CHAPTER I
CHAPTER I

Introduction

1.1 Background and Problem Formulation

Multimedia has been the buzzword of the nineties. Multimedia includes a variety of data types such as text, graphic drawings, images, sound and video. The features of such data are quite different from the data used in conventional databases. An important issue for all multimedia data is how to understand and interpret them. Raw multimedia data are a list of bytes that has to be preprocessed to identify the component objects. These objects may be someone's voice in a sound track, a person in an image or video, or a text in a document. Compared with traditional data the most noticeable features of multimedia data are their complex structural and semantic information, spatiality(spatial layout of objects and their relationships), temporality(time-dependence in media such as video and audio) and presentation requirements(because of synchronization between different media). The temporal and spatial relationships that exist between multimedia objects ought to be taken into account while modeling and presenting them. For example, multimedia document may have, inherited or artificially generated temporal and spatial relations among multiple media objects. The Multimedia Information System(MMIS) acts as a repository for such diverse data. MMIS is defined as a system designed to store, transport, display and in
general to manage such data and has more functionality and capability as compared to conventional information system. At the heart of MMIS lies a Multimedia Database Management system (MMDBMS). Traditionally, a database consists of logical collection of data related to a given set of entities, while Database Management System (DBMS) is a collection of interrelated data with a set of routines used to define, create, store, access, manage and query the database. Similarly, MMDBMS provides support for multimedia data types along with facility for their creation, storage, access, query and control. Methodologies to incorporate time-dependent and spatio-dependent multimedia objects into a database system are needed for the development of a MMDBMS and hence for MMIS. The conceptual model is one of the components in the design of any DBMS and MMDBMS is no exception. The complex structure and information about spatio-temporal relationships of multimedia data cannot be represented in the traditional data modeling paradigms. Thus we need to develop a conceptual model that will describe the spatio-temporal information along with structural information necessary to represent multimedia time dependencies and synchronization. Also, MMIS has to provide for presentation of the multimedia data that has been stored in an appropriate database. During this process of presentation, dependencies (both in time and space) which have been stored along with multimedia data should be satisfied.
1.2 Multimedia information

Multimedia data is often organized as a document. We encode a multimedia document as a collection of data of diverse media and a set of structure and synchronization relationships (with respect to space and time) that describes how data components are to be presented and manipulated. Thus multimedia information perceived by viewers consists of temporal, spatial and media information [71]. Temporal information specifies the relationships with respect to time that exist amongst the various media objects. Spatial information specifies the locations of media objects in space. Media information describes the individual media elements. Interactive media documents may also include unpredictable time events, such as user interaction etc. These exist as external information. Thus the information contained in a multimedia document can be depicted as:

![Diagram](image-url)

Different media objects have different characteristics and could be described differently, therefore the media information could further be decomposed into:
*Media content information* that describes the contents of the media.

*Media spatial information* that provides the relative positions of objects within media, for example, person in front of table.

*Media temporal information* which describes any type of change that happens over time. Each media type may not have all these categories of information. The table below shows various media types and the information types they include:

<table>
<thead>
<tr>
<th></th>
<th>text</th>
<th>graphics</th>
<th>animation</th>
<th>video</th>
<th>audio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C</td>
<td>C,S</td>
<td>C,S,T</td>
<td>C,S,T</td>
<td>C,T</td>
</tr>
</tbody>
</table>

where C-content, S-C-spatial; T-temporal

Media may be classified into static and dynamic, depending on absence or presence of temporal component to describe change over a period of time. Text, image and graphics are referred to as static (or non-continuous or aperiodic media) and video and audio are called dynamic (continuous or periodic media)[116]. Different types of temporal and spatial relations exist between two objects. These relations or their combination completely specifies how two or more objects are related with respect to time and space co-ordinates. For example, two objects *overlap*, object A *before* object B, two objects *start or end* simultaneously; object A is displayed to *right* of object B or object A is displayed towards *northwest* of object B.
After deliberating on multimedia data we would like to discuss about multimedia system and multimedia database management system.

1.3 Multimedia system

Several definitions for the term multimedia application or system are described in literature. Three criteria for the classification of a system as a multimedia system can be distinguished, viz. a viz. the number of media, the type of supported media and the degree of media integration. The simplest criteria is the number of media used in an application. Using only this criterion, a document processing application that supports text and graphics can be regarded as a multimedia system. The type of media supported is second criterion. Using this criterion some authors define multimedia system as a system that supports the processing of more than one medium with at least one time-dependent medium. (Time independent media object is usually presented using one presentation unit, for example a graphic; time-dependent media objects are presented by a sequence of presentation units, for example a video sequence presented frame after frame.) The degree of media integration is the third criterion. Integration implies that different types of media remain independent but can be processed and presented together. Combining all three criteria, the multimedia system is defined as a system or application that supports the integrated processing of several media types with at least one time-dependent medium [25].
1.4 Integrated vs non-integrated multimedia database

A fundamental problem in managing multimedia data lies in the handling of rich semantics of multimedia data. The description of rich semantics is crucial to build a powerful multimedia information system. Besides providing the services and capabilities of traditional databases, multimedia database management system which is an integral part of MMIS needs to support a variety of other features such as [30]

- Native support of multimedia data types
- ability to create a model of real world
- temporal and spatial modeling
- presentation ability

The size and complexity of multimedia data demands special treatment in storage, presentation, transmission etc. There are two approaches towards the design of a MMDBMS. The first approach advocates design of a multimedia specific DBMS from the scratch whereas in the second approach, a multimedia DBMS is usually built on top of an existing object-oriented or relational DBMS as they provide reliable support. For the latter approach an appropriate multimedia layer or interface has to be added to fulfil some specific requirements like presentation etc. Despite different approaches, a consensus exists that such efforts should be based on object-oriented techniques. A Multimedia DBMS Framework using the second approach is shown in fig.1.2.
In the above diagram preprocessor will carry out tasks like acquisition of multimedia data, compression of image data etc.

In a multimedia database, non-continuous media such as image and text are stored as native objects and use continuous media file servers as the underlying storage system to store continuous media such as audio and video[127]. This is a hybrid approach between integrated and non-integrated multimedia database system. Non-integrated systems store only meta-information about the multimedia objects in the database but store objects in files. Meta-information (or metadata) is data about a multimedia object, for example, for text it may be the format of the text, for image it may be pixel depth, color model, for audio it could be sample rate. Metadata may be considered as attributes of multimedia data, which may help in
its interpretation like format of text, pixel depth of image. Multimedia data may be historical or documentational in nature, like date associated with an image. Non-integrated approach keeps the database size small and easy to manage but it does not allow the use of basic DBMS services such as transaction management, access control and concurrency control for the management of multimedia objects. On the other hand in integrated approach the database system store multimedia objects together with meta-information in the database. In case of integrated approach, besides the advantage of using DBMS services, it is possible to develop a uniform interface for accessing information about document content and structure.

1.5 Review of literature

There has been considerable prior work on modeling multimedia data, both in the context of absolutely new multimedia document systems or as multimedia extensions to existing data base systems. Data models are central to multimedia database systems and a data model isolates users from the physical details. In studying representation of multimedia data appropriate conceptual data models are required to organize the various data types typically found in multimedia database(MMDB) systems. Multimedia data models (just like traditional data models) capture the static and dynamic properties of the database contents. The static properties include the objects that make up the data, the relationships between the objects, the object attributes and so on. Examples of the dynamic
properties include interaction between objects, operations on objects, user interaction and so forth. However, the unique nature of multimedia data requires certain considerations when choosing the data model. For instance, some multimedia data types(such as video) might require special data models for improved modeling efficiency and flexibility. Moreover, the importance of interactivity in multimedia systems makes their support by the data model an important issue. The conceptual model is one of the components required in the construction of MMDBMS and therefore for construction of MMIS. Below we give a brief review of some of the models and how they organize multimedia data.

Conceptual models for multimedia objects are classified into five categories - graphical models, petrinet based models, object oriented models, language based models and temporal abstraction models. In graphical models labeled directed graphs are used to represent information i.e. hyper text[161]. They provide good navigation facilities. However, in this study there is no specific mechanism to handle temporal synchronisations. Petrinet models developed by Little and Ghafoor represent component of multimedia objects and describe interrelationships as transitions. Further their model has been extended to form OCPN (object composition petrinets)[105]. The major strength of their model is the ability to capture the necessary temporal relation and OCPN can easily be extended to model databases, with each place representing an object with associated attributes operations and pointers. In case of object oriented models[119] the multimedia object has attributes, relationships and methods.
Class, instance, superclass, subclass, inheritance, generalisation/aggregation are
defined. They can capture temporal relationships and this approach can generate
multimedia database schema. Steinmetz[153] introduced a model based on
concurrent languages to specify multimedia synchronization requirements. The
major advantage is that they directly lead to an implementation. However, unlike
graphical models they are difficult to visualize and verify.

Vazirgiannis and Mourlas[165] have proposed a model for multimedia
composition. The model provides for spatial, temporal and spatio-temporal lists
along with identification of objects related to these lists. Thim and Rakow[159]
have modeled multimedia documents as a complex object. The component of this
complex object is modeled as object itself. Diamond also portrays multimedia
documents as a structured object which contains a collection of elements of
various media types alongwith information about how the elements are related to
each other and how they are presented[160]. Bulterman, Rossum and Liere[27]
have described a document structure for transportable dynamic multimedia
documents. The building blocks of this structure are data blocks, data and event
descriptors, synchronisation channels and synchronization arcs. In this model
document is depicted as a tree consisting of nodes and branches. Besides the
explicit synchronization provided by synchronization arcs, default synchronization
is provided as a result of node type i.e. sequential or parallel. Karmouch and
Emery[86] have developed a system called mediadoc where the logical structure
of the document is represented in a tree like shape. Abstract objects are used to
represent the document in aggregation hierarchy. These documents are stored in an object-oriented database. In the middle layers of the tree sequential, concurrent, alternate version and set object blocks (the global logical structure of the mediadoc provides for seven independent blocks) describe how their children are related temporally. The document profile lists information such as author name, title name, keyword etc. Papa, Ragucci et al[128] have described an object oriented model of multimedia documents based on the document's conceptual structure where each object consists of a unique id and may have a set of attributes and associated methods. Document's conceptual structure is composed of an aggregation of basic conceptual objects. A composite physical object and basic physical objects are also defined. Gibbs Damis and Tischritzis[57] have modeled a multimedia object as an active object with a number of ports through which the objects communicate. These objects are arranged in a hierarchy of multimedia classes and data types. All multimedia objects inherit from class multimedia object and active object. Herzner and Hocevar[68] describe an object oriented data base approach for storing multimedia documents in a distributed environment. They have defined five types of nodes - class, instance, elementary datatype, datanodes for default and method nodes. There are no separate categories for attributes and relationships. Twelve types of edges that can be regarded as binary relations or predicates have been defined. These are can-have-parts, can-have-types, can-have-attributes, can-have-relationships, has-relationship, has-method etc. They also distinguish between generic document
structure and specific document structure. Buchanan and Zellweger[26] have given an approach to describe the structure of interactive multimedia documents. The document level specification consists of media item, temporal constraints, operations, duration and costs and unpredictable event control. ODA[32] documents has two structures: a logical and a layout structure. The document structure forms a tree and its parts are called objects, the indivisible one being basic object and others are called composite objects. The concept of object class is also introduced. The set of logical object classes and layout object classes associated with a document are called generic logical structure and generic layout structure. The structures that are associated with a particular document are called specific logical structure and specific layout structure. This has also been extended later to provide for temporal synchronisation to the layout structure only[132].

The major challenge in multimedia information management is how to synchronize various types of data both in space and time. The synchronisation requirement plays a key role while designing a multimedia database. Vazirgiannis and Mourlas[165] refer to multimedia synchronization as a general term comprising of content relation, spatial relation as well as temporal relation. Though all the three types of synchronization relation are important, but we will discuss the temporal aspects in our study. (Now onwards we refer to temporal synchronization as synchronization) Synchronization implies that events are carried out in a particular order with respect to time. The synchronization specifications are a part of the description of a multimedia object and are captured at the time of object creation.
This specification describes all temporal dependencies of the related objects and sub-objects. Thus in the present context multimedia synchronization (sometimes referred to as orchestration) is a concept/technique which coordinates the presentation of various media streams in a time domain. The synchronization specification has been an area of extensive research in the field of multimedia because this specification determines the flow of presentation. There are various ways to describe synchronization specification. We study some of the techniques to specify synchronization requirements.

Blakowski and Steinmetz[25] have classified temporal synchronization specifications into interval-based, axes-based, control-flow-based, event-based and scripts. Another classification of multimedia synchronization was studied by Blakowski et.al.[23] in which they introduced the concepts of hierarchical synchronization where multimedia objects were regarded as a tree consisting of nodes denoting serial or parallel presentation; synchronization on a time axis in which single-media objects are attached to a time axis that represents an abstraction of time; and lastly, synchronization at reference points where single-media presentation are composed of sub-units presented at periodic times. In DARPA system the time coordination of a presentation is captured in the body of document, where the three controls are: sequential, simultaneous, and independent[138]. Sequential and simultaneous data is intended for synchronous viewing, whereas independent data can be presented in any time order. This falls into hierarchical specification category that is a subclass of control-flow-based. In
the document structure described by Bulterman, Rossum and Liere[27] synchronization information is encoded in terms of synchronization arcs. Each arc is a directed connection between two event descriptors, with a set of synchronization attributes associated with it. In their study, the basic tree structure of document also imposes a default synchronization that is based on the node type of the ancestor of a data node. This is a combination of hierarchical(control-flow-based) and event-based methods. Mediadoc system developed by Karmouch and Emery[86] includes a set of synchronisation specification types for creating the presentation schedule for the multimedia document. Mediadoc defines a scenario which comprises of the duration and synchronisation specifications. Scenarios are converted into scenario object rendering time (SORT) graphs. This model provides three temporal relation types: before (<), equals (=), and after (>). Here each multimedia document can be associated with several scenarios representing different ways it can be rendered. Layout information can also be added to scenario objects to describe their position and extent. Hirzalla, Falchuk and Karmouch[72] extended the work of Karmouch and Emery to model active multimedia documents graphically within a time line model to incorporate indeterminate scenarios by including partial specifications or interactive multimedia scenarios. Their extended model, based on the traditional timeline model, includes temporal inequalities between events. They also consider the problem of representing events whose start or end times are not known at authoring time. Wahl and Rothermel[169] presented a high-level, interval-based
temporal model whose powerful operators include both temporal equalities and inequalities between events. Gibbs, Damis and Tsichritzis[57] discussed the composition and synchronization of multimedia objects within an object-oriented framework for multimedia applications, where a multimedia object is an active object with a number of ports. This object-oriented framework includes a number of multimedia classes, and a hierarchy of multimedia datatypes and a composite is responsible for maintaining synchronization of its components. Each component of the framework has a synchronization mode attribute and four synchronization modes, namely, no-synch, demand-sync, test-sync and interrupt-sync, have been defined. The temporal relationship in a composite object are represented by a composite timeline diagram, one timeline for each output port within the composite. The timeline representation shows the concurrency within a composite due to superimposition of a number of multimedia channels (each horizontal bar in the timeline diagram). Composite objects handle Jump and Cue requests by forwarding requests to their sub-components. Vazirgannis & Mourlas[165] in their object-oriented model for interactive multimedia presentations have defined three types of multimedia compositions, i.e. spatial, temporal and configurational. Temporal relations can be before, during, overlaps, starts, ends, equal etc. There are two special events, one, the start point of a specific action, and second, the end of that action. Using above constructs sequential and parallel synchronizations have been defined. A composite item class - scenario list has been defined for modeling the data and behavior of complex multimedia presentations. Each tuple
of the scenario list has the form: <start_time, duration, action, synch_events, exception_handler, interrupt_handler>. Petra Hoepner[132] gave a general synchronization model for the description of presentation sequences of multimedia objects based on qualitative intervals. For the purpose of synchronized presentation of multimedia objects six path operators namely, parallel-last, parallel-first, sequential, selective, repetition and concurrency, are given. The attribute duration defines the presentation duration of a basic layout object in Basic Time Units (BTUs) with a default value of '0'. A static basic layout objects with a default duration is presented until it is terminated by some other instance, which means that the endpoint is undefined. A dynamic basic layout object with a duration of '0' is presented for its own inherent presentation time. Otherwise, dynamic basic layout objects are presented for a specific duration, which is defined by the presentation time of the content of the object, i.e. an audio sequence lasts five minutes. A static basic layout object also can be presented for a specified time, i.e. a picture is presented for one minute. MHEG provides a virtual coordinate system that is used to specify the layout and temporal relations of objects in space and time according to the virtual axes-based specification method[122]. MHEG lists four types of multimedia synchronizations i.e. elementary synchronization which is the synchronization of two objects(with regard to the same reference origin time, or one with regard to the other); chained synchronization refers to presentation of a set of objects one after another like a chain; cyclic synchronization is repetitive presentation of one or more objects; and
conditional synchronization is the presentation of an object after a certain condition is satisfied. Buchanan and Zellweger[26] in their Firefly system proposed a temporal specification scheme for the definition of general multimedia scenarios. Their model uses instant as a basic time unit. For the development of temporal specifications, a graphical representation is used. The synchronisation specification method is a combination of reference point and interval-based technique. The contextual information corresponds to the basic binary temporal relationships between instants (i.e. only a single relationship between two instants). Quantitative information can also be included if required. The specification of the temporal constraints distinguishes between media-level specifications, that describe the temporal behaviour of individual media objects and document-level specifications, that describe the temporal behaviour of a complete multimedia document. Little and Ghafoor[104,105] proposed the OCPN(Object Composition Petri Net) as a temporal specification scheme for the description of general multimedia scenarios. The graphical representation(syntax) is based on an extended type of Petrinet and retains most conventional petrinet semantics. The model of time in OCPN uses interval as basic time unit. The contextual information is both qualitative and quantitative. The temporal relationships considered are the thirteen basic binary temporal relationships give by Allen[8]. No indeterminacy can be expressed in the temporal scenario because the temporal relations are modified by quantitative information, which completely determines the scenario. Herzner and Kummer[69] have given a model for
synchronization based on events. Their prototype implementation integrates dynamic media into static documents and takes care of user interaction as well. The synchronization is event-based, where an event denotes a certain point in time. Cues have been defined to specify the conditions under which specific operations shall be performed. A cue can be conceptualized as a trigger that executes an operation when the condition is fulfilled. A cue may be named, so that it can be referenced by other cue expressions. Cue expressions represent boolean functions over time, whose values are always defined (during presentation) but may repeatedly change. The advantage of this approach over other approaches is that cue expressions provide a method to express temporal interdependencies; by specifying events relative to others rather than absolute to the begin of the (document’s) presentation.

A multimedia presentation is defined as a collection of streams that are coordinated with respect to time as well as space and which are synchronized and have a shared presentation control[13]. Multimedia can be viewed as an integration of data from such diverse media as text, graphics, audio, video etc. to be represented as a single information unit. This multiplicity implies that, if the various media objects are to convey the desired information to the user, there must be a way of coordinating how and when they are presented, both in terms of a single media object and as a group. Various ways have been suggested to achieve this coordination. Next, we discuss some of the work done regarding presentation of multimedia data.
Firefly, the multimedia document system developed by Buchanan and Zellweger[26], includes a scheduler to solve duration and temporal synchronisation constraints to produce a document rendering schedule for presentation. The objective of this system is to automatically generate consistent presentation schedules for interactive multimedia documents that comprises of media objects of predictable behaviour (like audio and video) and objects of unpredictable behaviour (like user interactions. To support interstream synchronization, skew control mechanisms are supported. They are based on dropping and duplicating data units. Vazirgannis & Mourlas [165] have given an O-O model for interactive multimedia presentations for developing complex interactive multimedia presentations. They define the scenario that may be viewed as a predefined spatial and temporal sequence of presentations of multimedia objects. Events are regarded as a special kind of information passing mechanism between objects. There are two special events - the start point of a specific action and the end of that action - that assist in achieving coordinated presentations. If two or more media objects are combined in a multimedia presentation the above events will be used in order to achieve the desirable temporal order of presentation i.e. sequential synchronization or parallel synchronization. More complex synchronization conditions can be formed if more media objects are combined for a multimedia presentation. Asynchronous interaction with the user can also be handled by this model. The handling of interrupts during presentations is also supported by the model. A composite item class is defined for modeling the data
and behavior of complex multimedia presentations. Thimm and Klas [158] have defined a stored presentation as a finite set \( M \subseteq M \) of \( L \) component media presentations,

\[
M = \{(i, \text{type}_i, R_i, E_i) \mid i = 1..L, E_i = \{e_i^k \mid e_i^k \in \mathbb{R}_0^+, k = 1...N_i\}\}
\]

where \( i \) denotes a component media presentation;

\( \text{type}_i \) is the type of media presentation identifying the media type (e.g. text, picture, audio, video) and the encoding of media data (e.g. ASCII, GIF, M-JPEG, MPEG, etc.)

\( R_i \) is a finite set of temporal relationships, such as starts, overlaps, ends, etc., media presentation \( i \) is participating in;

\( E_i \) is a finite set of media type specific explicitly defined external presentation constraints such as data rate, data quality, etc;

\( N_i \) denotes the number of external constraints for media type. The number varies for different media types since some constraints are only relevant for some specific media types and irrelevant for others (e.g. color depth)

Park, Lee, Cha and Nah [129] have proposed a technique that generates presentation plan for the query results. The schedules can be generated even if the result of the query does not belong to an already existing class. For generating schedules a Temporal/Spatial tree (TS-tree) representing the spatio-temporal structure of a class is used. More than one TS-tree may have to be combined to present results of a query. For representing spatio-temporal structures various
domain types namely, spatial composition, spatial sequence, temporal sequence, parallel collection etc. have been defined. The generation of presentation schedules is done in two steps, the preprocessing step and scheduling step. In the preprocessing step the schema level TS-tree is converted into instance level TS-tree. In the second step schedules are given in terms of list of actions. The Layered Multimedia Data Model given by Wynblatt [175] consists of four layers: DDL(Data Definition Layer), DAL(Data Arrangement Layer), DPL(Data Presentation Layer) and the CL(Control Layer). The DPL provides the abstraction of multimedia presentations to the control layer. The DPL maps the data to the output devices. The parameters of mapping such as playback rate and direction etc, as well as viewer which consists of a collection of parameters has been defined. The purpose of the viewer is to allow each presentation parameter to be defined statically or dynamically. The three special presentation parameters provided in this model are silence, wait and foreground priority. The presentation layer chooses the recovery methods to be used if two base events become unsynchronized. Provision is also made for specifying resynchronization action. Another function provided by this layer is to describe the format of playback. The DPL consists of two types of presentation templates - MovieBoard and ImageNarrative. Both can be instantiated with any one of the DAL layer events - WalkThrough and Collage event. Little and Ghafour [105] have given temporal access control algorithms which deal with playout management. These algorithms provide functionality for partial interval playout, reverse playout, and playout
deadline and playout deadline determination. For identifying the deadline of objects, the recursive traversal of the tree hierarchy is carried out. Also an algorithm is given algorithm to find out playout time, deadline, start time for objects using forward or reverse parameter. This algorithm uses as input a parameter $t_{\text{elapsed}}$ to indicate the current elapsed time throughout processing. Object's sub-nodes are identified in a recursive tree-traversal. As each sub-node is identified its temporal parameters are evaluated to generate a set of playout times. The deadlines can be generated in forward or reverse direction beginning at an arbitrary point in the temporal hierarchy. IMMPS designed by Shih and Davis [147] provides for construction of an intelligent presentation. The system presents a presentation specification language as well as a multimedia resource and presentation database management system. According to this model presentation should be considered from two perspectives, control view and knowledge view. From a control view, presentation is a graph with nodes as presentation windows and edges as communication links. From the knowledge representation view, a presentation is a hierarchy of windows sharing information such as the user's identity and background knowledge. This system uses a directed acyclic graph (DAG) for the knowledge inheritance structure. According to Rothermel and Dermier[140] most of the problems of synchronizing continuous media data can be traced to the timing constraints imposed on their presentation. They have discussed Quality of service parameters for presentation such as jitter and skew
which become critical because users expect the presentation of multimedia data to be continuous in time and thus may not accept discontinuities in their playout.

1.6 Summary of thesis

In chapter II we introduce a conceptual model that will describe structural information necessary to represent multimedia data as well as the temporal information required to describe time dependencies and synchronization. There are three levels in our model. The first level is given by the data types, provided by underlying object-oriented database management system; the second by the object class (document architecture) and the third by the interrelationships between two different types of object classes i.e. documents. Document is our unit of information. We describe a technique to specify the temporal interrelationships between media objects as a part of document structure. We also elaborate on how the components of our model can be mapped into class definitions for an object-oriented database.

Chapter III deals with the synchronization specifications, which in our model is a part of the description of a multimedia object and are captured at the time of object creation. This specification describes all temporal dependencies of the related objects and sub-objects. The objective of these specifications is to ultimately predict the presentation times of data units resulting in synchronized playout. The instant is our basic unit of time and we used event-based technique to model temporal relations. We also explain how the synchronization specifications
described in the structure of a multimedia object are sufficient to describe elements of a temporal model. In the end we define concept of triggers which are used to represent Allen's relations.

In chapter IV we propose a scheme which generates presentation schedules without passing the authoring effort to the user. The presentation schedules for the objects are generated by traversing the document tree and using the temporal relations defined previously. This schedule is stored as a table which is sorted before going to the playout phase.

Chapter V gives implementation details for generating presentation schedules and Chapter VI concludes our work and suggests guidelines for future work.

The functional requirements of Multimedia DBMSs can be represented as:
where DDF is Data Definition

SYN is Synchronization

PRN is Presentation

Thus we address the following problems in the present work: (1) modeling and management of multimedia data, represented by DDF above (2) incorporating temporal specifications in the structural model, represented by SYN above and (3) presentation of multimedia data, represented by PRN above. We suggest a layer on top of an existing Object-oriented DBMS which provides facilities for storing time-based objects and for building sequential or parallel presentations of multimedia data. The underlying database system allows to store and retrieve data and our extension provides a way to model multimedia data and build multimedia presentations expressing temporal synchronization constraints between objects.