in Fig 5.8.

Fig 5.8: User conducts Online Experiment on OnlineLab

The dashed line box represents some daemon processes, which are non-seeable to the user:

- a). **E-notification**: represents a daemon running on the server site to perform the service of notification. It happens when the user is trying to conduct experiment and the lab apparatus is busy. Normally, this daemon will send an Email or SMS to the queuing user if the lab apparatus becomes available.
- b). **Log Data**: represents a daemon to perform the service of collecting user command and experiment data during the lab session. Both command and data will be automatically collected and for downloading upon user finishes the lab experiment.

As one part of e-Learning, OnlineLab is password protected, which means that anyone who wants to conduct the online experiment must have valid user ID of the system. But, the introduction and information of the OnlineLab is available for any user. Upon the user ID is proved by administrator, user can login to the OnlineLab through the “User Login” link (see Fig. 5.8). To reduce time of system management, user-self registration is allowed so that the administrator could be released from the tedious work. The administrator is also responsible for the verifying and/or updating of user account.

After successful login, the user could find all available labs apparatus. To start a lab session, user can just click the laboratory ID that he/she wanted. In the real situation, many users may try to conduct the same experiment at the same time. To avoid the conflict, a “Time-Tabling” has been provided. For any user, who wants to conduct the online experiment, must book the lab session. The “Time-Tabling” will check the status of the selected apparatus. The user has to wait for the current user stopping the experiment if the apparatus is not idle (it may happen when administrator conducts the demo or test experiment). In this case, user could be notified by Email or SMS once the

apparatus become available (done by “E-notification”). Otherwise, user could start his/her online experiment immediately if there is no user controlling the apparatus. OnlineLab also provides the self-learning session if the user has no background or experience about the experiment.

During the lab session, a “Data Log” daemon will perform the data collection automatically in the server site. Hence, all experiment data and user actions can be downloaded for analysis or report work after the user finishes the experiment. The live video is also provided by the OnlineLab for user to watch and monitor the apparatus. More information about the system is available at the OnlineLab home page (see Fig 5.9).

Fig 5.9: Sample Webpage of OnlineLab

Chapter 6 Conclusions and Future Directions

In this thesis, we have presented the investigation of an e-Learning framework for Online Laboratories for future education. It can potentially be made available to a broad spectrum of students of an e-Learning system. It provides students opportunities for developing intuition and understanding theoretical knowledge far beyond the horizon of students today.

The proposed e-Learning framework in our research, which we named it as Lab-based Learning Management system (LLM), is an extension to ADL's SCORM with additional modules. Extra modules, such as Apparatus Virtual User Interface, Apparatus Run-Time Environment and Apparatus Time-tabling have been added to existing system. With these modules, the LLM could handle and manage the lab apparatus in a standard and uniform e-Learning environment. In order to reduce development cost, we define a set of Apparatus Application Programming Interfaces (APIs) in the Apparatus Run-Time Environment module. Upon the definition of these APIs, the system could easily initialize or terminate the connection to the hardware equipment. In addition, all user's commands and experiment data could be transmitted in a consistent manner.

We then focused on implementation of Lab-based courses for the proposed LLM. Since the Lab-based courses involve the use of real hardware, we have presented a Lab-based Learning Object (LLO) reference model to facilitate the development of LLO. The LLO separates the hardware and its control module so that we could make use of existing lab equipments to design the LLO for any e-Learning systems. In order to solve the reusability and interoperability issue of lab apparatus, we studied the IEEE LTSC's LOM for Learning Object and finally, we defined and implemented an Apparatus Markup Language (AML). The AML makes use of XML to describe the control User Interface

(UI) and communication interface of the hardware. Several related projects that employ XML to describe UI have been investigated in this thesis. The AML proposed in our research defines two categories of component. One is called ‘Indicator Variable’ to be used to show the experiment data and status. Another one is called ‘Control Variable’. It allows the user to change the value and submit to the apparatus. Furthermore, we have designed a communication message format for the LLO, which offers high efficiency in terms of the message process and transmission.

6.1 Achievements

A summary of achievements in this research project is given below:

- 1). Based on the proposed e-Learning framework, we have developed an impact LLM to illustrate the implementation of Online Laboratory in School of Electrical & Electronic Engineering, Nanyang Technological University. It is available at <http://onlinelab.ntu.edu.sdonlinelab.htm>. The OnlineLab has an apparatus management system so that we could import new apparatus into the system and remove the old one.
- 2). We have presented the Apparatus Markup Language for LLO to define and identify a Lab-based course. This is one of our major achievements to solve the reusability and interoperability of the lab apparatus when it is used in different LLMs.
- 3). We have also developed two Lab-based Learning Objects — Coupled Tank and Ohm’s Law board. The Coupled Tank is used to teach student the elementary feedback control principles, such as step responses, PID tuning. The Ohm’s Law board is designed to let student find the relationship of Resistance (R),

Amperage (I) and Voltage (V). Each of them has an AML file to specify the apparatus's properties, such as the name, communication port and control UI.

We have successfully imported these two LLOs into the LLM and they have been test-run by a dozen of students in our school.

6.2 Future Directions

Today, the e-Learning market is still at the very beginning, grappling with new systems, standards, and solutions. The e-Learning framework for Lab-based courses could lead to applications not limited to education, as it offers the opportunity for efficient collaboration in research and development. For example, collected data from hardware devices could provide a means for analyzing product performance, which could lead to improving devices and systems. Lessons learned from remote laboratory experiments could be applied to training and continuing education. Technologies developed for education could be transferred to industry for a wide array of applications, including remote control of appliances, instruments, dangerous processes, and smart structures. Typically, two potential cases are suitable to extend the use of the LLM, LLO and AML for remote monitoring and control instrument:

Case one:

All remote apparatus are connected with a centralized server (as shown in Fig 6.1), which provides the XML-Parser and Communication Module. For every apparatus, it has an AML file to describe the control user interface and communication interface of the hardware itself. In this case, the apparatus could be self-registered through network connection when it was plugged into the system (as plug-and-play fashion). After that, the centralized server could identify it as an available apparatus

in the system. The client could access the apparatus by downloading the corresponding AML file. Upon the generation of UI, the user could control or monitor the apparatus from a Web interface (Java Applet) provided by the centralized server. Since the XML-Parser is located in the centralized server, client site will not be affected if a new version of the XML-Parser is available or apparatus has been changed (user interface or communication interface).

Fig 6.1: Case one: controlling & monitoring apparatus via centralized server

Case two:

All remote apparatus are distributed with their AML files (as shown in Fig. 6.2), which describe the control user interface and communication interface of the hardware. A stand-alone Java application with a XML-Parser in the client machine could import the AML of the apparatus. The XML-Parser reads the AML and generates the control user interface in the local computer to allow user to interact with the apparatus. In this case, the centralized server is no longer needed. It is useful for a small network with less apparatus. The disadvantage of this is that client site must be updated if the XML-Parser or AML has been changed.

Fig 6.2: Case two: controlling & monitoring apparatus directly

In addition, further researches recommended might cover how to improve the performance of the system in terms of Internet time delays, apparatus usage efficiency, and framework scalability. Of course, one important direction is to standardize the LLM and LLO to facility the developing and deploying Lab-based courses. It may take years to become true. In this case, more elaborate research work has to be done.

Appendix

The Apparatus Markup Language for Coupled Tank

```
1  <?xml version="1.0" encoding="UTF-8"?>
2
3  <!--
4  AML for Coupled Tank. The UI will be update once every 10 seconds
5  automatically. Communication Port is 2001.
6  -->
7  <AML update="auto" port="2001" >
8
9      <!--
10     plot the water lever and pump voltage using a graph chart
11     -->
12     <graph name="Water Level"
13         location="0, 0"
14         size="600, 100"
15         label="Water Level"
16         axisX="0, 10, 200, 600"
17         axisY="0, 0.5, 1, 5"
18         background="238, 238, 238"
19         graphground="black"
20         grid="0, 150, 0"
21         text="black"
22         border="black"
23         grids="on"
24         borders="on" >
25         <chart id="SetPoint" color="red" label="SetPoint" />
26         <chart id="WaterLevel1" color="green" label="Tank1" />
27         <chart id="WaterLevel2" color="yellow" label="Tank2" />
28     </graph>
29     <graph name="Pump Voltage"
30         location="0, 150"
31         size="600, 100"
32         label="Pump Voltage"
33         axisX="0, 10, 200, 600"
34         axisY="0, 0.5, 1, 5"
35         background="238, 238, 238"
36         graphground="black"
37         grid="0, 150, 0"
38         text="black"
39         border="black"
```



```

40         grids="on"
41         borders="on" >
42         <chart id="FlowRate1" color="green" label="Pump1" />
43         <chart id="FlowRate2" color="yellow" label="Pump2" />
44     </graph>
45
46     <!--
47     display only
48     -->
49     <label name="connection"
50         location="0, 290"
51         label="Connection:"/>
52     <image name="led"
53         location="85, 290"
54         src="connected.JPG" />
55
56     <label name="ltime"
57         location="120, 290"
58         label="Time:"/>
59
60     <timer name="timer"
61         size="100, 20"
62         location="160, 290"
63         color="green" />
64
65     <tabs name="tabs1"
66         location="0, 320"
67         size="700, 200">
68
69         <!--
70         manual control panel
71         -->
72         <panel name="manual"
73             label="Manual Control">
74
75             <label name="p1"
76                 location="100, 40"
77                 label="Pump 1"/>
78
79             <spinner name="Pump1"
80                 size="40, 20"
81                 location="160, 40"
82                 min="0"
83                 max="5"
84                 step="0.1"/>

```



```

85
86         <label name="p2"
87             location="100, 80"
88             label="Pump 2"/>
89
90         <spinner name="Pump2"
91             size="40, 20"
92             location="160, 80"
93             min="0"
94             max="5"
95             step="0.1"/>
96
97         <image location="230, 10"
98             src="pump.JPG" />
99
100        <label name="l1"
101            location="360, 30"
102            label="Level 1"/>
103
104        <label name="l2"
105            location="360, 70"
106            label="Level 2"/>
107
108        <line location="195, 50"
109            size="210, 0"
110            thickness="3" />
111
112        <line location="195, 90"
113            size="210, 0"
114            thickness="3" />
115    </panel>
116
117    <!--
118    Closed loop control start here
119    -->
120    <panel name="closed"
121        label="Closed Loop">
122
123        <label name="sp1"
124            location="40, 40"
125            label="Set Point"/>
126
127        <spinner name="sp"
128            size="40, 20"
129            location="60, 60"

```

130	min="0"
131	max="5"
132	step="0.1"/>
133	
134	<line location="105, 70"
135	size="335, 0"
136	thickness="3" />
137	
138	<circle location="130, 60"
139	size="20, 20" />
140	
141	
142	<line location="445, 50"
143	size="165, 0"
144	thickness="3" />
145	
146	<line location="445, 90"
147	size="165, 0"
148	thickness="3" />
149	
150	<tabs name="tabs1"
151	location="160, 20"
152	size="160, 120">
153	
154	<panel name="onoff"
155	label="ON/OFF">
156	
157	<label name="amp"
158	location="10, 20"
159	label="Amplitude"/>
160	
161	<spinner name="amplitude"
162	size="40, 20"
163	location="90, 20"
164	min="0"
165	max="10"
166	step="0.1"/>
167	
168	<label name="dead"
169	location="10, 60"
170	label="Dead Zone"/>
171	
172	<spinner name="deadzone"
173	size="40, 20"
174	location="90, 60"

```

175         min="0"
176         max="5"
177         step="0.1"/>
178     </panel>
179
180     <panel name="pid" label="PID">
181
182         <label name="c"
183             location="50, 20"
184             label="Kc"/>
185
186         <spinner name="kc"
187             size="40, 20"
188             location="80, 20"
189             min="0"
190             max="50"
191             step="0.1"/>
192
193         <label name="i"
194             location="10, 60"
195             label="Ti"/>
196
197         <spinner name="ti"
198             size="40, 20"
199             location="30, 60"
200             min="0"
201             max="200"
202             step="0.1"/>
203
204         <label name="d"
205             location="80, 60"
206             label="Td"/>
207
208         <spinner name="td"
209             size="40, 20"
210             location="100, 60"
211             min="0"
212             max="50"
213             step="0.1"/>
214     </panel>
215
216     <panel name="mpc"
217         label="MPC">
218 </panel>
219

```

```

220     </tabs>
221
222     <label name="lbias"
223         location="340, 10"
224         label="Bias"/>
225
226     <spinner name="bias"
227         size="40, 20"
228         location="340, 30"
229         min="0"
230         max="5"
231         step="0.1"/>
232
233     <circle location="345, 60"
234         size="20, 20" />
235
236     <line location="355, 50"
237         size="0, 10"
238         thickness="3" />
239
240     <image location="480, 30"
241         src="pump.JPG" />
242
243     <label name="p12"
244         location="430, 30"
245         label="Pump 1"/>
246
247     <label name="l12"
248         location="570, 30"
249         label="Level 1"/>
250
251     <label name="p22"
252         location="430, 92"
253         label="Pump 2"/>
254
255     <label name="l22"
256         location="570, 92"
257         label="Level 2"/>
258
259     <line location="615, 70"
260         size="55, 0"
261         thickness="3" />
262
263     <line location="640, 70"
264         size="0, 85"

```

265	thickness="3" />
266	
267	<line location="140, 70"
268	size="0, 85"
269	thickness="3" />
270	
271	<line location="140, 155"
272	size="500, 0"
273	thickness="3" />
274	</panel>
275	</tabs>
276	</AML>

Table 6.1: AML for Coupled Tank Apparatus

The Apparatus Markup Language for Ohm's Law Apparatus

1	<?xml version="1.0" encoding="UTF-8"?>
2	
3	<!--
4	AML for Ohm's Law. The UI will be update when 'Measure' button is
5	clicked. Communication Port is 2002.
6	-->
7	<AML update="manual" port="2002" >
8	
9	<line location="130, 70"
10	size="210, 0"
11	thickness="3" />
12	
13	<!--
14	meter for display voltage only
15	-->
16	<meter name="V1"
17	location="160, 30"
18	size="150"
19	min="0"
20	max="10"
21	increment="2"
22	step="1"
23	precision="2"
24	indicator="on"
25	type="indicator" />
26	
27	<line location="130, 180"

```

28         size="210, 0"
29         thickness="3" />
30
31     <!--
32     switch for turning on or off the circuit
33     -->
34     <switch name="SW"
35         location="200, 150"
36         size="60, 60"
37         on="1"
38         off="0" />
39
40     <line location="130, 260"
41         size="680, 0"
42         thickness="3" />
43
44     <!--
45     meter for display current only
46     -->
47     <meter name="I"
48         location="360, 220"
49         size="150"
50         min="0"
51         max="0.01"
52         increment="0.005"
53         step="0.0025"
54         precision="4"
55         indicator="on"
56         type="indicator" />
57
58     <line location="130, 500"
59         size="680, 0"
60         thickness="3" />
61
62     <line location="130, 70"
63         size="0, 430"
64         thickness="3" />
65     <line location="340, 70"
66         size="0, 190"
67         thickness="3" />
68
69     <line location="540, 260"
70         size="0, 240"
71         thickness="3" />
72

```

```

73      <meter name="V2"
74          location="470, 340"
75          size="150"
76          min="0"
77          max="10"
78          increment="2"
79          step="1"
80          precision="2"
81          indicator="on"
82          type="indicator" />
83
84      <line location="670, 260"
85          size="0, 240"
86          thickness="3" />
87
88      <switch name="SW1"
89          location="640, 280"
90          size="60, 60"
91          on="1"
92          off="0" />
93
94      <line location="740, 260"
95          size="0, 240"
96          thickness="3" />
97
98      <switch name="SW2"
99          location="710, 280"
100         size="60, 60"
101         on="1"
102         off="0" />
103
104     <line location="810, 260"
105         size="0, 240"
106         thickness="3" />
107
108     <switch name="SW3"
109         location="780, 280"
110         size="60, 60"
111         on="1"
112         off="0" />
113
114     <label location="210, 340"
115         label="+"
116         color="red" />
117

```

```

118     <label location="210, 360"
119         label="V"
120         color="red" />
121
122     <label location="210, 375"
123         label="_"
124         color="red" />
125
126     <!--
127     a knob is used to change the power supply of the circuit
128     -->
129     <knob name="Vin"
130         location="60, 300"
131         size="150"
132         min="0"
133         max="10"
134         increment="1"
135         precision="2"
136         step="0.5"
137         indicator="on" />
138
139     <label name="r"
140         location="170, 250"
141         size="100, 20"
142         label="R"
143         background="red" />
144
145     <label name="r1"
146         location="660, 380"
147         size="20, 100"
148         label="R1"
149         background="red" />
150
151     <label name="r2"
152         location="730, 380"
153         size="20, 100"
154         label="R2"
155         background="red" />
156
157     <label name="r3"
158         location="800, 380"
159         size="20, 100"
160         label="R3"
161         background="red" />
162

```


163	<!--
164	update the UI using the new data read from server
165	-->
166	<button name="measure"
167	location="730, 520"
168	label="Measure"
169	value="Measure=1"
170	action="refresh" />
171	
172	</AML>

Table 6.2: AML for Ohm's Law Apparatus

References

- [1]. IEEE Learning Technology Standards Committee (LTSC), available at:
<http://ltsc.ieee.org>
- [2]. Advanced Distributed Learning (ADL), available at: <http://www.adlnet.org>
- [3]. IMS Global Consortium Inc (IMS), available at: <http://www.imsproiect.org>
- [4]. Alliance of Remote Instructional Authoring and Distribution Networks for Europe,
available at: <http://www.ariadne-eu.org/>
- [5]. Aviation Industry CBT Committee (AICC), available at: <http://www.aicc.org>
- [6]. IEEE LTSC, “Draft Standard for Learning Object Metadata”, 15 July 2002, IEEE
1484.12.1-2002
- [7]. ADL, “Sharable Content Object Reference Model (SCORM)”, 2004 Overview
- [8]. IMS, “IMS Learning Design Best Practice and Implementation Guide”, Version 1.0
Final Specification, 20 January 2003
- [9]. Maish Nichani, “LCMS = LMS + CMS [RLOs]”, 02 May 2001, available at:
<http://www.elearningpost.com>
- [10]. Lambertus Hesselink, Dharmarus Rizal, Eric Bjornson, Sandy Paik, Raj Batra, &
Peter Catrysse, Dan Savage, Anthony Wong. (2003). StanfordCyberLab: Internet
Assisted Laboratories, Journal of Distance Education Technologies, 1(1), 22-39,
Jan-Mar 2003.
- [11]. Benetazzo, L.; Bertocco, M.; Ferraris, F.; Ferrero, A.; Offelli, C.; Parvis, M.; Piuri,
V.; A Web-based distributed virtual educational laboratory. IEEE Transactions on
Instrumentation and Measurement, Volume: 49, Issue: 2, April 2000, Pages: 349 –
356

-
- [12]. F. J. González-Castaño, L. Anido-Rifón, J. Vales-Alonso, M. J. Fernández-Iglesias, M. Llamas Nistal, P. Rodríguez-Hernández and J. M. Pousada-Carballo (2001). Internet access to real equipment at computer architecture laboratories using the Java/CORBA paradigm. *Computers & Education*, Volume 36, Issue 2, February 2001, Pages 151-170
- [13]. Song You, Tianmiao Wang, Roy Eagleson, Cai Meng and Qixian Zhang (2001). A low-cost internet-based telerobotic system for access to remote laboratories, *Artificial Intelligence in Engineering*, Volume 15, Issue 3, July 2001, Pages 265-279
- [14]. Guimaraes, E.; Maffeis, A.; Pereira, J.; Russo, B.; Cardozo, E.; Bergennan, M.; Magalhaes, M.F.; REAL: a virtual laboratory for mobile robot experiments. *IEEE Transactions on Education*, Volume: 46, Issue: 1, Feb 2003, Pages: 37 – 42
- [15]. R.Pastor, J.Sanchez and S.Dormido. An XML-based Framework for the Development of Web-based Laboratories Focused on Control Systems Education. *International Journal of Engineering Education*. Vol. 19, No. 3 pp. 445-454, 2003
- [16]. G. D. Abowd; Classroom 2000: An Experiment with the Instrumentation of a Living Educational Environment, *IBM Systems Journal*, Volume 38, Issue 4, Pages: 508-530, 1999
- [17]. Ling K V, Lai Y K and Chew K B, An Online Internet Laboratory for Control Experiments, *Preprints of IFAC/IEEE Symposium on Advances in Control Education*, 17-19Dec 2000, Australia
- [18]. M. Hough, T. Marlin, Web-based Interactive Learning Modules for Process Control, *Computers and Chemical Engineering*, Volume 24, Issues 2-7, 15 July 2000, Pages 1485-1490
-

-
- [19]. Kuk-Hyun Han, Sinn Kim, Yong-Jae Kim and Jong-Hwan Kim, Internet Control Architecture for Internet-Based Personal Robot, Autonomous Robots Journal, Kluwer Academic Publishers, Vol. 10, No. 2, pp. 135-147, March 2001
- [20]. Ko, C.C.; Chen, B.M.; Jianping Chen; Zhuang, Y.; Chen Tan, K, Development of a web-based laboratory for control experiments on a coupled tank apparatus, Education, IEEE Transactions on, Volume: 44 Issue: 1, Feb 2001 Page(s): 76 – 86
- [21]. Anido, L.; Llamas, M., A contribution to the e-learning standardization, Standardization and Innovation in Information Technology, 2001 2nd IEEE Conference, 2001, Page(s): 295 –309
- [22]. Amaratunga, K. and Sudarshan, Raghunathan. A Virtual Laboratory for Real-Time Monitoring of Civil Engineering Infrastructure. International Conference on Engineering Education, August 2002, Manchester, UK
- [23]. Sanchez, J.; Morilla, F.; Dormido, S.; Aranda, J.; Ruiperez, P. Virtual and remote control labs using Java: a qualitative approach, IEEE Control Systems Magazine, Volume: 22 Issue: 2, April 2002 Page(s): 8 –20
- [24]. Dongil Shin, En Sup Yoon, Kyung Yong Lee, Euy Soo Lee. A Web-based, interactive virtual laboratory system for unit operations and process systems engineering education: issues, design and implementation Computer and Chemical Engineering, Volume 26, Issue 2, 15 February 2002, Pages 319-330
- [25]. Perrin, S.; Benoit, E.; Foulloy, L.; Automatic code generation based on generic description of intelligent instrument, Man and Cybernetics, 2002 IEEE International Conference on Systems, Volume: 6, 6-9 Oct. 2002
- [26]. S.H. Yang, X. Chen, J.L. Alty. (2002) Design issues and implementation of internet-based process control systems, Control Engineering Practice, Volume 11, Issue 6, June 2003, Pages 709-720
-

-
- [27]. J. Henry and C. Knight, Modern Engineering Laboratories at a Distance, International Journal of Engineering Education, Vol. 19, No. 3, pp. 403-408, 2003
- [28]. T. Kikuchi, S. Fukuda, A. Fukuzaki, K. Nagaoka, K. Tanaka, T. Kenjo, D. A. Harris. "DVTS-Based Remote Laboratory Across the Pacific Over the Gigabit Network", IEEE Transactions on Education, Volume 47, Issue 1, February 2004, Pages: 26 – 32
- [29]. Hoyer, H.; Jochheim, A.; Rohrig, C.; Bischoff, A.; A Multi-user Virtual-Reality Environment for a Tele-Operated Laboratory, IEEE Transactions on Education, Volume: 47, Issue: 1, Pages: 121 – 126, Feb. 2004
- [30]. A. A. Kassim, S. A. Kazi and S. Ranganath, A Web-based Intelligent Learning Environment for Digital Systems, International Journal of Engineering Education, Vol. 20, No. 1, pp. 13-23, 2004
- [31]. Eileen Scanlon, Chetz Colwell, Martyn Cooper and Terry Di Paolo, Remote experiments, re-versioning and re-thinking science learning, Computers & Education, Volume 43, Issues 1-2, August-September 2004, Pages 153-163
- [32]. Blackboard from BlackBoard Inc., available at: <http://www.blackboard.com>
- [33]. Swift Author from Gemini Performance Solutions Inc, available at: <http://www.nemini.com/>
- [34]. Authorware from Macromedia Inc., available at: <http://www.macromedia.com>
- [35]. TopClass from WBT Systems, available at: <http://www.wbtvsystems.com>
- [36]. Unified Modeling Language (UML) from Object Management Group, available at: <http://www.uml.org>
- [37]. Senvid's Summit System: Remote Laboratory, available at: <https://www.senvid.net/>
- [38]. Rensselaer Polytechnic Institute's AIM-Lab, available at: <http://nina.ecse.rpi.edu/shur/remote/>
-

-
- [39]. University Graduate Center's LAB-on-WEB, available at: <http://www.lab-on-web.com/>
- [40]. Chalmers University of Technology's I-Lab, available at: <http://www.ic.chalmers.se/ilab/>
- [41]. National University of Singapore's VLab, available at: <http://vlab.ee.nus.edu.sg>
- [42]. Massachusetts Institute of Technology's i-Lab, available at: <http://i-lab.mit.edu/>
- [43]. Gottfried Zimmermann, Gregg Vanderheiden, and Al Gilman, "Universal Remote Console Prototyping of an Emerging XML Based Alternate User Interface Access Standard", Eleventh International World Wide Web Conference, 7-11 May 2002, Honolulu, Hawaii
- [44]. Roland A. Merrick (IBM), "An XML Vocabulary for Describing User Interfaces. [Device Independent User Interfaces in XML]", 09 May 2001
- [45]. extensible Interface Markup Language (XIML), available at: <http://www.ximl.org>
- [46]. Extensible User Interface Language (XUL), available at: <http://www.mozilla.org/projects/xul/>
- [47]. OASIS User Interface Markup Language (UIML), available at: <http://www.oasis-open.org/committees/uiml/>
- [48]. NASA's Instrumentation Markup Language, available at: <http://pioneernsfc.nasa.gov/public/iml/>
- [49]. Nacimiento Software Corporation, Virtual Instrumentation Markup Language, available at: <http://www.nacimiento.com/VIML>
- [50]. Promoting Multimedia in Education and Training in European Society (PROMETEUS), available at: <http://prometeus.org/>
- [51]. ECMAScript Language Specification (ECMA-262), available at: <http://www.ecma-international.org/>
-

-
- [52]. XML.ORG, available at: <http://www.xml.org>
- [53]. IEEE Std 488.2-1992, IEEE Standard Digital Interface for Programmable Instrumentation, available at: <http://standards.ieee.org/>
- [54]. Semiconductor Equipment and Materials International (SEMI), available at: <http://www.semi.org>
- [55]. International SEMATECH Manufacturing Initiative (ISMI), available at <http://ismi.sematech.org/>
- [56]. Association Connecting Electronics Industries, available at: <http://www.ipc.org>

Author's Publications

- [1]. Bing DUAN, Keck-Voon LING and Habib Mir M. HOSSEINI, Developing a Framework for Online Laboratory Learning Objects, 2nd WSEAS International Conference on E-ACTIVITIES (Singapore, December 7-9, 2003), pp 467-471
- [2]. Bing DUAN, Keck-Voon LING and Habib Mir M. HOSSEINI, Developing and Implementing Online Laboratory for Control Engineering Education, ICARCV 2004, Kunming, China (March 2004, accepted)
- [3]. Habib Mir HOSSEINI, Keck-Voon LING, Bing DUAN, Robert Kheng Leng GAY, "Learning Object Model for Laboratory-Based Lessons", WSEAS Transactions on Computer, Issue 4, Volume **2**, Pages 1144-1148, October 2003.
- [4]. Bing DUAN, Keck Voon LING, Habib Mir M. HOSSEINI and Robert Kheng Leng GAY, An e-Learning Model for Lab-based Courses, International Journal of Engineering Education, March 2004 (Accepted)
- [5]. Bing DUAN, Habib Mir M. HOSSEINI, Keck Voon LING and Robert Kheng Leng GAY, An Architecture for Online Laboratory e-Learning System, International Journal of Distance Education Technologies, April 2004 (Submitted)