國立交通大學

資訊科學與工程研究所

碩士論文

一個電腦化互動多元評量系統的設計與實作

Design and Implementation of a Computerized Interactive Multiple Assessment System

研 究 生：李杰峰
指導教授：曾憲雄　教授

中華民國 九十九年七月

一個電腦化互動多元評量系統的設計與實作

Design and Implementation of a Computerized Interactive Multiple Assessment System

研 究 生：李杰峰 Student：Chieh-Feng Li

指導教授：曾憲雄 Advisor：Dr. Shian-Shyong Tseng

國 立 交 通 大 學
資 訊 科 學 與 工 程 研 究 所

碩 士 論 文

A Thesis
Submitted to Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer Science

July 2010

Hsinchu, Taiwan, Republic of China

中華民國九十九年七月

一個電腦化互動多元評量系統的設計與實作

研究生：李杰峰　　　　　　　指導教授：曾憲雄博士

國立交通大學

資訊科學與工程研究所

摘　要

科學探究能力是科學教育所希望培養學生具備的主要能力之一，因此透過評量老師可以了解學生探究能力的好壞並給予指導。而最適合並且最能發揮效果的評量方式，是透過學生在實驗操作的過程老師從旁觀察並給予指導。但是在實驗室進行實驗的成本是比較高的，而且進行實驗的時間時常不足。虛擬實驗(Virtual Laboratory)可以讓學生不斷嘗試錯誤、無器材耗損且實驗具安全性，同時解決老師時間不足無法兼顧每位學生的問題，因此適合用來進行科學探究能力的評量。但是要建構一個虛擬實驗並不容易，現有的編輯工具無法容易的編輯出不同的虛擬實驗。因此在這篇論文中，我們採用一個知識工程技術的方法，提出一套建構虛擬實驗的機制，透過對話式的方式引導老師利用 script language 完成實驗設計。在我們提出的 *Computerized Interactive Multiple Assessment Model* 中，將虛擬實驗的知識看作 default knowledge 和 experiment-specific knowledge 兩部份，透過知識擷取的方法將 default knowledge 進行前處理來降低設計實驗的成本。同時這個架構使學生的操作容易被記錄下來並且能提供後續診斷系統的研究。最後我們實做了一個系統驗證它能夠支援不同科學教育學科及不同實驗的編輯，而對老師進行的滿意度問卷調查及面談也顯示老師對編輯方式和虛擬實驗所呈現的效果相當滿意。

關鍵字：虛擬實驗、知識工程技術

Design and Implementation of a
Computerized Interactive Multiple Assessment System

Student: Chieh-Feng Li **Advisor: Dr. Shian-Shyong Tseng**

Institute of Computer Science and Engineering
Nation Chiao Tung University

Abstract

Scientific inquiry ability is of most importance in science education; hence the assessment of this ability can help teacher to assist students in developing this ability. The most effective way of this assessment is that teachers watch students conducting an experiment and give immediate feedbacks. However, taking a hands-on experiment is costly and time-consuming. The virtual laboratory can solve these drawbacks. But constructing virtual laboratories is still not easy using the existing authoring tools. In this thesis, a mechanism of virtual laboratory construction is proposed based upon knowledge-based approach. The teacher finishes experiment design by the proposed script language through a dialogue-based way. In the proposed *Computerized Interactive Multiple Assessment Model*, the knowledge of virtual laboratory is seen as default knowledge and experiment-specific knowledge. The default knowledge is preprocessed through a knowledge acquisition process to reduce the cost of construction. The proposed mechanism can log students' portfolio for further analysis. A prototype system is also implemented to evaluate the proposed model. The result shows this model can support various experiments in different domain, and teacher is satisfied about the experiment design process and the effect of generated virtual laboratories.

Keywords: virtual laboratory, knowledge-based approach

誌　謝

研究所這兩年的學習，是我人生中寶貴的經歷。從小到大的求學歷程，讓我早已習慣於標準答案的背誦，而很少嘗試去思考問題。因此首先我要感謝我的指導教授 曾憲雄博士，他總是不厭其煩的導正我的思路，一步步地教導我做研究的方法，培養我獨立思考的能力。感謝他願意花這麼多時間在我們這群學生身上，使我們獲益良多。同時，我要感謝口試委員：黃國禎教授、賀嘉生教授和袁賢銘教授，給予我的論文許多寶貴的意見，讓我的論文更加充實。

這篇論文能夠順利的完成，還要感謝實驗室的喚宇學長、蘇董學長、瑞鋒學長以及其他熱心的學長姐，他們在我研究的過程中提供我許多協助和建議，鼓勵著我繼續走下去。再來我要感謝實驗室的同學們：紹宜、金龍、嘉祥以及我最好的夥伴國彰和佳榕，因為有你們一起奮鬥，所以這段過程除了辛苦還有歡樂。

最後，我要感謝養我二十幾年的父母，謝謝他們對我的愛和包容。還有當年一起準備研究所考試的采韻，感謝你當時幫了我許多忙。也謝謝我的好朋友佩蓉、育賢和乃瑩，在研究所生涯中，我們彼此相互鼓勵。

Table of Content

List of Tables

List of Figures

Chapter 1. Introduction

Scientific inquiry, which is an ability of asking questions, investigating the natural world, and explaining based upon derived evidence, has become one of the most important learning objectives of science education [3]. Thus, teachers require an effective assessment to evaluate students' performance of scientific inquiry and offer appropriate supplementary instruction. A process of scientific inquiry incorporates multiple basic and integrated *science process skills (SPS)*, used to conduct investigations and conclude the results [4]. AAAS [5] defined thirteen skills of science process skills: Observing, Classifying, Using space/time relations, Predicting, Using numbers, Measuring, Controlling and identifying variables, Inferring, Formulating Hypotheses, Defining Operationally, Interpreting data, Communicating, and Experimenting. In order to assess these skills, letting students conduct real experiments for a specific question to verify their own hypotheses is an effective approach [1]. However, facilitating and monitoring experiments is time-consuming and costly for teachers, where students are asked to actively plan and investigate for their own questions. Moreover, some experiments are also dangerous for novices.

Some studies proposed virtual laboratories to reduce both cost and danger of hands-on experiments. These virtual laboratories could also monitor students' inquiry processes and provide logs of experiments for teachers to evaluate students' performance. However, some virtual laboratories [2] offering only demonstrations of experiments lacked flexibility for students to manipulate and control the variables. Some studies [9,10,12,13,14,15] constructed environments of specific experiments for scientific inquiry, but constructing a hard-coding virtual experimental environment for each specific experiment was costly and time consuming. In order to overcome this issue, some studies proposed authoring tools [7,8] to edit virtual laboratories of specific scientific domains, such as chemical experiments. These domain-specific authoring tools could facilitate teachers to construct various virtual laboratories of the specific domain, but could only construct very narrow kinds of experiments. Thus, proposing a new approach to facilitate teachers to design virtual laboratories without limitations of specific domains is an important issue.

The knowledge-based system can reduce construction cost by modifying knowledge to generate variously different virtual laboratories without modifying the program. In virtual laboratories, the default knowledge which means experimental environment settings like devices and operations has stereotype that is fit to be modeled using frame knowledge representation. To preprocess the default knowledge content can reduce construction cost since experiment designer can reduce the reworking in default knowledge and focus on the specific experimental setting. A proposed knowledge acquisition method is used to preprocess the default knowledge. To facilitate teachers in the experiment design, a dialogue-based script language is used to help teachers in designing.

In this thesis, a Computerized Interactive Multiple Assessment System (CIMAS) based upon a knowledge-based approach was proposed to model a virtual experiment as default knowledge and experiment-specific knowledge. The default knowledge expressed the experimental environment settings for specific domains, so replacing the default knowledge could vary the domain of a virtual experiment. Various predefined default knowledge was constructed by a proposed default-knowledge-preprocessing process via knowledge acquisition using modified repertory gird and shared for teachers. The experiment-specific knowledge of an experiment was the major objects and operations for the experiment. A dialogue-based knowledge acquisition was proposed to assist teachers in designing experiment-specific knowledge and reusing default knowledge to construct a new virtual experiment.

Finally, a prototype system of CIMAS was constructed to evaluate the ability of the proposed mechanism in supporting various virtual experiments of different domains. An experiment toward teachers was to evaluate the satisfaction about constructing a new virtual experiment and performing the virtual experiment for assessment. The case study showed that the system could support various virtual experiments in different domains. The experimental results showed the mechanism was effective in assisting teachers in designing a new virtual experiment and the performed virtual experiment could satisfy the requirements of a science process skills assessment.

The remainder of the article is organized as follows: Chapter 2 introduces current studies of virtual laboratories for assessment of science process skills. Subsequently, the CIMAS and CIMAM are described in Chapter 3. Chapter 4 introduces the dialogue-based knowledge acquisition, facilitating teachers to construct a new virtual experiment. The implementation of the prototype system and experiments are discussed in Chapter 5. Finally, Chapter 6 gives a conclusion and future works.

Chapter 2. Related Work

In this chapter, we will review the previous research results of virtual lab including the ready-made virtual lab and the authoring tool for virtual lab to assess science process skill.

2.1 Virtual Lab

In [6], virtual lab has been used to familiarize students on distance education with the real laboratory of school, where it can familiarize students with laboratory having the same architecture as that in the school laboratory as a preparation for doing hands-on experiment in the school. Before the students take their first hands-on experiment in school laboratory, the virtual lab helps them have basic concept about laboratory architecture and experiment devices. The virtual lab consisting of all the required virtual devices which can be found in the school lab does not really have the ability for students to conduct an experiment, but provides student a virtual laboratory environment to explore and get familiar with laboratory architecture.

In [11], there are three interactive learning applications which are used to enhance students' learning of hard-to-grasp concepts in dentistry.

The virtual lab mentioned above focusing on concepts and scientific knowledge learning did not emphasize the operations used in conducting experiment. So this kind of virtual lab are not suitable for science process skill assessment.

In the website [2] which was founded by Ministry of Education of Taiwan, there are plenty of virtual experiments in different scientific domain like chemistry, physics, biology, earth science and so on. The virtual experiment is most likely a demonstration which aims to validate knowledge or concepts written in textbook;

since the experiment flows are fixed, student should follow the predefined experiment flow to validate the result. This kind of virtual lab is still not suitable for science process skill assessment since every student would use same steps predefined by system to finish the experiment.

There were plenty of customized virtual laboratories that constructed environments of specific experiments for scientific inquiry like those studies in [9,10,12,13,14,15], but constructing a hard-coding virtual experimental environment for each specific experiment was costly and time consuming.

2.2 Authoring Tools of Virtual Lab

[7] provides teacher an authoring tool to construct diagram-based free-response assessment in electronics like logic circuit design. While teachers design the question, they can also set the marking file which is used to input their system for scoring of students' answer. The teacher can use this authoring tool to create different diagram-based assessment in electronics.

In [8], the authoring tool provides teacher the flexibility to add new chemicals and chemical equations. Students can operate predefined devices to conduct a chemical experiment, doing actions like mix chemicals or heat device to observe the reaction.

Those authoring tools of virtual lab mentioned above provide an easy way in constructing a specific experiment type. But the knowledge of experiments including device, medical and reaction are hard code in the program, all the implementation efforts of previous cases cannot help to largely reduce the effort of applying the authoring tool to a similar problem domain.

Based on related work survey, we find that if we want to use a virtual lab as the assessment of science process skills, we need a new platform to construct virtual experiment with enough expressive power and reduce construction cost in the same time.

Chapter 3. Computerized Interactive Multiple Assessment System

In Computerized Interactive Multiple Assessment System (CIMAS), a Computerized Interactive Multiple Assessment Model (CIMAM), including default knowledge and experiment-specific knowledge, was used to represent a virtual experiment. This model was proposed from the survey of virtual laboratory related works and experiments case study to induct. As shown in Figure 1, default knowledge defined common environments and operations of a specific experimental domain. The knowledge consisted of devices, structures, operations, and scenes. Devices were tools used in a kind of experiment; structures were assembled using devices for specific functions; operations were actions of using devices in an experiment; and scenes were the place where experiments were implemented. Experiment-specific knowledge represented experimental objects, experimental actions, and experimental settings. Experimental objects were major objects for manipulating and observing in experiments. Experimental actions were operations of manipulating experimental objects for specific experiments. The reactions of experimental objects after a period of time also belonged to experimental actions. Experimental Setting defined the composition of experimental objects and devices to form the structure for observing. To simplify our discussion, Scenario 1 is given and used in the rest of this thesis.

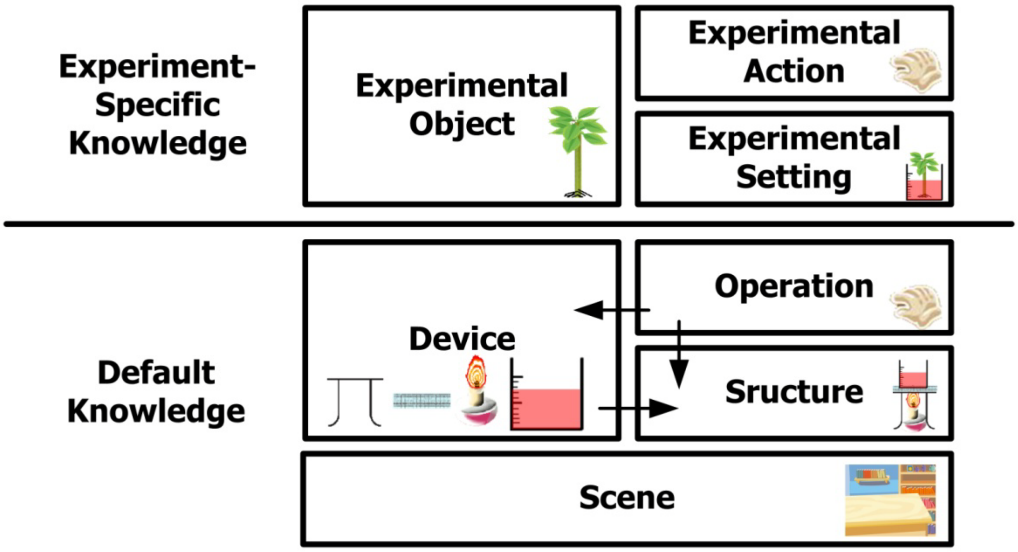


Figure 1. Computerized Interactive Multiple Assessment Model

Scenario 1: An Experiment Scenario and Its Mapping to CIMAM

The goals of the experiment "*The related factors in transpiration speed of plants*" originated from the natural curriculum of junior high school are to identify factors influencing transpiration speed of plants and determine relationship between these factors and the speed. In this experiment, students could use devices, such as a knife, a scale, a ruler, a graduated cylinder and vats to manipulate experimental objects, such as celery. Since celeries can be cut stem or pluck leaves the ways to change these variables of experimental object are experimental actions, like using knife to cut stem or using hands to pluck leaves. After students finishing the manipulations of the celery, they put the celery in a graduated cylinder filled with red ink. The "put" action, an experimental action to complete the experimental setting "celery in the graduated cylinder", can be conducted in two scenes, laboratory and playground, where different temperature of the scenes would influence the result. After finishing the experimental setting, students wait for a period of time, they observe the trend of the decrease of the quantity of red ink to inference the relation between factors and transpiration speed.

This experiment scenario has been constructed by our prototype system with a screenshot shown in Figure 2. The screenshot of the experiment result is shown in Figure 3.

Figure 2. The screenshot of "The Related Factors in Transpiration Speed of Plants"

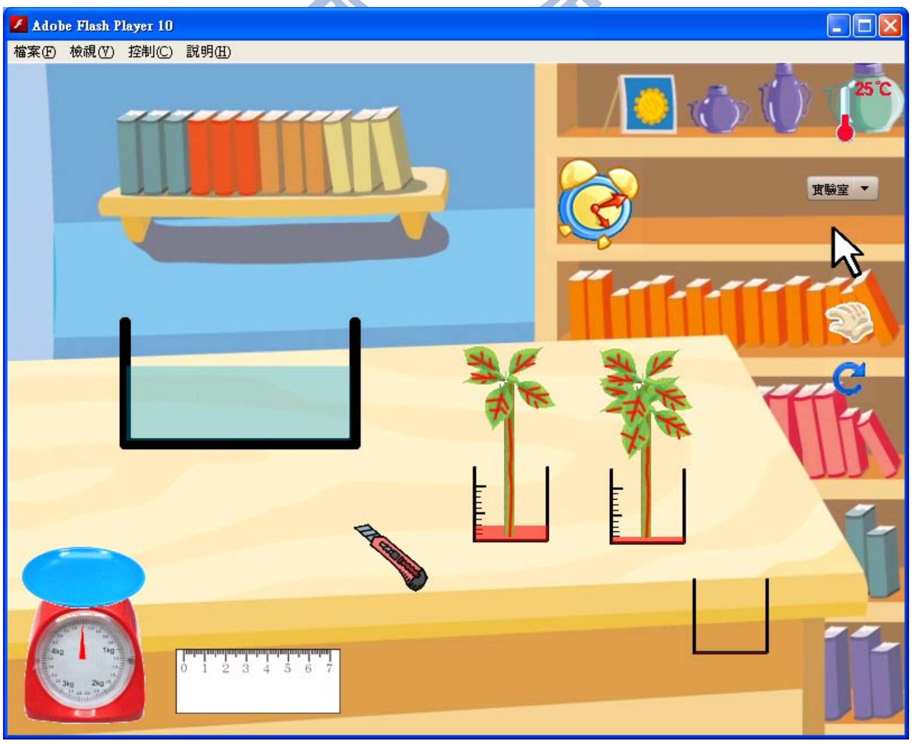


Figure 3 The screenshot of the experiment result

In order to facilitate modeling and executing the knowledge of virtual experiments, a frame-based representation was used to express the whole CIMAM because the proposed default and experiment-specific knowledge in all kinds of experiments was stereotyped and the hierarchical knowledge structure was appropriate to manage devices and experimental objects. The frames describe a knowledge entity using a set of slots, each slot is a 3-tuple contains slot name, slot type and slot value. The slot name is used to depict the attribute of the knowledge entity, each attribute has a value and the value belongs to one kind of data types. The frame representation has hierarchy property. A frame entity inherits slots of its parent frame, and is able to override those slots if needed. A frame also has the ability to represent embedded rules, which are attached to slots and are triggered when conditions are satisfied. The embedded rules have some types like IF-ADD, IF-NEEDED, IF-DELETED and so on.

Accordingly, system architecture of CIMAS, as shown in Figure 4, included a preprocessing phase, a construction phase, and a runtime phase. In the preprocessing phase, knowledge engineers generated frames of default knowledge, and stored them in a default-knowledge base. In construction phase, a dialogue-based knowledge acquisition assisted experiment designers in referring default knowledge and expressing their experiment-specific knowledge by the proposed script language. The script then was translated by a script translator, and the experiment-specific knowledge and used default knowledge were integrated to a frame-based virtual experiment, which was stored in an experiment pool for students to execute. In the runtime phase, a process of virtual experiment execution, incorporating a frame engine, could conduct a virtual experiment using these knowledge frame instances for students, and students' operation portfolios were logged for further diagnosis [16].

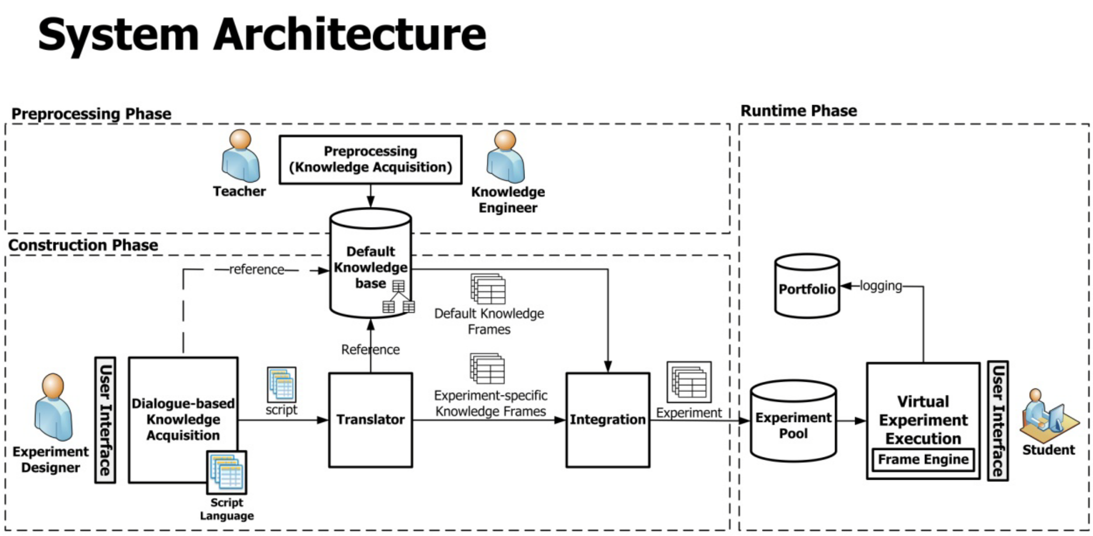


Figure 4. System Architecture of CIMAS

3.1 Computerized Interactive Multiple Assessment Model

In CIMAM, default knowledge represents default experimental environments and operations, and experiment-specific knowledge represents the specific objects and actions of an experiment.

Each component in default and experiment-specific knowledge of CIMAM are introduced in the following section.

3.1.1 Device

Device means frequently used devices in laboratories. Each device can use a Device Frame to depict. A Device Frame has attribute slots and action slots, where attribute slot represents some physical attributes of the device, like capacity, weight and so on, and action slot represents the action it can do by itself, like a “cut” action of a knife, a “heat” action of a beaker. There are two types of effect. One is to change attribute slot value of itself. For instance, the “heat” action of a beaker would change its temperature attribute. The other effect type is to change its appearance, as listed in

Table 1. Each action would correlate to one or more effects. Device changes its appearance to reflect the result after an action for student to observe, as shown in Table 2, where each attribute slot has a slot type of Number, String, Boolean, Enumeration and Record, and each action slot has a slot type of Procedure.

There is inheritance relation between devices. The child device frame inherits attributes and actions from the parent device frame. The child device can add new slots for attributes or actions. For example, a "beaker" can be seen as a subclass of "container." The beaker inherits attributes and actions like capacity, pour, and add new attribute slot of "graduation" and new action slot of "heat."

Table 1. Effect of Device

Effect Name	Description
Change Picture	Each device has a slot called picture list. It contains a list of picture file name. The Change Picture effect could change the device's displaying picture to others in the list.
Move	Moving to new position of the stage.
Rotate	Rotating device.
Scale	Scaling the device. The effect has some scale types like magnify along X-axis, magnify along Y-axis, shrink along X-axis, shrink along Y-axis and so on.
Change color	Changing the device color.
Change transparency	Changing the transparency of device.
Change visible	Changing the device into visible or not visible.

Table 2. Slot Type

Slot type name	description	Example
Number	A number that could be decimal.	10.5
String	Words.	Long
Boolean	Being one of two values: true or false	True
Enumeration	Enumerating the possible values.	{big, middle, small}
Record	A record is an n-tuple, with each element has a data type between Number, String, Boolean,	(water, 500, transparent)

	Enumeration.	
Rule	Checking restrictions of attribute slots, trigger action if the checking passed	IF $attrbute_i$=$value_1$ THEN trigger $action_j$
Procedure	Enable one or some effects. Effect has two type of changing attributes or changing appearance.	Change picture of $device_i$

Example 1: a device frame "graduated cylinder"

The graduated cylinder has attribute slots of Picture list, Capacity, Graduation and Liquid. The "Increasing liquid" and "Decreasing liquid" are the action slots of this device. The increasing liquid action has two effects, one is to change the liquid attribute and the other is to scale liquid like liquid increasing.

Table 3. a device frame "graduated cylinder"

Slot name	Slot type	Slot value
Name	String	graduated cylinder
Picture list	String	cylinder.gif
Capacity	Number	100
Graduation	Number	1
Liquid	Record	(none, 0)
Increasing liquid	Procedure	Changing attribute "liquid" Showing effect "scale liquid"
Decreasing liquid	Procedure	Changing attribute "liquid" Showing effect "scale liquid"

3.1.2 Structure

Structure means the composition of devices for the purpose to observe the result. Each Structure uses a Structure Frame to depict. A Structure Frame seems like a Device Frame, but has an additional slot called "components", it records device frames' name which compose the structure. The attribute slots and action slots in the Structure Frame have two types according to the slot value source. One is the same like Device Frame, that is, have initial value since the structure has been created. The other type, the slot value will come from the assembled devices. In this situation, an embedded rule IF-NEEDED attached in the slot of Structure Frame can provide value when the slot is referred.

Each device frame has a default slot called “composition”. This slot records the composition state with other devices currently. And this slot has an embedded rule IF-CHANGED, it will check Structure Frame if a structure has been set up since this operation. If current condition matches a structure frame then create a structure instance, or decompose a structure in the inverse condition.

Example 2: a structure frame “celery in the graduated cylinder”

The structure composed of celery and graduated cylinder in Components slot has two attribute slots of Leaves number and Capacity. The Inhale red ink is an action slot. In this case, both the attribute slot value and action slot value are come from devices of Components.

Table 4. a structure frame “celery in the graduated cylinder”

Slot name	Slot type	Slot value
Name	String	celery in the graduated cylinder
Components	String	celery, graduated cylinder
Leaves number	Enumeration	IF-NEEDED: get value from child device “celery”
Liquid	Number	IF-NEEDED: get value from child device “graduated cylinder”
Decreasing liquid	Procedure	IF-NEEDED: get action from child device “graduated cylinder”
Inhale red ink	Procedure	IF-NEEDED: get value from child device “celery”

3.1.3 Operation

There are two types of operation: two-device operation and one-device operation. The former means user uses actor device to do the action onto target device as shown in Table 5. Once the operation is triggered, the system checks the conditions and trigger actions of actor device or target device if the conditions of actor device attributes, target device attributes and scene's context attribute (introduced later in 3.1.4) are satisfied. The process of the latter is similar except having only target device. Besides trigger actions, an operation may change the "composition" slot in a device frame, which means changing composition state of a device.

Operation will be translated as a slot with slot type "rule" in the target device's frame. The content of the rule is "IF condition THEN trigger reaction."

Table 5. Manipulation Type

Manipulation Type	Description
Drag in	An operation between two devices. Simply drag $Device_i$ onto $Device_j$
Drag then Click	An operation between two devices. Drag $Device_i$ onto $Device_j$, then the mouse cursor would be replaced by $Device_i$, then using mouse to click would test if do any reaction
Drag with line	An operation between two devices. Drag $Device_i$ onto $Device_j$, then the mouse cursor would be replaced by $Device_i$, then using mouse to drag, the condition check would take track into account
Click	An operation toward one device. Simply using mouse cursor to click device
Drag	An operation toward one device. Simply using mouse cursor to drag device
Rotate	An operation toward one device. Using rotation mark cursor to click device.
Drag out	An operation toward one device. Using hand mark cursor to drag device.

Example 3: an Operation “pour” append to device frame “graduated cylinder” as a slot

The slots with white background color belong to original device frame of graduated cylinder. Now we have an operation called “Pour” which means a beaker pours liquid into this graduated cylinder. The operation will be appended as a rule slot like the last slot with gray background color showing in this case.

Table 6. an operation “pour” appended to device frame “graduated cylinder” as a slot

Slot name	Slot type	Slot value
Name	String	graduated cylinder
Picture list	String	cylinder.gif
Capacity	Number	100
Graduation	Number	1
Liquid	Record	(none, 0)
Increasing liquid	Procedure	Changing attribute “liquid” Showing effect “scale liquid”
Decreasing liquid	Procedure	Changing attribute “liquid” Showing effect “scale liquid”
Pour	Rule	IF manipulate type=drag then click & beaker-liquid > 0 & graduated cylinder-liquid < capacity THEN trigger “Increasing liquid” of “graduated cylinder” and “Decreasing liquid” of “beaker”

3.1.4 Scene

Scene is the place where the experiment conducts. Each scene uses a Scene Frame to depict. A scene frame contains context attribute slots like temperature, humidity and so on.

Example 4: a Scene frame “laboratory”

In this case, the scene frame using context attribute slots to record temperature, humidity, pressure and illumination of the laboratory.

Table 7. a scene frame “laboratory”

Slot name	Slot type	Slot value
Name	String	laboratory
Temperature	Number	25
Humidity	Number	70
Pressure	Number	1
Illumination	Number	300

3.1.5 Experimental Object

Experimental Object defines the target to manipulate, using Experimental Object Frame to represent. The Experimental Object Frame has same attributes like Device frame.

Example 5: an experimental object “celery”

The celery has attribute slot from “Number of leaves” to “Immerse in water”, and has action slot from “Pluck leaves” to “Cut stem”.

Table 8. an experimental object frame “celery”

Slot name	Slot type	Slot value
Name	String	celery
Picture list	String	celery1.gif,celery2.gif,celery3.gif,celery4.gif
Number of Leaves	Enumeration	10 of {10, 5, 0}
Stem length	Number	20.5
Stem cut	Enumeration	not cut of {not cut, straight cut, oblique cut}
Immerse in water	Boolean	false
Pluck leaves	Procedure	Changing attribute “Leaves number” Showing effect “change picture”
Inhale red ink	Procedure	Showing effect “change picture”
Cut stem	Procedure	Changing attribute “Stem cut” Showing effect “change picture”

3.1.6 Experimental Action

Experimental action consists of the operation about experimental object or experimental setting. Since time factor can be consideration as an action in an experiment, experimental action also means the experimental object's reaction in different situation after a period of time. An experimental action would be appended to an experimental object frame as a slot.

Example 6: an experimental action "cut celery" appended to experimental object frame "celery" as a slot

The slots with white background color belong to original experimental object frame of celery. Now we have an experimental action called "Cut celery" which represents using a knife cutting the celery stem to get rid of root. The experimental action will be appended as a rule slot like the last slot with gray background color showing in this case.

Table 9. an experimental action "cut celery" appended to experimental object frame "celery" as a slot

Slot name	Slot type	Slot value
Name	String	celery
Picture list	String	celery1.gif,celery2.gif,celery3.gif,celery4.gif
Number of Leaves	Enumeration	10 of {10, 5, 0}
Stem length	Number	20.5
Stem cut	Enumeration	not cut of {not cut, straight cut, oblique cut}
Immerse in water	Boolean	false
Pluck leaves	Procedure	Changing attribute "Leaves number" Showing effect "change picture"
Inhale red ink	Procedure	Showing effect "change picture"
Cut stem	Procedure	Changing attribute "Stem cut"

		Showing effect "change picture"
Cut celery	Rule	IF actor device = knife & target device = celery & manipulate type=drag with line & celery-stem cut = not cut THEN trigger action "cut stem" of celery

Example 7: an experimental action "after 10 minutes" appended on "celery in the graduated cylinder" experimental setting frame

The experimental action shows the change of the celery in the graduated cylinder after 10 minutes of the action. The experimental action is appended as a rule slot to the experimental setting frame.

Table 10. an experimental action "after 10 minutes" appended on "celery in the graduated cylinder" experimental setting frame

Slot name	Slot type	Slot value
Name	string	celery in the graduated cylinder
Components	string	celery, graduated cylinder
Number of Leaves	Enumeration	IF-NEEDED: get value from child device "celery"
Liquid	Number	IF-NEEDED: get value from child device "graduated cylinder"
Decreasing liquid	Procedure	IF-NEEDED: get action from child device "graduated cylinder"
Inhale red ink	Procedure	IF-NEEDED: get action from child device "celery"
Time effect	Rule	IF time passed 10 minutes & Liquid quantity > 0 THEN trigger Decreasing liquid & trigger Inhale red ink

3.1.7 Experimental Setting

Experimental Setting defines the composition of experimental objects and devices to form the structure for observing. The process is similar to that in 3.1.2. The difference between structure and experimental setting is that structure is the composition of devices in default knowledge, while experimental setting refers to the composition includes experimental object with other devices that is for specific experiment.

3.2 Generation of Virtual Experiment

Based upon the components of CIMAM given in Section 3.1, we will show how to generate the CIMAS virtual experiment from those frames.

A complete experiment stored in an experiment pool database as a set of frames firstly needs to acquire all the referenced default knowledge frames and its specific experiment frames, where a frame engine is required to handle the interaction between frames. The system provides some predefined user interface like a selector of scenes. While students are doing the experiments, the operation and experimental action would be logged. The attributes of devices and experimental object would also be logged; hence the system keeps a complete student portfolio in doing the experiment. Here is some examples to show the interactions of frames in the runtime phase.

Example 8: using a knife to cut the celery

This example shows the process to handle an operation. The operation is "using a knife to cut the celery." Firstly student drags knife onto celery with predefined manipulation type "drag with line." The manipulation would cause collision slot of celery changes its slot value. Then it triggers embedded rule IF-CHANGED to find out if there exists a suitable rule. Since the condition is match, "cut root" action be executed. It calls the "cut root" procedure, showing suitable effect on celery like change picture, and return the result to modify attribute slot.

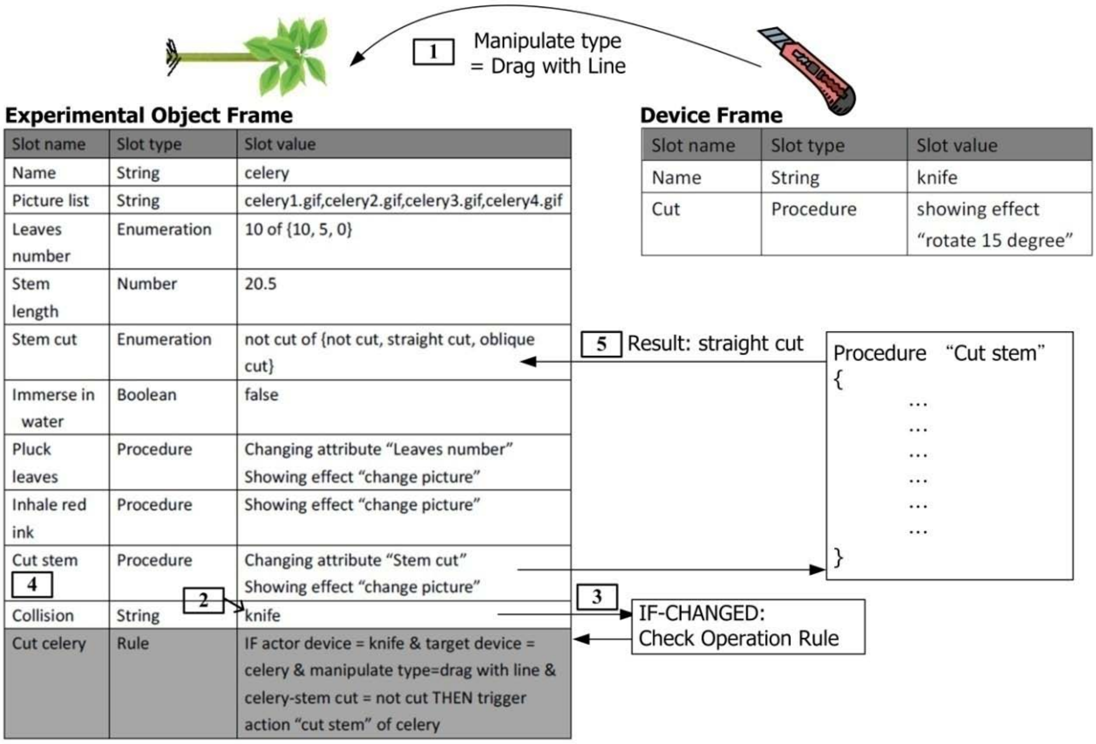
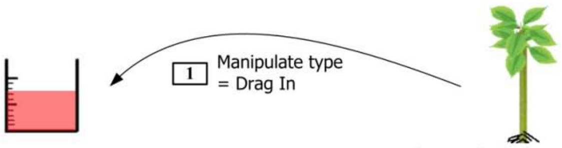


Figure 5. Example of the action "cutting stem"

Example 9: put the celery into graduated cylinder

This example shows an operation to composite a structure. Firstly student drags the celery into the graduated cylinder. The composition slot of graduated cylinder add new object celery, then trigger embedded rule IF-CHANGED to check if this operation composite a structure. A structure is matched or not, by checking components slot of the structure frame. Then system creates the structure instance.



Device Frame

Slot name	Slot type	Slot value
Name	String	graduated cylinder
Picture list	String	cylinder.gif
Capacity	Number	100
Graduation	Number	1
Liquid	Record	(none, 0) 2
Composition	String	graduated cylinder, celery
Increasing liquid	Procedure	Changing attribute "liquid" Showing effect "scale liquid"
Decreasing liquid	Procedure	Changing attribute "liquid" Showing effect "scale liquid"

3 IF-CHANGED: Check Structure

Experimental Object Frame

Slot name	Slot type	Slot value
Name	String	celery
Picture list	String	celery1.gif,celery2.gif,celery3.gif,celery4.gif
Leaves number	Enumeration	10 of {10, 5, 0}
Stem length	Number	20.5
	Enumeration	not cut of {not cut, straight cut, oblique cut}
Immerse in water	Boolean	false
Pluck leaves	Procedure	Changing attribute "Leaves number" Showing effect "change picture"
Inhale red ink	Procedure	Showing effect "change picture"
Cut stem	Procedure	Changing attribute "Stem cut" Showing effect "change picture"

5 Create structure frame instance

Structure Frame

Slot name	Slot type	Slot value
Name	String	celery in the graduated cylinder
Components	String	celery, graduated cylinder ← 4 Match
Leaves number	Enumeration	IF-NEEDED: get value from child device "celery"
Liquid	Number	IF-NEEDED: get value from child device "graduated cylinder"
Decreasing liquid	Procedure	IF-NEEDED: get action from child device "graduated cylinder"
Inhale red ink	Procedure	IF-NEEDED: get value from child device "celery"

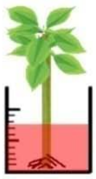
Figure 6. Example of the action "put the celery into graduated cylinder"

Example 10: the reaction of "celery in the graduated cylinder" after 10 minutes

This example shows the reaction of "celery in the graduated cylinder" after 10 minutes of simulation. Firstly student clicks the clock to choose ten minutes. Then the collision slot of this structure adds this action information and triggers embedded rule IF-CHANGED to find if there exists any matched rule. Since it can find a matched rule and satisfy those conditions, the actions of the corresponding rules are triggered.

1 Click the clock to choose "after 10 minutes"

Structure Frame



Slot name	Slot type	Slot value
Name	string	celery in the graduated cylinder
Components	string	celery, graduated cylinder
Leaves number	Enumeration	IF-NEEDED: get value from child device "celery"
Liquid	Number	IF-NEEDED: get value from child device "graduated cylinder"
Decreasing liquid	Procedure	IF-NEEDED: get action from child device "graduated cylinder"
Inhale red ink	Procedure	IF-NEEDED: get action from child device "celery"
Collision	String	Time=10
Time effect	Rule	IF time passed 10 minutes & Liquid quantity > 0 THEN trigger Decreasing liquid & trigger Inhale red ink

4 trigger

2

3 IF-CHANGED: Check Operation Rule

Figure 7. Example of the reaction "celery in the graduated cylinder after 10 minutes"

3.3 Default Knowledge Preprocessing

As we know, there is default knowledge in every experiment; e.g., the default knowledge related to devices is the major portion of default knowledge. Since the devices have properties of stereotype and inheritance, use frame is suitable to represent knowledge. Besides, the pre-defined default knowledge can reduce the construction cost because it reduces the reworking when designing a new experiment. In this thesis we propose a method to do knowledge acquisition for the preprocessing of devices and output the pre-defined default knowledge that is the knowledge frame hierarchy of devices with their attribute slots and action slots.

The proposed knowledge acquisition method is based on repertory gird which is a widely used method for knowledge acquisition. The original purpose of using repertory grid is to extract the knowledge to distinct different elements. In our approach, we use repertory grid in order to get the attribute slots and action slots of device frame in constructing the hierarchy of devices. The proposed algorithm KAofDK is shown in Algorithm 1.

In Step 1 we list all the frequently used devices in the domain of specific curriculum level. Then the teacher can add new devices or remove some devices of them. In Step 2 the teacher can classify those devices into some subclasses by their heuristic. In Step 3, we handle each subclass respectively. In Step 4 the teacher lists the common attributes and actions for each subclass, and that are the reasons why the teacher can classify those devices in the same class in Step 2. Those attributes and actions listed in this step would be the device frame slots and form the first level of device hierarchy. From Step 5 the repertory grid is used to acquire attributes and actions of each device in a subclass. First, add all devices in the subclass as elements of the repertory grid. Then repeatedly choose three of the elements, ask teacher to describe the difference between one and the other two, the differences would be the constructs of repertory grid. After finishing all the iteration of asking, teacher then fills the repertory grid with three kinds of value:

1: means the device tends to have trait.

2: means the device has no tendency about trait or opposite.

3: means the device tends to have opposite.

In order to cut down attributes that have similar meaning but have different notation, we compute the row similarity. If the similarity is greater than threshold, we provide teacher the information to decide whether to combine the two attributes. With the aim to group devices as a second level of device hierarchy, the column similarity is used to provide teacher the information of grouping devices with higher similarity over given threshold.

Algorithm 1. The knowledge acquisition to form the device frame hierarchy

Algorithm: KAofDK

Input

Devices that are frequently used in the domain.

Output

The slots for each device frame and the device frame hierarchy of default knowledge.

Step 1. List all of the frequently used devices.

Step 2. Classify listed devices into some subclasses by heuristic.

Step 3. For each subclass, repeat from step 4 to step 6.

Step 4. Ask teacher the common attributes and actions of this subclass. Form the first level of device frame hierarchy.

Step 5. Use repertory grid to get device frame slots.

Step 5.1. Add the devices to be the elements of repertory grid.

Step 5.2. Repeatedly choose three of the elements, ask teacher the attributes to distinct one element from the other two.

Step 5.3. Fill the repertory grid with three kinds of value.

1: means the device tends to have trait.

2: means the device has no tendency about trait or opposite.

3: means the device tends to have opposite.

Step 6. Compute the similarity to merge attributes and form the hierarchy.

Step 6.1. Compute the row similarity for attributes merge.

Step 6.2. Compute the column similarity to group devices and form the second level of device frame hierarchy.

Step 7. Finish all subclasses, Output the device frame hierarchy of default knowledge and the slots of each device frame.

Example 11: Knowledge acquisition process using five containers

Step 1.

We have devices of condenser, U-model tube, test tube, wide mouth bottle, beaker, graduated cylinder, phosphoric acid, nitric acid, Petri dish, test tube rack, separatory funnel rack, burette clamp, centrifuge tube holder and wash bottle.

Step 2.

The devices in step 1 were classified into five subclasses.

1.Container: wide mouth bottle, beaker, graduated cylinder, test tube, wash bottle

2.Tube: condenser, U-model tube

3.Rack: test tube rack, separatory funnel rack

4.Clamp: burette clamp, centrifuge tube holder

5.Chemicals: phosphoric acid, nitric acid

Step 3.

From now on we focus on the subclass of container.

Step 4.

Asking teacher the common attributes and actions of those containers.

The teacher may answer:

Container has common attributes: capacity

Container has common actions: pour in, pour out

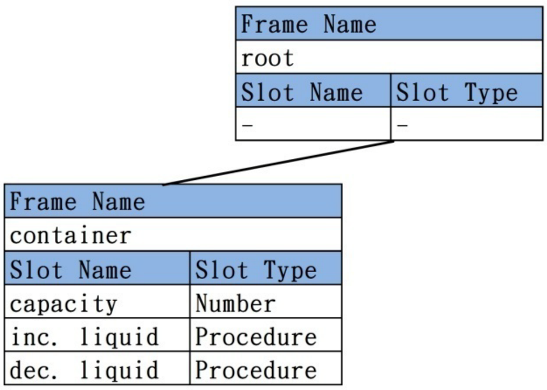


Figure 8. the first level of container in device frame hierarchy

Step 5.1

Add devices as elements of repertory grid.

Table 11. Fill out the element of repertory grid

	wide mouth bottle	**beaker**	**graduated cylinder**	**test tube**	**wash bottle**	

Step 5.2

Repeatedly choose three of the elements, ask teacher the differences between the three, the differences would be the construct of repertory grid.

Q1.What is the difference between wide mouth bottle, wash bottle and beaker?
A1.(The teacher answers) heat, temperature
Q2.What is the difference between beaker, wash bottle and graduated cylinder?
A2. graduation, heat
Q3.What is the difference between graduated cylinder, beaker and test tube?
A3. heat, graduation
Q4.What is the difference between test tube, wide mouth bottle and wash bottle?
A4. cover, acid endurance

Table 12. The original construct of repertory grid

	wide mouth bottle	beaker	graduated cylinder	test tube	wash bottle	
heat						**can't heat**
temperature						**no temperature**
graduation						**no graduation**
cover						**no cover**
acid endurance						**no acid endurance**

Step 5.3

Fill the repertory grid with three kinds of value.

Table 13. Filling out the values of repertory grid

	wide mouth bottle	beaker	graduated cylinder	test tube	wash bottle	
heat	**3**	**1**	**3**	**1**	**3**	can't heat
temperature	**2**	**1**	**2**	**1**	**2**	no temperature
graduation	**2**	**2**	**1**	**3**	**3**	no graduation
cover	**1**	**1**	**1**	**1**	**1**	no cover
acid endurance	**1**	**1**	**1**	**1**	**3**	no acid endurance

Step 6

Compute the column similarity, since beaker and test tube have higher similarity, we suggest teacher to group them, as “heatable container.”

Table 14. Count the column similarity to group devices

	wide mouth bottle	beaker	graduated cylinder	test tube	wash bottle	
heat	3	**1**	3	**1**	3	can't heat
temperature	2	**1**	2	**1**	2	no temperature
graduation	2	**2**	1	**3**	3	no graduation
cover	1	**1**	1	**1**	1	no cover
acid endurance	1	**1**	1	**1**	3	no acid endurance

Step 7

The output of container subclass frame hierarchy with attributes slot and action slots is shown in Figure 9. The first level of device frame “container” is obtained from step 4. The second level of device frame comes from step 6. In level 2, each device inheritances slots from parent frame, and has additional slots that has value “1” in repertory grid.

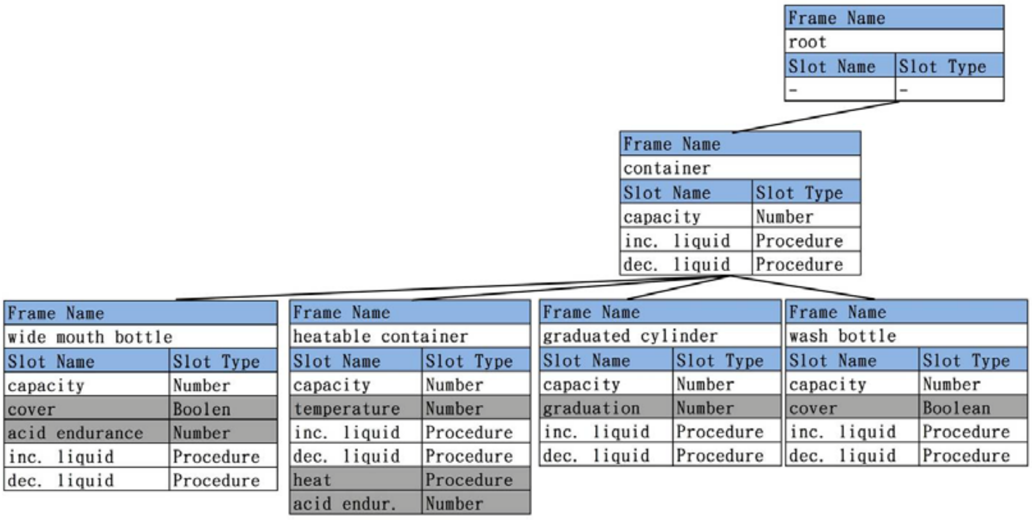


Figure 9. The result of frame device hierarchy

Since the frame knowledge representation is understandable, through a mechanism teachers can easily add new device frame or modify slots to maintain the knowledge.

Considering other components of default knowledge, operation can be appended to previously constructed device frame hierarchy. Since the structure and scene did not have attributes of inheritance, they can be preprocessed or learned by surveying experiment cases.

Chapter 4. Dialogue-based Script Language Construction

This chapter introduces how to construct an experiment. The proposed script language is used to depict an experiment including experiment-specific knowledge and referenced default knowledge. The goal of script language is to provide enough information for experiment construction. A dialogue-based knowledge acquisition is used to make the construction facile, where the dialogue emphasizes the facility of human-like thinking flow. The script language format and dialogue-based knowledge acquisition would be introduced in next two sections.

4.1 Script Language Approach

In this section, we will show the format of script language. In a hands-on experiment design, teachers often do not need to describe something like default knowledge because it is common sense. But the virtual experiment cannot work without understanding the common sense. To show all the information that construct a virtual experiment needed, the script language based on CIMAM does not distinctly separate experiment-specific knowledge and referenced default knowledge in the script format. For the part that is referenced from preprocessing default knowledge, some slots just need to copy and paste without starting from zero. In Figure 10, mapping between script language and CIMAM shows the system can get all the needed information from the script language.

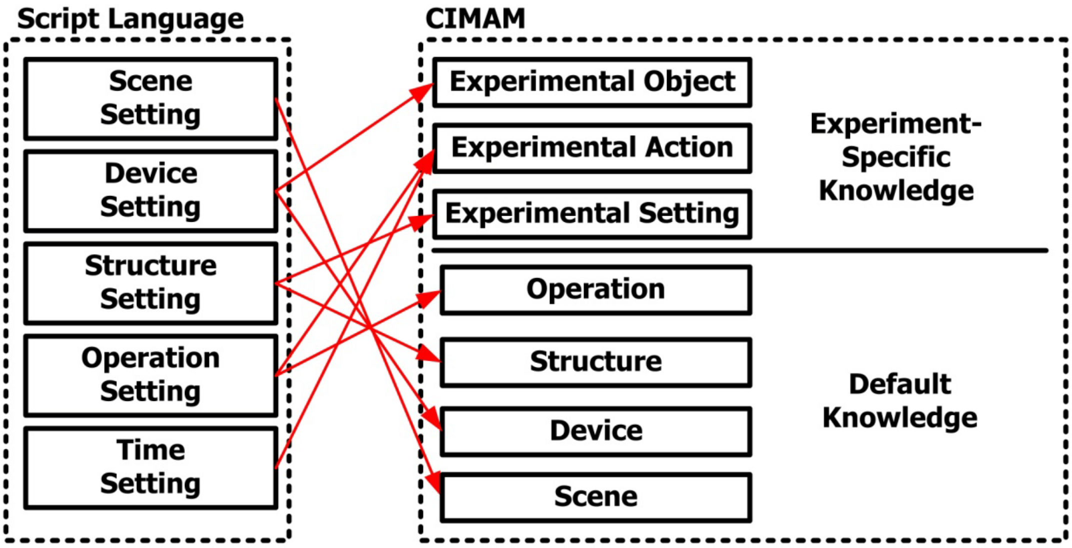


Figure 10. The Relation Between Script Language and CIMAM

The script language contains five parts which are scene setting, device setting, structure setting, operation setting and time setting. Scene setting in Table 15 describes which places the experiment conducting. The first column is a place where the experiment conducts. Each place may have many context attributes defined in the second column. Then in the third column teachers give the values to each context attribute.

Table 15. Part 1 of Script Language Format

Scene Setting		
Place	**Context Attribute**	**Value**
$place_1$	$context_1$	$value_1$
	$context_2$	$Value_2$

The device setting in Table 16 defines the devices and objects that can be manipulated in this experiment. The first column is the name of device. The number is set in the second column. Each device may have several attributes and actions. Designers choose a data type and give an initial value for each attribute. For each action, it has a type of procedure. The fifth column about an action has several action effects. Action effect has two type of changing attribute of itself or showing effect as mentioned in 3.1.1.

Table 16. Part 2 of Script Language Format

Device Setting				
Device Name	**Number**	**Attribute / Action Name**	**Attribute / Action Type**	**Attribute Value / Action Effect**
device$_1$	number$_1$	attribute$_1$	Number	value$_1$
		attribute$_2$	String	value$_2$
		action$_1$	Procedure	effect$_1$ effect$_2$
		action$_2$	Procedure	effect$_1$ effect$_2$

The structure setting in Table 17 defined the composition of devices to form an observing structure. In the column 2, teachers define the devices which compose the structure. The devices are chosen from previous defined in device setting phase. The structure may have some attributes and actions. Their names are listed in column 3. If an attribute or action is come from its component, teachers just need to fill the column 4 of the device name and finish the configuration of this attribute or action. If an attribute or action is newly generated, just skip column 4 and using column 5 and column 6 to set attributes and actions is similar to the way in device setting phase.

Table 17. Part 3 of Script Language Format

Structure Setting					
Structure Name	**Components**	**Attribute / Action Name**	**Attribute / Action Source**	**Attribute / Action Type**	**Attribute Value / Action Effect**
structure$_1$	device$_1$+object$_1$	attribute$_1$	device$_1$	n/a	n/a

		attribute$_2$	n/a	String	value$_1$
		action$_1$	device$_1$	n/a	n/a
		action$_2$	n/a	Procedure	effect$_1$

The operation setting in Table 18 means user's manipulation toward devices. It can happen under some conditions and cause the reaction. Every operation has a manipulation type defined in Table 5. If this operation is correlated to any two devices, using column 1 and column 3. If this operation is only correlated to one device, just fill the name in column 3. In the column 4, a list of condition is used to define the operation would happen under what condition. The conditions contain attributes of actor, target and environment. In the fifth column teachers define the reaction after students do this operation. It can be two types of trigger action of device or change device composition.

Table 18. Part 4 of Script Language Format

Operation Setting				
Actor	**Manipulate**	**Target**	**Conditions**	**Reaction**
actor$_1$	manipulate$_1$	target$_1$	condition$_1$ = (device$_1$, attribute$_1$, operator$_1$, value$_1$), condition$_2$ = (device$_2$, attribute$_2$, operator$_2$, value$_2$)	reaction$_1$

The time setting in Table 19 depicts how the time factor influences the devices in the experiment. If the device or structure in column 2 satisfies the condition of column 3, then after the time defined in column 1, the reaction of the target would be triggered.

Table 19. Part 5 of Script Language Format

Time Setting			
Passed time	**Target**	**Condition**	**Reaction**
$time_1$	$target_1$	$condition_1$ = ($device_1$, $attribute_1$, $operator_1$, $value_1$), $condition_2$ = ($device_2$, $attribute_2$, $operator_2$, $value_2$)	$reaction_1$

4.2 Dialogue-based Experiment Design Process

Since contents in script language have many dependence relations, and many contents reference preprocessed default knowledge, we use a dialogue-based knowledge acquisition to help teacher fill out the script. The dialogue-based approach has two objectives, one is to reduce cost in experiment construction, and the other is to keep the identical of notations that have dependency in this script.

A dialogue algorithm is proposed in Figure 11. In the first phase, teachers need to specify experiment elements which include scenes, devices, structures and variables. Teachers just name the elements without configuration of details. The phase is similar to design a hands-on experiment. Teachers don't need to describe the details, because most of details belong to the default knowledge which means default values of hands-on experiments. The details setting would be finished through the following dialogue phase. In the second phase, teachers will be asked to set the value of variables that have relation to scenes. The third phase is for the configurations of devices. Since most of devices belong to default knowledge, the attributes and actions are predefined, teachers will be asked to set values of the attributes without defining attributes of them. There are still experimental objects need to define attributes and actions. We use variables defined in the first phase to help teacher add the attributes of experimental object. In the fourth phase, teachers define the structure. We will guide teacher to chose devices for composition. Then for attributes and actions of every composed device, the teacher is asked to determine whether the attributes and actions of the composed devices are inherited by this structure. The teacher could still extend new attributes or actions in the structure. The fifth phase is operation setting phase. We hint teacher with each variable defined in the first phase. The teacher will be asked the operation flow to change the variable. In the last stage, the teacher could define rules to describe how the time could affect the experimental results.

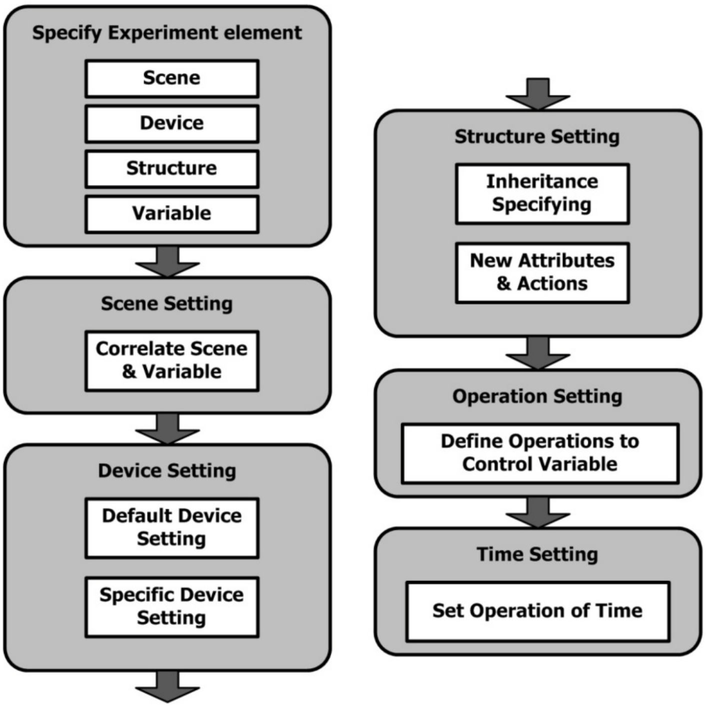


Figure 11. Dialogue Algorithm

Example 12: Using dialogue-based script language to edit the experiment "The related factors in transpiration speed of plants"

<u>First phase: Specify Experiment Element</u>

Teachers name the four parts as they design a hands-on experiment.

Scene: laboratory, playground

Device: graduated cylinder, knife, water vat, ruler, red ink, celery

Structure: celery in the graduated cylinder

Variable: temperature, leaves number of celery, stem length of celery

Second phase: Scene Setting

Consider the variables that have relation to scene. Since temperature has relation to scenes, teachers are asked to decide the temperature of each scene as follows.

Table 20. The script content in second phase of dialogue-based experiment construction

Scene Setting		
Place	**Context Attribute**	**Value**
laboratory	temperature	25
Playground	temperature	30

Third phase: Device Setting

In this experiment, devices like graduated cylinder, knife, water vat, ruler and red ink are belong to default knowledge. Thus we provide predefined attributes of teachers to fill out the value. For the device graduated cylinder, teacher only has to set the values which are underlined in the following table.

Table 21. Part 1 of the script content in the third phase of dialogue-based experiment construction

Device Setting				
Device Name	**Number**	**Attribute / Action Name**	**Attribute / Action Type**	**Attribute Value / Action Effect**
graduated cylinder	1	picture	String	measure.gif
		contained liquid	Record	(none,0,0,0)
		capacity	Number	100
		material	String	glass

Then we consider the celery. Celery belongs to experiment-specific knowledge. Thus teachers have to define its attributes and actions. We use variable to guide teacher in adding the attributes and actions. For variable “leaves number of celery”, we ask teacher the way of how to change leaves number. First the celery need to have an attribute called leaves number, and the celery need some different pictures to show different leaves number. After the action “pluck leaves” is triggered, the picture of celery is replaced by the corresponding picture according to the number of leaves.

Table 22. Part 2 of the script content in second phase of dialogue-based experiment construction

Device Setting				
Device Name	**Number**	**Attribute / Action Name**	**Attribute / Action Type**	**Attribute Value / Action Effect**
celery	1	picture	string	celery1.gif, celery2.gif, celery3.gif
		Number of Leaves	Number	10
		stem length	Number	11
		stem cut	String	None
		In the water	Boolean	false
		Pluck leaves	Procedure	1.Change picture 2.change “leaves number” attribute
		Cut stem	Procedure	1.change picture 2.change “stem cut” attribute
		Inhale red ink	Procedure	1.change picture
		Immerse in water	Procedure	1.change “in the water” attribute

<u>Forth phase: Structure Setting</u>

In the first phase teachers already name a structure called “celery in the graduated cylinder." In this phase first step is to list all the composing devices. Then we will guide teacher via asking them if those composing devices have attributes or actions need to keep in the structure. Those inheritance attributes and actions are automatically filled in third column and fourth column.

Table 23. The script content in forth phase of dialogue-based experiment construction

Structure Setting					
Structure Name	**Components**	**Attribute / Action Name**	**Attribute / Action Source**	**Attribute / Action Type**	**Attribute Value / Action Effect**
celery in the graduated cylinder	Celery, Graduated cylinder	Leaves number	celery	n/a	n/a
		Contained liquid	Graduate cylinder	n/a	n/a
		Inhale red ink	celery	n/a	n/a
		Pour out	Graduate cylinder	n/a	n/a

<u>Fifth phase: Operation Setting</u>

We ask teacher the operations to achieve the change for each variable. For variable "stem length of celery", it can be achieved by two operations. The two actions are immersing celery into water and cut the celery stem. Thus it forms two operation settings.

Table 24. The script content in fifth phase of dialogue-based experiment construction

Operation Setting				
Actor	**Manipulate**	**Target**	**Conditions**	**Reaction**
celery	Drag in	water vat	none	Trigger celery "immerse in water" action
knife	Drag with line	celery	Celery's attribute	Trigger celery "cut

			"stem cut" has value "none"	stem" action

Sixth phase: Time Setting

In this experiment teachers define the red ink decreasing amount in different conditions.

Table 25. The script content in sixth phase of Dialogue-based experiment construction

Time Setting			
Passed time	**Target**	**Condition**	**Reaction**
10	celery in the graduated cylinder	Leaves numbers of celery = 10	trigger action "Inhale red ink" and "Pour out" with degree equals 12
10	celery in the graduated cylinder	Leaves numbers of celery = 5	trigger action "Inhale red ink" and "Pour out" with degree equals 6
10	celery in the graduated cylinder	Leaves numbers of celery = 0	trigger action "Inhale red ink" and "Pour out" with degree equals 2

Chapter 5. Implementation and Experiment

5.1 System Implementation

For evaluation of the CIMAS, we implement a prototype system based on CIMAM. The system is written by Adobe Flash actionscript. It can input an XML configure file that conforms to CIMAM and include pictures as device resources. The system then generates the designed experiment of xml configure file. This XML-based configuration file is coming from the dialogue process using script language. Since there is no authoring tool for visualizing configure like coordinates of devices, we manually transform the script to XML configure file.

We conduct case study and experiment based on the prototype system. In case study, we generate five experiments to evaluate the expressive power of the CIMAM. In the experiment we invite a teacher to design a virtual experiment, using the proposed system. After the process, we use teacher satisfied questionnaire and interview to invest the teacher's feeling about the construction facility and the effect of generated virtual experiment.

5.2 Case Study

We design five experiments, and make them into xml configure files. The prototype system input these files to generate experiments, as shown from Figure 12 to Figure 16. The five experiments belong to three different domains in science education, including biology, chemistry and physics. These experiments show the CIMAM has enough expressive power in supporting experiments of different domains.

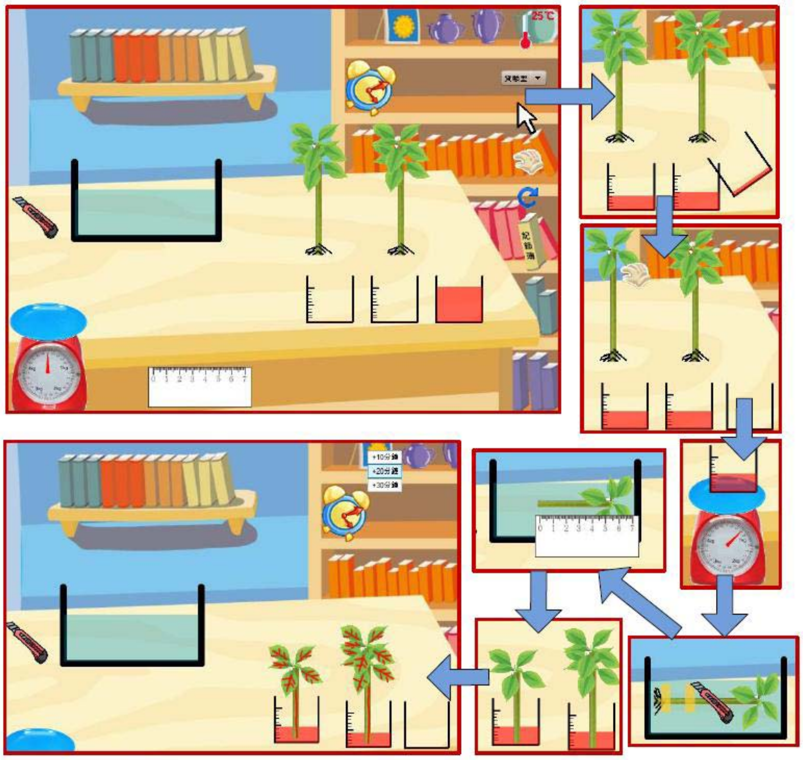


Figure 12. The Related Factors in Transpiration Speed of Plants, Type 1

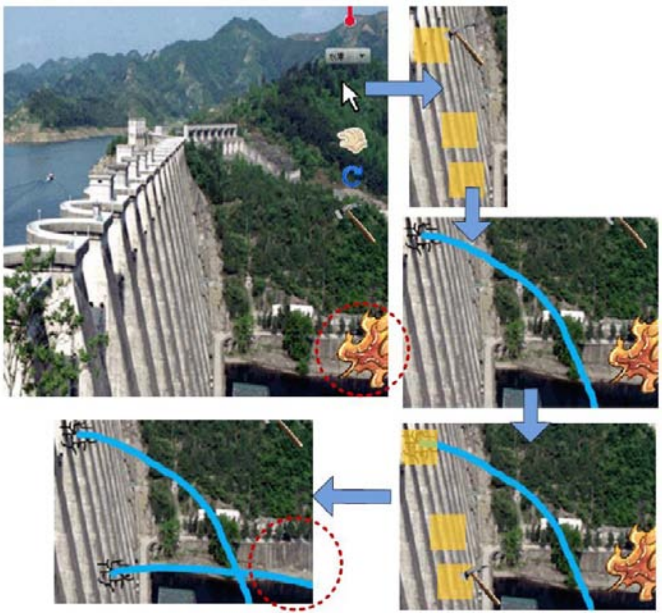


Figure 13. The water pressure experiment

Figure 14. The Related Factors in Transpiration Speed of Plants, Type 2

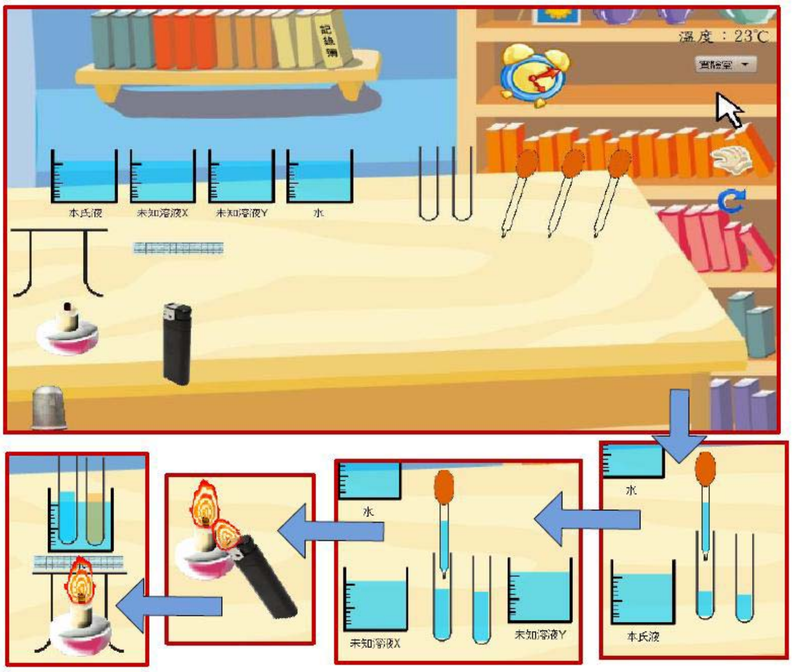


Figure 15. The measure of glucose

Figure 16. The nerve conduction experiment

5.3 Experiment

We invite a teacher who teaches physics in junior high school to design an experiment using the dialogue-based script language. The designed experiment is called “The interaction between marble and hydrochloric acid”. The main purpose of this experiment is to observe if marble’s pallet size or the concentration of hydrochloric acid would influence the speed of interaction. After the teacher uses dialogue-based script language to finish experiment design, we use satisfied questionnaire and interview to discuss with teacher. The satisfied questionnaire result is shown in Table 26, the choice which has bigger number means more agreement about the statement.

Table 26. A teacher’s satisfied questionnaire result of construction facility

Satisfaction about design process	1	2	3	4	5
It is easy to comprehend experiment as scene, device, structure and operation	☐	☐	☐	☑	☐
Follow the order of dialogue-based guiding to design an experiment is easy	☐	☐	☐	☑	☐
This approach could represent the experiment which I	☐	☐	☐	☐	☑

intend to design					
It is easy to design various experiments by this approach	☐	☐	☑	☐	☐
Modify an exist experiment to another experiment is easy	☐	☐	☐	☑	☐
The pre-defined devices could help in reduce construction cost	☐	☐	☐	☐	☑

The result shows teacher is satisfied in general to the design process. However the question "It is easy to design various experiments by this approach" has lower scores. According to the interview, the teacher thinks it is because there are still many details need to depict in experiments, and it is not easy to finish by only applying one iteration. If there is an authoring tool that can show the result immediately after each phase, it can increase the facility of experiment design.

We transformed the experiment design script to XML configure file. The XML configure file were executed by the prototype system. Another teacher satisfied questionnaire is conducted to investigate if the teacher is satisfied about the experiment effect.

Table 27. A teacher's satisfied questionnaire result of experiment effect

Satisfaction about generated experiment	1	2	3	4	5
The display is attractive	☐	☐	☐	☐	☑
The operation is smooth	☐	☐	☐	☑	☐
This experiment reflects my design faithfully	☐	☐	☐	☑	☐
This experiment reflects my design of scenes	☐	☐	☐	☐	☑
This experiment reflects my design of devices	☐	☐	☐	☑	☐
This experiment reflects my design of structures	☐	☐	☐	☑	☐
This experiment reflects my design of operations	☐	☐	☐	☐	☑
Students had operation flexibility in manipulating and controlling variables	☐	☐	☐	☑	☐
The virtual experiment could achieve the goal of assessing scientific inquiry skill	☐	☐	☐	☐	☑

The result shows the teacher is satisfied about the experiment representation and the assessment effect. The teacher also give some comments from the interview: “It is a good way to use virtual laboratory to reduce the cost of hands-on experiments. Virtual laboratory has the advantages of no devices consumption, available for students doing repeatedly trying, keep safety in experiment and saving the time. The ready-made virtual laboratories only provide experiments based on textbooks. If there is a system to help teachers in creating virtual experiments that different from existing experiments in textbook, teachers will have the chance to enhance their teaching.”

Chapter 6. Conclusion

In this thesis, we proposed a CIMAS using knowledge-based approach. Creating new experiments just need to modify the knowledge without modifying the program, thus the separation of knowledge and program reduces the construction cost. The proposed CIMAM can represent the knowledge structure of virtual experiments. The model of default knowledge in the CIMAM makes the system be able to support experiments in different domain. It is also a proposed knowledge acquisition method to preprocess the default knowledge. The preprocessing of default knowledge also reduces the construction cost by the reuse of knowledge. The proposed script language, representing adequate information of a virtual experiment, can facilitate teachers to describe the desired experiment. The dialogue-based knowledge acquisition provides teachers a systematic way to describe an experiment design.

In the experiment, it is found that the dialogue-based script language construction method can guide teachers to finish the experiment design. Since many details of the experiment design still need to be specified, an authoring tool to show the result of each stage may help teachers in construction. It is the future work of this study.

Reference

[1] 陳文典，「我國國中、國小學生科學知能學習成就評量之研究」，國教學報，第四期，81 年 6 月。

[2] 教育部，六大科學教育學習網。http://science.edu.tw/

[3] National Research Council (NRC), *"National Science Education Standards"*, Washington, DC: National Academies Press, 1996.

[4] Daniel P. Shepardson, *"Assessment in Science: A Guide to Professional Development and Classroom Practice"*, Springer, 2001.

[5] American Association for the Advancement of Science (AAAS), *"Science: A process approach"*, 1967.

[6] Barney Dalgarno, Andrea G. Bishop, William Adlong, Danny R. Bedgood Jr., *Effectiveness of a Virtual Laboratory as a preparatory resource for Distance Education chemistry students*, Computers & Education, 2009.

[7] Colin A. Higgins, Brett Bligh , Pavlos Symeonidis, Athanasios Tsintsifas, *"Authoring diagram-based CBA with CourseMarker"*, Computers & Education, 2009

[8] David Yaron, Michael Karabinos, Karen Evans, Jordi Cuadros‡, Jodi Davenport, Gaea Leinhardt†and James G. Greeno, *"The ChemCollective Digital Library"*, in ConfChem, 2008

[9] Diane jass Ketelhut, Brian C. Nelson, Jody Clarke and Chris Dede, *"A multi-user virtual environment for building and assessing higher order inquiry skills in science"*, British journal of Educational Technology, 2010.

[10] Andreas Holzinger, Michael D. Kickmeier-Rust, Sigi Wassertheurer and Michael Hessinger, *"Learning performance with interactive simulations in medical education: Lessons learned from results of learning complex physiological models with the HAEMOdynamic SIMulator"*, Computers & Education, 2009.

[11] Florin D. Salajan, Susanne Perschbacher, Mindy Cash, Reena Talwar, Wafa El-Badrawy and Greg J. Mount, *'Learning with web-based interactive objects: An investigation into student perceptions of effectiveness"*, Computers & Education, 2009.

[12] Eric. N. Wiebe, James Minogue, M.Gail Jones, Jennifer Cowley and Denise Krebs, *"Haptic feedback and students' learning about levers: Unraveling the effect of simulated touch"*, Computers & Eductaion, 2009.

[13] Edys S. Quellmalz and James W. Pellegrino, *"Technology and Testing"*, Science, vol 323, p.75-79, 2009.

[14] Zacharias C. Zacharia and Georgios Olympiou, *"Physical versus virtual manipulative experimentation in physics learning"*, Learning and Instruction,

2010.

[15] Lucília Domingues, Isabel Rocha, Fernando Dourado, Madalena Alves and Eugénio C. Ferreira, "*Virtual laboratories in (bio)chemical engineering education*", Education for Chemical Engineers, 2010.

[16] Chia Jung Lu, "*Portfolio assessment and diagnosis for online Scientific Inquiry Experiments*", Master's thesis, National Chiao Tung University, 2010.