December 1993

An Object-Oriented Development Process for Multimedia Videotex Application

Chien-Chih Yu
National Chengchi University

An Object-Oriented Development Process for Multimedia Videotex Applications

Chien-Chih Yu, Department of Management Information Systems,
National Chengchi University, Taipei, Taiwan, R.O.C.

ABSTRACT

Current videotex systems provide text and picture based public information services by means of sequences of pre-edited frames. In order to integrate diverse types of multimedia information such as text, graphics, images, animation, audio and motion-video and to provide a friendly and flexible user interface for videotex applications, the existing database structure, system functions, as well as operating environment of the videotex system should be reformed and enhanced. In this paper, an object-oriented development process is proposed with a three layered system architecture to facilitate the construction of a multimedia videotex information model and physical system environment. A prototyped multimedia information system for Chinese videotex services is also presented for validating the proposed system architecture and development process.

This research is supported in part by the Ministry of Communications, Executive Yuan, R.O.C. under Project: MOC-STAO-RD2601.

INTRODUCTION

Developed in Europe in the late seventies, videotex is in general a two-way computer communication system in which a large population of users can remotely access categorized information from a large central database resident in the videotex center as well as from multiple external databases connected to the network. Users can view selective information or receive special services through making choices from pages of linked menus that are displayed on their own output devices such as modified TV sets, special visual display units or high resolution computer monitors.

In the past decade, many public and private videotex systems have been established and promoted in governmental and enterprise sectors around the world. Table 1 lists names of some commercialized public videotex systems in different countries [2,6,13]. Some international corporations adopting videotex systems as company-wide communication and document distribution channels include Digital and McDonald.

Table 1 Countries and Videotex Systems

Great Britain	Prestel
France	Teletel, Antiope
Germany	Bildschirmtext
Netherlands	Viditel
Japan	CAPTAIN
Canada	Telidon
U.S.A.	Viewtron, CompuServe, The Source
R.O.China	CVS

In recent years with the advancement in the broadband communications and multimedia database technologies, the demands for distributed multimedia applications such as telemarketing systems, geographic information systems, CAD/CAM systems, corporate publishing systems, medical video response systems and a variety of military application systems have increased rapidly. These systems have been expected to be capable of allowing users to retrieve, manipulate, exchange and present information in the form of electronic multimedia documents in which data in diverse types such as text, graphics, still-image, animation, audio and full-motion video can be integrated and organized dynamically in specified formats [1,4,5,11,16,19]. Applying the emerging multimedia related information technologies to videotex applications should enhance the contents and functions of the system. For instance, the simultaneous combination of text, high resolution graphics, images as well as audio and motion video data in the videotex system makes the information services process more attractive, helpful, and easier to be used and understood. Background narration with foreground visualized pictorial displays on video screens will be more effective than the traditional plain-text videotex frames. The use of corporate videotex to create and distribute electronic information to internal employees and to support video conferencing for a group of separated audiences can produce significant savings in terms of both cost and time and thus greatly improve the efficiencies and effectiveness of organizational communications. Links to customers and suppliers with a multimedia videotex system provide caltaloging and ordering services which can also help a company achieve competitive advantages in today's dynamic marketplace. As a result, multimedia videotex applications can be considered as one of the most promising forms of communication and information services and have been included as major features of the broadband integrated digital services networks (BISDN).

Current and potential multimedia videotex services can be classified into the following categories [2,9,13,23] :

1. information retrieval, including daily news, weather reports, want ads, library caltalogs, transportation schedules, telephone directories, financial and stock market information, tour guides and consumer based information,...etc.;

2. commercial transaction processing, such as electronic banking, teleshopping, airplane reservation,...etc.;

3. electronic messaging, such as electronic mail, bulletin board, electronic material delivery, teleconferencing, closed user group services,...etc.;

4. telesoftware, including home computing, computer-aided educational softwares and video games;

5. intelligent on-line supporting services, such as interactive problem-solving assistant, telemonitoring, and gateway services.

Many technical considerations and problems have to be taken into account and solved in the process of designing and building sophisticate multimedia services for a videotex system [19,23]. These problems and design issues include:

1. interconnection of heterogeneous networks,

2. digitization of non-digital data,
3. establishment of high speed networks,
4. standardization of communication and transport protocols,
5. installation of powerful multimedia servers and workstations,
6. arrangement of high capacity storage units,
7. network-wide integration of multimedia input/output devices ,
8. efficient compression and decompression of multimedia data,
9. organization and presentation of multimedia documents,
10. file conversion and exchanging among multiple multimedia software tools,
11. synchronization of diverse multimedia information objects,
12. comprehensive conceptual architecture for developing multimedia applications,
13. formal definition, powerful manipulation and consistent control over multimedia database,
14. flexible data modeling for multimedia applications,
15. effective multimedia data fragmentation and file allocation,
16. query optimization for accessing multimedia information,
17. transaction management in distributed multimedia environment,
18. provision of friendly and intelligent user interfaces,

......and so on.

In the current literature, most videotex related researches concentrated on solving physical and environmental problems [6,8,10,17,24] or describing and discussing commercial potentialities [9,13]. Investigation of conceptual and implementation design aspects including data model, system architecture, and development processes for constructing multimedia videotex systems are very rare, if not completely absent. Typical videotex systems organize and present information contents as a hierarchical structure or frames. Only restricted types of information, mainly text and simple graphics, can be created and put into pages of frames. Users browse through desired information by traversing a series of menus or by entering keywords or page numbers to retrieve and display the information pages. This tree structure of frames for holding information is rather static and less flexible to handle a wide variety of user needs and is slow in response to the rapidly changing data. Besides, data redundancy is high and reusability is low. The flat style of character-based menu and information presentations may cause users to get lost in the middle of navigating page sequences. And above all, the current videotex information structure as well as presentation formats are not suitable for modeling complex multimedia applications.

The scope of this paper focuses on dealing with the critical conceptual and implementation design problems. Major purposes are to propose a system architecture in accordance with an object-oriented development process for effectively modeling and constructing multimedia videotex applications and to integrate into the application level of the system framework the hypermedia technologies for providing a friendlier and intelligent graphical user interface in which diverse types of icons, menus, and windows are available and can be combined. In the following section, a brief review of object-oriented modeling concepts and design methods as well as hypermedia technologies is provided. A three-layered system architecture consisting of application, conceptual and physical layers as well as a corresponding object-oriented system development process for multimedia videotex applications are presented in section 3. In section 4, a prototyped multimedia chinese videotex system is illustrated with discussions on physical mappings. A conclusion and suggestions of future extensions are provided in the final section.

OBJECT-ORIENTED MODELING CONCEPTS AND HYPERMEDIA TECHNOLOGIES

Many different data models have been adopted for modeling multimedia database and hypertext or hypermedia systems. Examples include extended relational data model [18], node-link model [1], object composition petri net [4], hypergraph data model [20], and the object-oriented data model [3,7,12,14,15,21,22]. Among them, the object-oriented approach which is well suited to handle data of diverse types, from different sources and in various forms has been expected to become the major modeling technique for the future database applications. Employing the object-oriented concepts, every type of information unit, such as text, image, video, animation and sound, as well as a multimedia document. (which is an external representation of a combination of data in multiple types), can all be treated as an object associated with a set of properties and behaviors.

Detail discussions of the object-oriented concepts and methods are beyond the scope of this paper and can be found elsewhere. In this section, only a brief description of key elements is provided.

(1) Object and object identifier
Object is defined as an abstract representation of a real world material or entity, e.g. a person, a map or a document. An unique object identifier is associated with each object for precisely specifying the object.

(2) Attributes and methods
Every object can be characterized by state and behavior. A set of attributes, also termed as properties or instance variables, is contained in an object with their values carrying the state of the object. Object behavior is reflected by a set of methods or procedures operates on the state of the object. Both attributes and methods are encapsulated in an object and are invisible to users. Such an encapsulation catches the advantages of information hiding, data independence, and modularity in system design.

(3) Messages and message passing
Object attributes and operations can be accessed or invoked from external requests called messages. Messages indicate related information for executing proper actions. These information include specifications of the target object (receiver), the chosen method (selector), and arguments for activating the method. The mechanism of message passing among objects allows program procedures to be bound at runtime and hence achieves the advantage of dynamic binding.

(4) Class
Objects that share the same attributes and methods are collected to form a class. Every object is an instance of exactly one class and this constitutes an instance-of relationship. Classes are structured in a hierarchical way by defining relationships among them.

There are three types of inter-object relationships in the object-oriented data model.

(1) Generalization
Based on an existing class, a new class, called a subclass of the original class, can be defined to have special properties and operations by adding new attributes and methods to it while inherits all the attributes and methods from its ancestor class which is called the superclass. Generalization and specialization refer to the relationships between the superclass and its subclasses in opposite directions. The generic object of the superclass includes the common properties and attributes that are inherited by the category objects of the subclasses. When a class can inherit from only one superclass, it is called single inheritance. In the case of having multiple superclasses, we have multiple inheritance. The inheritance hierarchy or lattice establishes a number of connected is-a relationships which may minimize data redundance, simplify maintainance, and promote incremental design.

(2) Aggregation
When a combination of related objects forms a portion of the attributes group for a high-level object, or conversly, an object has some attributes being objects with their own set of attributes, these aggregate relationships between the higher-level object and the collection of its component objects represent class-attribute links and give rise to a nested structure of the class definition. These so called part-of or have-a relationships establish a class-composition hierarchy that is orthogonal to the inheritance hierarchy described above.

(3) Association
When a class refers to properties and/or activates methods of another class independent to it, this establishes an association or reference relationship. Referencing is done by explicitly passing messages between objects to access desired attributes or to invoke proper methods.

Using the object-oriented paradigm, a number of analysis and design methods have been proposed in the literature for developing database applications. Different approaches have different concerns and orientation in what has to be done and how to do it. Some focused on identifying object contents, others on behaviors or functions or system life cycles. Yet there are no trivial connections between these methods and the development of multimedia applications. Analyzing content and format of multiple multimedia documents using an object-oriented approach should result in the specification of information, functional, and process requirements and then lead to the identification of objects and relationships. The following object-oriented conceptual design process should carry out the classification of objects and perform class integration to form a logical class structure representing the abstraction of those multimedia documents. Extensibility, flexibility, modularity, reusability, information hiding and dynamic binding are some of the technical advantages to be achieved. In the next section, a system architecture and associated development process will be presented to formalize the analysis, design and implementation procedures for developing multimedia videotex applications.

Hypermedia technology, originated from hypertext, is a concept and methodology for constructing and representing non-linearly linked pieces of multimedia data and for allowing users to navigate the information network in a self-controlled procedure. This technology has been applied to many different areas including electronic books, computer-aided instruction, and office presentations. Characteristics of a hypermedia based system can be classified by the following features:

(1) iconic as well as mouse/menu driven user interface,
using menus, icons and mouse clicking to drive the information processing procedure;
(2) multiple-windowed screen design,
relating information of the same branch by presenting these information in multiple windows of different sizes;
(3) multimedia presentation,
representing information in the richness of multimedia contents and formats;
(4) node-link data model for view integration,
integrating multimedia information objects and views by defining nodes and links to form a nonlinear information structure;
(5) non-sequential information browsing,
accessing and linking information through a non-sequential procedure to generate diversified viewing paths;
(6) user controlled information processing,
allowing users to navigate through information structure and to tailor information in a self-controlled pattern.

Hypermedia technologies will be applied to the user interface design of the multimedia videotex applications.

SYSTEM ARCHITECTURE AND DEVELOPMENT PROCESS

In order to efficiently and effectively develop a multimedia videotex application system, a system architecture is needed for standardizing the analysis and design specifications and formalizing the development process. A three layered system architecture including application, conceptual, and physical layers is shown in Figure 1. An object-oriented development process in correspondance with the system architecture is proposed to facilitate the analysis, design and implementation procedures and to achieve the benefits of extensibility, modularity, and flexibility in system evolution, as well as availability, integratability, sharability, diversity and integrity in information utilization. In the following, contents and tasks involved in the three layers are described and followed by a complete integrated object-oriented system development process.

(1) Application layer

The application layer stands for the multimedia information browsing and representation level of the system. It focuses mainly on using the system easily and productively. This layer is composed of two principal parts: multimedia external user views and iconic user interfaces. A multimedia user view is the simultaneous representation of a combination of heterogeneous information objects that are organized in a specific format and sequence. It provides information with the desired contents and layouts to meet the users' requirements. Collecting and identifying user views as well as defining attributes and operations associated with these views are major tasks to be performed in the analysis stage of the application system development process. A friendly hypermedia-based graphic user interface equipped with functional capabilities such as menus, icons, and windows provides the means to easily access and manipulate these user views. User views and interfaces are often integrated to constitute the dynamic and static portions of the screens. Menus can be presented in

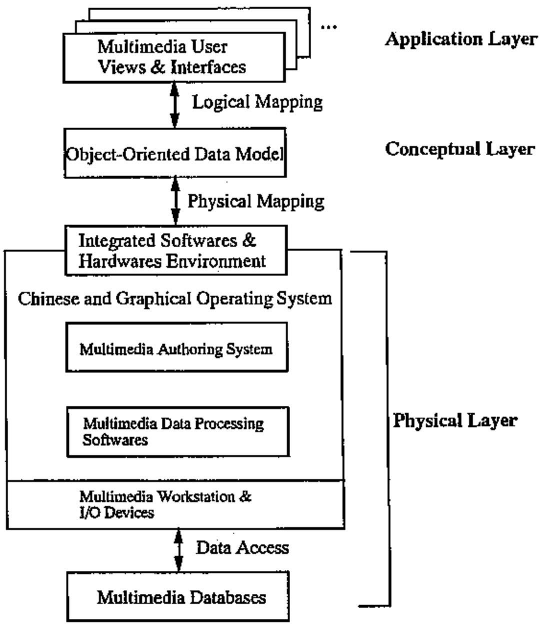

Figure 1. Three layered system architecture

different forms including line bar, pull-down, pop-up and full-screen menus. Windows can be specified to have different sizes and scrollabilities. An operable icon has a macro program attached to it for executing a string of precoded commands. User views, menus, icons, windows, and other system components or information units such as relational data files, graphics, images, sequences of sound, and videos can all be treated and identified as objects. In general, the application layer defines contents of the multimedia information and the ways they are presented and accessed to clearly specify the information, functional, and operational requirements of the application system .

(2) Conceptual layer

The conceptual layer focuses on utilizing flexible data modeling techniques for classifying and organizing multimedia information objects as a logical structure to support all the requirements specified in the previous layer. In this layer, an object-oriented data model is used to represent the entire multimedia user views and interfaces in an abstract level. From the results obtained in the application layer, user views, interface components, and their combinations are formally defined as objects with attributes and methods. Objects having the same attributes and methods are grouped into a class. Relationships among classes are then identified and the whole integrated class hierarchy is established. This results in the first-staged accomplishment of an object-oriented conceptual design process. Paths to generate every user view from the entire conceptual model are pointed out with their performances being estimated. The class structure is then reviewed and refined if necessary to gain better performance. To indicate mappings from the object-oriented conceptual model to every multimedia user view constitutes the logical mapping process. Also constructed in this layer are the functional structures and the operational proess. The functional structure presents the structural arrangement of system functions such as query, browse, modify, update, insert, delete, and print,..etc.. The operational process illustrates the framework relating operational sequences, types of user interface used, and the connected information objects

(3) Physical layer

The physical layer focuses on successfully implementing the conceptual model to an integrated software, hardware and communication environment. Physical multimedia databases are created and the operable application system is constructed. For the Chinese videotex applications, the software environment contains various kinds of compatible software: a Chinese operating system, a multimedia authoring system, a group of multimedia data processing softwares, and device drivers. The hardware environment contains multimedia servers or workstations, a group of multimedia data input, output and storage devices and interface cards. The physical multimedia databases consist of relational databases and databases for different types of data, such as images, graphics, voice, animation, and motion video. High speed networks and transport protocols are included in the communication environment. To implement the object-oriented conceptual model to the physical multimedia databases and system environment for successfully generating the desired user views is the objective of the physical mapping process.

Following the proposed architecture, steps to perform the analysis and design process for multimedia videotex applications are outlined as follows:

1. collecting and defining multimedia external user views and interfaces;
2. identifying objects;
3. defining attributes and operations for each object;
4. building object directory;
5. identifying relationships;
6. defining integrated class structure;
7. defining logical access mapping;
8. designing functional structure;
9. designing operational process;
10. designing multimedia representation ;
11. designing physical mapping procedures;
12. implementing, testing and tuning system.

Steps 1.- 4. correspond to tasks ought to be done in the application layter. Steps 5.- 9. belong to the conceptual layer while the other steps belong to the physical layer. Using the proposed system architecture and development process, the advantages we mentioned above can be achieved. For instance, the logical access mapping specifies the control over massage passings to access or generate desired user views and the physical access mapping guides the implementation of the system models to various software and hardware platforms. This helps achieving logical as well as physical data independences and hence reduces efforts on system development and maintenance.

A PROTOTYPED MULTIMEDIA VIDEOTEX SYSTEM

In this section, we present a prototyped multimedia tour guide system for Chinese videotex services to illustrate the process of developing multimedia videotex applications using the proposed system architecture and object-oriented methodology. Descriptions will follow the proposed layered architecture of the system.

(1) Application layer

After reviewing the existing Chinese videotex system and interviewing people from the tourism bureau and travel agencies, a number of related user views and operational requirements are identified for the prototyped system. Figure 2, Figure 3 and Figure 4 display some typical multimedia user views and graphical user interfaces of the prototyped videotex services using hypermedia technologies. A line-bar main menu showing categories of traveling related information interests always appears on top of the screen. Each main menu option is associated with a pull-down menu listing detailed selective items in each category. The combination of line-bar and pull=down menus allow users selecting and switching from branches of information interests.

Figure 2 illustrates a branch of views for amusement and recreational activities. Part (a) is a screen menu for different outdoor recreations including gulfing, swimming, fishing, and springs bathing. Part (b) is a Taiwan map with names, locations and types of 23 attractive hot/cold springs areas. A couple of pop-up menus listing certain health conditions and rules for picking proper springs areas are placed in the lower right corner of the screen. The resulting chosen springs areas are revealed by changing the colors of their name boxes on the map. Part (c) contains commonly requested information about a particular springs area such as name, location, brief introduction, and facilities as well as further information on dinning, driving, and transportations. Part (d) shows a related regional guiding road map for self-driving to a target springs area.

Figure 3 shows a city tour branch for Taipei downtown area. Part (a) is a screen menu in which a group of information selectors are outlined upon the background Chiang Kai-Shek Memorial Hall. Part (b) posts a Taipei city map with spots of high-ranked hotels. Part (c) lists room services information of a chosen hotel. Part (d) provides information about various bus routes to the user chosen destination.

Figure 4 presents a branch of main sightseeing lines and spots (part(b)) in the Taipei city area (part(a)) with multiple levels of details including hike trails of the YangMingShan national park (part(c)).

As can be seen, a videotex view, no matter what size of window it resides, usually contains 3 principle parts: a self explainable graph, a few text fields, and several functional icons. Background voice narration and windowed video can also be inserted. Text, graphics, images, and voices are diverse types of data that can be identified. Line-bar menu, pull-down menu, pop-up menu, and screen menu constitute a menu family. Windows can be classified into different sizes and with or without scrolling capability. Icons can be either global or local according to their usages across a variety of views. Icons such as return to the previous view, and standard tourism marks for getting dinning and driving information are global through the entire system. Local icons such as choosing certain trails in a park can only be activated in the local view.

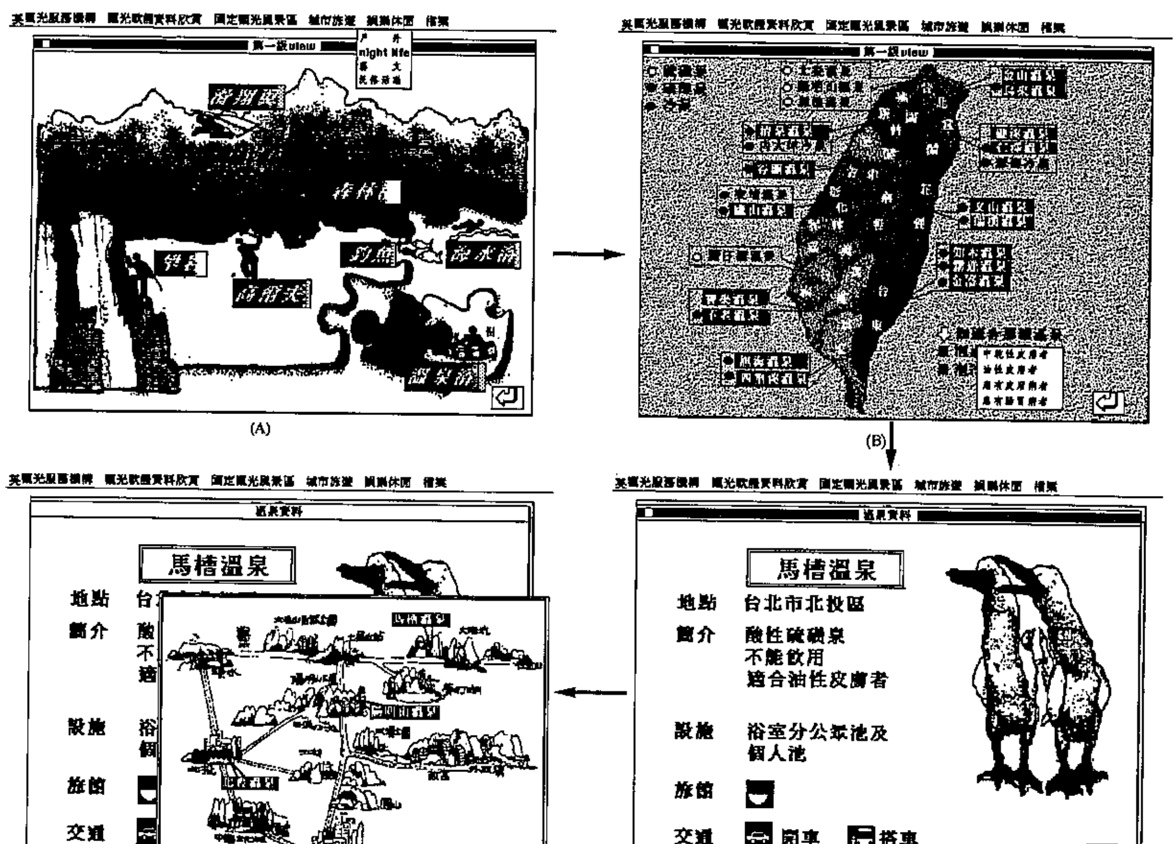

Figure 2. Branch views of amusement and recreation

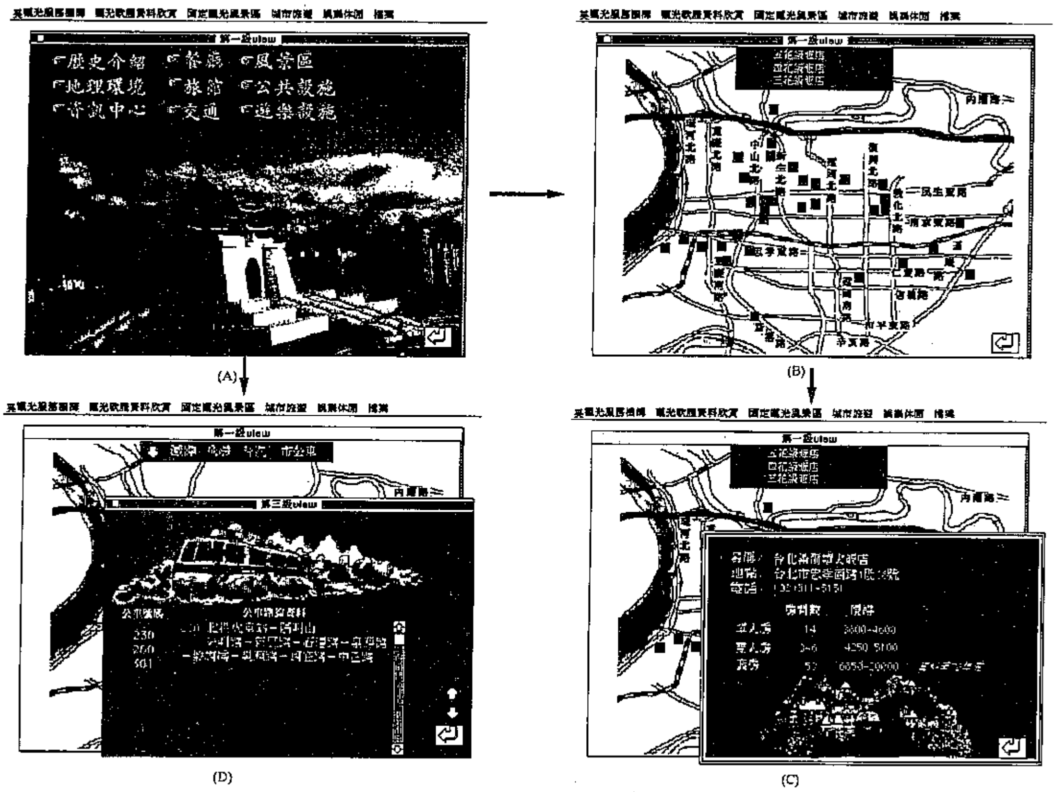

Figure 3. Branch views of Taipei city tour

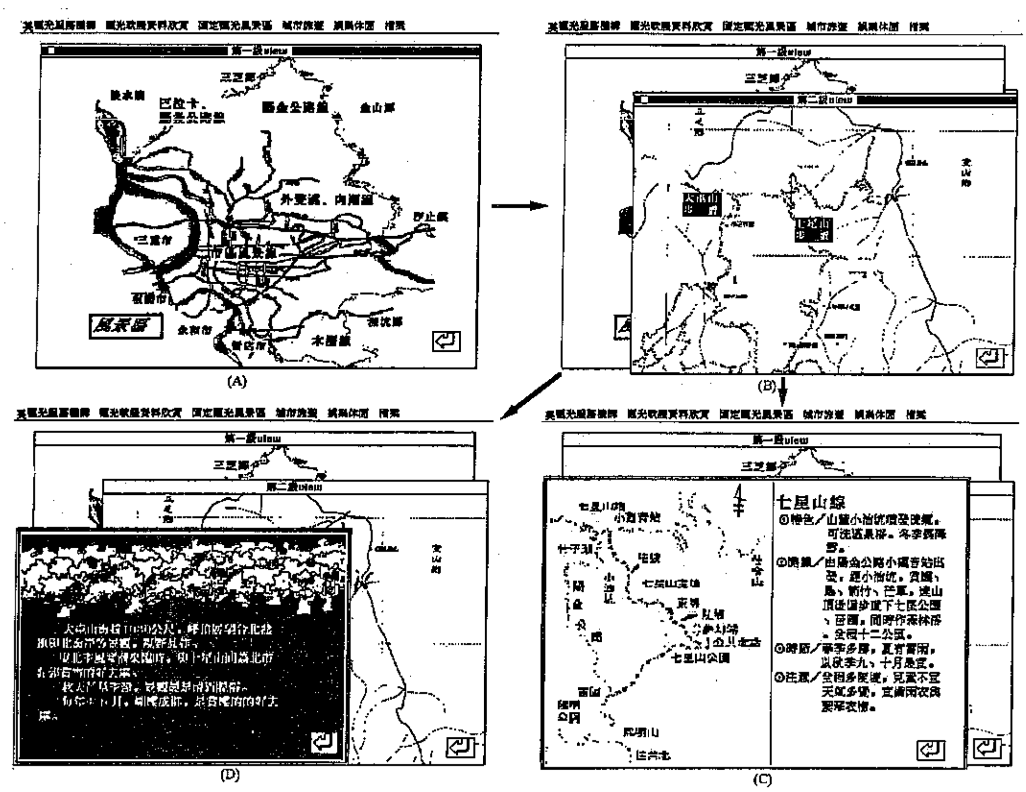

Figure 4. Branch views of sightseeing lines

(2) Conceptual layer

According to the specifications from the previous layer, classes and relationships are to be identified and the integrated class structure is to be constructed. A standard sheet for defining class is shown in Figure 5 and the whole class structure containing attributes, methods, and relationships is illustrated in Figure 6..

Reviewing the styles of contents, representations, and operations commonly used in the application layer, three types of object groups can be classified: menus, views of information, and physical databases. In the abstract level of the menu family, the Menus class is defined as the base class having the Menu Items class as its component class and thus establishes a has-a relationship. The Menus class also has three subclasses: Line-Bar Menus, Pull-Down/Pop-Up Menus and Screen Menus classes. Each inherits both attributes and methods from their superclass and constitutes an is-a relationship. An User Views class is defined to cover general attributes and methods of the user view objects.

Class Name : User Views	Class Number: 1.6	
Superclass: none	Composite Class: none	
Attributes :	Data Types	Explanation
name	string	
screen size	number	1: full screen 2: large size 3: middle size 4: small size
name of embeded menu	string	
name of background graphics	string	
Methods:		
show the user view		
fetch embeded menu		

Figure 5 A standard sheet for defining class

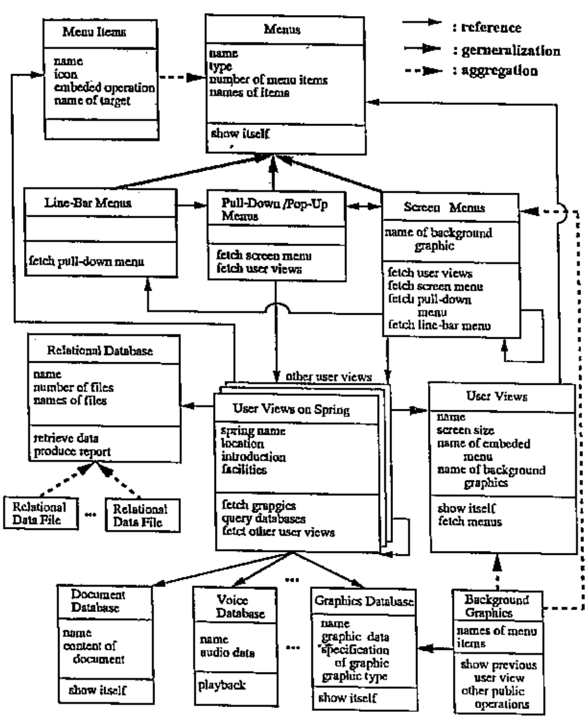

Figure 6. An object-oriented conceptual model

All specific purpose user views classes such as the User Views on Spring class are its subclasses. Both Screen Menus and User Views classes have Background Graphics class as their component class. From a screen menu or an user view, other menus and user views can be opened and thus entails reference relationships among menu and user view classes. Each specific user view class may contain multiple attributes with diverse types of data that are defined and stored in different databases. Thus, a number of media-oriented database classes are identified including Relational Database class, Graphics Database class...etc. They can be referred by the user views classes.

An example logical access mapping from the conceptual model to the user views of specific springs such as shown in Figure 2 is as follows (Rf stands for reference)

Line Bar Menus Rf--- Pull-Down Menus Rf--- Screen Menus Rf--- (part (A))
Taiwan Springs User Views Rf--- Springs Info User Views Rf--- (part (B),(C))
Graphics Database (part (D))

The information browsing and retrieval process is controlled by users through pointing to proper menu items and/or iconic buttons. A portion of the information and operation framework showing the relationships among three class groups with information contents and interfaces is given in Figure 7.

(3) Physical layer

Two prototyped systems have been implemented on PC 486 and on Macintosh IIfx computer environments. Connected to the multimedia workstations are a Microtek color scanner, a HP Laser Jet III printer, and a Tektronix ink jet color printer. Under the Macintosh system, the integrated software environment contains MacChina II (Great Eastern Software Co.) as a Chinese operating system, SuperCard (Silicon Beach Software) as a multimedia authoring system, and PhotoMac (Avalon Development Group), Pixel Paint (SuperMac Software), MacRecorder (Farallon Computing), MacroMind Director (MacroMind) as image, graphics, sound and animation processing tools respectively. For the PC 486 environment, CWindows (Microsoft) is used as a Chinese operating and windowing system. Toolbook (Asymetrix) is used as a multimedia authoring system. And, Photostyler (U-Lead), CorelDraw (Corel), Sound Blaster (Creative), and Animation Works (Gold Disk) are used as image, graphics, audio, and animation processing tools respectively. Since the conceptual structure is independent to the physical environments, only the implementing issues of the Macintosh system will be discussed without loss of the generality.

SuperCard is chosen to play a key role in multimedia data composition and representation primarily because of its capabilities to deal with colors and to manipulate graphics, windows, menus as objects. There are totally 9 types of objects in SuperCard. They are project, menu, menu item, background, card, graphic, field, and button. All objects can be defined and processed by using a hypermedia-style manipulation language called SuperTalk.

As for physical mapping, card is the basic unit for holding instance level information. Menus, fields, graphics, and buttons can be defined onto a card. Attributes and methods are resided in fields and buttons (and/or menu items). Command string of a method is written into an executable SuperTalk script. A Pull-

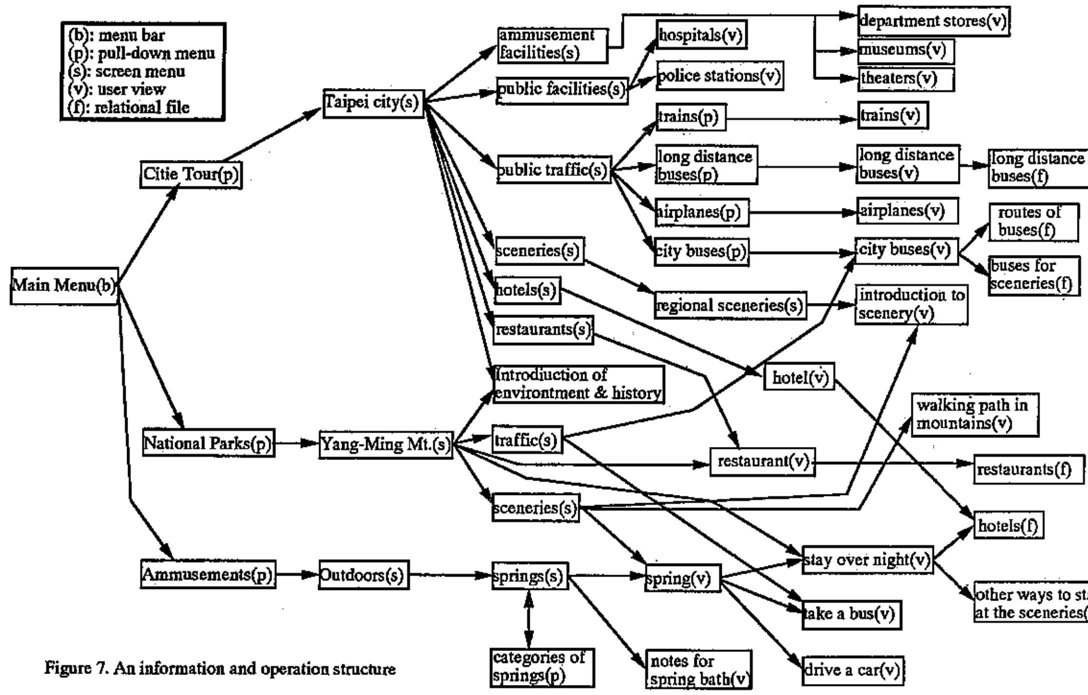

Figure 7. An information and operation structure

Down/Pop-Up Menus object can map to a SuperCard menu object in a natural way by simply mapping the component menu items to SuperCard menu items. A Line Bar Menus object is thus a group of organized SuperCard menu objects. A Screen Menus object maps to a card while screen menu items map to buttons of the card.

Background is the upper level object than card. A background may contain menus, fields, graphics, and buttons and can be shared by various cards. Window objects correspond to the class level User Views and specific User Views objects and may contain several backgrounds and cards. Consisting of many windows, a project can be treated as a special application level information branch in which multiple related user views are included.

Relationships are implemented at run-time by two types of operations. The first type of operation, usually is utilized as a direct reference, simply open a specific card or set of cards when a method is invoked. An example script is as follows:

```
on mouseUp
    open card "xxx" of window "yyy" of project "zzz"
end mouseUp
```

The second type of operation performs searching for and retrieving of relative information and then presents them simultaneously in a defined format and/or sequence. The following script illustrates interactions among different objects (if_then_else, repeat, set, and other internal/external commands are omitted for simplicity).

```
on mouseUp
    put field 1 of card "xxx" of window "yyy" of .... into
     temp1
    put temp1 into line i of field 1 of card "kkk" of ....
end mouseUp
```

Relationships such as generalization, aggregation, and reference can all be processed in the same way. Functions such as multiple criteria search, update, and print can also be coded as methods using the SuperTalk script language.

Implementing the prototyped system to the ISDN communication environment involves more technical problems to be handled and will be discussed in separate papers. A subsidiary research team with a testing group has been assigned to evaluate the performances of the prototyped system in comparison with the existing Chinese videotex system. Data collected from the performance testings indicate that the prototyped object-oriented multimedia videotex system with hypermedia user interfaces is easier and sooner in information processing and retrieval.

CONCLUSIONS

In this paper, we propose a three-layered conceptual architecture and an object-oriented methodology for modeling and developing multimedia videotex applications. A prototyped Chinese videotex services system is implemented on an integrated software/hardware environment. Consequences reveal a few benefits and improvements in the development and operating process in comparison with the traditional frame-based videotex systems. Firstly, the resulting conceptual data model is compact and understandable, yet is sufficient to model complex practical applications. Secondly , the structure is expandable and information as well as operations are reusable. Thirdly, the three-layered architecture guides the development process naturally and provides independencies and flexibilities among layers. Fourthly, multimedia data can be dynamically linked and bound at the run time. Finally, the information contents and presentations are enriched and the viewing process can be friendly controlled by users. Further studies will focus on policies and methodologies for integrating multimedia database management functions into the operating system and for implementing a refined version of this proposed architecture to the evolving distributed communication environment for actual commercial use.

References

[1] Akscyn, R.M., McCracken, D.L., and Yoder, E.A., "KMS : A Distributed Hypermedia System for Managing Knowledge in Organizations," _Communications of the ACM,_ 31(7), July 1988. pp.820-835.

[2] Alber, A.F., _Videotex/Teletext Principles and Practices,_ McGraw-Hill, 1985 ,pp.1-36.

[3] Bailey, Jr. C. W., "Intelligent Multimedia Computer Systems: Emerging Information Resources in the Network Environment," _Library Hi Tech,_ 29(1), 1990, pp.29-41.

[4] Bapa Rao, K.V., Gafni, A., and Raeder, G., "Dynamo: A Model for a Distributed Multi-Media Information Processing Environment," _Proceedings of the 22nd Hawaii International Conference,_ 1989, pp.800-809.

[5] Berra, P.B., Chen, C.Y.R., Ghafoor, A., Lin, C.C., Little, T.D.C., and Shin, D., " Architecture for Distributed Multimedia Database Systems," _Computer Communications,_ 13(4), May 1990, pp.217-231.

[6] Cheung, T. Y., and Sablatash, M., "A Functional Network Model for Analytical File Management in ISDN Systems From Generalization of Videotex Systems," _Computer Networks and ISDN Systems,_ 16, 1988/1989, pp.299-310.

[7] Cordes, R., Hofmann, M., Langendorfer, H., and Buck-Emden, R., "The Use of Decomposition in an Object-Oriented Approach to Present and Represent Multimedia Documents," _Proceedings of the 22nd Hawaii International Conference,_ 1989, pp.820-828.

[8] Hehmann, D. B., Salmony, M. G., and Stuttgen, H. J., "Transport Services for Multimedia Applications on Broadband Networks," _Computer Communications,_ 13(4), May 1990, pp.197-203.

[9] Hice, G. F. and Andriole, S. J., " Artificially Intelligent Videotex," _Applications in Artificial Intelligence,_ Andriole, S. J. (ed.), Petrocelli, 1985, pp.295-309.

[10] Huitema, C., "The Challenge of Multimedia Mail," _Computer Networks and ISDN Systems,_ 17, 1989, pp.324-327.

[11] Irven, H.H., Nilson, M.E., Judd, T.H., Patterson, J.F., and Shibata. Y., "Multi-Media Information Services : A Laboratory Study," _IEEE Communications Magazine,_ 26(6), June 1988, pp.27-44.

[12] Klas, W., Neuhold, E. J., and Schrefl, M., "Using an Object-Oriented Approach to Model Multimedia Data ," _Computer Communications_ 13(4), May 1990, pp.204-216.

[13] Kusekoski, G., "Corporate Videotex: A Strategic Business Information System," _MIS Quarterly,_ December 1989, pp.447-456.

[14] Moeller. E., Scheller, A., and Schurmann, G., "Distributed Multimedia Information Handling," _Computer Communications,_ 13(4), May 1990, pp.232-242.

[15] Meyrowitx, N., "Intermedia: The Architecture and Construction of an Object-Oriented Hypermedia System and Application Framwork," _OOPSLA'86 proceedings,_ September 1986, pp.186-201.

[16] Naffah, N., "Multimedia Applications," _Computer Communications,_ 13(4), May 1990, pp.243-249.

[17] Popescu-Zeletin, R., "From Broadband ISDN to Multimedia Computer Network," _Computer Networks and ISDN Systems,_ 18, 1989/1990, pp.47-54.

[18] Shetler, T., "Birth of the BLOB," _Byte,_ February 1990, pp.221-226.

[19] Sugimoto, M., Taniguchi, M., Yokoi, S., and Hata, H., "Videotex: Advancing to Higher Bandwidth," _IEEE Communications Magazine,_ 26(2), February 1988, pp.22-30.

[20] Tompa, F. W. M., "A Data Model for Flexible Hypertext Database Systems," _ACM Transcations on Information Systems,_ 7(1), January 1989, pp86-100.

[21] Woelk, D., Kim, W., and Luther, W., "An Object-Oriented Approach to Multimedia Databases," _ACM SIGMOD'86,_ May 1986, pp.311-325.

[22] Yoon, B. D., Suzuki, F., Lshikawa, H., and Makinouchi, A., "Experimental Multimedia DBMS Using an Object-Oriented Approach," _Proceedings of the Eleventh Annual International Computer Software & Applications Conference,_ Tokyo Japan, October 7-9, 1987, pp.632-641.

[23] Yu, C. C., and Chen, M. P., "A Distributed Multimedia Environment for Chinese Videotex Applications," _Journal of National Chengchi University,_ 63, December 1991, pp. 597-612.

[24] Ziegler, C., and Weiss, G., "Multimedia Conferencing On Local Area Networks," _IEEE Computer,_ September 1990, pp.52-61.