Preface

The need for multimedia information systems is growing rapidly in a variety of fields including business, manufacturing, education, CAD, CAE, medicine etc. Due to the diverse nature of multimedia data, systems designed to store, transport, display, and manage such data must have considerably more functionality and capability than conventional information management systems. Multimedia applications can be very complex as regards the number of objects involved, their transformations in the scope of an application, and the relationships among them. These features become even more salient with the advent and wide usage of the WWW as the main medium for information dissemination.

The issue of modeling, authoring, and presenting multimedia documents has attracted significant research and industrial efforts in the last years. In this book we present an integrated framework for Interactive Multimedia Documents (IMDs). We cover most of the stages of the life cycle of an IMD: *data modeling, authoring, verification and querying, execution and rendering*. The basis of the framework is a model for IMDs that widely covers the issue of interaction and spatiotemporal composition. According to the model, an IMD is defined in terms of *actors, events,* and *scenario tuples*. The actors (video, sound, image, text, and buttons) represent the participating media objects and their spatiotemporal transformations to align to the presentation requirements. The events are the interaction primitives and they may be atomic or complex. They are generated by user actions, actors' state changes, or by the system. The basic constituent of an IMD is the scenario, namely a set of scenario tuples. A tuple is a fundamental entity of functionality in the scenario conveying information about the event(s) that start (or stop) a set of synchronized media presentations.

Based on this IMD model we develop an *authoring* methodology, clearly identifying the stages of interaction and spatiotemporal presentation specification. During the authoring process, it is vital to provide the author with *verification* facilities to review the document design and thus avoid errors and improve the design quality. The verification has to deal with the temporal and spatial dependencies and constraints of the document. We propose a methodology for verification of the spatiotemporal content of an IMD. A related issue is the ability of an author to query an IMD as regards the spatiotemporal dependencies that are inherent to it. There the requirement for efficiency is apparent. We have worked on a spatiotemporal indexing scheme that efficiently processes queries with intensive spatiotemporal content.

The result of IMD specification under this approach is a declarative script, which has to be executed whenever an IMD session is initiated. The execution (*rendering*) of an

IMD may be a complicated task due to the multitude of events occurring and the potentially large sets of different presentation options.

We developed two approaches for IMD rendering, one based on a single-thread approach and another generic multi-threaded architecture that detects and evaluates events that occur in the presentation context and triggers the appropriate synchronized presentation actions. Both schemes are implemented.

Book Overview

The book is organized in chapters, each one describing one of the aforementioned issues.

In the *first* chapter, the issue of multimedia document modeling is addressed. Moreover, a review of the research efforts aiming at modeling the various aspects of multimedia documents is provided as well as a substantial presentation of current document standards (HYTIME, MHEG, and SMIL).

In chapter *two*, the integrated IMD model is presented. The modeling approach takes into account the various aspects of a multimedia database and applications. Namely: multimedia transformations, composition of media objects in space and time, events modeling, scenario definition. Indeed, an interactive multimedia application includes media objects modified accordingly and presented according to a predefined spatiotemporal sequence or according to some interaction. The participating media objects are rarely used as they are. In most cases parts of them (cropping procedure takes place) are transformed (spatially and/or temporally) according to the authors needs. The next stage is the definition of the application functionality. We distinguish between two types of functionality: the pre-orchestrated (i.e. the spatiotemporal presentation of the participating objects is predefined) and the interactive one (i.e. the flow of the application is dependent on the interaction actions that will occur). As for the pre-orchestrated case, the spatiotemporal composition specifications have to be defined, where as for the interactive one the events that will trigger the corresponding action have to be defined. The two aforementioned procedures will produce the overall application scenario and this has to be stored in the appropriate database.

In chapter *three*, we present the authoring methodology that results from the IMD model. The authoring procedure is carried out in three consecutive phases corresponding to the basic modeling primitives: i. selection and transformation of the media objects to participate; ii. definition of the events (atomic and complex) that the IMD session will consume; the author defines the atomic events that the current IMD will exploit for the scenario; iii. specification of the IMD scenario in terms of scenario tuples.

In chapter *four*, we present the verification tools and methodologies that are available in the context of our framework. The tools may be used both for the *prototyping* and the *verification* of multimedia presentations or spatiotemporal compositions in general. Emphasis is on the flexible definition of spatial and temporal relationships of the participating entities. The *verification* procedures are supported by multiple tools allowing designers to preview their applications, in various ways: *spatial layouts* of the application window, *temporal layout* of the application, indicating the temporal duration and relationships among the participating objects and *animation (rendering)* of the application.

In chapter *five*, we present a generic framework for transformation of the IMD declarative script to an algorithmic form, in order to obtain an executable form of the document. The declarative IMD script resulting from the authoring process, is translated into a set of procedures corresponding to the constituents of the scenario, namely: the events and the scenario tuples (including the synchronized presentation actions). The result of this process is a set of procedures specific to the IMD concerned. This set of procedures is integrated in a predefined IMD template, and the result is the specific algorithmic form of the IMD document.

In chapter *six*, we address the issue of rendering an IMD. Rendering is the process of enforcing the presentation specifications of the IMD scenario, i.e. when, where, for how long, and under what transformations each media object will be presented. Here, we present two implemented designs we have carried out. A single threaded generic one and a multithreaded one appropriate for the latest technological demands arising from the WWW. Both schemes carry out the following tasks: i. detect events generated by the system, the user, or the actors and subsequently evaluate these events against the start/stop event expressions of each scenario tuple; ii. activate the appropriate tuples asynchronously and perform synchronized presentation actions according to the scenario tuples' specifications; iii. handle exceptions.

In chapter *seven,* we present a scheme for managing IMDs that involve a large number of interrelated objects. We propose an indexing scheme based on spatial indexing techniques. Thus fast retrieval of queries regarding spatiotemporal queries is feasible. Moreover spatial or temporal application layouts may be obtained.

Finally, in the appendices there is an extended example application as it is represented through the model proposed in previous chapters (Appendix A) and the formal definition of the model in terms of BNF grammar (Appendix B).

Acknowledgements

The procedure for developing ideas, making them concrete, verifying, and communicating them to the research community is a long and arduous procedure. During this procedure I was not alone. There were people (friends and colleagues) whose contributions I would like to acknowledge.

First, I want to thank the following colleagues who contributed with their implementation skills to the development of the concepts into running prototypes: Y. Stamati, M. Trafalis, D. Tsirikos, Markousis, T. Mouroulis, and Y. Politis (in chronological order).

Also I would like to thank D. Tsirikos, T. Markousis, and I. Mirbel for their kind effort in reviewing preliminary versions of parts of the book and contributing their comments, corrections, and insight.

I should also mention my co-authors in the publication history that served as the basis for this book: C. Mourlas, M. Hatzopoulos, Y. Theodoridis, S. Boll, and Y. Stavrakas (again in chronological order).

The contribution of Prof. T. Sellis to my evolution in the research domain, after my Ph.D., has been critical. I am grateful to him for offering me the friendly environment and the resources of his laboratory and for his manifold support in the various aspects of what we call research work.

Also I would like to acknowledge the support and contribution of Dr. T. Rakow for giving me the chance to work in GMD/IPSI and also for his contribution to the initiation of this work.

I am grateful to my parents (Ignatios and Angela) for their long term and no-matter-what support so far. They both have devoted to me more resources than I could expect.

Finally, I have to acknowledge Victoria’s support, patience, and love throughout this effort.