TECHNICAL RESEARCH REPORT

CHIMP: A Framework for Multimedia Documents

by K.S. Candan, B. Prabhakaran, V.S. Subrahman ian

T.R. 97-45

Sponsored by
the National Science Foundation
Engineering Research Center Program,
the University of Maryland,
Harvard University,
and Industry

CHIMP: A FRAMEWORK FOR MULTIMEDIA DOCUMENTS*

K. Selçuk Candan
B. Prabhakaran
V.S. Subrahmanian
Department of Computer Science,
Institute for Advanced Computer Studies &
Institute for Systems Research,
University of Maryland,
College Park, Maryland 20742.
Email: {vs,candan,prabha}@cs.umd.edu.

ABSTRACT

A multimedia document consists of different types of media objects that are to be presented at different instants of time and for different durations. The media objects may be stored in computer systems connected by a network, thereby rendering the document distributed. The authors may wish to collaborate in a distributed manner to edit a multimedia document. Hence, an authoring system needs to identify a retrieval schedule that describes the time instants at which the objects have to be retrieved as well as the logical paths to be followed by the objects. In this paper, we consider a collaborative multimedia document authoring system with the above mentioned features. We propose a difference constraints based temporal specification for the multimedia document. This approach allows us to generate flexible schedules for retrieving objects over the computer network. This flexible retrieval schedule can handle variations in system parameters such as available network throughput and buffer resources.

INTRODUCTION

This paper is part of the Collaborative Heterogeneous Interactive Multimedia Platform (CHIMP) project, which has the goal of studying the technical aspects of collaborative multimedia document authoring and presentation, as well as building a system based upon these results.

Suppose we consider a team of individuals jointly *authoring* a multimedia document. In order to successfully author such a document, the authors must:

- Identify the objects (e.g. audio objects, video objects, text objects, etc.) that will be part of the authored multimedia document. This is studied in detail by Marcus and Subrahmanian[9, 10] who showed that a fragment of Datalog queries may be used to identify objects of interest.
- Specify how these objects should be presented to an end-user wishing to view the final multimedia document. This specification includes, amongst other things, the temporal constraints used to generate the presentation. In contrast to previous work[8, 3, 10] that describes how arbitrary constraints may be used to specify presentations, CHIMP benefits from the use of a small class of constraints called difference constraints that are adequate for specifying very flexible presentations.

In addition to the presentation constraints, when a set of objects are scattered across the network, we need to generate a *Retrieval Schedule* that specifies how the CHIMP retrieval engine will retrieve the desired objects from other locations by interacting with remote servers. This too, constitutes parts of the CHIMP project, and includes the study of resource reservation

*This research was supported by the Army Research Office under grant DAAH-04-95-10174, by the Air Force Office of Scientific Research under grant F49620-93-1-0065, by ARPA/Rome Labs contract Nr. F30602-93-C-0241 (Order Nr. A716), and by an NSF Young Investigator award IRI-93-57756.

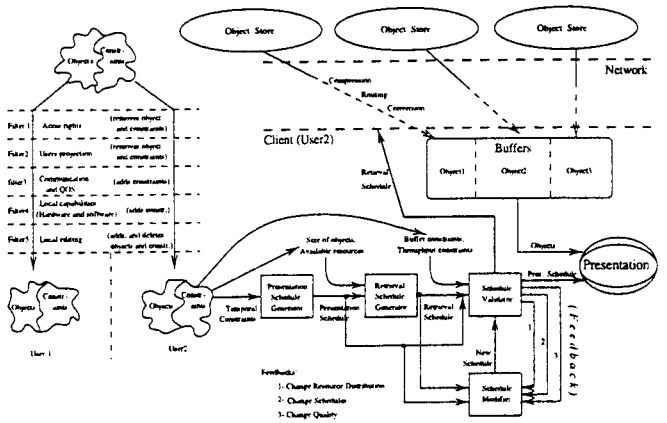

Figure 1: Collaborative Multimedia Document Authoring Architecture

algorithms, as well as servers that can be used for networked delivery of multimedia objects keeping in mind, the available resources (which include bandwidth resources, buffer resources, available viewing formats at different nodes, etc.). CHIMP is currently implemented on the SUN/Unix platform, and includes some, but not all the services listed above.

SYSTEM ARCHITECTURE

Figure 1 shows the overall architecture of the collaborative multimedia document authoring system. In this section we briefly describe the components of this architecture.

Temporal Specifications: In our architecture, we use flexible temporal relationships where the time instants and durations of presentations of objects are allowed to vary as long as they preserve certain specified relationships. These relationships are specified using difference constraints. Flexible temporal specifications imply that the presentation of the multimedia document can can have a *set* of presentation schedules that satisfy the given temporal constraints. Each member of this presentation schedule set describes one possible *view* of the multimedia document.

Filters for Accessing the Multimedia Document: Depending on their interest and their needs, authors editing a document can have different *filters* of the same document. These filters describe the specific portions of the multimedia document that can be viewed or edited by an author. The filters that can possibly be applied to a document depend on (1) user access rights, (2) local system capabilities, and (3) user's projection of the document.

Conversion of Objects to Appropriate Formats: The multimedia document can involve objects of different formats. The local system may or may not have the capability to present an object using the same format in which it is stored. In this case, objects might have to be converted from the stored format to the desired format. This conversion can be done by the system that stores the object or by intermediate system(s), i.e, the object can be shipped to intermediate system(s) that convert the object to the desired format, and then transferred to the author's system.

Presentation Schedule Generator: This component picks a presentation schedule that satisfies the given difference constraints based temporal specification.

Retrieval Schedule Generator: Depending on the temporal specification and the access filters, a schedule for retrieving objects has to be identified. The retrieval schedule specifies the time instants at which the author's system should make a request for retrieving the objects that compose the multimedia document. This retrieval schedule depends on (1) the presentation schedule, (2) the size(s) of the object(s), (3) throughput (or the bandwidth), and delay of the communication channels, and on (4) the buffer availability. The flexibility in the temporal specification of the multimedia document helps in deriving a flexible retrieval schedule that can adopt to the system dependent factors such as available throughput and buffer resources.

Schedule Validator: Given a temporal schedule, system constraints, and a retrieval schedule, this module checks the validity of the generated retrieval schedule based on the application dependent and system dependent constraints. If the schedule is valid with respect to the specified constraints, then the *validator* returns them as the final solution. However, if the schedule does not satisfy the system constraints, this module suggests modifications that can be made to the current solution in order to satisfy the system constraints.

Schedule Modifier: This module modifies the *current* solution for retrieval and presentation schedules based on the suggestions made by the schedule validator module. The modified solution is given back to the validator module to check against the system constraints (throughout and buffer constraints). This process of

solution-feedback and validation is repeated till a valid schedule is generated. In case a valid schedule cannot be arrived at, then the best schedule found so far can be used as the solution. Objects whose schedules do not satisfy the system constraints can be dropped from the presentation, provided the viewer agrees to it.

RELATED WORK

One of the major differences between multimedia information and text information is the temporal nature of video and audio. In [1, 2], James F. Allen proposes an algorithm based on intervals which, given a set of relationships among the intervals in the database, can infer the relationships among all intervals. Due to its simplicity, this model formed the basis for many multimedia authoring systems. However, the timeline model is too rigid for many purposes; users must specify exactly when an event occurs.

In [8], Little and Ghafoor propose an interval based model based on object composition petri net (OCPN), a modification on timed petri net model. Prabhakaran and Raghavan, then, extend the OCPN model to handle similar user interactions [11]. Due to its higher flexibility and simplicity, this model proved to be very useful in multimedia authoring and multimedia simulating applications. Li, Karmouch, and Georganas, on the other hand, propose a Time Flow Graph (TFG) based model to model temporal presentation scenarios [7]. Their model, again, is based on intervals, and it can represent fuzzy presentation scenarios. In their framework, fuzziness can be due to unknown object presentation durations or unknown relative timing of the events.

Buchanan and Zellweger use temporal constraints to specify temporal relationships among the multimedia objects within a document in their system called FireFly [3, 4]. Their work is based on the use of difference constraints for the specification of temporal information, like ours. However, they use the more expensive simplex algorithm (well known to take exponential time in the worst case) to solve for the presentation schedules. In our work, we use highly efficient graph algorithms for this purpose.

CONCLUSION

In this paper, we have presented a unified framework for multimedia document authoring and presentation that facilitates the articulation and use of flexible temporal specification, access filters, local editing, object format conversions and flexible retrieval schedules for presenting objects over a computer network. Our main contributions are the following:

- Difference constraints based flexible temporal specification that allows the following :
  - Generation of flexible presentation schedules to adapt to the system requirements.
  - Handling of inconsistencies in temporal specifications.
  - Incremental modification of temporal constraints to facilitate local editing and access filters.
- Flexible retrieval schedule generation methodology that handles :
  - Variations in system parameters such as available network throughput and buffer resources.
  - Modifications of the presentation schedules due to local editing and access filtering.

The main difference between our work and that of others is that we utilize the flexibility in the temporal presentation specification to generate retrieval schedules which in turn can handle variations in network throughput and buffer resources.

References

[1] J.F. Allen (1983) *Maintaining Knowledge about Temporal Intervals*, Communications of the ACM, vol. 26, no. 11, pp. 832-843, November 1983.

[2] J.F. Allen (1984) *Towards a General Theory of Time and Action*, Artificial Intelligence, 23, pp. 123-154.

[3] M.C. Buchanan and P.T. Zellweger (1993) *Automatic Temporal Layout Mechanisms* ACM Multimedia 93, pp. 341-350.

[4] M.C. Buchanan and P.T. Zellweger (1993) *Automatically Generating Consistent Schedules for Multimedia Documents*, ACM/Springer-Verlag Journal of Multimedia Systems, vol. 1, no. 2, 1993.

[5] K. Selçuk Candan, V.S. Subrahmanian, and P. Venkat Rangan (1996) *Towards a Theory of Collaborative Multimedia*, IEEE International Conference on Multimedia Computing and Systems, Hiroshima, Japan.

[6] K. Selçuk Candan, B. Prabhakaran, and V.S. Subrahmanian (1996) *CHIMP : A Framework For Supporting Distributed Multimedia Authoring and Presentation*, Proceedings of ACM Multimedia '96 Conference, Boston, November 1996.

[7] L. Li, A. Karmouch and N.D. Georganas (1994) *Multimedia Teleorchestra With Independent Sources: Part 1 and Part 2*, ACM/Springer-Verlag Journal of Multimedia Systems, vol. 1, no. 4, February 1994, pp.143-165.

[8] T.D.C. Little and A. Ghafoor (1990) *Synchronization and Storage Models for Multimedia Objects*, IEEE J. on Selected Areas of Communications, vol. 8, no. 3, April 1990, pp. 413-427.

[9] S. Marcus and V.S. Subrahmanian. (1995) Towards a Theory of Multimedia Database Systems. in: *Multimedia Database Systems: Issues and Research Directions* (eds. V.S.Subrahmanian and S. Jajodia), Springer Verlag.

[10] S. Marcus and V.S. Subrahmanian. (1996) Foundations of Multimedia Database Systems, JOURNAL OF THE ACM, Vol. 43, 3, pps 474–523.

[11] B. Prabhakaran and S.V. Raghavan (1994) *Synchronization Models For Multimedia Presentation With User Participation*, ACM/Springer-Verlag Journal of Multimedia Systems, vol.2, no. 2, August 1994, pp. 53-62. Also in the Proceedings of the First ACM Conference on MultiMedia Systems, Anaheim, California, August 1993, pp.157-166.