인과성 모델에 기반한 멀티미디어 동적 저작시스템 구현

An Implementation of the Multimedia Dynamic Authoring System based on Causality Model

신 현 산*
Hyun-san Shin

요 약

본 논문은 인과성 모델에 기반한 멀티미디어 동적 저작시스템을 구현하였다. 사용자가 원하는 시나리오를 직관적이고 자연스럽게 프로그램 하도록 2가지 명세를 정의하였다. 인과성을 바탕으로 한 시간 명세는 프리젠테이션에 참여하는 미디어 객체간의 시간관계를 정의한 것이며, 두 번째 명세는 스크린상에 실행되는 미디어들의 상대적인 위치 관계를 토대로 정의한 공간 명세이다. 이 두 가지 사양을 토대로 프리젠테이션 시스템을 구성하였다. 본 시스템의 구성은 멀티미디어 전자문서 생성 과정을 반영한 것이며 그 단계로는 1차원 스트링 리스트 생성 단계, 시간 및 공간 트리 생성 단계, 복합 트리 생성 단계로 구성 처리하였다.

Abstract

In this paper, we implement the multimedia dynamic authoring system based on causality model. we define two specifications which support user to specify intuitively and naturally what he/she wants. The temporal specification describes causal-based temporal relationships between presentation objects, and the spatial specification describes relative layout structure among objects on the screen. Using the specifications, the system processes for multimedia documentation are one-dimensional string list, relational trees, such as temporal, spatial, and annotated composition tree generation phases.

☞ Keyword : causality, authoring system, temporal specification, spatial specification, user interaction

1. Introduction

Many presentations and authoring systems were developed for users or artists in various fields. The common problems of the authoring systems for users are how to author as easily as possible for a presentation scenario. Three major problems face users trying to create multimedia documents [2].

The first problem is authoring easy during document creation and maintenance [1,12-14]. Most systems allow users to cut and paste presentation objects or actions via button click and drawing. Authoring system should be able to satisfy these relationships automatically by positioning the media on time and space domain, inserting delays, or performing alternative actions. In addition, users should be able to change a multimedia document as easily as possible.

The second problem is flexibility of media object synchronization [3,5,11]. The FLIPS [11] model supports the specification of flexible presentations. Barriers and enablers of the FLIPS specify the coarse-grain synchronization among the set of media objects and events. A multimedia document should be able to causally represent the temporal and spatial relationships and include asynchronous information, such as user interaction (skip, stop, restart, fast, etc). So, although users may modify the

* 정 회 원 : 충청대학 인터넷정보학부 부교수
shs01@ok.ac.kr(제 1저자)
☆ 본 연구는 충청대학 2003년도 학술연구비 지원에 의하여 수행되었음.

media duration during presentation, the system should be able to not change the temporal relationships between media objects.

The third problem is to provide unknown and variable duration of media objects [5,10]. Existing temporal models for multimedia can be divided into two classes: point-based and interval-based [13]. In point-based models, the elementary units are points in a time space. The model is well suited for temporal composition of media segments of known durations, however it falls short for unknown durations. Interval-based models consider elementary media entities as time intervals ordered according to some relations. Existing models are mainly based on the relations defined by Allen [9] for expressing the knowledge about time. However, the relations were designed to express exiting relationships between intervals of fixed duration and not for specifying relationships that must be always satisfied even when interval durations and changed [10]. With the incorporation of variable media, there will not always be a single fixed presentation. Users can jump to different points within the presentation, affecting the duration of some objects but not others that are displayed at the same time.

To address these problemswe have implemented a multimedia document system called the Presentation Jockey(PJ). Our aim is to enable users to create flexible multimedia documents more easily. Presentation Jockey can be characterized by the following properties.

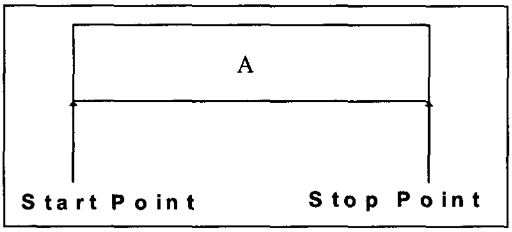
- Our system is able to provide flexible synchronization using the causal relationship among the media objects.
- Our system proposes internal representation such as tree structure, that it uses structured parsing method.
- Our system is able to provide to author easy during multimedia document creation and maintenance using the temporal and spatial specifications.

The organization of the paper is as follows. Section 2 defines the causality and system model, temporal/spatial specifications, user interactions. In section 3, we describe the structure of PJ system and parsing mechanism, i.e. there are one-dimensional string list, consistency check, temporal/spatial tree, annotated composition tree, and schedulers. Section 4 summarizes the results and discusses some of the contributions of the paper.

2. Dynamic Authoring Model

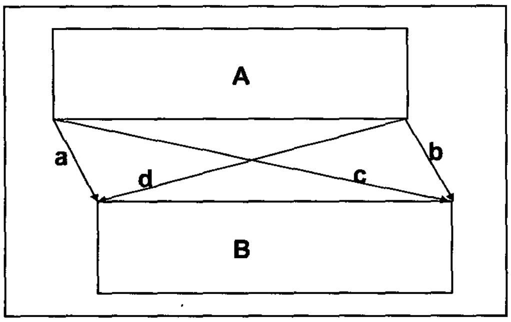
2.1 Causality

Many Multimedia data have causal dependency between objects that represents temporal coincidence. Figure 1 represents two synchronization point of an object. The start point means the presentation start of the object, and the stop point means the presentation stop.



〈Figure 1〉 Two Synchronization Point

Figure 2 describes the causal relations between two objects. For the synchronization point, one object(A) is cause object, and the other object(B) is effect object. "a" means that the starting of an ob-

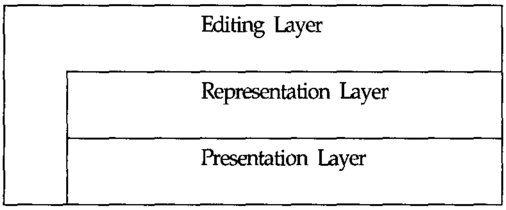


〈Figure 2〉 Causal Relation between objects

ject A causes the starting of an object B. "b" means that the stopping of an object A causes the stopping of an object B. "c" means that the starting of an object A causes the stopping of an object B. "d"means that the stopping of an object A causes the starting of an object B.

2.2 System Design

Figure 3 represents the PJ system model for multimedia documentation. The figure involves three layers i.e. editing, representation, and presentation layer. The role and function of the layers are as follows.



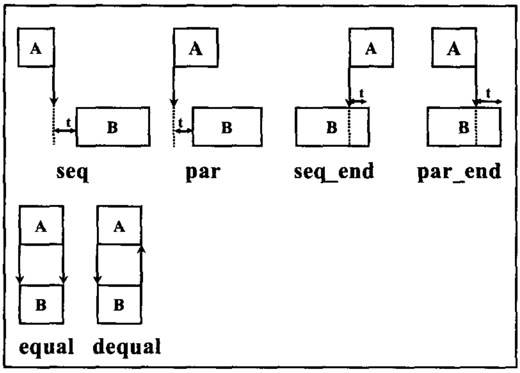
〈Figure 3〉 System Design Model

- Editing Layer : This layer represents the role of user interface, which allow user to specify intuitively and naturally what he/she wants, verify the consistency of scenarios, provide the preview of the authoring.
- Representation Layer : This layer represents the internal mechanism and structure of the internal representation that made by binary tree structure.
- Presentation Layer : This layer represents the scheduler that has scheduling algorithm that is determined the starting time of a media object using the internal data structures.

2.3 Temporal Specifications

Many existing models for temporal relationships of multimedia objects are based on the Allen relation's [9]. However, the relations were designed to express exiting relationships between intervals of fixed duration and not for specifying relationships that must be always satisfied even when interval durations are changed. And the relations are descriptive, so that they depend on known interval duration [10]. We define a set of causal-based temporal relationships between object intervals.

There are four possible relations with a single action involving A and B, including delay time t. In addition we can consider two relations with equality of interval duration. The distinction between the two relations is important, because any of the



〈Figure 4〉 Causal-based Temporal Relations

〈Table 1〉 The Syntax and Semantics of the Temporal Relations

Syntax	Semantics
A seq(delay) B	the end of interval A starts interval B after t delay time
A par(delay) B	the beginning of interval A starts interval B after t delay time
A seq_end(delay) B	the beginning of interval A stops interval B after t delay time
A par_end(delay) B	the end of interval A stops interval B after t delay time
A equal B	the beginning of interval A starts interval B and the end of interval A stops interval B
A dequal B	the beginning of interval A starts interval B and the end of interval B stops interval A

intervals may be of unknown duration.

The causal-based temporal relationships are defined as Figure 4. And the syntax and semantics of the temporal relations is defined as Table 1.

Using the causal-based temporal relationships, we are able to define the processing of event handling

〈Table 2〉 The Processing of Event Handling

Temporal Relation	Synchronization
A seq(delay) B	the stop(cause) event of interval A -> the start(effect) event of interval Bthe end of interval A starts interval B after t delay time
A par(delay) B	the start(cause) event of interval A -> the start(effect) event of interval Bthe beginning of interval A starts interval B after t delay time
A seq_end(delay) B	the start(cause) event of interval A -> the stop(effect) event of interval Bthe beginning of interval A stops interval B after t delay time
A par_end(delay) B	the stop(cause) event of interval A -> the stop(effect) event of interval B the end of interval A stops interval B after t delay time

〈Table 3〉 The Properties between Allen Relation's and Causal-based Relation

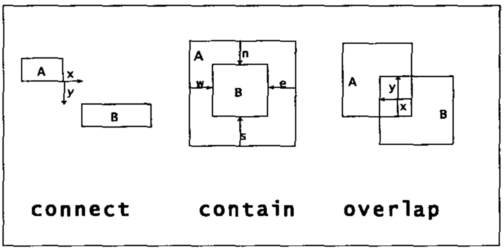
Allen Relation's	Causal-based Relation
before	seq(t)
meet	seq(0)
start	par(0)
overlap	par(t)
during	par_end(t)
finish	par_end(0)
equal	equal, dequal

for our system synchronization. Table 2 represents the processing of synchronization between objects based on causal relation.

Table 3 represents relation properties between Allen relationship and our causal-based relation

2.4 Spatial Specifications

We have described how to capture the relative ordering of a set of multimedia objects in the time domain. The same argument works in the space domain. The spatial relationships among a set of objects are described inthe space domain. Most of the authoring systems allow users to specify spatial information of presentation objects directly on the screen via interactive drag and drop tools. Our system defines three spatial relations among media objects. So users can create a document draft using the spatial relations. After creating, users are able to



〈Figure 5〉 Spatial Relations

〈Table 4〉 The Syntax and Semantics of the Spatial Relations

Syntax	Semantics
A connect(x,y) B	the right bottommost point of object A connects the left top-most point of object B with x, y distant (x,y ≥0).
A contain(e,w,s,n) B	four borders of object A contain border of object B with e,w, s,n size. (e,w,s,n ≥0).
A overlap(x,y) B	the space of object A overlaps space of object B with x,y size. (x,y>0).

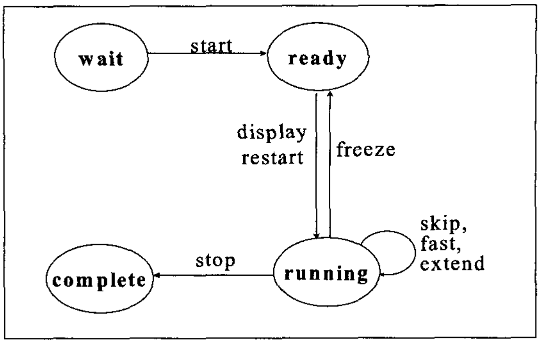
change a part of the layout, instead of re-positioning most of the objects on the screen. Similar to temporal specifications, spatial relations are defined as Figure 5.

The syntax and semantics of the spatial relations is defined as Table 4.

2.5 User Interactions

Generally, the schedulers of a multimedia authoring system manage both the external and the internal event. The types of the internal events are start, stop, and display, and it is more related to temporal-spatial synchronization. For example, the display event causes to access spatial information for layout on the screen and the stop event causes to delete the spatial information. The external event will be occurred by user interaction, such as skip, freeze, extend, fast, and restart.

Our system is able to provide three types of user interactions, freezing/restarting, forward/backward skipping, and extension. The form of user interaction is the use of on-screen button or keyboard selection. For example, if user selects the freeze event (by button), the display of all object stops, and then the system wait for restart event. If user



〈Figure 6〉 State Transition Diagram for Objects

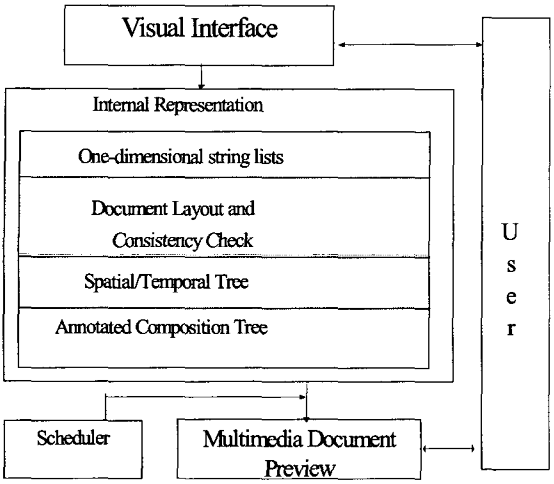
selects the skip (or fast) event, the display of the object skips (repeat skips). If user selects the extend event, the display of the object extends. The presentation window in the PJ system provides several buttons for user interaction. Users can reduce (or extend) the presentation time, using interaction buttons without knowing the details of the scenario format. As the events become enabled, the state of an object changes in Figure 6. And the schedulers of the PJ manage the state transition according to several events.

The states of media object are wait, ready, running, and complete. The states are defined as follows.

- wait : when the object has not yet begun.
- ready : when the object has delayed for running.
- running : when it is being displayed.
- complete : when it has ended.

3. Presentation Jockey

The system provides users friendly graphical user interface to create and modify presentation scenarios. The user interface provides icons that represent media objects, temporal and spatial relationships based on system design model [6,8].



〈Figure 7〉 The Structure of the PJ

Our system consists of three components: visual interface, internal representation, and scheduler. Figure 7 shows the structure of the PJ. Users can create a presentation scenario and define object attributes using parameter windows that they are consisted of not only temporal, but also spatial specifications. A scenario created using icons is translated into one-dimensional string lists by scanning process. After consistency check process, one-dimensional string lists is translated into the temporal/spatial tree. In the translation process, the temporal/spatial tree is translated into the annotated composition tree by adding semantic actions. Finally, the scheduler decides the start time of the objects using the annotated composition tree and also manages the states of the objects on presentation.

3.1 Internal Representation

A scenario created using icons is easily translated into one-dimensional string lists by the scanning algorithm. Input to the algorithm is a scenario on screen and output from the phase is one-dimensional string lists. After the scanning phase, the PJ system processes document layout and consistency algorithm for the string lists that represent the temporal and spatial relationships between objects. The consistency algorithm can detect on some contradictory that have the temporal inconsistency between objects. After the consistency process, the temporal/spatial tree is translated from the nested expression of the temporal/ spatial linked list. Finally, the system produces the annotated composition tree that includes a set of presentation objects, and. semantic attributes for the presentation [4,7].

1) one-dimensional string list

Input to the scanner is a set of media objects as well as temporal/spatial relations that hold between them. The input data can be thought as a graph with primitive objects that has nodes and relations.

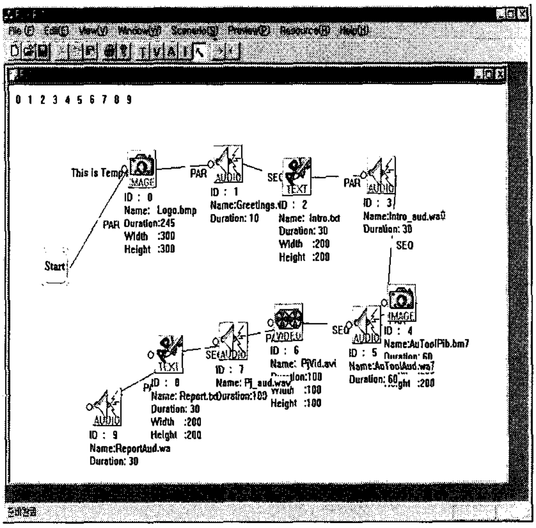
Figure 8 represents the form of the input data to the scanner.

By Depth First Search algorithm, a scanner translates the graph into one-dimensional string list.

• <u>Scanning algorithm</u>

1. START icon is visited.
2. Search minimum media ID that has not visited. If ID is selected, express list with prefix nested expression. If not, translation process is ended.
3. Search a media ID that has temporal/spatial relation of the selected media. If ID is selected, express list with prefix nested expression. If not, go to 2.

In this example, figure 8 indicates that the tree is translated into nested expression of the presentation sequence, such as



〈Figure 8〉 The Example of Temporal Scenarios

START par (Logo seq (Intro par (Intro_aud seq (AUTool par (AuTool_aud seq (PJ_vid par (PJ_aud seq (Report par Report_aud)))))))

2) Document Layout and Consistency Check

Input to the phase is one-dimensional string lists.The phase processes consistency algorithm for the string lists that represent the temporal relations between object icons. The consistency algorithm can detect on some contradictory that have the temporal inconsistency between media objects. The consistency check algorithm is mainly searched for a derived temporal relationship among media objects. Naturally, the causal temporal relations have derivation temporal relational triangulation table. The consistency check algorithm is defined as follows.

• Consistency check algorithm

1. Read the one-dimensional string list.
2. If the visited icon is a media icon, search the derived relation in the relational triangulation tablebetween the media object and dependent media that has temporal relation of the media,
3. If agree, go to 1.
4. If not, output consistency error message.

〈Table 5〉 The causal derived relational triangulation table

1st \ 2nd	S	P	SE	PE
S	S	S	S,PE	S,PE
P	P	P	P,SE	P,SE
SE	P	P,SE	SE	P,SE
PE	S	S,PE	PE	S,PE

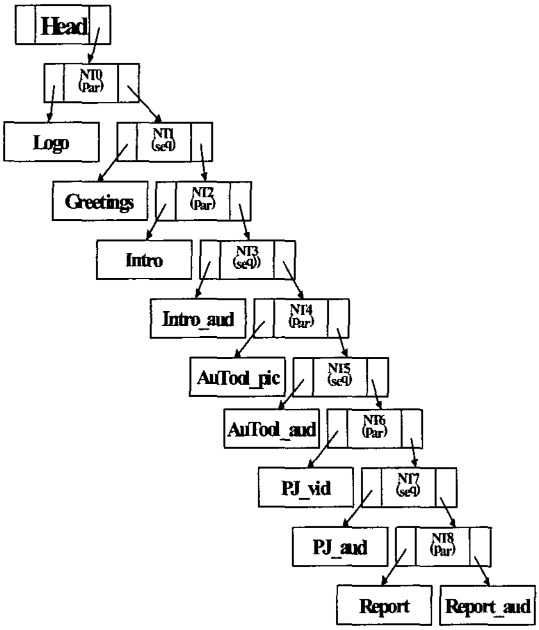
(S:seq, P:par, SE:seq_end, PE:par_end)

3) Temporal/Spatial Tree

The parser's goal is to build a temporal/spatial tree that covers the input such as one-dimensional string list. The temporal tree consists of terminal node, non-terminal node that includes temporal relations. A node in the tree is the leaf or terminal node. Each node has attributes that indicate node type, such as terminal and non-terminal, media type, text, image, video, audio, and anchor, the translation information for semantic translation, and a pointer that indicates the location of the data for presentation. The other node of tree is the non-terminal that has a slightly different structure. Attributes for this node include node type, the relational information for time and layout structure, left and right child pointers. The temporal tree generation algorithm is defined as follows.

• Temporal Tree Generation algorithm

1. Read the refined one-dimensional string list.
2. If the visited icon is a media, assign the media icon to left terminal node and assign the



〈Figure 9〉 The Example of Temporal Tree

other media dependent on the media to right child node.

3. If the visitedicon is a temporal/spatial relational icon, assign the icon to nonterminal parent node
4. If all of media icon is visited, this process is end.

In this example, figure 9 represents the form of the temporal graph that has translated.

4) Annotated Composition Tree

For spatial-temporal synchronization, our system translates the temporal tree into the annotated composition tree (ACT). ACT is executable graph that include spatial, temporal, and media characteristics. Based on temporal tree, ACT composes of root node, terminal node, non-terminal node that includes information for temporal-spatial synchronization, such as par, equal, dequal. ACT generation algorithm is defined as follows.

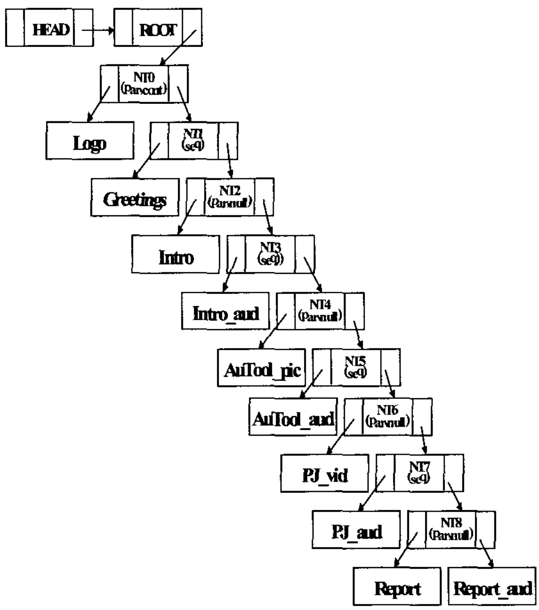
- <u>Annotated Composition Tree Generation algorithm</u>

1. Based on temporal binary tree
   A. Define the root node, which has document name and information.
   B. On the traverse the temporal tree, add spatial relational information to (par, equal, and dequal) temporal nonterminal node.
   C. If all of node is visited, this process is end.

In this example, figure 10 represents the form of the ACT that has translated.

3.2 Schedulers

The scheduler decides the start time of the media objects using the annotated composition tree. The start time of the media is determined by media du-

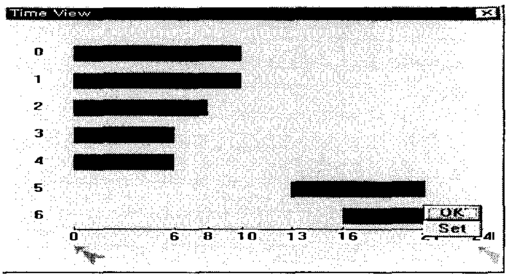


〈Figure 10〉 The Example of ACT

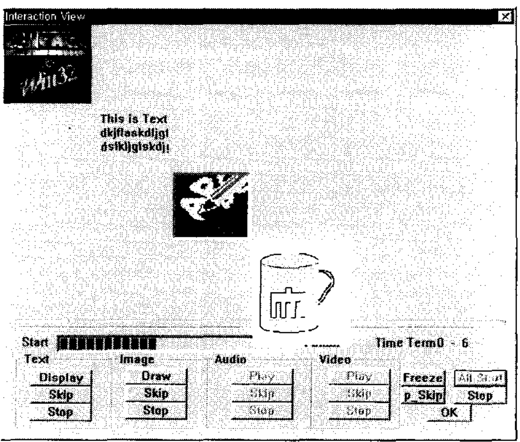
ration value and it's delay time. Also the scheduler manages the state of the objects using state management table on presentation. That is, the values of the table are present states of the media objects. The states of the media object are wait, ready, running, and complete.

3.3 Multimedia Document Previews

Our system can provide a multimedia documents that have two types of program result, temporal preview, and presentation preview. Figure 11 shows temporal preview window that represents the duration of media objects (06) on presentation. Figure 12 shows presentation preview window that includes user interaction buttons that is reduced (or extended) the duration of a media object in presentation.



〈Figure 11〉 The Temporal Preview Window



〈Figure 12〉 The Presentation Preview Window

3.4 Implementation

For this point, we have implemented the visual interface and a part of parsing process. The implementation of the Presentation Jockey is under Microsoft Windows using Visual C++, MFC (Microsoft Foundation Class) Library, and MCI (Media Control Interface) in win32.api. The toolbar, icons, and pull-down menus provide an easy access to create media objects and relationships between media objects, and to define the relationship and content information for the media objects. In the parameter window, users can define and modify the properties for a selected object using the browser that include defining the exact location for the media file. The control window can be used for user interaction using buttons on presentation run time.

4. Conclusions and Applications

Our aim is to enable users to create flexible multimedia documents more easily. Our system can be characterized by the following properties.

It provides for user to author easily multimedia document using the temporal and spatial specif-ica-tions.

It defines a causal relation that can manage changes of media duration.

It provides an internal representation using tree structure and thus supports effective document parsing.

It supports a presentation engine for effective presentations and user interactions in runtime.

Many existing models for temporal relationships of multimedia objects are based on the Allen relation's [9]. However, the relations are descriptive and depend on known duration, so they couldn't manage changes of media duration. We define a set of causal-based temporal relationships between object intervals. There are six possible relations with a single action involving A and B, including delay time t. In our model, duration can be changed according to reference point and thus a presentation is flexibly synchronized. To support this model, we developed an authoring system. The system consists of components such as visual interface, internal representation, and schedulers. The system was implemented on Microsoft Windows using Visual C++.

Our system can be available in various fields. We expect that the authoring system is able to contribute to remote education of cyber campus, the industry of game, education, kiosk, and presentation.

References

[1] S.K. Chang, M. J. Tauber, Bing Yu, "A Visual Language Compiler", IEEE Trans. On Software Engineering, Vol.15, No.5, May 1989.

[2] J. F. Koegel, J. M. Heins, "Improving Visual Programming Languages for Multimedia Authoring", EDMEDIA, 1993.

[3] P.D. Stotts, R.Furuta, "Petri-Net-Based Hypertext: Document Structure with Browsing Semantics", ACM Trans. On Information Systems, Vol.7, No.1, Jan. 1989.

[4] L.Weitzman, K. Wittenburg, "Automatic Presentation of Multimedia Documents using Relational Grammars", Multimedia, 1994

[5] Thomas D.C. Little, A. Ghafoor, "Spatio-Temporal Composition of Distributed Multimedia Objects for Value-Added Networks", IEEE Computer, 1991.

[6] R. Steinmetz, K. Nahrstedt, Multimedia: Computing, Communications, and Applications, Prentice Hall P T R.

[7] Aho, A.V., R. Sethi, and J.D. Ullman, Compilers: principles, Techniques, and Tools, Addition-Wesley, Reading, MA., 1986.

[8] Hyun-san Shin, Yoo-jae Won, etc, "Presentation Jockey: Multimedia Authoring System for Automatic Presentation", IASTED Applied Informatics'97, pp. 259-262, Feb. 18-20, 1997.

[9] J. F. Allen, "Maintaining knowledge about temporal intervals", CACM, Vol.26, No. 11, Nov 1983.

[10] Andrzej Duda and Cherif Keramane, "Structured Temporal Composition of Multimedia Data," International Workshop on Multimedia Database, pp. 136-142, Aug 1995.

[11] J. Schnepf, J. Konstan, and D. Du, "Doing FLIPS: Flexible Interactive Presentation Synchronization," IEEE Journal on Selected Areas in Commpp 114-125, Jan 1996.

[12] G. Blakowski, J. Hubel, U. Langrehr, and M. Muhlhauser, "Tool Support for the Synchronization and Presentation of Distributed Multimedia," Computer Comms, Vol. 15, No. 10, pp. 611-618, Dec 1992.

[13] Timothy K. Shih, Chin-Hwa Kuo and Hwei-Jen Lin, "Visual Multimedia Presentation Designs" DMS'96, pp. 306-315, 1996.

[14] M. Cecelia Buchanan and Polle T. Zellweger, "Scheduling Multimedia Documents Using Temporal Constrains," Workshop on Network and Operating Systems supported for Digital Video and Audio, pp. 237-249, Nov 1992.

◐ 저 자 소 개 ◑

신 현 산
1984년 충남대학교 계산통계학과 졸업(학사)
1986년 충남대학교 대학원 전산학과 졸업(석사)
1998년 충남대학교 대학원 전산학과 졸업(박사)
1988~1995 한국전자통신연구원 선임연구원
1995~현재 충청대학 인터넷정보학부 교수
관심분야 : 멀티미디어저작시스템, 멀티미디어 응용, 의료정보처리.
E-mail : shs01@ok..ac.kr