ISSN 2047-3338

Structure of Multimedia Documents: A Theoretical Approach

R. N. Jugele[1] and Dr. V. N. Chavan[2]

[1]Department of Computer Science, Science College, Congress Nagar, Nagpur, Maharashtra, India
[2]Department of Computer Science, S.K. Porwal College, Kamptee, Nagpur, Maharashtra, India

***Abstract*– The most frequently used computer application is the classical document handling system, electronic document and at this stage is the exact equivalent of a traditional paper document. It comprises information represented by text and graphical images structured in a desired fashion. Upcoming technological developments are expected to considerably advance the way in which users interact with their systems as well as the way they communicate among each other. Increase in processing and communications power allows integrating new media types, which capture the naturalness of information as experienced in everyday's life. This feature will make these new media in a range of fields and a strong impact can be observed in document handling. Document handling will have to support new interaction schemes. The development gives the directions and considerably enhances the way documents are handled. Other current developments concerning documents are the move towards hypertext and hypermedia, although some of the presented techniques apply to these as well. The study is based on evaluating results of recent and ongoing research; including results which have already lead to new document standards. The goal design for study to provide a comprehensive overview of requirements and solution approaches, to show up issues requiring future research and the aspects are treated: information structuring, communication support and data management.**

***Index Terms*– Document, Structure, Logical Structure, Layout Structure, Physical Structure, Hyperoda, ODA, SGML, Hytime, MHEG and Multimedia Object**

I. INTRODUCTION

MULTIMEDIA and multimedia documents varies greatly, some consider graphics and text to be a case of multimedia, others use this notion only if time dependent media like audio and video are included. Here the term multimedia denotes a combination of time dependent and time independent data [1]. It distinguishes the following basic data types in documents:

text, as the basic representation for written documents,

tables, give an overview on selected information,

raster graphic, for representation of included pictures,

vector graphic, support technical drawings,

animated graphic, for the demonstration of moving graphics,

audio, include sounds and voice annotations

video, include moving pictures to a document.

Multimedia documents reflect the inclusion of multimedia data into documents and the resulting impact with respect to handling support. There is no standard consistent that how a multimedia document is to be defined. The following are some assumptions and understandings:

A document is a self-contained entity of information belonging semantically together, stored persistently and may be accessed at any time. As for time-dependent media, it means that the data has to be available from a stored file and not from a live source and to pure data a document comprises structuring information. This refers to the content of a document and its presentation; it also includes the specification of user interaction during the presentation of a multimedia document.

II. STRUCTURE OF MULTIMEDIA DOCUMENTS

The documents are distinguished between three different structures.

- The layout structure specifies how the components of a document like titles, paragraphs or figures relate visually, where the document is to be presented on paper or on screen.
- The logical structure identifies the components of a document and defines logical relations between them.
- The physical structure represents the separation of a document into several parts that may be stored separately.

These structures are the main part of a document architecture and additional considerations cover specification of the basic content types like ASCII text, graphics format, coding and the relationship between the structures.

In most document architecture cases logical structure and layout structure cannot be distinguished. The logical structure is inherently expressed by the layout structure e.g., different line spacing for paragraph demarcation or different font sizes for title identification. Clear distinction between the structures offers a greater deal of flexibility concerning handling and processing e.g., different layout specification for the same document.

Multimedia documents extend the paradigm of traditional documents by containing both time independent and time dependent content portions. This requires consideration of temporal relationships between the two. Similar to specifying the spatial relationship of paragraphs, figures and titles when it comes to presentation also the temporal relationship between paragraphs, figures, audio sequences and video clips must be specified.

Another issue concerns referencing of external document parts. Text or graphic content with respect to data quantity and processing constraints are different from video or audio sequences. In a richer physical structure this encourages separation of such content portions from the rest of the document in order to store and handle them on special purpose systems.

In computer applications like office systems, desktop publishing or programming environments, writing and editing documents is one of the basic integrated task. The existence of a plethora of programs and products in this field is quite natural due to the large number of different applications they are used for. This diversity has its counterpart in the formats and concepts that are used to represent documents. The exchange of documents between different products is likely to fail and this problem became the more serious because more computer systems were connected to each other simplifying data exchange in the form of documents.

Converter based approach between different formats does not fulfill the requirements of openness which led to two international standards namely ODA [2] and SGML [3]. Which support to enable open document interchange for the office and publishing area specially covered traditional documents that contain text, graphics and images. Whereas emerging multimedia data portions in documents will face the same difficulty concerning exchange probably to a greater extent since the formats and concepts used are more diverse.

III. CONCEIVING IDEA

It is a mechanism to specify relationships between certain units of a document, generally there must be means to identify certain document portions (e.g. footnote text) and to specify the semantics or properties of the relationship (e.g. reference). The integration of time dependent content portions the mechanisms have to be extended considerably. Specification of temporal relationships requires richer means of identification of certain document portions and semantics specification.

The smallest document portion to be identified is equivalent to some structural building block (e.g., a paragraph) there must be means to identify parts of building blocks. This may not be straightforward in the case of audio and video sequences.

Figures, chapters or entries within the bibliography are refers traditional document architectures where the semantics of relationships are mainly references. Whereas when it comes to expressing temporal behavior, things become more complex. Two audio sequences may be presented concurrently or sequentially, starting at the same time or with some delay.

Quite sophisticated mechanisms are required to allow specification of relationships, especially in describing temporal dependencies where flexibility is an important issue. Multimedia documents are rather complex accumulations of content portions that belong to different information types. Each information type needs dedicated tools that allow for generation, modification and presentation.

Object oriented model provides structuring masses of data into smaller modules called objects, used to access and process the object's data which leads to rather encapsulated units. Every unit has an interface supporting invocation of methods designed in a uniform manner throughout all objects. Thus handling and processing of different information types is facilitated as tasks like presentation, access or modification lead to present, access or modify calls that are the same for all objects as the interface is the same. Usage of such autonomous objects enables their sharing among multiple documents and becomes more appealing and more effort has been invested in creating the objects.

IV. STANDARDS

The standards play an important role in information technology and are the basis on which complex applications may be developed without having to design and specify everything from the very beginning. The developers of computer hardware and software are on the way to define and establish standards that take off the burden of designing and implementing certain modules or formats each time from scratch, but enable at least usage of common architectures, data exchange formats and API's. Following is a survey of standardization in the field of multimedia documents.

A. ODA / HyperODA

Office applications need to exchange information in form of documents came up early in the environment, it enable the “blind document interchange”, i.e. the exchange of electronic documents concerning their content and their logical and layout structure without the need to define special agreements in advance. The Open Document Architecture (ODA) addressed this wider application area. ODA is a CCITT Standards (T.400 series) as well as ISO/IEC (IS 8613).

The ODA standard specifies the logical and layout structures of a document and how they are encoded in ASN.1 for communication purposes. Standardization portions is done by referring to the existing standards mainly text, graphics and image and specifying their interface as content portions to ODA and it also define processing model for the formatting process. An ODA document consists of:

- *Logical structure:* It specifies the document in terms of a document root, composite object and basic object that are used by the editing person of a document to structure. To ease the control of the logical structure of a concrete document, a generic logical document structure is specified, that describes the type of a document. The editing itself generates a specific logical structure that is valid for one concrete document and obeys to the generic structure.

- *Layout structure:* It specifies the structure directly, establishes the form of the document as printed or displayed after the formatting process has taken place. To enable the mentioned standardized automatic formatting process the layout of a document must be specified. This specification is called generic layout structure, whereas the structure generated by the formatting process of an actual document is called specific layout structure.

• *Content:* It is the basic information carrier and belongs to both the logical and layout structure. ODA is extensible for new content types in specifying content only in terms of an abstract interface to the ODA structures and it enables handling of content portions during the processing of the ODA structures. This is done by different attributes of content portions concerning all necessary aspects of their participation in logical and layout structures, editing, formatting and presentation.

Logical and layout structures are associated and a generic logical object refers to a generic layout object to specify how specific logical object should be laid out. The automatic formatting process scans through the specific logical structure of the document, traversing the layout styles defined in the generic layout objects to generate the specific layout objects.

The logical and layout structures are specified via attributes named subordinates to build the hierarchical structure and layout category refer to the generic layout objects. ODA also refer to bundles of attributes called styles that link the content portions to the basic logical and layout objects and set user defined references.

ODA document has different special standardization concerning attributes of the whole document and document transmission in various forms and Document Application Profiles (DAPs) specify the usage of ODA for specific application areas [4] including multimedia information.

ODA requires mainly the following aspects:

- Definition of interfaces for new content portions audio and video.
- Definition of a model for document content layout in time and its integration with the conventional layout in space.
- Definition of user interaction for the time dependent presentation of a multimedia document.
- Inclusion of non-linear structures into the whole document model.
- Extension of the addressing capabilities of the ODA model with respect to partial documents.

As an addition to the ODA standard an audio content architecture [5] and the HyperODA [6] activities have proposed extensions for temporal relationships, non-linear structures and fragmentation of documents. Handling of their temporal presentation in conjunction with spatial extent layout requires a revised layout process. Audio content has no spatial extent but has a location in a document; this location could be the position between other document parts that do have spatial extent or the document part to which audio is linked as an annotation. Similar to an image, video has an appearance in the spatial dimension.

B. SGML / HyTime

The Standard Generalized Markup Language has been designed for usage in environments where the logical structure has great importance and the layout remains a local matter. The author is responsible for content and structure whereas the publisher is concerned with layout considerations that usually require high quality standards. It also allows for structuring documents by inserting special markup in order to identify logical document parts and provide some relation between them. It does not bother about the application semantics of the identified document portions.

Hypermedia/Time-based Structuring Language [7] is an SGML application. Thus a HyTime document constitutes first of all an SGML document. The HyTime standard provides semantic specification governing basic constraints in the field of hypermedia documents.

SGML: This document consists of:

- A document type definition (DTD) that provides the specification of a certain document class. The DTD governs the logical and physical structure of the document; it defines element types and their relation to each other which forms a hierarchy. Nesting of elements must always be consistent with the DTD, it allows the document content to be of any format or coding. The DTD distinguishes between parsable character data defined by the standard and called non-SGML data that is unknown to SGML. Element have any number of attributes that are specified within the DTD and attributes in SGML are unstructured. It does not standardize the attributes semantics.
- A document that belongs to a certain document class specified by the DTD and the content of a document is enriched with readable markup that specifies
  - logical structure = element structure
  - attribute values of individual elements
  - entity references

SGML document is subdivided into several pieces called entities, which are internal or external. Internal provide a means by which document parts which appear several times, may be declared once in the DTD and can then be referenced several times in the document. External gives hints, where the content may be found. An entity may also be used as a container of data that is unknown to SGML. In such cases, the entity declaration should contain a notation specification which is complemented by a separate notation declaration intended to tell the application how to process the data.

Entities may be referenced by other entities which lead to a hierarchical entity structure. It is up to the application to define an appropriate entity structure. In many cases the DTD will be an external entity. This avoids transfer of the DTD of the document also possesses the affiliated DTD.

The logical structure represented by the element nesting and the physical structure represented by the entity declarations are independent of each other, thus an entity may contain any portion of the logical structure. It knows nothing about layout and consequently does not contain a layout structure. It is the task of a formatting application to provide layout specifications that apply to the elements defined in the DTD and every document may be formatted accordingly. Similarly applications may be defined for certain document type definitions [8], [9].

As content portions may be of any format or coding, document instances containing stills and audio or video sequences are within the scope of SGML which is applicable for specification of multimedia documents in a limited manner. However there is no concept of relating elements expressing temporal relationships and hence HyTime comes in.

HyTime: This document support hypermedia applications that are useful in most applications. It includes addressing aims to identify a certain piece of information within some source, linking facilities are to establish connections between pieces of information and alignment specification serves to describe relationships between information with some finite coordinate system.

HyTime is a construction kit of architectural forms; form represents a template of an SGML element declaration plus an associated attribute list. It defines its semantic; an application designer creates an appropriate DTD, selecting those architectural forms necessary to describe the intended information. Within a HyTime each element corresponds to an architectural form. Several elements may correspond to the same form and may omit ones that are not explicitly required and add private ones. Identification of the associated architectural form is achieved by a special attribute and private attributes require the application to know about the DTD designer's intention as their semantic cannot be derived from the standard.

It only supports generic facilities that may be applied to any information base so usage of it is not restricted to any specific information format or coding and applications do not have to support the complete standard as HyTime is built of modules. The location address module contains the mechanisms and HyTime offers in order to identify certain pieces of information with respect to the structure of the information base. A coordinate location scheme may be used, if measuring along some coordinate space is feasible. A location is identified with a two marker concept, the first marker defining the start of the location and the second one defining the extent, for this purpose HyTime offers dedicated architectural forms.

If the location to be addressed carries some name then namespace location scheme will be applied and finally if both of the former fail the semantic location mechanism may be used. Locations are addressed by some application specific construct; in general location elements are chained one defining the environment for the next. The last element provides a pointer to the identified portion of information which may then be used to establish relationships i.e., Hyperlink.

Hyperlink is used for building link connections where different architectural forms are supplied. These are distinguishing as:

1) contextual links, one link end location is the original location within the data stream.
2) independent links, connecting several link end locations.
3) aggregation links, treating multiple link end locations as one location.

The finite coordinate space (FCS) used for aligning pieces of information and HyTime does not know about the dimension of information objects as it does not know about format or coding. Thus HyTime places bounding boxes within the finite coordinate space that reference their content and is called an event. Content may be anything from text or graphic pieces to stills, audio and video sequences but placing is done with the two marker concept, either absolutely in coordinate space units or relatively referencing the markers of other events. Alignment in time and space is handled uniformly and a finite coordinate space is built of axes which measure in fractions or multiples of predefined international measurement units like the System International Second (SIsecond) or System International Meter (SImeter). Events may be grouped to event schedules. FCS may contain any number of event schedules [10], [11].

C. MHEG

Multimedia and Hypermedia Information Coding Expert Group (MHEG) [16]. In most cases people refer to the evolving standard as the MHEG standard, while MHEG originally named the standardization group occupied and MHEG has the status of a working draft. MHEG aims at specifying the coded representation of final form multimedia and hypermedia information objects. The standards specifications follow the object oriented model using related terms like class, message and method. However MHEG does not define any representation in an actual system implementation. Therefore realization of MHEG conformant systems may decide to follow a non object oriented approach.

MHEG distinguishes between basic objects and composite objects and in other part between input, output and interactive objects. Output objects contain some content and a projector that guides the presentation. Content may be text, graphic, audio etc. Each type has its dedicated basic output class. Input objects are to receive some information from the user. They contain some introduction of what reaction is expected from the user, specification of what kind of input is requested and an input projector, controlling the means by which input will be gained. Composite objects contain basic objects as well as other composite objects. They define relationships between their content objects with respect to hyperlink connections and temporal behavior. Interactive objects contain both input and output objects and are therefore inherently composite.

Each class defines a set of attributes and a set of methods that work on the attributes. All the classes form a hierarchy with subclasses inheriting attributes and methods from their parent class. The individual set of attribute values makes up an object instance. Basic output objects have one attribute carrying the object content, a set of attributes defining an encoding scheme and decoding parameters and both derived from the information type. Attributes of composite objects specify the relationship between all the content objects e.g. synchronization [15]. The standard provides a basic coding representation for each class using ASN.1 and an alternative representation using an SGML DTD.

V. EXISTING SYSTEMS

By implementing nonstandard prototype systems, many research teams have put considerable effort into making multimedia applications actually work, but very few systems contain the notion of multimedia documents. Many systems focus on other aspects than data modeling. However, two approaches give some description of how to structure multimedia data and do not describe the whole prototype system but emphasize on data modeling.

MINOS: Multimedia Information System (MINOS) [12] is a integrated multimedia documents containing text, image and voice portions and it follows the object oriented model in structuring data. Objects belong to classes that are part of subclass and superclass relations. Objects may be aggregated to more complex objects; there are dependent and independent objects. It distinguishes between a logical model and a physical model. The logical model represents the logical structure of documents, defining object classes like chapter, image or voice classes and their relationships and the physical model describes the presentation structure of a multimedia document on a physical device such as the screen of a workstation.

Furthermore there exists a mapping between logical structure and physical structure, specifying which components from one model are mapped to which components of the other. Thus a MINOS multimedia document consists of a logical document instance, a physical document instance and a mapping instance.

A logical document instance may consist of attributes, images, voice sections, text sections and annotations. There may be several relationships between objects of the different parts. A physical document contains sequences of pages that may be subdivided into rectangles and attributes may describe presentation properties with regard to rectangles or objects, ranking higher in the aggregation hierarchy. A mapping object can tell where a certain logical object has to be presented within a certain physical object.

MINOS contains one time dependent information type i.e. voice. Voice segments appear in two types with respect to presentation these are may either be associated with a voice indicator or may exist as a so called narration.

In the first case the voice segment is related to a logical object such as a paragraph or image. The indicator is displayed within the physical object while presenting the paragraph or image. The user may trigger voice reproduction by selecting the indicator. In the second case the segment is related to a page object. Reproduction is initiated by turning the associated page while displaying the document.

Multimedia Objects: Little and Ghafoor [13], [14] provide a synchronization and storage model for multimedia objects. The synchronization model is based on an enhancement of the Petri net model called Object Composition Petri Net (OCPN). An OCPN corresponds to an ordinary Petri net enhanced by durations and resources associated with the places of the net. Transitions in the net represent synchronization points, while places indicate processing of some media. Therefore, places have associated with the required presentation resource and the time required to output presentation data. They showed that the enhanced Petri net model is able to express arbitrary synchronization relationships.

The storage model tells how OCPN concept may be mapped to a logical database structure and a database scheme is implicitly defines the structure of multimedia objects. This structure consisting of terminal and nonterminal nodes and terminal nodes have various attributes including the medium type and the location of the data to be presented. Terminal nodes represent granularity of synchronization and nonterminal nodes have a number of attributes, pointers to the left and right child as well as temporal specifications. In synchronization model the binary hierarchical structure is sufficient to express any time relation between multiple objects that belong together. This motivates the two child pointers and temporal specification relates the child objects.

In many cases the binary structure becomes complicated and carries redundant information so the authors introduce meta nodes which are mostly the same as nonterminal nodes but may contain any number of pointers to child objects. As all the relationships between the children have to be clarified, multiple child objects meta objects also have a complex time specification and hence multimedia objects may be stored in the database by inserting valid pointers and synchronization specification.

VI. CONCLUSION

The basic concepts of data modeling within multimedia documents are discussed. Then the standardization activities in Open Document fields are surveyed and focusing on a description of the main ideas. A brief introduction of two non-standard approaches within the research community follows. Finally the synchronization aspect is analyzed, classifying standard and non-standard approaches and end the section concludes with a short discussion.

REFERENCES

[1] R. Steinmetz, J. Ruckert and W. Racke. Multimedia-Systeme. Informatik Spektrum, 13(5):280–282, 1990.

[2] Information processing - Text and Office Systems - Office Document Architecture (ODA) and Interchange Format, 1989. ISO/IEC 8613.

[3] Information processing - Text and office systems - - Standard Generalized Markup Language (SGML), 1986. ISO/IEC 8879.

[4] R. Hunter, P. Kaijser and F. Nielsen. ODA: document architecture for open systems. Computer Communications, 4 1989.

[5] A. Scheller, E. Wilde, et al. Definition of an Audio Architecture including a proposal for an ODA Audio Content Architecture. BERKOM-Project:"Multi-Media-Document in ISDN-B", 1990.

[6] HyperODA - Extensions for Temporal Relationships, 1992. ISO/IEC JTC 1/SC 18/WG 3/N2232.

[7] Information Technology - Hypermedia/Time-based Structuring Language (HyTime), 1992. ISO/IEC 10744.

[8] F. Charles, Goldfarb. The SGML HANDBOOK. Oxford University Press, New York, 1990.

[9] Joan M. Smith. Standard Generalized Markup Language and related standards. Computer Communications, 12(2), 4 1989.

[10] F. Charles Goldfarb. HyTime: A Standard for structured hypermedia interchange. Computer, 8 1991.

[11] R. Steven Newcomb, Neill A. Kipp, and Victoria T. Newcomb. The Hypermedia/Time-based Document Structuring Language "HyTime". Communications of the ACM, 11 1991.

[12] S. Christodoulakis, M. Theodoridou, F. Ho, M. Papa, and A. Pathria. Multimedia Document Presentation, Information Extraction, and Document Formation in MINOS: A Model and a System. ACM Trans. on Office Inf. Syst., 4(4), 10 1986.

[13] T. D. C. Little and A. Ghafoor. Network Considerations for Distributed Multimedia Object Composition and Communication. IEEE Network Magazine, 11 1990.

[14] T. D. C. Little and A. Ghafoor. Synchronization and Storage Models for Multimedia Objects. IEEE Journal on Selected Areas in Communications, 8(3), 3 1990.

[15] R. N. Jugele and V. N. Chavan, Synchronization Scheme In Multimedia and Hypermedia Application Development. Submitted to known International Journal in 2012.

[16] Information processing - Coded Representation of Multimedia and Hypermedia Information Objects (Working Draft), 1992. JTC1/SC29/WG12.

R. N. Jugele received his B.Sc. degree from Nagpur University and M.Sc. degrees in Computer Science from Marathwada University, Aurangabad, Maharashtra, India in 1991 and 1993 respectively. Currently, he is working as a Associate Professor in Department of Computer Science, Science College, Congress Nagar, Nagpur, Maharashtra.

Dr. V. N. Chavan Head, Department of Computer Science, S. K. Porwal College, Kamptee, Dist: Nagpur. Maharashtra. He is research guide in Computer Science subject in various universities and has wide knowledge in research field since last 22 years. He is a member of various bodies in Computer Science subject.