SoftLab: A Virtual Experiment Environment on WWW

Ivory Chung, Maiga Chang, Jia-Sheng Heh
Dept. of Info & Comp. Eng., Chung Yuan Christian Univ., Chung Li, Taiwan
e-mail: s8227120@cycs01.ice.cycu.edu.tw, maiga@ms2.hinet.net,
jsheh@cycs01.ice.cycu.edu.tw

摘要

虛擬實驗是爲了使學習者能在任何時間與地點實地操作，以及學習實驗課程中的知識。本篇論文提出並實作一個虛擬實驗環境，稱爲 SoftLab ，讓使用者於網際網路上合作做實驗。根據物件導向的方法，一個含有特定知識的實驗由知識物件與視覺化物件組成。我們可以找到各個實驗所含知識之間的關係，並且形成一個知識圖，於是一個完整的實驗會構成像 Petri net 的狀態選擇圖。

關鍵字： *Virtual Experiment, Learning Environment, Knowledge Object, Visual Object, Petri-net*

ABSTRACT

A Virtual Experiment tends to let learners practice and learn the embedded knowledge in practicum at any time and at any place. This paper proposes and implements a Virtual Experiment Environment, called SoftLab, on World Wide Web for doing experiments cooperatively. Based on object-oriented approach, Knowledge Objects and Visual Objects compose an experiment, called Experiment Object, which contains some specific knowledge. Relationships in the embedded knowledge of experiments can be found out and then be used to form knowledge graphs, thus a complete experiment is probably constituted with a Petri-net-like Configuration switching graph.

Keywords: *Virtual Experiment, Learning Environment, Knowledge Object, Visual Object, Petri-net*

VIRTUAL EXPERIMENT AND ITS OBJECTS

In order to let learners do physical experiments at any time and at any place, a so-called *Virtual Experiment* is developed to simulate physical experiments on computer network. Furthermore, this kind of Virtual Experiment can achieve some difficult or even impossible experiments in the real world, such as planet motion. (Bret E. Peterson, 1995)(Maiga Chang, *et al,* 1996)

There are two kinds of objects: Visual Objects and Knowledge Objects, involved in the process of Virtual Experiment. Through manipulating *Visual Objects* and observing their changes, learners can practice and then obtain the knowledge of a Virtual Experiment. On the other hand, the physical formulae which control the status of this experiment are embedded in its *Knowledge Objects*. (Yu-Wei Jeng, *et al,* 1996)

To analyze the characteristics of Virtual Experiment, some concepts of Object-Orientation are introduced:

1. Visual Objects and Knowledge Objects encapsulate the outlook of experiment equipment and the knowledge of physics, respectively.
2. These two kinds of objects are both reusable. A ball, for example, might be used in not only "Gravity Experiment," but "Collision Experiment."
3. In real world, the knowledge pieces of nature laws possess some precedence relationship. By ordering

the knowledge pieces of experiments, it is probable to find the inheritance relationship among Knowledge Objects, then to form a *Knowledge Graph*. Accordingly, Visual Objects can have similar relationship.

The above two kinds of objects constitute an *Experiment Object*, which represents a Virtual Experiment. By the reason that the learners in an experiment control the experiment equipment (Knowledge Objects) to obtain its specific embedded knowledge, we will analyze Knowledge Objects first, then design a Virtual Experiment Environment, called SoftLab, and implement it on World-Wide Web (WWW).

FROM KNOWLEDGE OBJECTS TO EXPERIMENT OBJECTS

Those pieces of knowledge corresponding to a Visual Object is a special kind of Knowledge Objects, called a *Module*, which represents the behavior of this object. Given a set of input / output states, a Module can generate its corresponding next states. A ball with some specific properties is a ball and the states are position, velocity, acceleration, and so on. Modules with incremental physical laws form a class hierarchy. For example, a ball with two-dimensional motion inherits from a ball with one-dimensional motion ball; whereas, a frictional ball is also one descendant of frictionless ball.

Practically, there are many *Interactions* within different Modules. Two balls may collide and a free-falling ball may hit the ground. These Interactions can be collected with their Modules to form a *Configuration* of an experiment:

$$F = \begin{bmatrix} \ddots & \cdots & \cdots & \cdots & \cdots \\ \cdots & M_i & \cdots & I_{ij} & \cdots \\ \cdots & \cdots & \ddots & \cdots & \cdots \\ \cdots & I_{ji} & \cdots & M_j & \cdots \\ \cdots & \cdots & \cdots & \cdots & \ddots \end{bmatrix},$$

where M_i is the i^{th} Module and I_{ij} is the Interaction between M_i and M_j. A Configuration holds only when some *Constraints* are satisfied; for example, two balls keep going only when their in-between distance is larger than the sum of their radii. These switching processes from one configuration to another constitute a *complete experiment* (an Experiment Object), which can be illustrated by *Configuration switching graph*, as Figure 1 shown.

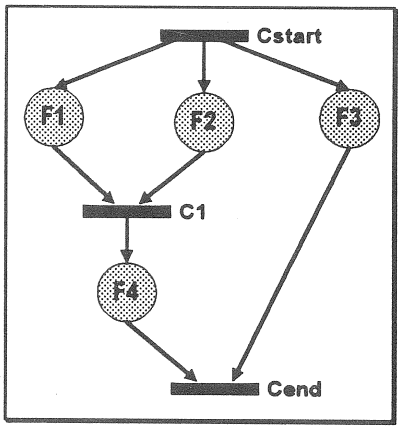


Figure 1 Configuration Switching Graph of an experiment

This graph is one kind of *Petri net* (James L. Peterson, 1981), with places being Configurations and transitions being Conditions. Note that an experiment has a unique starting point (Cstart) and a unique ending point (Cend).

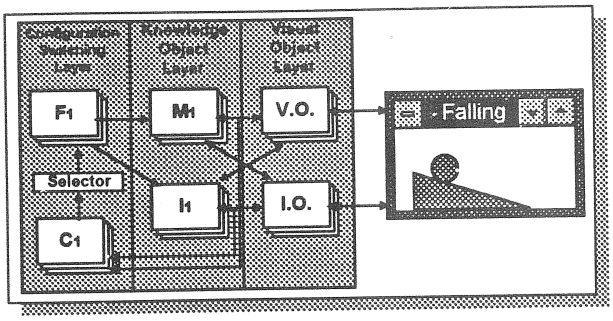


Figure 2 Architecture of SoftLab

SOFTLAB: THE VIRTUAL EXPERIMENT ENVIRONMENT

SoftLab is an *Virtual Experiment* Environment (abbre. as V.E.E.) in which learners can do experiment cooperatively through computer network, so that distance Virtual Experiment can be held as in a real laboratory. According to the analysis above, such an environment is established in three-layered structure: Configuration Switching Layer, Knowledge Object Layer and Visual Object Layer, as Figure 2 shown.

Given the Configuration switching graph of an experiment, an Experiment Object is encapsulated with its Configurations *(Fi)* and Constraints *(Ci)*, in which its designated Configuration Fi is controlled (through Selector in Figure 2) by the corresponding Constraint Ci's. Based on the Configuration Fi, the corresponding Knowledge Objects (Module Mi's and Interaction Iij's in Knowledge Object Layer) will be activated. Each Module exhibits its behavior through Visual Object *(V.O.)*; whereas, each Interaction shows the interaction among Visual Objects through Interaction Object *(I.O.)*.

Obviously, the activation sequence Configuration/Constraint → Module/Interaction → Visual Object / Interaction Object with the a path feedbacking states in Visual Object Layer into Constraint make SoftLab a closed-loop system, which interprets the Configuration switching graph shown in Figure 1. On the other hand, the states of Visual Objects (I.O. and V.O.) are all used to display the experiment results in a visual style, which makes SoftLab a visual learning environment. .

WWW IMPLEMENTATION OF SOFTLAB AND AN EXAMPLE

A prototype of SoftLab is constructed on WWW (World-Wide Web). Different parts of Experiment Objects, such as Visual Objects and Knowledge Objects, can be implemented by different techniques, HTML (HyperText Markup Language), CGI (Common Gateway Interface) or Java, depending on real needs. Here, the demo system chooses Java language as the development tool for the three layers of SoftLab. The visual layout of the whole experiment environment is accomplished through HTML, as shown in Figure 4.

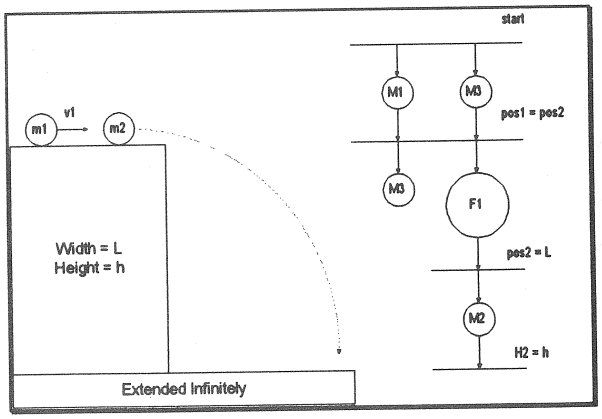


Figure 3 An Example of Virtual Experiment

An example is made in Figure 3, with one inelastic ball B1 colliding with another ball B2 on a higher block, then one ball falling to the ground. The air resistance is neglected. Figure 3 also depicts the corresponding Petri net, where two modules representing these two balls are established in one configuration, then ball collision and free-falling are other two configurations. When these configurations and modules are fed into the established SoftLab, a Virtual Experiment will be held, as Figure 4 shown.

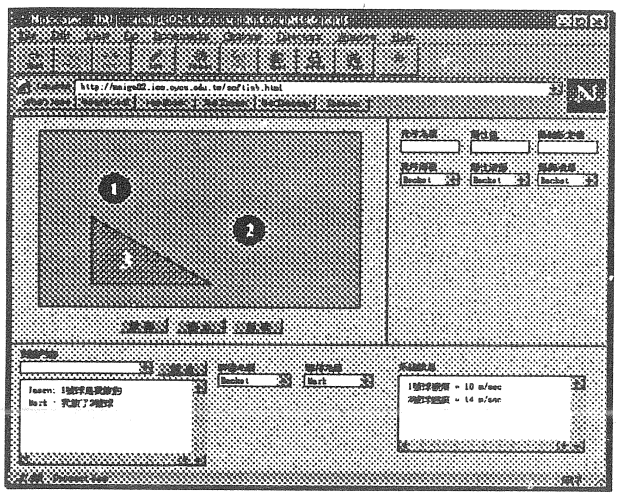


Figure 4 An Outlook of Virtual Experiment Environment

CONCLUSION

The idea of Virtual Experiment design is proposed for learners to do experiment conveniently and distancely. A Virtual Experiment is analyzed through an object-oriented approach, designed as a Petri-net-like Configuration switching graph, and implemented by World-Wide Web technology. On the platform of SoftLab, a Virtual Experiment Environment, an example experiment is established to prove our idea.

By the inheritance property of Knowledge Objects, the knowledge pieces in experiments can be ordered as a knowledge graph. When a set of experiments are designed, the related Experiment Objects can also form a learning graph by the ordering of their embedded Knowledge Objects. Therefore, for those users doing experiments in different orders, SoftLab will be possible to suggest different learning paths.

REFERENCES

Maiga Chang, Ivory Chung, Yu-Wei Jeng, Jia-Sheng Heh (1996), A Cooperative and Visualized Environment for Virtual Experiment, *International Conference on Computer Symposium*, 16-23

Yu-Wei Jeng, Maiga Chang, Ivory Chung, Jia-Sheng Heh (1996), Designing Objects for Virtual Experiments, *Object-Oriented Technologies and Applications*, 331-339

James L. Peterson (1981), *Petri Net Theory and the Modeling of Systems*, Prentice-Hall, Inc., Englewood Cliffs

Bret E. Peterson (1995), Exploration and vitural experimentation in a local neuroscience database, *Journal of Neuroscience Methods*, 159-174