行政院國家科學委員會專題研究計畫成果報告

計畫編號：NSC 87-2213-E-032-019- 多媒體派翠網模型及其應用
執行期限：86 年 8 月 1 日至 87 年 7 月 31 日

主持人：施國琛 淡江大學資工所

Abstract

This paper focuses on two perspectives. First, there are many multimedia presentation systems that do not provide dynamic presentations. These presentations generated by authoring systems will stay as the form they were created. In order to improve the drawback, we allow dynamic presentations in our multimedia presentation system. In order to let our multimedia presentation system also deal with the problem of synchronization, our designing tool uses timed Petri Net.

Key words: Multimedia Presentation, DFD/CFD, CASE, Petri Net, Software Engineering,

1 Introduction

In software development, we apply the concept of software engineering. A multimedia presentation can also be designed by using this approach, especially using structured analysis and design. We can develop our system via the refinement gradually. This methodology is introduced by data flow diagrams and control flow diagrams. The initial design of a system can be refined step by step in multi-level data flow and control flow diagrams. We support a CASE tool to let an engineer to organize the data flow and control flow diagrams. Audiences can design their dynamic multimedia presentation. Incorporating with the proper browser and IMMPS [6], he/she can get the layout and the representation of media that he/she wants.

Although we can construct a dynamic multimedia presentation, there are still some unsolved problems, such as the problem of multimedia synchronization. In order to solve the problem, we use the timed Petri nets. According to this reason, our CASE tool must also deal with the problem of media synchronization by timed Petri nets. Therefore, our system involves the function of constructing the Petri net diagrams. We can't do any refinement to any Petri net because it's the last level of design.

This paper is organized as the following. In section 2,we mention the other well-known systems and discuss the drawbacks of these systems. Section 3 proposes the dynamic multimedia presentation. The proposed multimedia data flow and control flow diagram and graphical components are discussed in section 4. Section 5 summarizes the timed Petri nets model to solve the problem of synchronization and describes those graphical components of timed Petri nets. A short conclusion describes our contributions is presented in section 6.

2 Related Works

Before the design of our system, we refer to many other systems of the same function. Sony Corporation developed a hypermedia prototype system (SAL)[3] for multimedia authoring, which is based on a link and node model used in most authoring systems. The Layered Multimedia Data Model (LMDM)[5] allows the reuse of presentation templates which is important for improving the efficiency of multimedia presentation designs. The work discussed in [1] proposes an architecture and data model for integrated multimedia presentations. The architecture provides a homogeneous strategy to access, process, and exchange multimedia documents generated by

different authoring and presentation systems. Diamond[7] in a multimedia message system built on a distributed architecture for creating, editing, transmitting, and managing multimedia documents. An open hypermedia system is discussed in [2]. The system supports heterogeneous multimedia data types and allows new types to be added. A platform independent multimedia presentation composition system is discussed in [8].

We also looked at the following commercial products about multimedia authoring or presentation designs:

1.Authorware by Macromedia, Inc.
2.Multimedia Viewer by Microsoft
3.Multimedia Toolbook by Asymetric Corporation
4.Hypermedia Authoring and Playback System by ITRI (Taiwan)
5.Action! by Macromedia, Inc.
6.Audio Visual Connection by IBM
7.Astound by Gold Disk Inc.

Authorware uses an event control flow diagram allowing the presenter to specify presentation objects and controls, which can be decomposed into several levels in a hierarchy structure. Other systems (i.e., 2,3,and 6 above) also provide script language and API functions. Hypermedia System and Action! use a time line table allowing actions or objects be dropped in a particular time slot. Most systems allow users to cut and paste presentation objects or actions via button click and drawing. Multimedia Viewer also provides a set of medium editing tools.

However, none of the above systems allows dynamic presentations. Different audience always watches the same presentation. If a presentation can take user interactions and mutate itself, the presentation is more diversified. Besides, not many systems focus on the stepwise refinement of presentation designs. A presentation designer who uses the above systems must have his/her presentation script ready before using the systems. On the other hand, our system helps the designer to design his/her presentation step by step until the final version is created.

3 Dynamic Multimedia Presentation

When a typical multimedia presentation is designed, the layout, and the navigation sequences are all fixed until the presentation is re-designed by the presentation again. We want to improve this approach by allowing dynamic replacement of resources, layouts, and controls of the presentation. Our system provides the function that presentation genarator have the ability of computing the presentation representation at the run-time. Hence, the possible dynamic changes show as follows:

- asserting/retracting of information
- changing resources
- changing layouts
- changing navigation controls

We use the data flow diagram and control flow diagram to change those items dynamically. We describe the multimedia type system in the first subsection. Our system will deal with different presentation windows which handle those four changes. And we will discuss the multimedia resource attributes that help a user to select useable resources in the last subsection.

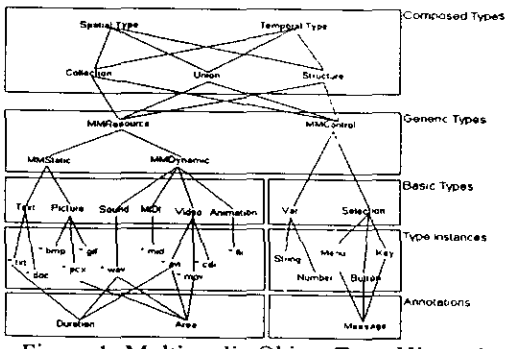

Figure 1: Multimedia Object Type Hierarchy

3.1 Multimedia Type System

Multimedia presentations are collections of multimedia objects. Besides the resources, we need messages as control objects. A multimedia resource has a type which describes what device needs to be

used to present the resource. To ensure the objects in the database are using correct devices, type checking is necessary. Thus we need to know the type hierarchy and type inference rules. Figure 1 illustrates the type hierarchy of multimedia objects.

3.2 The Multimedia Resource Attributes

In order to make a nice multimedia presentation, one has to use a set of high quality multimedia resources. These resources are saved on the disk and reused in different presentations. Of course, these resources also include text file. Each resource has a number of attributes. Possible attributes for multimedia resource are shown as below:

- ◆Name:
- ◆Usage:
- ◆Model:
- ◆Hardware limitation:
- ◆Resolution:
- ◆Resource description:
- ◆Temporal endurance:
- ◆Synchronization tolerance:
- ◆Keyword:
- ◆Medium:
- ◆Detectability:
- ◆Version:
- ◆Start/End time:
- ◆Date/Time:
- ◆Startup delay:

Recently, Data mining has became a hot research topic in the community of database systems. The existence of mutual dependence among the above multimedia attributes infers us that it is possible to analyze these dependence and use Data Mining techniques to improve our system. We are constructing an interactive database subsystem to collect the ways that presentation designers use our database.

4 Graphical Component of Multimedia DFD/CFD

Figure 2 shows all components of our multimedia DFD/CFD. And each graphical component represents its meaning which is different to the traditional DFD/CFD. A multimedia presentation contains a number of presentation windows. Each presentation window contains a layout definition, a collection of multimedia resources, some state variables, and navigation control messages. A presentation window must be refined.

A multimedia DFD/CFD is to help a presentation designer to analyze the script structure of a presentation. Incorporated with an interactive multimedia Petri net diagramming mechanism, the last level of a presentation window is refined to a Petri net. We will discuss this viewpoint in the next section.

5 About Petri Net

We introduce the timed Petri net for multimedia synchronization in our tool, we refine the presentation to the last level, which is the timed Petri net diagram.

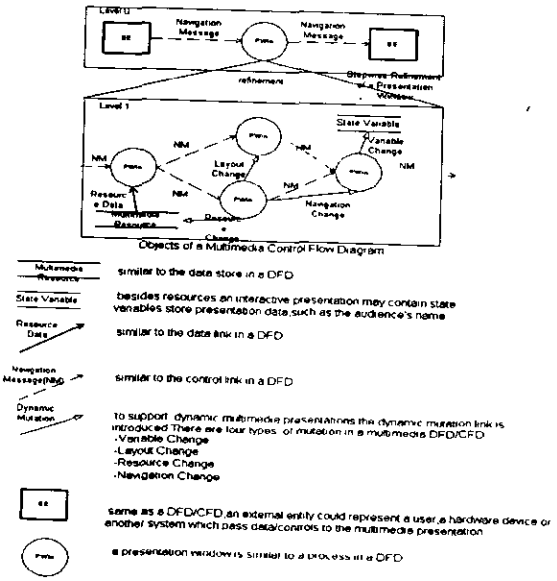

Figure 2:Components of Multimedia DFD/CFD

In figure 3, a timed Petri net is a bipartite graph, with transitions and places connected to each other by arcs. A transition controls synchronization and a place holds a token and a time duration. A transition is fired only after each place adjacent to the transition releases the token. A place holds a multimedia resource to be played for a time duration. Transitions and places are connected by sync arc. We add user transitions and user arcs to the timed Petri net. A user transition receives a navigation message from the user before it is fired. A user transition is directly connected to some transitions. The activation of the user

transition interrupts the demonstration of the presentation window and causes the activation of the connected transitions simultaneously. On the other hand, the timed Petri net diagrams also can do the dynamic mutation. Of course, our system can do this because of the relation of inheritance. Timed Petri net diagrams must inherit the attributes DFD/CFD have. The environment includes all of the functions described in this section.

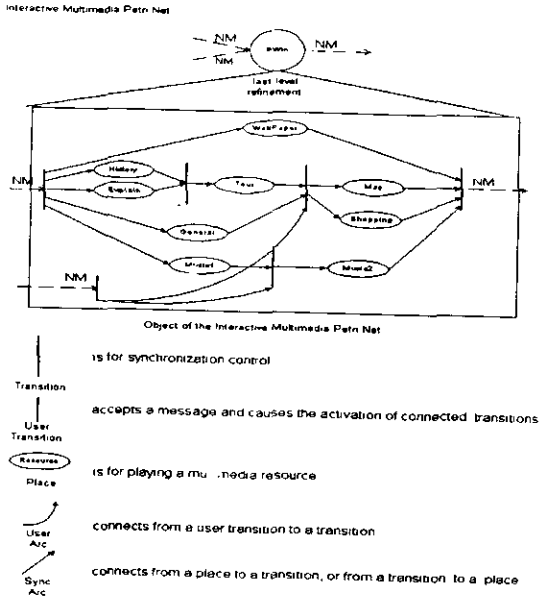

Figure 3: An Interactive Multimedia Petri Net and Components of the Petri Nets

6 Conclusions

We use Visual Basic and Visual C++ to implement the tool. The preliminary experiences show that, the proposed multimedia DFD/CFD is easy to learn because DFDs have been used by many engineers and managers for years. We have some contributions in this paper. Firstly, we revised structured analysis/design methodology and DFDs/CFDs for the use of multimedia presentation designs. Secondly, we introduce the timed petri net to solve the problem of synchronization for dynamic multimedia presentations. Finally, a prototype system was developed on MS Windows 95 or Windows NT to justify our approach. The system can be used for general purposed presentations as well as education software, demonstrations, and others.

References

[1] H.Khalfallah and A. Karmouch. "An architecture and a data model for integrated multimedia documents and presentational applications". Multimedia Systems, Springer- Verlag, 3:238-250, 1995.

[2] T. Kirsta and W. Hubner. "An Open Hypermedia System for Multimedia Applications".In Eurographic Seminars, Tutorials and Perspectives in Computer Graphics, L. Kjelldahl(ED.) Multimedia Systems,Interaction and Applications Chapter 17, chapter 17. 1991.

[3] A. Lundeberg, T. Yamamoto, and T. Usuki. "SAL, A Hypermedia Prototype System". In Eurographic Seminars, Tutorials and Perspectives in Computer Graphics, L. Kjelldahl(ED.) Multimedia Systems,Interaction and Applications Chapter 10, chapter 10. 1991.

[4] R.S. Pressman. "Software Engineering: A Practitioner's Approach". McGraw Hill,1992.

[5] G. A. Schloss and M. J. Wynblatt. "Presentation Layer Primitives for the Layered Multimedia Data Model". In Proceedings of the IEEE1995 International Conference on Multimedia Computing and Systems. May 15-18, Washington DC, Pages 231-238, 1995.

[6] T. K. Shih. and R. E. Davis. "IMMPS: A Multimedia Presentation Design System". IEEE Multimedia, Summer, 1997.

[7] R. H. Thomas and e. a. Harry C. Forsdick. "Diamond : A Multimedia Message System Built on a Distributed Architecture". IEEE Computer, December: 65-78, 1985.

[8] M. Vazirgiannis and C.Mourlas. "An Object-Oriented Model for Interactive Multimedia Presentations". The Computer Journal, 36(1), 1993.

[9]Tadao Murata, "Petri Nets: Properties, Analysis and Applications". Proceedings of the IEEE, Vol. 77, No.4, 1989.