Communication Facilitators for a Distributed Collaborative Engineering Environment

by

Karim Mohie El Din Hussein

B.S., Electrical Engineering and Computer Engineering
Carnegie Institute of Technology, Carnegie Mellon University (1992)

Submitted to the Department of Civil and Environmental Engineering
in partial fulfillment of the requirements for the degree of

Master of Science
in Civil and Environmental Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1995

Author ....
Department of Civil and Environmental Engineering
May 26, 1995

Certified by ....
Feniosky Peña-Mora
Postdoctoral Associate
Thesis Supervisor

Certified by ....(FPM)...................
Duvurru Sriram
Principal Research Scientist
Thesis Supervisor

Accepted by ....
Joseph Sussman
Chairman, Departmental Committee on Graduate Students

Communication Facilitators for a Distributed Collaborative Engineering Environment

by

Karim Mohie El Din Hussein

Submitted to the Department of Civil and Environmental Engineering
on May 26, 1995, in partial fulfillment of the
requirements for the degree of
Master of Science
in Civil and Environmental Engineering

Abstract

Availability of professionals and communication of corporate expertise are among the key competitive factors in a modern corporation. However, due to the growing size and geographic dispersion of corporations, making optimal use of their human resources and expertise is becoming increasingly difficult. Since a significant portion of an employee's time is spent in meetings or conferences to coordinate group efforts, conference management is a critical component of a collaborative engineering effort. This thesis describes, CAIRO (**C**ollaborative **A**gent **I**nteraction control and synch**RO**nization system), a system for managing participants in a distributed conference. We have drawn from various models of group interaction and social communications theory in order to develop CAIRO. While most conference systems have focused on the technical issues of communicating information between computers, we have also emphasized the role of the computer as a mediator and conference control mechanism. CAIRO provides both media synchronization, i.e. insuring that all information conveyed from one participant to another is synchronized, and agent synchronization, i.e. insuring effective structuring and control of a conference.

Keywords: Conference Management, Collaboration, Multimedia Synchronization, Conference Control, Floor Control.

Thesis Supervisor: Feniosky Peña-Mora
Title: Postdoctoral Associate

Thesis Supervisor: Duvurru Sriram
Title: Principal Research Scientist

Acknowledgments

I would like to thank Professors F. Peña-Mora and D. Sriram for their support and guidance throughout this study. I have particularly appreciated the latitude and initiative they have granted me in pursuit of my research. I would also like to thank Professor Madnick and Dr. Siegel who have expanded my research horizons and provided me with financial support throughout my stay at MIT.

I would then like to thank all at the Department of Civil Engineering, the Sloan School and at MIT at large who have contributed in making of my stay in the Institute a most enlightening experience. I have met here, among faculty and students, a special breed of dedicated people who will always remain a source of inspiration.

Special thanks to Ziad Kanaan, Hisham Kassab, Ahmed Mitwalli, Amr Mohammed and Amine Tazi Riffi who were always there to assist me in difficult times. I would also like to thank my undergraduate roomate Michael Wu for the valuable insights he has provided through our continuous e-mail and phone conversations. A sincere thanks to the members of the MIT Arab Students Organization and the B-entry Bunch for making my stay at MIT pleasant and exciting.

To my parents, Mohie El Din Hussein and Mervat Badawi, thank you for all the support throughout my life. They have taught me the value of education and have provided me with the means and unconditional support I needed to achieve my goals. They forever will be the inspiration.

Finally, I would like to dedicate this thesis to my grandmother Amina ElKosheiry who is unfortunately unable to share with me the pride and joy in my accomplishments.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

There will only be two kinds of businesses in the 90's, the quick and the dead.

Andy Grove, CEO, Intel Corporation.

Availability of professionals and communication of corporate expertise are among the key competitive factors in a modern corporation. However, due to the growing size and geographic dispersion of corporations, making optimal use of their human resources and expertise is becoming increasingly difficult. Corporations have expanded geographically to respond to more specialized and localized consumer demand. This trend has also forced the restructuring of organizational structures towards a decentralized model. Furthermore, the need to create customized products for specific markets and the need to keep up with technological change has shortened product development cycles, thereby adding serious temporal constraint on corporations. Trends towards decentralization and customization combined with the growing complexity of modern products have strained existing manufacturing and design processes. Several new paradigms – such as total quality management, agile manufacturing and just-in-time systems – are increasingly being adopted by corporations to adapt to modern market conditions. However, these paradigms address manufacturing processes and do not address the product design process.

The traditional product design process involves several professionals working in a shared work space and coordinating activities through continuous design discussions. This work process is no longer applicable to the realities of modern product development. Current products are increasingly complex and involve a large number of professionals from multiple disciplines. These individuals can not realistically be located in a single work space. Due to the progressive complexity of the design tasks project coordination has become more critical yet increasingly more difficult to do through meetings since the professionals are distributed. This results in design delays and poor response to consumer demand. Furthermore, the lack of communication and coordination results in inefficiency and wasted design effort. The current design also does not properly accommodate the distributed nature of the modern corporation.

Improving the design process necessitates the increasing use of information technology. The Distributed Integrated Computer-aided Engineering (DICE) project [Sriram and Logcher, 1993] provides extensive support for an integrated computer based design work-space (See Section 1.1). The DICE system allows distributed access by a multidisciplinary design team to a shared work space, thereby eliminating one of the key difficulties in modern product design. However, communication and coordination among team members was not addressed by the DICE system.

Coordination and communication through meetings has been an effective strategy for coordinating group collaboration on projects with moderate complexity. However, the current meeting process significantly degrades productivity when the design problem addressed is complex and/or when the design team is distributed. Hence, a reformulation of the meeting process is necessary for corporations to remain competitive (See Section 1.2).

This thesis is a description of CAIRO (**C**ollaborative **A**gent **I**nteraction control and synch**RO**nization system), a component of the DICE effort that provides for distributed meetings over a computer network. CAIRO provides effective communication and coordination tools for a distributed environment. The system addresses the problems with communication and coordination apparent in current meeting structures.

The DICE and CAIRO systems combined provide a complete distributed replacement to the prevalent product design processes.

1.1 Overview of the DICE Project

The aim of the DICE project, under which this research was undertaken, is to provide a distributed system to manage collaborative engineering work. Since the focus of DICE has been on Civil Engineering related applications the client applications have generally focused on this field. At the core of the DICE system lies a shared work-space manager (SHARED [Wong, 1993, Wong and Sriram, 1993]), and a design rationale capture and management system (SHARE-DRIMS [Peña Mora, 1994, Peña Mora et al., 1995]). Several client applications have been developed on top of the SHARED framework: GNOMES [Sriram et al., 1991] (a non-manifold geometric modeler), CONGEN [Gorti and Sriram, 1993] (a concept generator), and COSMOS [Sriram et al., 1994](a knowledge base design tool). The DICE system has been designed to support multiple views of the work space knowledge. This allows the collaboration of architects, structural engineers, construction managers and clients in an integrated and coordinated environment.

Most of the research effort within the DICE project has been concentrated in the representation and presentation of highly structured information, however free and informal information is clearly a critical element of any collaborative effort. Project members must discuss and negotiate revisions in a more personal fashion so as to appropriately update the final decision plan.

1.2 Reformulation of Meetings

Engineers in a design group typically spend more than 35% [Doyle and Straus, 1993] of their workdays in meetings with management or among themselves to generate ideas, debate design decisions and plan tasks. This process is characterized as informal communication. Physical group meetings are typically inefficient and require significant preparation in order to communicate ideas among the participants. Each participant is removed from his typical work environment and does not have access to his typical tools. The purpose of this thesis is to propose a better informal communication system (CAIRO) that would replace the existing physical meeting paradigm and remove the constraint of collocation on the members of a team.

The physical meeting paradigm is centered around a meeting event. The meeting event must be scheduled and the work process must be disrupted in order to coordinate group effort. However, in order for meetings to become most effective they must become a part of the work process and not an independent event. Some benefits of such a reformulation are enumerated below:

1. Each member of a design team has more to offer since he is next to his tools.

2. Professionals within the team are accessible on demand and provide instant feedback to design decisions.

3. The meeting process would not be a discontinuity that would decouple the members from the work process.

In order to integrate meetings within the design process the essence of meetings must be understood. A meeting involves an infra-structure that collects the design team (eg. a meeting room), a mutual understanding among the members, and a coordination of a process to achieve a common goal. Hence, these processes must be replicated in order to provide the functionality of a modern meeting.

In the context of a distributed collaborative environment the functionality described above can be interpreted as: collocation, cooperation and coordination services. Each of these critical elements of group collaboration is detailed below:

Collocation involves dealing with the network infrastructure to provide seamless communication among distributed clients in a conference. This layer provides naming services to identify client locations as well as interaction with the network protocols to transmit data across the network between the clients.

Cooperation involves the sharing of information among clients in a team. Due to differences in software and capabilities of the various clients, translations need to be performed in order to provide a coherent view of the data among the clients.

Coordination involves control of the workflow and communication process. This allows for efficient control mechanisms to coordinate group effort. The coordination layer acts as a "virtual manager" of the conferring clients.

The research discussed in this thesis is primarily focused on the coordination layer of services. Conference control mechanisms implemented in CAIRO provide tools for structuring teams in a collaborative effort. Furthermore, CAIRO provides enhancements to the collocation layer to allow for synchronized communication of multiple media across the network.

1.3 The CAIRO Solution

Providing visual and verbal communication services among distributed individuals is commonly referred to as distributed (tele, video or computer) conferencing. The concept of distributed conferencing systems has existed for over two decades. Significant developments have been made in communication infrastructure to allow for real time interactions over computer networks. Various tools have been created to allow for distributed conferencing over a network of computers (eg. AT&T RAPPORT^{TM} [Ahuja and Ensor, 1992], Invision^{TM} [InVision, 1994]). However, few systems address much more than the technical issues involved in transmitting the media across network channels (see [Egido, 1988] for a detailed discussion of the failures of video conferencing to support group work). Their main emphasis has been on acquiring audio, visual, and textual information, transmitting it across a network to the appropriate destination and then playing it back. Several studies [Egido, 1988] indicate that video images do not provide adequate communication queues for effective coordination of meetings. Hence, the computer can not be used simply as a medium for communication much as a telephone line. The inherent processing power of the medium should be exploited to enhance the communication process.

The CAIRO system provides extended support for conferencing in a distributed collaborative environment. The work described herein addresses solutions to synchronization problems among the various media (i.e. audio, video, text) used by participants in a conference as well as conference collaboration control mechanisms. Media synchronization is essential to retain the meaning of the transmitted information (eg. imagine a lecturer pointing at a diagram five minutes after he/she has finished discussing the diagram). CAIRO includes extensible collaboration tools encoded with various coordination methodologies to allow the computer to take a more active role in the control of the mediation and collaboration process. These tools provide a basis for the moderation of a conference and for effective floor control.

1.4 Thesis Overview

A background study of literature associated with distributed conferencing is presented in Chapter 2. This chapter concludes with the requirements identified for a distributed conferencing system for group collaboration. Chapter 3 provides a survey of existing commercial and academic distributed conferencing tools. Chapter 4 describes the system architecture of our communication tool (CAIRO) based on the requirements outlined in Chapter 2. The multimedia synchronization mechanism in CAIRO is further elaborate in Chapter 5. Chapter 6 presents the CAIRO conference control mechanism that enables computer mediation of conferences. Chapter 7 describes a sample bridge design scenario that illustrates the use of the CAIRO tools. Finally, Chapter 8 provides concluding remarks and contributions as well as a guideline for future work.

Chapter 2

Conferencing Requirements

A distributed computer conferencing system spans two diverse disciplines, i.e. social communication theory and computer communication theory. The multidisciplinary nature of conferencing systems is shared with a broader group of collaborative computer systems commonly referred to in the literature as "groupware". "Groupware" has been defined as computer-based systems designed to support multiple users engaged in a common task, thereby providing an interface to a shared environment [Ellis et al., 1991]. The major classes of groupware include videoconferencing systems (eg. AT&T RAPPORT^{TM} [Ahuja and Ensor, 1992]), shared document managers (eg. Lotus Notes^{TM} [Marshak, 1990]), shared work-space managers (eg. DICE), as well as process and task coordination tools. Key aspects of groupware include: the underlying technologies necessary to enable communication; models for group dynamics and social interaction; and effective coordination of group tasks. CAIRO is a "groupware" system designed to enable informal real-time multimedia communication among group members as well as provide coordination of group meetings. Therefore, CAIRO draws on research in technical issues (Section 2.1) related to multimedia transmission over a network as well as social issues (Section 2.2) related to group coordination and collaboration control.

2.1 Technical Issues

Information exchanged in a shared environment is comprised of a variety of media (i.e. multimedia). Typical exchanges between members of a group involve speech, gestures, documents and sketches. Such interactions occur in real time, as in a meeting, or offline, in the form of memos and more recently e-mail. This thesis focuses on the real time aspects of geographically distributed group interaction. Real time interaction is inherently taxing on both system and communication resources. Furthermore, the multimedia nature of human interaction necessitates a synchronization mechanism between the media channels to preserve the time dependence of the initial user input (see Section 2.1.2). For example, consider watching a movie where audio and video do not match (commonly referred to as "lip-sync"). The lack of synchronization can prove to be irritating as well as highly confusing if there is a significant delay between the two channels of communication.

A real time conferencing system is highly reliant on the available network infrastructure. Although, many advanced protocols such as ATM (Asynchronous Transfer Mode) and BISDN (Broadband Integrated Services Digital Network) have been proposed, the Internet remains the prevalent high bandwidth network today. The CAIRO system is based on the internet (see Section 2.1.1) and its underlying TCP/IP protocols. A major difference between current networks and future networks is the determinism of the network. Networks based on ATM will be deterministic (i.e. will have prespecified packet delay times) which greatly simplifies the communication subsystem in the multimedia communication facilitator proposed in this thesis. However, CAIRO assumes that the underlying network is non-deterministic and methods have been developed to accommodate this inadequacy which are based on real time scheduling techniques (described in Section 2.1.2).

2.1.1 Distributed Networks - Internet

The Internet is a collection of interconnected nodes (machines) that interact via a common protocol that is TCP/IP [Comer and Stevens, 1993]. Due to the nature of the protocol as well as the packet transmission and routing mechanisms prevalent on the internet, the internet is a non-deterministic network. Hence, interpacket arrival time is unpredictable due to varying network traffic. In a real time application – an application with prespecified time dependence – such random delay patterns can render the application useless. Insuring real time communication via the internet requires a series of delay compensation techniques discussed within this thesis. These heuristics reduce the amount of variability in the underlying network as well as provide the end user with near real time performance.

2.1.2 Real Time Scheduling

Synchronization of the various media inherent in a multimedia conference requires real time scheduling support by the conference support tools. Most current operating systems[1] do not provide adequate support for real time scheduling. Real time system theory [CMU S.E.I., 1993] addresses the scheduling of multiple independent channels or streams of data. These channels may have unpredictable arrival rates, although they must be subject to specific timing constraints (see Figure 2-1). Real time scheduling assures that all media channels are communicated within a given time period or frame, eliminating all "lip-sync" effects. Due to the possibility of losing packets or delays in packet transmission by the medium, a queuing mechanism is required to enforce the real time constraints.

Real time (RT) systems are commonly classified as hard or soft real time systems. Hard RT systems have critical deadlines that must be met, otherwise a catastrophic system failure would occur (eg. an aircraft control system). On the other hand, in soft RT systems, it is undesirable to miss a deadline. However, it is not catastrophic to system operation if some deadlines are missed. A conferencing system is a soft RT system since some video and audio frames may be dropped without significant consequences to the overall performance of the system.

[1]Notable exceptions are MACH RT and RTOS.

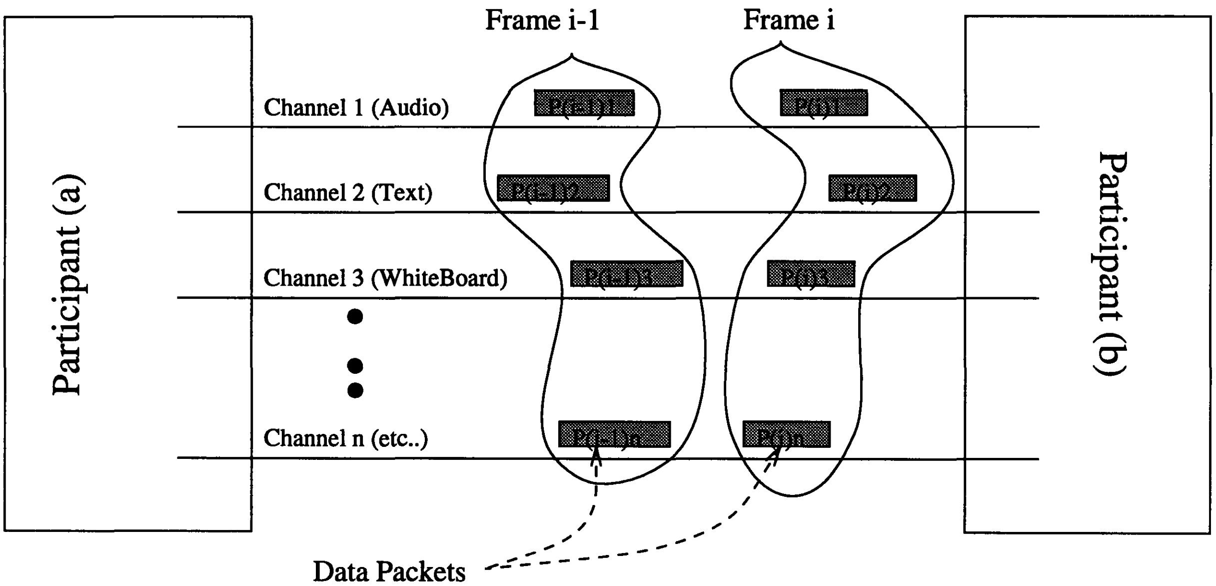

Figure 2-1: Synchronization issues between two conference participants

2.2 Social Issues

The emergence of a variety of communication tools has required a deep examination and study of human social behavior. Various models have been proposed to describe human social interaction. A firm understanding of these models is a prerequisite to designing effective tools for group or team work, due to their great interaction with and support of humans working in teams.

Significant research efforts have been undertaken on group behavior by social scientists, organizations and consulting firms [Ellis et al., 1991]. The efficiency of an organization's groups and teams is critical to the performance of the organization, hence, much time and effort has been spent determining effective methods to coordinate group effort. Models of speech acts, meeting cycles, and group life cycles are among the key conceptual frameworks that assist in understanding group dynamics. These models help define the mechanisms that must be supported by a conferencing tool in order to support group interaction.

2.2.1 Speech Acts

Searle's theory of speech acts which has been commonly referred to as the language/action approach has been widely used for analyzing communication by individuals within a group [Searle, 1969]. A speech act is a basic unit of communication that provides the context for the following data. There are three main categories of speech acts: propositional, illocutionary, and perlocutionary. Propositional acts are simply expressions of facts such as "The world is round" or "Pre-stressed concrete carries a higher load than regular concrete." Illocutionary acts consist of expressions that imply an intention, eg. "I will write a memo to my boss" or "I will finalize the bridge design by tomorrow." This form of act implies a commitment to a future behavior by the party performing the act. Perlocutionary acts produce an effect on the feelings, attitudes or behavior of the receiver. They are in the form of requests, orders or suggestions such as "Please pass me the salt" or "Please recalculate the joint stresses." The analysis of typical speech acts provides the basis for designing a user interface in a conversation based groupware tool, since it aids in the categorization of the information flowing from one participant to another. The categorization of conversation can be used as an indexing mechanism for the proceedings of a meeting and also enable the tracking of a specific member's contribution.

2.2.2 Meeting Cycles

The most basic structure for group coordination and interaction is a meeting. Meetings among individuals involved in product design tend to follow a similar cycle. Researchers at Xerox PARC have isolated three stages in a meeting cycle to aid in the development of their groupware product Cognoter (a collective presentation preparation tool)[Stefik et al., 1987]. The three stages consist of *brainstorming*, *organization* and *evaluation*. During the *brainstorming* stage various ideas generated by members of a team are laid out on a shared work-space (eg. a chalkboard). The second stage, *organization* consists of extracting the essential ideas and grouping and sequencing the various ideas presented. Finally, during *evaluation*, the ideas are further refined and tasks assigned to the members of the team. In the Cognoter model, brainstorming is carried out on a loosely controlled shared board (no erasing is allowed). During the *organization phase*, there is strict control on the meeting and only one person is allowed to access the board at any one time. At the final stage, when the tasks have been appropriated to the participants, each individual can refine his/her section and accept suggestions from the other participants. The Cognoter model in addition to the background research on meeting cycles suggest that interaction among members of a group or team varies as the meeting or project progresses. The control structure for a conference among these members must also evolve accordingly. Thus, a conference cannot have a static control structure but rather must be allowed to evolve as the needs of the participants evolve.

2.2.3 Group Life Cycles

Cole and Cole describe the cycle of group formation in [Cole and Nast-Cole, 1992]. This cycle consists of five stages: *forming*, *storming*, *norming*, *performing* and *adjourning*. Understanding the dynamics of each stage is critical for the realization of a distributed group interaction system since they each have a distinct form of conversational interaction. Furthermore, each stage involves a distinct authority structure. In the *forming* stage, members get to know each other and the tasks assigned to them. *Storming* involves the definition of roles within the group. At this stage, significant tensions may arise between the members as authority is asserted by a few of them and subtasks are determined. At the *norming* stage all roles are settled and the group focuses more intensely on the priorities of subtasks as well as procedures and methods to tackle them. The *performing* stage is when real work gets done, goals are achieved and the group becomes productive, energetic and effective. The group finally loses its structure in the *adjourning* phase when the work is completed, the group is reorganized or the members are assigned a different mission. During this wrapping up stage, groups reflect on their learning experiences and document their work to retain it in corporate memory. It is clear from the above description that the form of the interaction among members varies significantly in the five stages. The conversation among the members varies from chaotic and informal (*forming* stage) to a more structured and focused form (*performing* stage). Furthermore, at each stage authority and control structures are reformulated. Therefore it is critical that a groupware tool provide the flexibility to adapt to the various situations.

2.3 Summary of Requirements

The social models described in Section 2.2 provide a sound basis for the development of an effective communication tool that would easily fit into the accepted social structure. Speech acts theory provides a basis for understanding the context of conversation. The meeting cycles theory provides an overview of typical engineering approaches to group problem solving. Finally, the group life cycle model provides insight into the organization of teams and their evolution. Furthermore, Section 2.1 provides an overview of the technical constraints within which the communication tool must operate. Both the social and technical constraints contribute to the necessary list of requirements for an effective distributed informal communication tool enumerated below:

(i) Multiple media channels are required since group communication is generally comprised of audio, textual, and visual data.

(ii) Multimedia channel synchronization is essential due to random delays inherent in the underlying network.

(iii) A conference control mechanism is required to provide efficient group interaction.

(iv) The system must be adaptable to different conference styles (from informal, unstructured conversation to a stringent and formal conversation control mechanism).

(v) Ability to support groups in the various stages of formation, i.e. the ability to have hierarchically structured groups that are easily expandable.

(vi) Ability to retain group memory to build corporate experience as specified by the *adjourning* phase in the group life cycle.

Chapter 3

Survey of Computer Conferencing (Studies on Collaboration Tools)

The emergence of high speed communication networks and improved visualization techniques has laid the foundation for computer based collaboration. Various collaboration tools have been developed by academic institutions, office system manufacturers and communication companies. In this chapter, the major conferencing systems that have been developed are reviewed.

The review is not limited to distributed conferencing systems but also includes advanced electronic meeting systems. Distributed conferencing systems are tools used to simulate a meeting environment among geographically distributed participants. Electronic meeting systems (EMS), on the other hand, involve computer assisted meeting environments, i.e. computer equipped meeting rooms with enhanced group applications. Both types of conferencing systems were evaluated to highlight the key requirements for collaboration. Although CAIRO is a distributed conferencing system it draws on many of the techniques used in EMS to provide effective conference controls.

Descriptions of each system include:

- a brief overview of the architecture of the system,

- a description of the various media supported by the system (eg. X-window, audio, and video).
- Support for temporal dependence among various media channels, i.e. multimedia synchronization.
- support for higher level protocols to control meeting structure and floor control, i.e. collaboration and floor control.
- support for meeting logging and efficient retrieval mechanisms, i.e. process history support.
- effective transmission of information, i.e. addressability, reasonable delay times and minimal information loss (reliability).

3.1 Electronic Meeting Systems

3.1.1 Xerox PARC Collab

The Xerox PARC Collab Project's [Stefik et al., 1987] main emphasis is on collaboration control mechanisms for a shared board. Their work provides valuable insights into meeting cycles and social interaction during a group meeting. However, Collab is lacking in multimedia communication and assumes all participants are physically collocated.

Architecture: The Xerox Collab project is comprised of several tailor made shared applications for specific meeting functions (Board Noter, Cognoter etc...). There is no meeting or name server incorporated within the system

Media Support: The Collab system is a highly specialized system and therefore has only one shared application (i.e. a Whiteboard).

Multimedia Synchronization: None, since only one media is present.

Collaboration and Floor Control: Complex floor control mechanisms describe in detail in Section 2.2.2.

Process History: Personal notes and Snapshots of screens are allowed. Activity on the shared board is also continuously logged.

Reliability: Closed network (LAN) system with very high reliability.

3.1.2 GroupSystems EMS

The University of Arizona / IBM GroupSystems EMS (Electronic Meeting System) joint effort [Nunamaker et al., 1991] extends the work undertaken in the Xerox Collab project. They provide mechanisms for retaining organizational memory, process support and structuring, task planning and structuring as well as control support for three basic meeting types (chauffeured, supported and interactive). As in Collab, GroupSystems EMS does not support multimedia communication and assumes a collocated meeting. Many of these EMS systems have been set up in convention centers to allow speedy issue resolution among top executives.

Architecture: GroupSystems EMS consists of a network of computers in a specialized meeting room with a large projection screen. Specialized software runs on each machine to provide support for process design and scheduling.

Media Support: Process support and structuring applications are provided (eg. Electronic Brainstorming, Electronic Discussion, Idea Organizer, Issue Analyzer, Vote Selection, Policy Formation).

Multimedia Synchronization: No synchronization is required since meetings are carried out face to face.

Collaboration and Floor Control: Three meeting types are supported:

1. Chauffeured - Single person enters group information.

2. Supported - All group members can enter comments, however, there is a central control on group memory access.

3. Interactive - All group comments and actions are logged in group memory.

Process History: Very detailed process support and structuring and extensive group memory maintenance (queuing and filtering).

Reliability: Closed network (LAN) system with very high reliability.

3.2 Distributed Conferencing Systems

3.2.1 WVU MONET

MONET (Meeting On the NETwork) [Srinivas et al., 1992], developed by CERC at West Virginia University, is among the first and most complex research efforts in conferencing systems. This project was supported by the Darpa DICE initiative.

Architecture: The MONET system is comprised of application sharing servers, conference servers, multimedia servers and a directory server. The application sharing server (COMIX [Babadi, 1990]) intercepts XClient calls from any X application and broadcasts them to the members of a conference. The conference servers handle membership, invitation processing and archiving for an active conference. Multimedia servers' key function is inter-media synchronization, however, this portion of the MONET system has not been fully implemented. Finally, a directory server maintains lists of registered participants as well as characteristics associated with those participants. The MONET system has a simplistic user interface that is quite cumbersome to use.

Media Support: MONET provides support for audio, video and shared X applications. Audio and video capabilities are limited, however, and are comprised mainly of annotations to text rather than as an effective real time communication mechanism. The shared X system allows all participants access to any X application.

Multimedia Synchronization: Although multimedia synchronization is mentioned as a goal for MONET, no indication of synchronization was provided. Much of the effort has been focused on providing operating system and hardware support for synchronization.

Collaboration and Floor Control: MONET provides three basic floor control mechanisms: chairman control, time-limited FIFO, and a combination of the two. There is no support for extension or design of more complex mechanisms.

Process History: MONET provides no conference logging facility.

Reliability: No information available.

3.2.2 NCSA Collage

NCSA's Collage [NCSA Collage, 1994] conferencing tool has a characteristically clean interface similar to NCSA Mosaic. NCSA Collage was designed with a focus on visualization applications and hence has complex image visualization and manipulation mechanisms incorporated within it.

Architecture: NCSA Collage is based on a strict client-server model. All participants initiate NCSA Collage sessions and NCSA Collage Servers are created as each conference is initiated. All future communication by participants in a conference are passed through the newly created NCSA Collage server. NCSA Collage lacks any form of directory service. NCSA Collage is also available on Macintosh and Windows platforms which greatly enhances its usefulness.

Media Support: Whiteboard, Text, Animation, and Image visualization tools are the core media supported by the NCSA Collage system. NCSA Collage also incorporates an effective screen capture mechanism. No support for audio and video is included in the current system.

Multimedia Synchronization: Due to the lack of audio or video media in NCSA Collage, no synchronization mechanism is incorporated within the system. All media drivers have no temporal dependence.

Collaboration and Floor Control: No floor control protocol is provided with the NCSA Collage system. All clients have access to the shared application and all interactions are broadcast to members of a conference.

Process History: No history of a conference session is maintained by the NCSA Collage Server. However, local snapshots of conference proceedings can be maintained by each client.

Reliability: No directory service provided.

3.2.3 SRI CECED

The Collaborative Environment for Concurrent Engineering Design (CECED), developed by SRI International, provides mechanisms for informal communication and history capture of informal stage in the specification and design process. The work undertaken has detailed the requirements for effective conferencing systems. SRI's approach has been to ensure that the conference system is non-intrusive and as natural as a standard meeting conversation.

Architecture: CECED builds on the MOSAIC platform (Multimedia Open System for Augmented Interactive Collaboration [Craighill et al., 1992], [Garcia-Luna et al., 1987]). As in the MONET and XTV systems, CECED distributes existing unmodified X applications. This is performed by specialized Collaboration Management Agents (CMA). A Shared Tool Event CMA provides broadcast capability to existing XClient applications. A connection CMA handles all underlying network protocol translations. A Session Manager acts as a user interface to the conference tool. An Information Store CMA provides archiving and data access control for the collaborative conference. CECED also incorporates Collaboration Aware Tools (CAT). These are specifically developed tools for the CECED system. The current prototype has an audio CAT to allow for audio communications in the collaborative environment.

Media Support: CECED supports any X-based application as well as limited audio capability through the audio CAT.

Multimedia Synchronization: Synchronization can be implemented as a CMA among various X-applications. However, the CECED prototype does not include any inter-media synchronization

Collaboration and Floor Control: CECED provides synchronous multi-user access. The access control protocol is similar to the Ethernet concept. It involves a listening process that waits till the line is free and then allows the participant to speak. This process known as COMET [Garcia-Luna et al., 1989], is a distributed activity sensing floor control algorithm that guarantees a single stream of input to unmodified single-user applications.

Process History: CECED provides only a complete logging of conference proceedings. Furthermore, CECED provides logging of semantic changes in the conference as well as raw data.

Reliability: Completely distributed. No directory service is provided.

3.2.4 AT&T RAPPORT

The AT&T RAPPORT [Ahuja and Ensor, 1992] system focuses on the network communication issues of conferencing and on effective user interface design for conferencing tools. It provides synchronized video, audio and data communication, however, RAPPORT uses heterogeneous networks for each mode of communication (PBX for audio, coax cable for video, LAN for data). RAPPORT also lacks effective support for conference control.

Architecture: Proprietary.

Media Support: Provides voice, video and shared X applications.

Multimedia Synchronization: Synchronization is not necessary since the system has virtually no communication latency. Currently, Rapport runs on three separate networks: a LAN for data transmission, a specialized coax video network for video, and an ISDN system for audio communication.

Collaboration and Floor Control: Chalk passing is the only control mechanism suggested.

Process History: No capture of process history is captured aside from screen snapshots and note-taking applications.

Reliability: Highly reliable communication with no data loss due to the nature of the network. However, the system is prohibitively expensive and not easily scalable.

3.2.5 XTV

The XTV [Abdel-Wahab, 1993](X Teleconferencing and Viewing) effort focuses primarily on providing reliable transfer of data among shared X systems. XTV incorporates a very simple floor control mechanism and does not provide support for non-X media communication.

Architecture: The XTV system is comprised of three key components: information daemons (ID), conference announcers (CA) and user interfaces (UI). ID's maintain communication among the UI's and the CA's. ID's are equivalent to meeting rooms in a physical conference. UI's are each individual participant in a conference. A UI is an X-application used by a conference member that can be shared among all participants. CA's maintain conference membership lists and process conference invitations.

Media Support: XTV only supports X-based applications.

Multimedia Synchronization: No synchronization among X applications is provided by the XTV system.

Collaboration and Floor Control: Chalk passing protocol with a chairman override capability.

Process History: No explicit process history capture mechanism is provided by XTV.

Reliability: Provides redundant servers to insure fault tolerance and employs sophisticated protocols to insure reliable information transfer.

3.3 Concluding Remarks

The above systems provide an overview of the wide array of conferencing software available on the market. Two classes of conferencing software were discussed in the preceding sections: electronic meeting systems and distributed conferencing systems. The electronic meeting systems (GroupSystems EMS and Xerox Parc Collab) are in general more focused on meeting organization and coordination, since the architecture of the system depends on the physical collocation of the group members. The distributed conferencing systems, on the other hand, ignore the coordination problem and concentrate primarily on collocation facilitators (i.e. multimedia information transmission). The approach taken in this thesis is to achieve an appropriate balance of distributed meeting collocation and coordination technologies. Table 3.3 provides a simple comparison chart for the various systems surveyed.

The CAIRO system described in this thesis focuses on providing many of the complex collaboration control mechanisms associated with electronic meeting systems in a distributed conferencing system. The CAIRO system architecture also provides control mechanisms that are less rigid than the ones provided by the two electronic meeting systems surveyed. Furthermore, CAIRO provides synchronized multimedia information transmission using the standard internet protocol.

Table 3.1: Comparison Table for various conferencing systems[1]

Feature	*Conferencing System*						
	`MONET`	`Collage`	`CECED`	`Collab`	`RAPPORT`	`GroupS EMS`	`XTV`
Distributed?	+	+	+	–	+	–	+
Video Support	0	–	0	NA	+	NA	–
Audio Support	0	–	0	NA	+	NA	–
X Support	+	0	+	–	0	–	+
Synchronization	0	–	0	NA	+	NA	–
Floor Control	0	–	+	+	–	0	–
Process History	–	–	0	0	0	+	–
Reliability	N	0	0	+	+	+	+

[1](+) = extensive support, (0) = limited support, (–) = minimal or no support, (N) = No information available, (NA) = Not Applicable

Chapter 4

System Architecture

The CAIRO system is comprised of several interlinked modules and servers (see Figure 4-1). Each participant engaged in a CAIRO conference spawns a *Collaboration Manager* (shown as a dashed box) which is comprised of media drivers (shown as pictograms of the media – i.e. video camera, microphone and X display) and message servers (indicated by the acronym 'MSG'). The media drivers satisfy requirement (i) specified in Section 2.3. The message servers package data for transmission over the network and enforce synchronization constraints during media play-back thereby enforcing requirement (ii). *Forum servers* are processes that maintain control of a conference among several individuals (requirement (iii)) and enforces membership constraints (requirement (v)). Furthermore *forum servers* log all conference proceedings (requirement (vi)). *Forum servers* are spawned by *forum managers* (not shown) that define a specific control methodology. Forum managers also provide mechanisms for converting a *forum server's* control strategy thereby satisfying requirement (iv). Finally, the *name server* maintains a directory of all participants, forum managers and forum servers within the CAIRO system. It allows each participant to easily address any other member or forum in the CAIRO system.

The following sections describe the key components of the CAIRO system architecture. Section 4.1 defines terms that will be used throughout the architecture description. The collaboration manager module is then described in detail (Section 4.2).

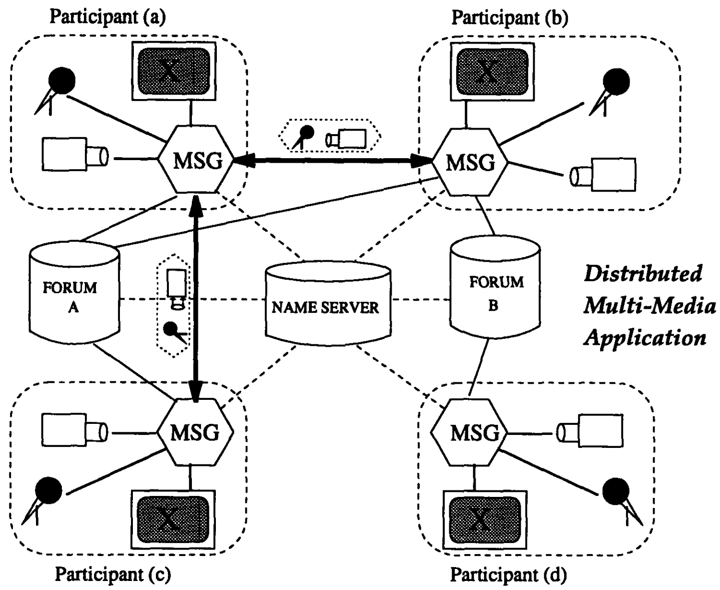

Figure 4-1: The CAIRO System: An overview.

Section 4.3 describes the mechanism by which collaboration control methodologies are enforced by forum servers upon collaboration managers. The following sections describe media drivers and the components of the message servers that enable communication among the participants in a conference. The functionality of forum managers and forum servers are detailed in Section 4.6 and 4.7. Finally, the functionality of the name server is illustrated in Section 4.8.

4.1 Definitions

Participant a user who has the ability to participate in a multimedia session.

Conversation a multichannel connection between two or more participants.

Forum a set comprised of participants and other forums. Associated with a forum are a variety of access control and collaboration control parameters. An atomic forum is a single participant.

Media Source a device or application that provides a channel in a multi-channel, multimedia conversation.

4.2 Collaboration Manager

The Collaboration Manager incorporates the CAIRO user interface and maintains lists of available media resources and forum servers (see Figure 4-2). The Collaboration Manager also has a snapshot facility that allows each participant to retain portions of the meeting for his/her own personal notes. It also enforces conference controls associated with the forums in which the user is participating. For example, a specific forum may not allow any conversations with users outside of the forum or may not permit any private side conversations with other members of the forum.

4.3 Token-Based Control

All restrictive controls on the participants in a forum are provided via token access. The Collaboration Manager cannot issue any communications without having received a token granting access priveledge to that specific speaker. Token controllers on both the Collaboration Managers and Forum Servers must be secure and trusted code. Methods to enforce this abound, see [Popek, 1974, Wood and Kochan, 1985] for a more detailed discussion. Forum Servers issue two commands related to tokens: a Grant_Token command (specifying write or read rights to a communication channel with another participant) and a Retrieve_Token command (retracting read or write rights specified by a Grant_Token). Collaboration Managers respond with an Accept_Token or Reject_Token message depending on conflicts with other active forums on that user's workstation (eg. engagement in another forum that does not permit multiple parallel forums). Tokens have internal time-out counts after which tokens expire. Specialized tokens denote ability to participate in side conversations, external conversations, and interjection rights. These side and external conversation tokens can be used to maintain confidentiality within a conference and to minimize group distractions. Interjection tokens allow for emergency situations.

Tokens are granted upon request submitted to the Forum Server by a Collaboration Manager. Such tokens can be granted automatically using a predetermined com-

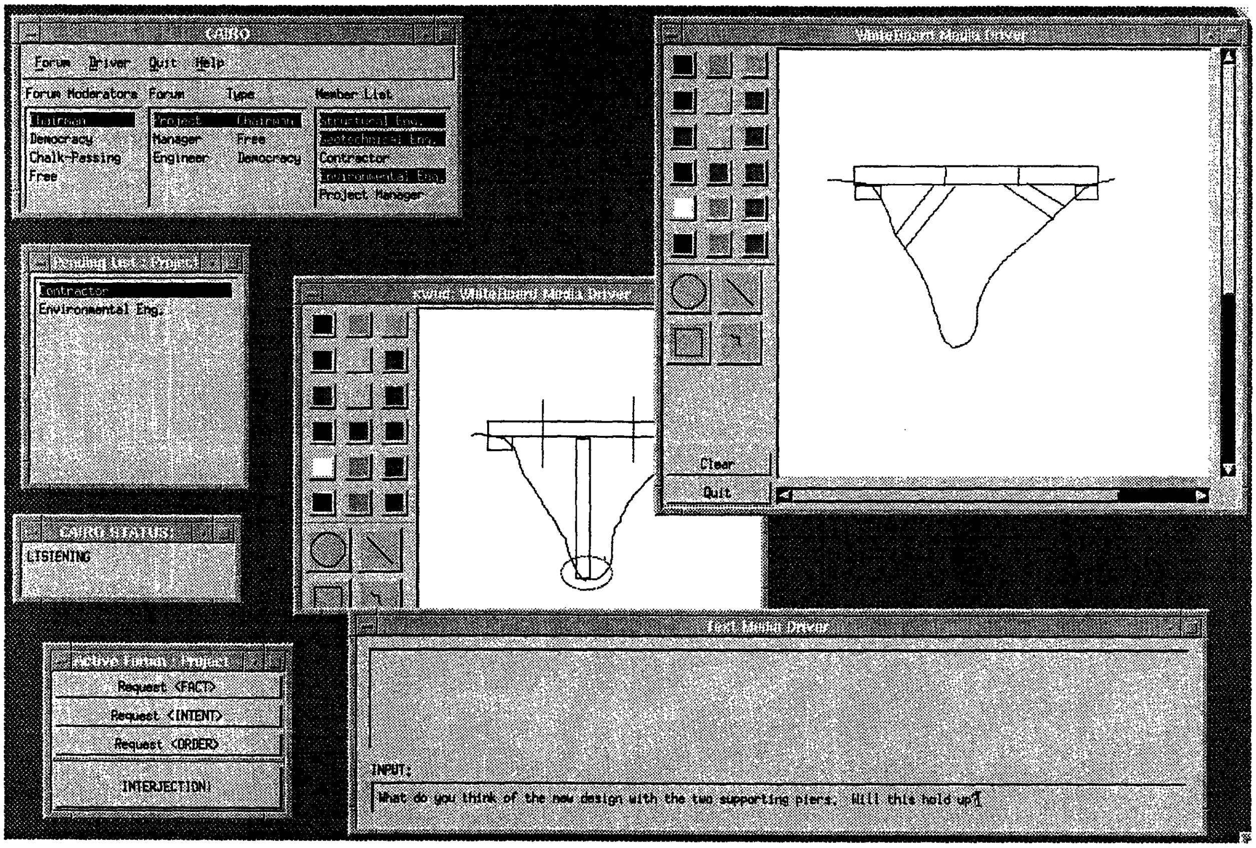

Figure 4-2: A Sample Session of CAIRO: Bridge Design Case (Contractor Perspective)

puter moderation scheme or can be granted manually by a moderator. Furthermore, conference logging is achieved via a specialized token requesting communication sent to the Forum server where all interactions are logged for future browsing and editing. This mechanism satisfies the process history support requirement (requirement (iv)) described in Section 2.3.

The token structure provides a centralized control yet distributed communication structure for conferencing. Hence, all high bandwidth communication is decentralized and direct, while all floor control requests are centralized by the forum server.

4.4 Media Drivers

Media drivers handle all I/O between the Multimedia collaboration system and the underlying media channel. Each driver is tailored specifically to the underlying media represented. Each driver is responsible for data acquisition and frame compilation for transmission and replay. This module must also provide the multimedia server with synchronization information, frame size, and delay and error tolerances. Three media drivers have been implement: a controller of the SUN audio system using the AudioFile extensions to X11/R5, a shared X whiteboard for sketching, and a text entry and display tool.

4.4.1 Audio Driver

This is a driver implemented using AudioFile, a standardized interface similar to the X programming toolkit. The code should be portable to a variety of workstations that support AudioFile.

4.4.2 Text Driver

This is a driver that allows the exchange of short text messages among the participants. Lengthy text entries may also be pasted into the text entry input box for transmission to conference participants.

4.4.3 Shared X Whiteboard

This is a driver for an application that simulates a Blackboard in an office environment. It can be shared among the members of a forum to communicate visual information such as sketches of various product design ideas. It can also used to transfer bitmaps of images on the user's screen to the rest of the team.

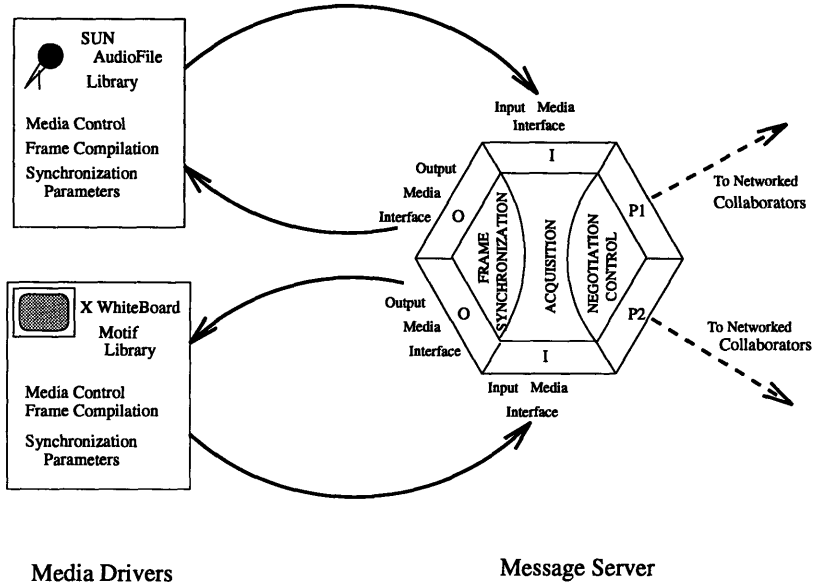

Figure 4-3: Message Server Overview: Media drivers for audio and whiteboard devices

4.5 MultiMedia Message Servers

Each user in the CAIRO collaborative environment is associated with at least one multimedia message server. This server (see Figure 4-3) handles all communication between users, schedules transmission and display of channel data, as well as maintains membership on the various forums the user wishes to be associated with. The components of the multimedia server are described below:

4.5.1 Synchronization Engine

The engine maintains the input and output buffers and ensures that all channels are assembled before play-back by the media drivers. Figure 4-4 describes the essential functionality of the synchronization engine as well as its relation to the input queue and multimedia frame output queue.

Multimedia Frame Output Queue: Storage of multiple frames prior to transmission on the ethernet. This is required to allow for complete channel frame transmission.

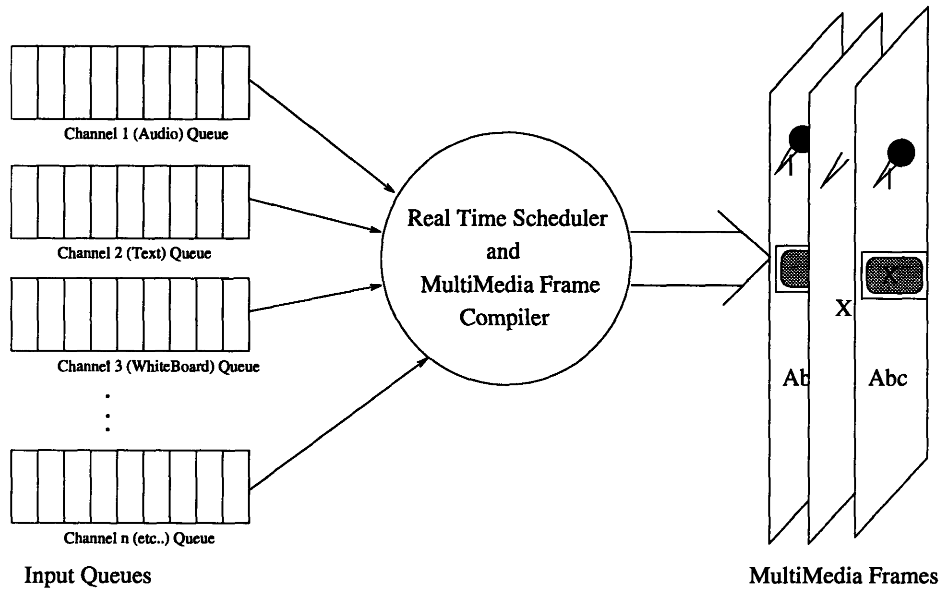

Figure 4-4: Multimedia Frame Assembly from Input Media Channel Queues.

Input Media Channel Queue: Storage of incoming data packets on each media channel. These are stored until they can be compiled into a multimedia frame.

Connection Manager: This object takes care of low level calls to the TCP/IP layer for maintaining socket connections and sending datagrams across the internet.

Correspondence Cache: A cache of addresses associated with all participants the user will broadcast to given that he/she is a member of a specific forum. Update requests are periodically transmitted to maintain cache coherence between the forum server and multimedia message server.

4.5.2 Message Protocol

Appendix B provides a complete listing of the messages exchanged between the various component of the CAIRO system. All messages are TCP/IP datagrams and are asynchronous. Each component of the system has an interrupt handler that manages incoming and outgoing messages and appropriately routes the messages to the appropriate objects.

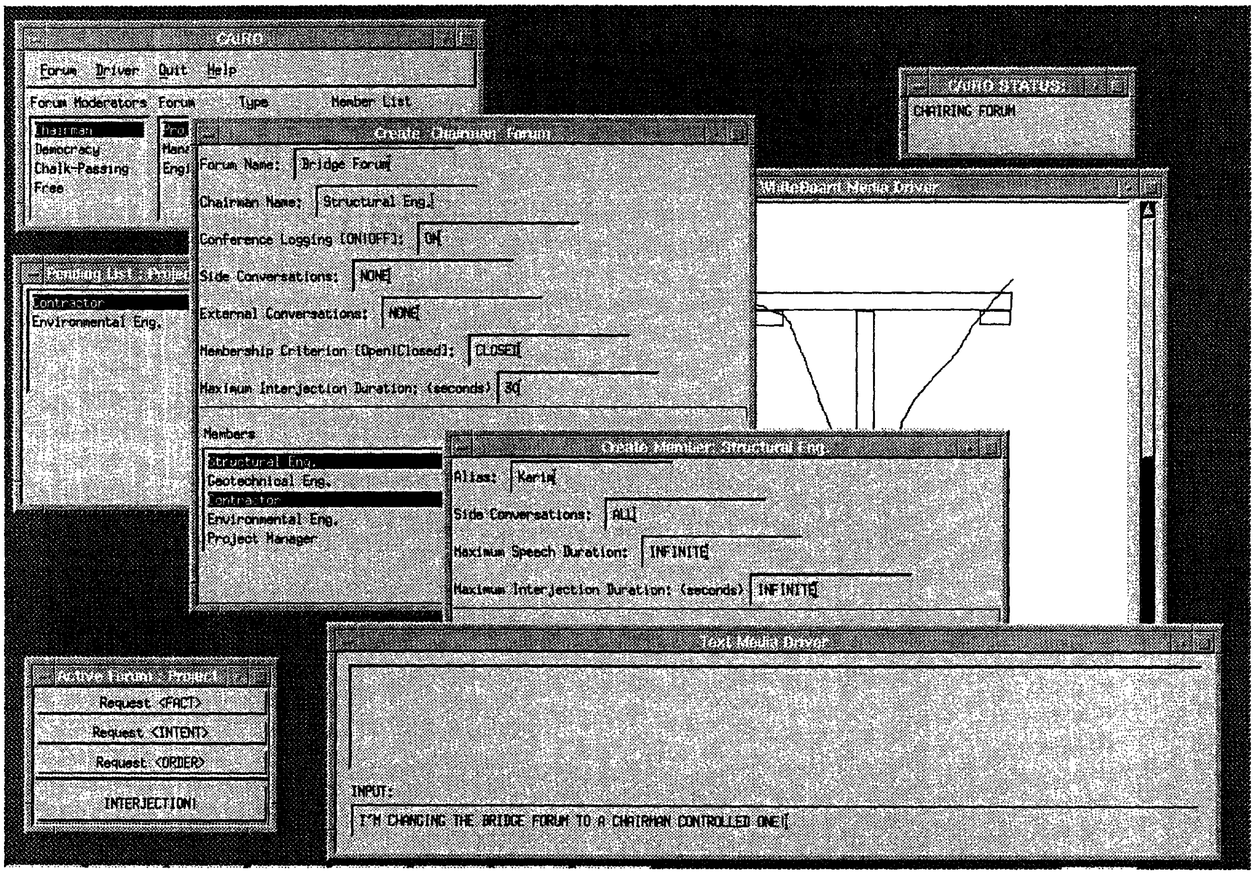

Figure 4-5: Forum Manager User Interface

4.6 Forum Manager

Forum managers contain information on a particular type of meeting. They spawn off instances of forums that comply with the forum manager control mechanisms but with varying memberships. Currently, four such forum managers have been designed however the system is extensible and future systems need only comply to a predefined message protocol to enter into CAIRO. Chapter 6 describes the various control schemes and the underlying primitive control structures. Among the necessary provisions are membership request processing, membership grant, token request, token grant, as well as participant privilege explication. These parameters allow a forum manager to specify membership constraints as well as floor controls for a conference.

4.7 Forum Server

A forum is a structured group of participants involved in a collaborative effort. The forum server maintains a list of all participants in a specified forum as well as the privileges associated with each participant. Each forum member is listed in one of three states in the forum: active (logged in and listening to conference), speaking (actively participating in conferencing, i.e. has control over the floor), or non-active (not logged in and not receiving any conference communications).

Forum servers have two key functions: subscription control and speaker control. Subscription control may be a predefined list of allowable conference participants or it could be through a vote by existing participants or it may be a forum maintainer with the right to revoke and grant membership to potential members. Speaker control is the process by which a forum server maintains an orderly conversation among the members of the forum. Speaker control or floor control of the forum is achieved through the granting and revoking of conversation tokens as described in Section 4.3.

4.8 Name Server

The name server is an independent server that acts as a global directory for the CAIRO conference system. The following information is listed in the name server for each participant and each forum and may be queried by any participant or forum server.

1 Participant Name and Location: including media driver locations and media descriptors.

2 Participant Status: each participant is either in an active or non-active state. Active denotes that the user is logged into the conference system via a Collaboration Manager on his/her workstation. Non-active status is given to users who are subscribers to the CAIRO system but are not reachable.

3 Forum Manager Name and Location: including a brief description of control style.

4 Forum Name and Location: including a listing of shared media drivers.

5 Forum Status: each forum is either in an active or non-active state. Active forums imply a conversation is occurring among the participants of the forum. Non-active forums are structured meeting skeletons with membership lists for a meeting that is not currently in session.

Chapter 5

Media Synchronization

The CAIRO system is designed to support multiple media channels in a conversation. Due to delays in the transmission of the packets across the internet, packet arrival times are unpredictable (see Figure 5-1). Therefore, each multimedia frame does not arrive at the destination as one chunk. The receiver must then reassemble the frame and ensure that play-back of the frame is synchronized such that it reflects the initial input from the source. Figure 4-3 illustrates an overview of the media channel synchronization subsystem of CAIRO. Media synchronization is base on the synchronization parameters (Section 5.1) supplied by each media driver. Each media driver also supplies temporal relations with respect to the other media drivers in the system (Section 5.2). Given these parameters the system can compensate for skews in the delivery time of messages through the heuristics described in Section 5.3. A real time scheduler is then invoked to determine the schedulability of the input media streams (Section 5.4). A synchronization engine combines the synchronization heuristics and parameters to play-back the multimedia data to the receiver in as similar a form to the original data as possible (Section 5.5).

5.1 Synchronization Parameters

The following are parameters that define the quality of service for a particular channel. These parameters are provided by each media driver involved in the CAIRO system

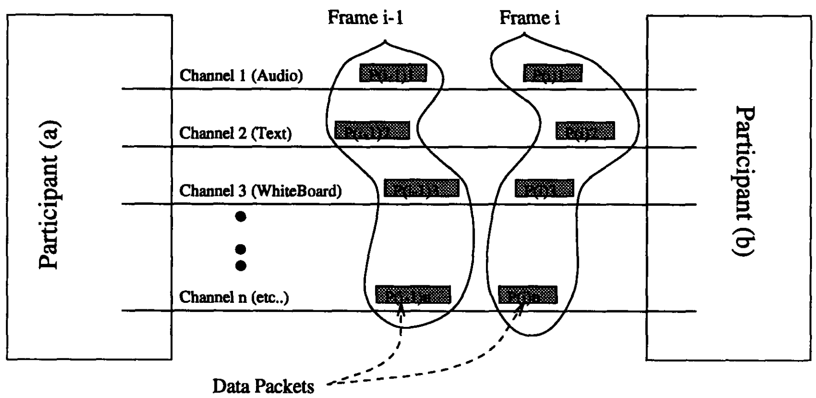

Figure 5-1: Synchronization issues between two conference participants

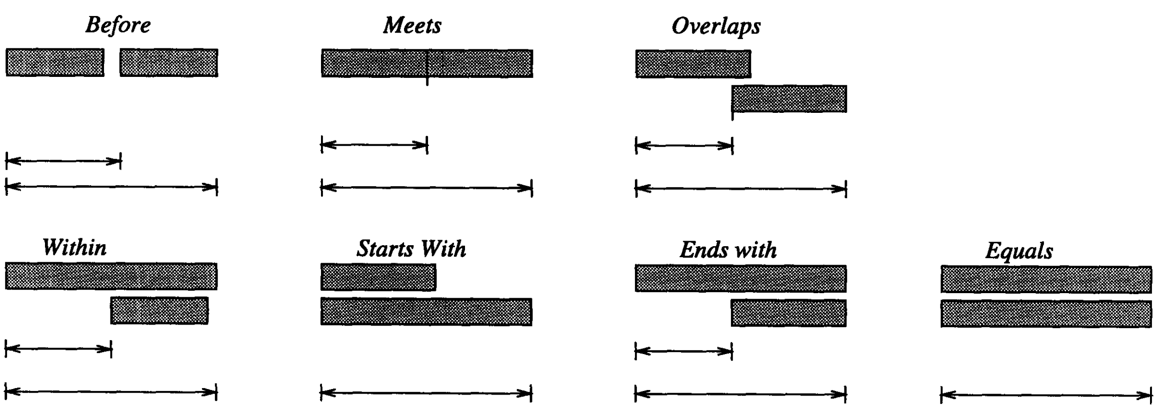

Figure 5-2: Various methods for combining channels. Adapted from [*Little and Ghafour*, 1993]

and are required for scheduling of the media channel transmission by the message server.

Frames per Second (FPS): The average number of frames per second, along with average frame size, are critical for appropriate scheduling of media transmission and display. Example: Audio has been set at 40 FPS.

Average Frame Size: The size of each frame in bytes after all compression has been performed. Example: Audio is sent after μ-law encoding at 200 bytes per frame.

Inter-Glitch Spacing: The maximum allowable spacing between missed or corrupted frames in the transmission stream. For audio this is typically 1 in 20.

Delay Time: The maximum amount of time a frame can be skewed with respect to the frame boundary.

Temporal Relation to other Channels: A listing of all other media driver channels with which this channel must be synchronized. Synchronization can occur in many forms which are discussed in Section 5.2.

5.2 Temporal Relations between Channels

Each channel in a multimedia conference must include a parameter that describes its temporal relation to each of the other channels. Little and Ghafour have developed a conceptual model for capturing temporal relationships among various media channels [Little and Ghafoor, 1993]. Figure 5 provides a graphical overview of the temporal relations between channels. A subset of the temporal relations between two channels α and β are described in Table 5.1.

T_α is the duration of a transmission on channel α and T_β is the duration of a transmission on channel β. T_δ is the difference in time between the beginning of

Table 5.1: A subset of temporal relations between a channel α and a channel β.

Relation	T_α	T_β	T_{TR}
before	$< T_\delta$	$\neq 0$	$T_\beta + T_\delta > T_\alpha + T_\beta$
meets	T_δ	T_α	$T_\alpha + T_\beta$
overlaps	$< T_\beta + T_\delta$	$\neq 0$	$T_\beta + T_\delta < T_\alpha + T_\beta$
overlaps	$> T_\beta + T_\delta$	$\neq 0$	T_α
starts	$< T_\beta$	0	T_β
equals	T_β	0	T_α

transmission on channel α and the beginning of transmission on channel β, and T_{TR} is $max(T_\alpha, T_\beta + T_\delta)$.

5.3 Delay Compensation Heuristics

All media channels are synchronized using the compensation heuristics described below as well as the real time scheduling algorithm described in the following section. The work undertaken is based to a large extent on [Ravindran and Bansal, 1993][1].

Frame Interpolation: If a current missed frame time is greater than the inter-glitch spacing. Then replay the last frame and continue.

Handling of Persistent Slippage: If continuous loss of frames then switch to a lower resolution or lower frame rate (i.e. graceful degradation of a channel).

Advance to Next Temporal Interval: Retain frame until next frame arrives as long as the skew is not too far off the temporal interval boundary.

Control of Frame Time-outs: If packets on a channel are delayed by a specific amount then delay all subsequent packets, in order to try to only have one skew period.

[1]Extensions have been added in our implementation to enhance throughput on our network. Those that we have introduced will be indicated by an asterisk (*).

Output Queuing*: Compile and store multiple frames prior to transmission on the ethernet until a prespecified number(i.e. the queue length) of complete multi-channel frames are ready to for transmission.

Input Queuing*: Store incoming packets of data, until they can be compiled into a multimedia frame and there exists two subsequent multimedia frames that can continue the multimedia play-back.

Variation in Queue Length*: Increase the length of Input Queues of the various media channels to allow for enhanced scheduling of the multimedia frames.

5.4 Soft Real Time Scheduling

Real time scheduling is the scheduling of multiple concurrent tasks to be performed within a given temporal interval or frame. Each task i has an associated computation time C_i as well as a period T_i. The rate monotonic algorithm provides a conservative estimate as to the number and type of tasks schedulable on a system (see [Gomaa, 1993] for a more detailed discussion).

$$U(n) = \sum_{i=1}^{n} \frac{C_i}{T_i} \leq n(2^{\frac{1}{n}} - 1) \tag{5.1}$$

As each media device registers with the message server system $U(n)$ is checked for consistency with the above equation. Once all task computation times and deadlines are determined the scheduler operates on an earliest deadline first policy. That is within a given time unit the highest priority tasks to be scheduled are those that have the highest periodicity.

5.5 Synchronization Algorithm

The preceding sections provided a description of the necessary parameters and task constraints for multimedia synchronization in a distributed conference. Furthermore, Section 5.3 described the basic heuristics employed by the synchronization engine. This section describes in detail the synchronization mechanism implemented in CAIRO. The base data structures, multimedia frames and media device input queues, are discussed followed by the description of the synchronization mechanism.

5.5.1 Frames

Multimedia frames transmitted by a source participant are encoded with a frame sequence number and a time stamp. Furthermore, the initial and final frames in a conversation are uniquely tagged to aid the synchronization and scheduling mechanism as discussed in Section 5.5.3. Temporal constraints described in Table 5.1 are encoded with respect to a single frame. Each frame is composed of multiple channels of media data for a given period of time. In order to ensure the arrival of all packets in a single frame, a delay in play-back at the destination must be introduced. CAIRO enforces a delay of .5 seconds although this may be varied as the network infrastructure changes.

The synchronization engine enforces three types of temporal constraints: before, after, and during. All three constraints are determined on the transmission side and the appropriate frame sequence numbers are chosen for each channel to reflect the constraint. For example, if text was required to appear after audio, and audio was sampled in frames i to $i+10$ then the text sequence number would be $i+11$.

5.5.2 Queues

All packets arriving on the receiving end are placed in input buffer queues by the media drivers (Appendix A provides a detailed diagram of object structures in CAIRO). The queues store up to $fmax$ frames ($fmax = 100$ in the CAIRO prototype). Incoming data is placed in the queue according to the frame sequence number. The queue index is equal to the frame sequence number modulo $fmax$. Each media channel has its own queue structure (eg. audio queues have 10 audio clips per queue element, text queues have 1 string per queue element) see Figure 5-3. The queue structure is

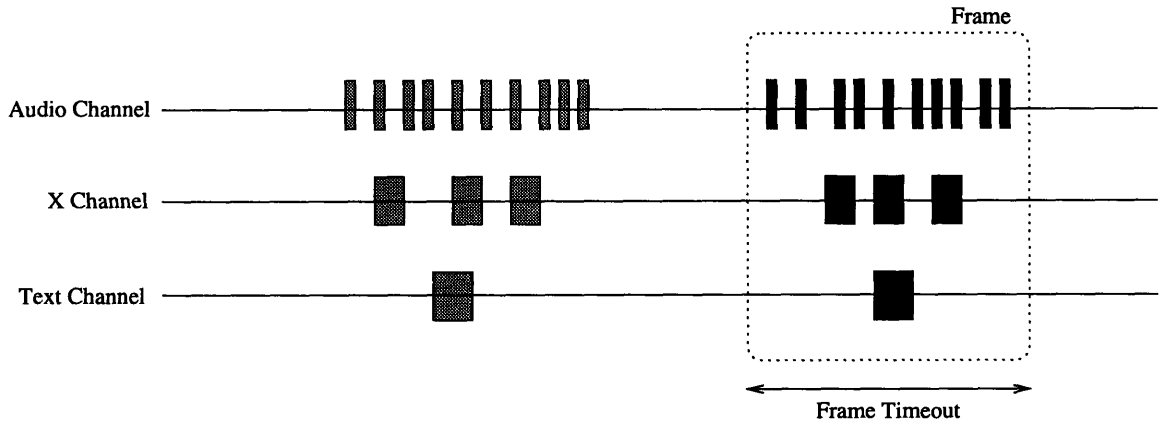

Figure 5-3: Multi-channel frames

a list of lists. The toplevel list is indexed by sequence. For each sequence index in the toplevel list a secondary list contains the media packets indexed by source (this allows a receiver to listen to data from multiple sources). Hence, a single sequence index can be associated with multiple elements. Each element in the queue is also tagged with a time-stamp and a source specification.

5.5.3 Scheduling

The scheduler operates on the basis of frames. The scheduler is invoked periodically based on the frame time-out period. The frame time-out period is arbitrarily set at a $\frac{1}{4}$ second. Each frame contains several packets on each media channel (see Figure 5-3). At each interval the scheduler polls each queue and retrieves a list of complete frames. If a complete frame exists and it is has the smallest sequence number the frame is scheduled for replay. However, if the frame with smallest sequence number is incomplete, the scheduler employs the delay compensation heuristic that is applicable. If none of the heuristics are applicable the user is notified that the communication channel can not support the quality of service requested and suggests decreases in the quality thresholds.

There are two exceptions to the behavior of the scheduler. As discussed earlier there are two special frame identifiers, initial and final. The scheduler should not replay a frame unless three frames are available for replay, unless the final frame is among the last three frames. This buffering of frames ensures that data will usually be available for replay at the frame time-out.

The synchronizer then takes a single frame and passes it on to the real time scheduler. The scheduler then posts the appropriate events to replay the frame. The events are posted based on an earliest deadline first policy. The scheduler is implemented on top of the X event handler.

Chapter 6

Collaboration Control

Structuring and control of group meetings enhances the efficiency of a collaborative team. The following sections discuss the hierarchical meeting structure of CAIRO (Section 6.1) in addition to the collaboration primitives defined in the system (Section 6.2) and the collaboration schemes built upon these primitives (Section 6.3). Finally, an operational description of key processes in the CAIRO system is provided to illustrate the enforcement of the collaboration controls (Section 6.4).

6.1 Hierarchical Forum Model

Forums maintain a conference among individuals. Each forum is associated with a *forum moderator* that defines the control behavior of the conference. The forum server processes requests for membership to the forum as well as requests to speak by participants within the forum. As shown in Figure 6-1, a forum is comprised of individuals and other forums. The forum `Management` that is a member of another forum `Project` must be at least as restrictive as the forum `Project`. Any restrictions on membership and communication must be upheld by the child forum, `Management`.

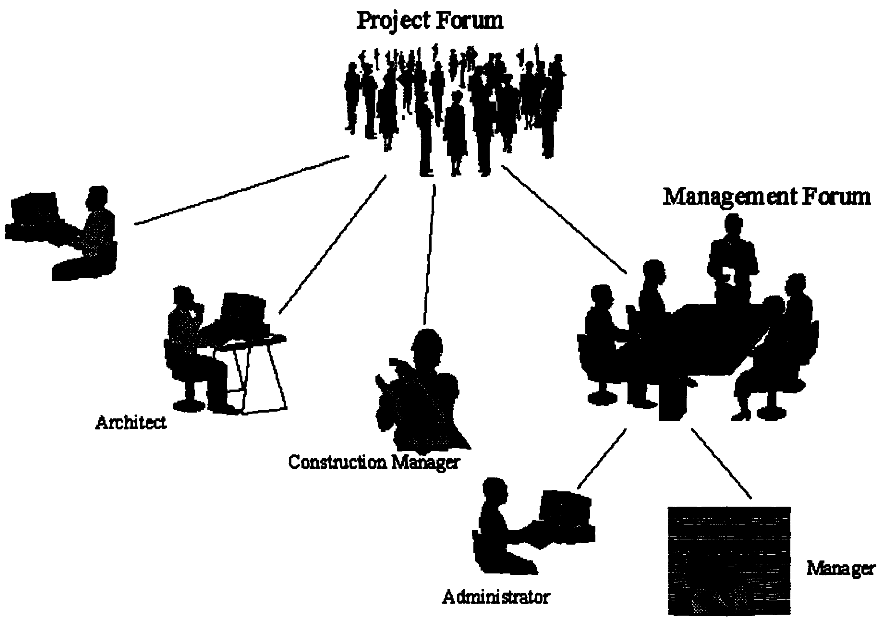

Figure 6-1: Hierarchical Forum Structure

6.2 Collaboration Primitives

During a meeting or conversation a particular participant can be in one of three states: *active* (i.e. speaking or demonstrating), *pending* (i.e. awaiting his/her turn to speak), or *inactive* (i.e. passive observer or listener). Each participant's state is relative to a specific forum and is stored in the forum server. The following is a list of communication primitives that are part of the participant user interface to aid in transition between the three states.

Speech Act: This is a qualification by the participant as to the intent of his/her speech according to the speech act theory described in Section 2.2.1. The intent of the speech must be provided before a speaker request is issued.

Speaker Request: This is equivalent to a professional raising his/her hand in a meeting situation. It indicates to the forum moderator and to the other members of the forum the participant's intent to speak. A speaker request is accompanied by a qualification of the speech act the speaker intends to perform. The forum server would then place the participant on a list of *pending* speakers depending on his/her qualifications. In a democratic forum a participant becomes *active* if a majority agrees to his/her request to speak. Furthermore, the computer can automatically moderate (i.e. choose the *active* speakers from the *pending* queue) a forum based on precompiled speaker qualification data.

Interjection: This is a mode of conversation in which the participant can interrupt an ongoing conversation for a limited amount of time.

6.3 Group Primitives

Group meetings can take on multiple characters and structures. As described in Section 2.2, group formation and meeting cycles require various group control procedures and paradigms. Below is a list of primitive controls on each forum from which a more complex collaboration control mechanism may be devised. The forum creator may choose to over-ride any of these primitives for a particular forum member.

Interjection Duration: Within the parameters specified for a forum is the length of time allowed for interjections. An interjection time of zero indicates no interjections are allowed. Conversely an infinite interjection time allows for complete unstructured free-form conversation.

Maximum Speech Duration: Within the parameters specified for a forum is the length of time allocated to a single member to hold the floor of the conference.

Maximum Number of Active Speakers: This parameter indicates the number of concurrent speakers allowable during the conference.

Chairman: A designation of a participant or group of participants who hold a privileged status within the forum. They may preempt speakers and arbitrarily choose active speakers.

Side Conversations: Side conversations are two-way or multi-way conversations among a subset of the forum members. Forums may be created that do not allow such side conversations to exist.

External Conversations: External conversations are conversations between a member of a forum and other non-members while a forum is active. This form of conversation may also be restricted by the forum.

Logging Mode: Currently the system only provides either continuous logging or no logging at all of the ongoing conference.

Speaker Evaluation: A voting mechanism has been implemented to evaluate participant acceptance of a specific topic or to determine participant value to a conference. The results of this evaluation may be used to determine the order of speaker priority for a conference.

Speaker Ordering: The ordering of the *pending* speaker queue may be on a first come first serve basis or other evaluation criteria. These include: ordering of speakers based on value determined by the participants, as described in *Speaker Evaluation*; or ordering based on chairman choice in a chairman controlled conference. This control mechanism satisfies the requirement for free form and structured conferencing.

6.4 Sample Collaboration Schemes

The collaboration primitives discussed above are combined to form a collaboration scheme or mechanism. The CAIRO system can easily be extended to provide many different collaboration schemes. Below are the list of schemes that have been implemented.

Free: All participants may talk at any time. Completely uncontrolled all speakers may speak at once. That is Chairman='none', side conversation = ALL, external conversation = ALL, Speaker Ordering = 'first-come-first-serve'.

Democracy: Choice of the active speaker is based on a vote by all other participants. That is Chairman='none', side conversation = ALL/NONE, external conversation = ALL/NONE, Speaker Ordering = 'highest vote'.

Chalk-Passing: Last active speaker chooses next person to be a designated active speaker. Each speaker may only speak for the time allotted by the *Maximum Speech Duration* parameter specified above. In this scheme: Chairman='last speaker', side conversation = ALL/NONE, external conversation = ALL/NONE, Speaker Ordering = 'chosen by chairman'.

Chairman Control: A specific privileged participant (Mr. X) has the ability to choose the participant who should address the conference at any specific time. In this scheme: Chairman='Mr. X', side conversation = ALL/NONE, external conversation = ALL/NONE, Speaker Ordering = 'chosen by chairman'.

Modified Delphi: The system polls all participants in the collaboration on their views regarding a specific design problem. The results are compiled and presented to the conferring experts and the participants are then re-polled. This process is repeated by the questioner until the experts provide a consistent analysis. The Delphi method is used extensively in polling experts on directions in hi-tech industry. In this control strategy there exists a moderator as well as a questioner. A quicker more dynamic method using our collaboration methods is proposed. In this scheme: Chairman='moderator/questioner', side conversation = ALL/NONE, external conversation = ALL/NONE, Speaker Ordering = 'round robin'.

6.5 Operational Description

The CAIRO collaboration control mechanism is composed of several interacting servers and modules. A brief description of the operations of these modules/servers is provided in this section. The operations are listed in the order in which they would naturally occur.

6.5.1 Forum Creation

A forum is initiated by invoking a forum manager. Forum managers can be invoked by executing the appropriate forum manager program or by choosing the *New Forum* command from the CAIRO control panel (see Figure 6-2). menu. A series of dialog boxes and menus then guide the forum creator through the creation process. Figure 4-5 shows the forum manager user interface. The forum creation process involves specifying the group primitives described in Section 6.3 as well as specifying the members of the forum and their associated privileges. The specified parameters are then stored in a forum specification file (see Appendix C) that is used by the forum server when instantiated.

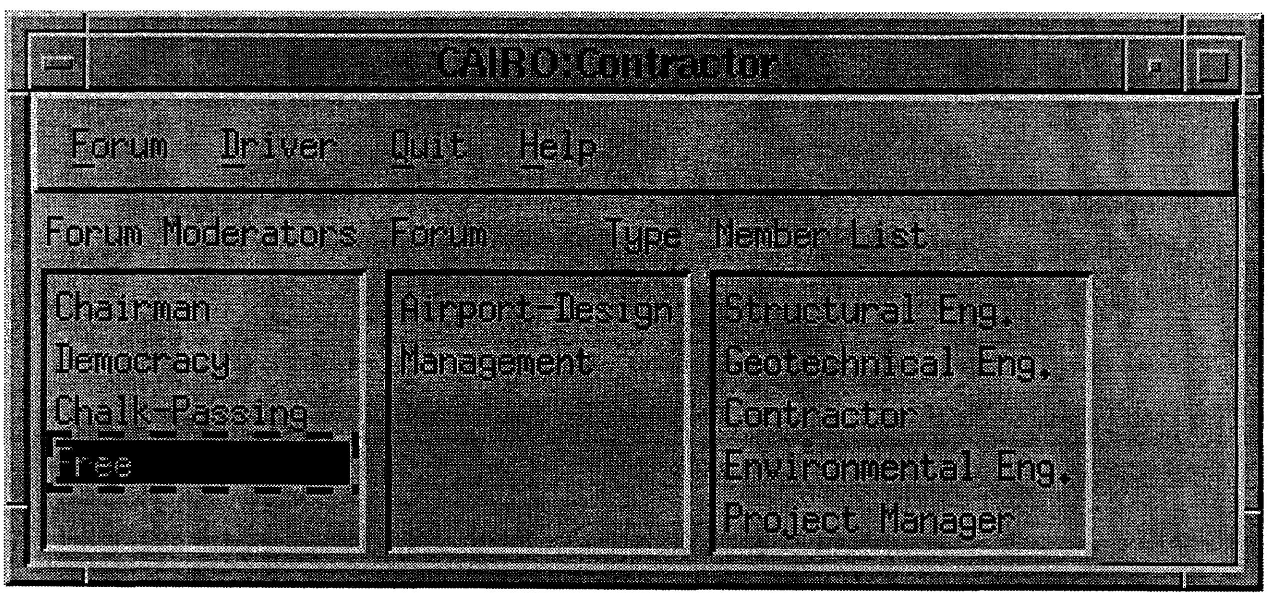

Figure 6-2: The CAIRO Control Panel

Forum managers can also be used to transfer an existing forum from one control scheme to another. The forum manager loads the forum specification file from the existing forum and prompts the user for any additional information required by the new forum control scheme.

6.5.2 Forum Startup

Forum Servers are instantiated by forum managers. As described earlier forum managers extract the necessary parameters for forum instantiation from the forum creator. The forum manager stores all parameters in a file according to the format described in Appendix C. The forum server is then started as an independent process. Upon startup the server reads the parameter file and initializes all internal objects accordingly. The server then register's itself with the name server. It is then ready to accept any login or membership requests from users of the CAIRO system.

The forum server maintains a membership list that includes an identification of each member's state. A forum member can be in any of the four states described below.

1. Member - the user has been specified as a person who is allowed to join the forum.

2. Logged In (active) - the user is actively engaged in a forum discussion.

3. Waiting to Speak - the user has requested the right to speak but has not yet acquired the enabling token.

4. Speaking - the user has the floor (i.e. the user possesses a speech token) and has the ability to transmit information to any number of forum members

Each state described above assumes that the user was in the preceding state before transition.

6.5.3 Participant Startup

Users of the CAIRO system must each start a collaboration manager (CM) process on their workstations. The manager provides an interface/control panel to the CAIRO distributed conferencing system. Upon startup, the CM registers with the nameserver. The CM then requests a list of the available forum managers and forum servers from the nameserver. Finally, the information is displayed in the first two list boxes in the CAIRO control panel. The control panel also provides the following functionality:

1. Local conference logging control (including recording and retrieval).

2. Screen capture.

3. Forum server creation via the forum managers.

4. Instantiation of media drivers according to the local workstation's capabilities.

6.5.4 Accessing Forums

Once the two key components (i.e. forum servers and collaboration managers) are running, conferences can be started on the CAIRO system. The initial step in entering a conference is accessing a specified forum. This can be done by simply clicking on the appropriate forum name in the forum list box in the CAIRO control panel. Once a forum is selected a login message is sent to the forum server, whose address has been supplied by the name server. The forum server then determines if the participant logging in has the appropriate access rights (i.e. the participant is on the membership list for a closed forum). An acknowledgement is returned to the collaboration manager if the user has been successfully logged in, otherwise a rejection message is transmitted to the user. Furthermore, if the login was successful, the forum server's *active* list is updated and all *active* members of the forum are informed of the addition to the community.

6.5.5 Retrieving Active/Pending List

The active member list box on the right side of the CAIRO control panel shows the currently logged in members of the forums highlighted in the forum list box. As described in the section above the forum server automatically updates all active members when any forum members have logged in or logged out (the messages involved are described in Appendix B). Two additional windows are generated when a user logs in to a forum. The first window provides speech request buttons for the three speech acts described in Section 2.2.1. The other window displays the queue of members of the forum that are waiting to speak.

6.5.6 Requesting to Speak

Speech requests on the CAIRO system involve two steps: selecting the audience and describing the speech intent. Audience selection simply involves selecting the appropriate recipients from the active member list box on the CAIRO control panel. Forums that do not allow side conversations will automatically have all items highlighted in the active member list box. A speech intent is indicated by pressing one of the speech request buttons.

As soon as a speech request button is depressed token requests are sent to the forum server. A token request is sent for each highlighted member in the active member list box. The forum server then processes the token requests. The server's response is dependent on the forum control scheme that is encoded in the forum server. According to the control scheme the forum server decides whether to place the speaker request on the pending queue or to automatically grant tokens to the requestor. For example, in a chairman controlled scheme, all requests are placed on the pending queue. When the chairman allows a specific user to speak, his/her name is transferred from the pending queue to the speaking queue and tokens are granted to the user. Any changes in the contents of either the pending queue or speaker queue are automatically broadcast to all members of the forum.

6.5.7 Communicating with other Participants

Once the previous steps have been completed successfully (i.e. a participant logs onto an existing forum server and is granted one or more communication tokens) real time multimedia information can be shared with other members of the forum. The user can then use any of the media drivers available (i.e. audio, text, X whiteboard) at his/her workstation to send data via all connections for which the user has tokens (the tokens act as keys that unlock a point to point connection). The data generated by the drivers is transformed into TCP/IP packets and tagged with a time stamp and frame sequence number. The data receiver then replays the packet as per the algorithm described in Section 5.5.

6.6 Concluding Remarks

All conference control mechanisms described above are generic and can be applied to any conference control scheme. Although only a limited set of control schemes has been implemented (see Section 6.4) simple tools are provided for control scheme extensions to the CAIRO system. Furthermore the tokenized control mechanism described in this chapter is highly efficient and eliminates any bottlenecks associated with a centralised communication and routing center.

Chapter 7

Example Scenario

The functionality of the CAIRO architecture described in Chapters 4–6 can be better illustrated through an example conferencing scenario. A bridge design problem is introduced in Section 7.1 and will be used a basis for the scenario. Section 7.2 describes the interactions between the design group members that would typically occur in the design of the bridge. Finally, Section 7.3 will illustrate how the CAIRO tool will enable effective communication among the design group members to achieve their design goal.

7.1 Definition of Task

A design team comprised of a structural engineer (SE), an environmental engineer (EE), a geotechnical engineer (GE) and a contractor (CN) have been charged with the task of designing a bridge across the Sollecks river (See [White et al., 1972] for a more detailed description of the design problem). The professional background of each team member is clearly different and each member provides valuable insight into key design decisions (eg. the GE provides the team with information regarding the load that the soil can bear.) Furthermore, in the AEC (Architecture, Engineering, Contracting) industry, the members of the design team would be employees of separate professional companies collaborating on the overall bridge design project. It is assumed that these separate companies would not be located in the same area, in fact they are typically not even in the same state.

Coordination of the bridge design among the professionals is a difficult and expensive task due to their geographic distribution. Design team meetings would incur significant travel and time expenses. Furthermore, the number of meetings would be limited (perhaps once a month) thereby decreasing the productivity of the whole group since the design loop is much larger. The design loop length is critical in a large-scale design project due to the iterative nature of the design process. The following section describes some of the interaction loops involved in the design of the bridge.

7.2 Group Interaction

Design teams in the AEC industry are generally centered around the structural engineers who have the final word in design decisions. In this bridge design problem it is assumed that the SE is central to the design process and generates most design plans. During each design loop, the initial bridge schematic generated by the SE is reviewed by each of the other professionals. The SE then adjusts the design according to the suggestions of the other team members and repeats the design loop until a design is ratified by the whole team. A small set of interactions are enumerated below:

(I) The SE designs a bridge consisting of a single spanning member and a support pillar in the center of the river (see Figure 7-1). A meeting would be arranged among the team members to discuss the design or alternatively the schematic would be sent to each professional for comments. The other members would generate the following comments:

Geotechnical Engineer: The design is reasonable.

Environmental Engineer: The central pier through the river would adversely affect the aquatic life in the river.

Contractor: Suggests the use of steel girders instead of a central pier due to construction problems.

(II) Given these comments the SE returns to the drawing board. He/she then generates a second design (see Figure 7-2) that involves a single pre-stressed concrete spanning member supported at the ends since steel girders are not a viable option. This design must be reviewed by his peers again resulting in the following recommendations:

Geotechnical Engineer: The design is reasonable.

Environmental Engineer: The design is environmentally friendly.

Contractor: Rejects the design on the basis that no facilities are available on site for manufacturing such a large spanning member of pre-fabricated concrete, nor can it be transported due to its large size.

(III) The SE then revises the design again resulting in the design shown in Figure 7-3. The bridge now consists of smaller pre-stressed concrete members and two vertical pier supports. The design loop is then reiterated resulting in the following recommendations from the other members of the team:

Geotechnical Engineer: The design is rejected due to the large shearing force in the slope rock.

Environmental Engineer: The design does not seriously affect the aquatic life.

Contractor: The proposed design is feasible.

(IV) Finally, the SE arrives at the design shown in Figure 7-4. This design is composed of two concrete members and two inclined piers to remedy the problem with the shearing force generated by the earlier design. All members of the team then agree to the design and the drawings are approved.

The process described above is highly time consuming and inefficient. Each of the professionals operates in almost complete isolation with limited interaction. Full drawings must be generated before feedback can be solicited from the other members of the design team due to the expense associated with communication among the members. Furthermore, during the limited time in which they do meet, they would

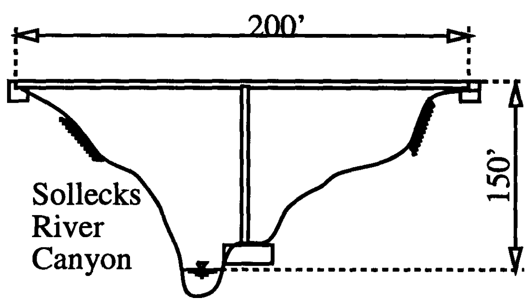

Figure 7-1: Bridge Design Phase I

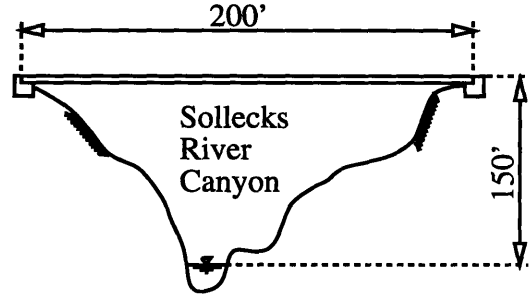

Figure 7-2: Bridge Design Phase II

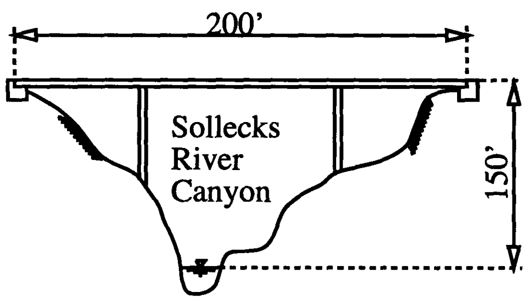

Figure 7-3: Bridge Design Phase III

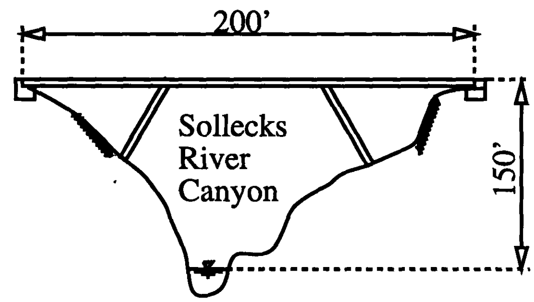

Figure 7-4: Bridge Design Phase IV

be typically away from their ideal work environment and can not make proper calculations without their analysis tools. The following section provides an alternative approach to the design process with the benefit of the CAIRO tools described in this thesis.

7.3 Execution of Task

The group interaction paradigm that is supported by the CAIRO platform is that of continuous interaction (i.e. virtual collocation). Hence, the design process described above would be radically changed with the introduction of the CAIRO conferencing system. Response to any designs by the SE would be immediate and incremental (i.e. the SE would not have to generate a full design before it could be reviewed by the other professionals.) This eliminates the waterfall model of design described in Section 7.2. The group could now be continuously in design meetings, commenting and participating when necessary.

The SE will still have a leading role in the design process and hence is designated as the moderator or chairman of the group. The overall collaboration process using CAIRO consists of three stages: Meeting Setup, Design Discussion, and Design Finalization.

7.3.1 Meeting Setup

The CAIRO meeting through which the professionals will interact presumes that each professional has an engineering workstation that is hooked up to the internet. Each participant in the meeting would then startup the CAIRO collaboration manager on each of their workstations. Upon startup of the collaboration manager, each user is registered with the name server (which is a process that is presumed to be always active and at a prespecified location). The collaboration manager then lists all existing forums and all members logged in to CAIRO as shown in Figure 7-5 and 7-6. The list of forums and members is provided by the nameserver to each participant. The two forums that already exist are irrelevant to the design problem and a new forum must be created to address the bridge design problem.

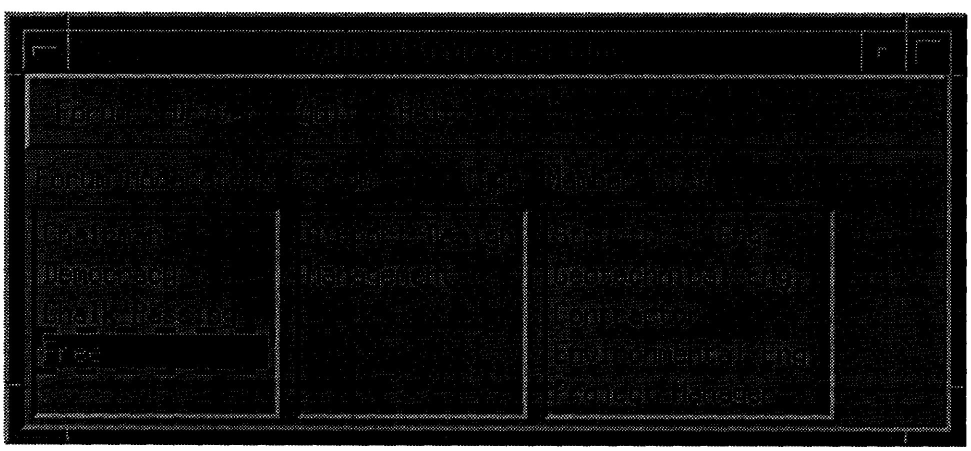

Figure 7-5: CAIRO Control Panel: Structural Engineer

Since the SE is the prinicipal coordinator, he/she must instantiate the forum server that will encompass the bridge design team. This is simply done by selecting the new forum option from the forum menu on his/her control panel. The SE is then asked to

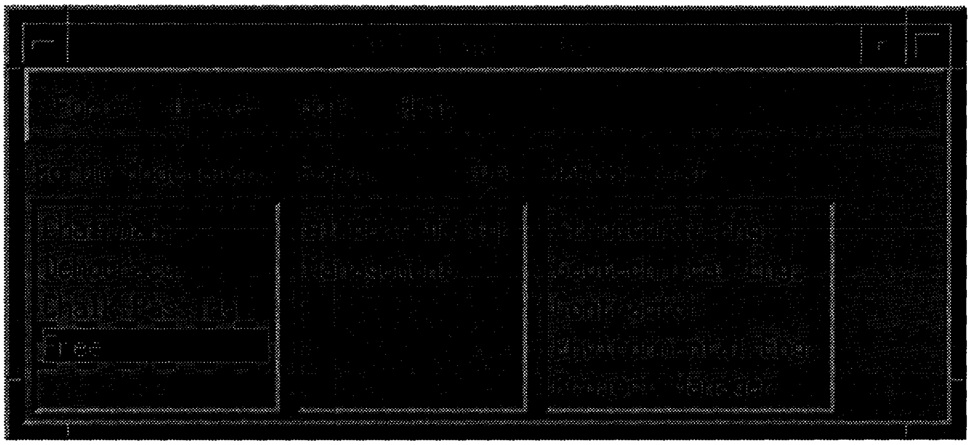

Figure 7-6: CAIRO Control Panel: Contractor

specify a set of parameters associated with the forum as shown in Figure 7-7. Once all the parameters have been entered and the SE clicks on the Create button, the forum Bridge-Design is instantiated. All the participants' screens will then reflect the existence of the new forum, since the new forum is registered with the name server (the name server has the responsibility of notifying all participants when any forum or participant registers.)

The members specified in the forum creation box are each requested via a pop up screen to join the forum. Joining the forum simply involves selecting the forum name Bridge-Design from the forum list. The collaboration manager then sends a login message to the forum server and the server will acknowledge the login if the participant is authorized to access the forum. Furthermore, three new windows appear that describe the status of each user as well as the list of pending speakers in the forum and a push-button panel that allows the participant to request to speak (See Figures ?? and 7-8). The forum now has a number of participants and can be used to exchange information among the members of the bridge design team.

7.3.2 Design Discussion

The forum initially created for the bridge design team is a 'free' forum. 'Free' forums as described in Section 6.4 allow any participant to address the forum at any time.

The only constraint on this type of forum is that only N speakers can speak at any one time ($N = 2$ was specified by the SE during forum creation.) This forum strategy is ideal in the brainstorming stage of design since it opens up the discussion to all participants. The SE could post the initial design shown in Figure 7-1 to the forum and each speaker could mark it up and suggest design alternatives. Figure 7-9 shows a typical session where a bridge design is marked up by the various professionals. Although the figure only shows textual and graphical interaction, verbal interaction also takes place among the various individuals in order to explicate some of the design constraints. The following section details the processes that take place behind the scenes during a conversation among the participants.

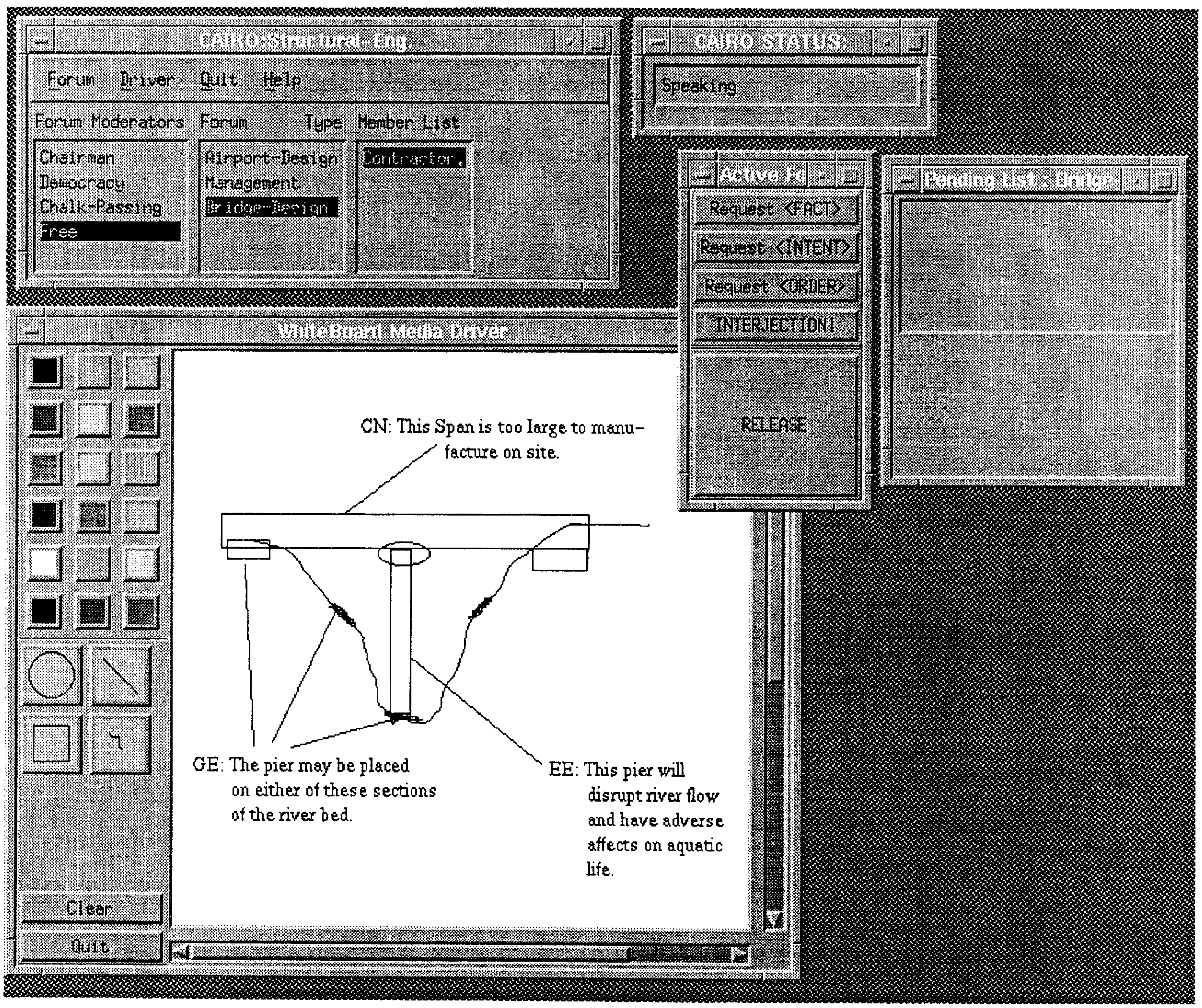

Figure 7-9: A typical CAIRO session with some markups

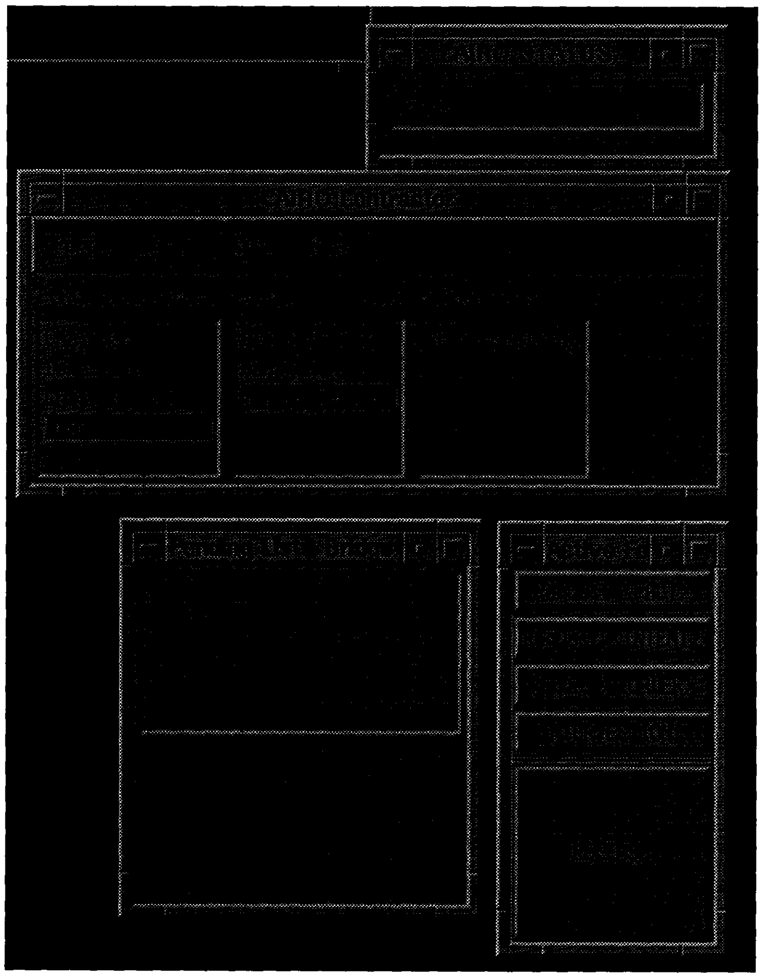

Figure 7-8: CAIRO Control Panel: Contractor

The only constraint on this type of forum is that only N speakers can speak at any one time ($N = 2$ was specified by the SE during forum creation.) This forum strategy is ideal in the brainstorming stage of design since it opens up the discussion to all participants. The SE could post the initial design shown in Figure 7-1 to the forum and each speaker could mark it up and suggest design alternatives. Figure 7-9 shows a typical session where a bridge design is marked up by the various professionals. Although the figure only shows textual and graphical interaction, verbal interaction also takes place among the various individuals in order to explicate some of the design constraints. The following section details the processes that take place behind the scenes during a conversation among the participants.

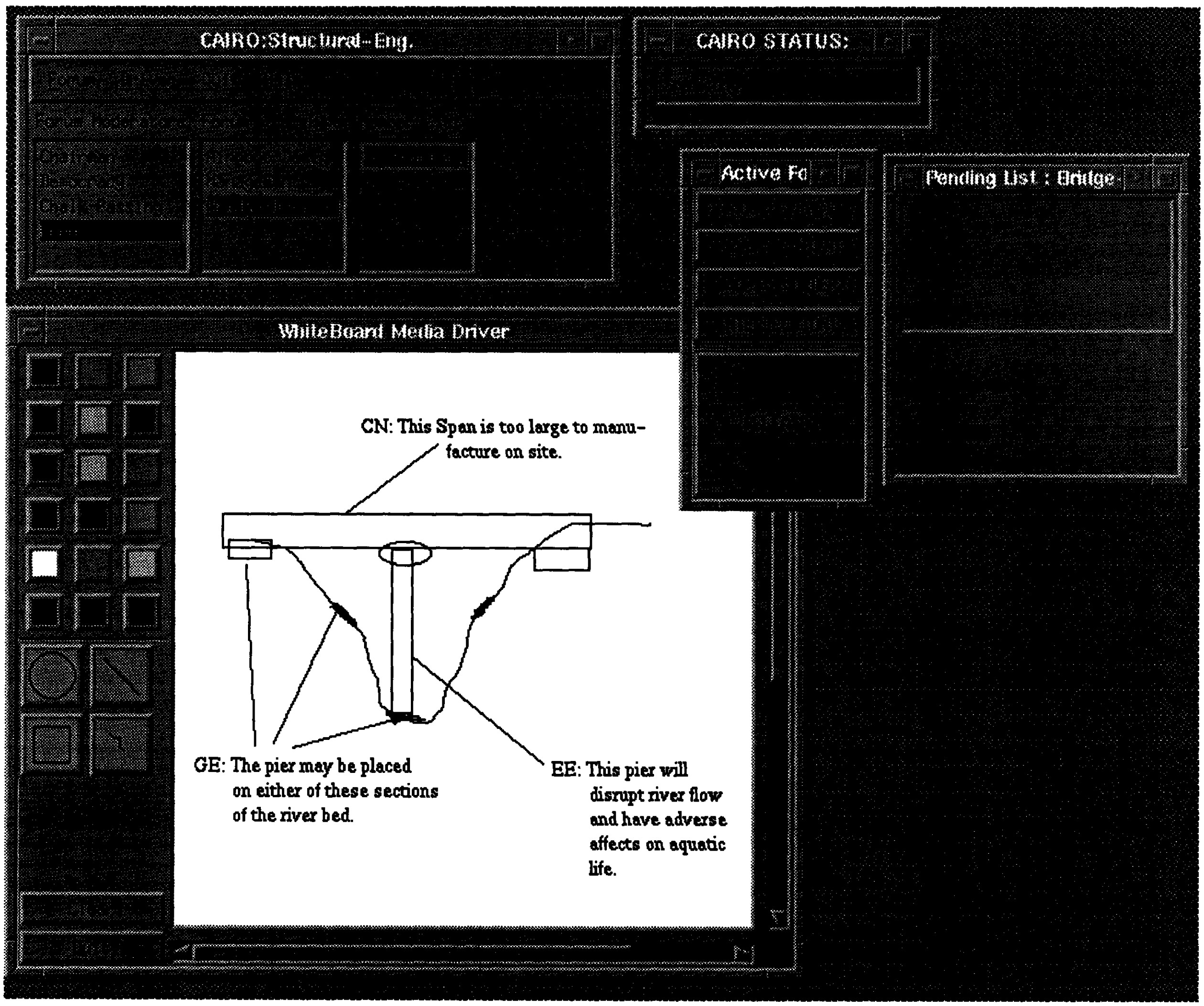

Figure 7-9: A typical CAIRO session with some markups

7.3.3 Details of a Conversation

For a detailed analysis of the conversation process, consider a request issued by the SE to the Geotechnical Engineer. The SE asks the GE to look into the possible locations where support piers could be placed along the river bank. The GE analyzes the soil mechanics and suggests several locations where the piers could be placed.

For the initial request posed by the SE, he/she would click on the button marked "Request Order". This initiates a token request by the SE's collaboration manager to the forum server. The message is received by the forum server and the SE is placed on the *pending* queue if two people are already having a conversation. When less than two people are conversing, the forum server issues a conversation token to the SE's collaboration manager. This unlocks a channel of communication between the SE and the GE. The status bar on the SE's display will indicate that the SE is in "Speaking" mode. The SE would then speak into the workstation's microphone as well as circle the points of interest on the river bed. The audio data would be digitized and packetized by the audio driver and labeled with a frame sequence number. The data from the X whiteboard is also packetized and labeled. The complete frame would then be sent out to the GE upon compilation. The GE's collaboration manager would then receive the packets, reorder them according to sequence and frame number and replay the data in a synchronized fashion (i.e. retaining the original temporal relations among the data elements). The SE would then release the token by clicking on the "Release!" button (this causes the collaboration manager to return the token to the forum server).

The GE would then perform an analysis on the soil using any necessary computational tools on his/her workstation and click on the "Request Fact" button when his answer to the SE's query is ready. Again the forum server is notified of the GE's intent to speak. A token is awarded to the GE as soon as one is available to the forum server. The GE's status line would then show a speaking message and the GE would circle the areas where a pier could be placed. The GE could also describe verbally the reasoning behind his/her decision. The packets would then be transmitted as described for the SE case. Finally, the GE would release the token as soon as his/her input has been completed.

7.3.4 Design Finalization

In the discussion phase described above the SE would have arrived at the design shown in Figure 7-4. At this stage, the SE may require more detailed recommendations from each of the professionals. Since these details are of minimal concern to most of the professionals, the requests may be directed to specific participants in a more organized fashion. During this phase the SE opts for a "Chairman" controlled forum where he/she can focus the group's efforts more efficiently. Switching the forum control strategy simply involves selecting the forum and selecting modify forum from the forum menu (see Figure 7-10). The necessary parameters for the new forum type are then entered and all other attributes of the new "Chairman" forum will be inherited from the previous "Free" forum.

The SE can then address questions to any of the participants simply by selecting the receivers from the member list and clicking on one of the speech request buttons. For example, the SE may be interested in knowing from the GE the maximum depth the piers can be driven into the river bed. The SE will automatically be allocated a token by the forum server since he/she is the chairman. Any other participants requesting to speak will be placed on the *pending* queue by the forum server. The chairman then selects those participants he/she wishes to hear by selecting their name from the pending list on his/her control panel. The forum server then issues tokens to the specified participant, thereby enabling the participant to take the floor.

7.4 Concluding Remarks

The new design process enabled by the CAIRO tools (as described in Section 7.3) provides a significant improvement over the original cumbersome process described

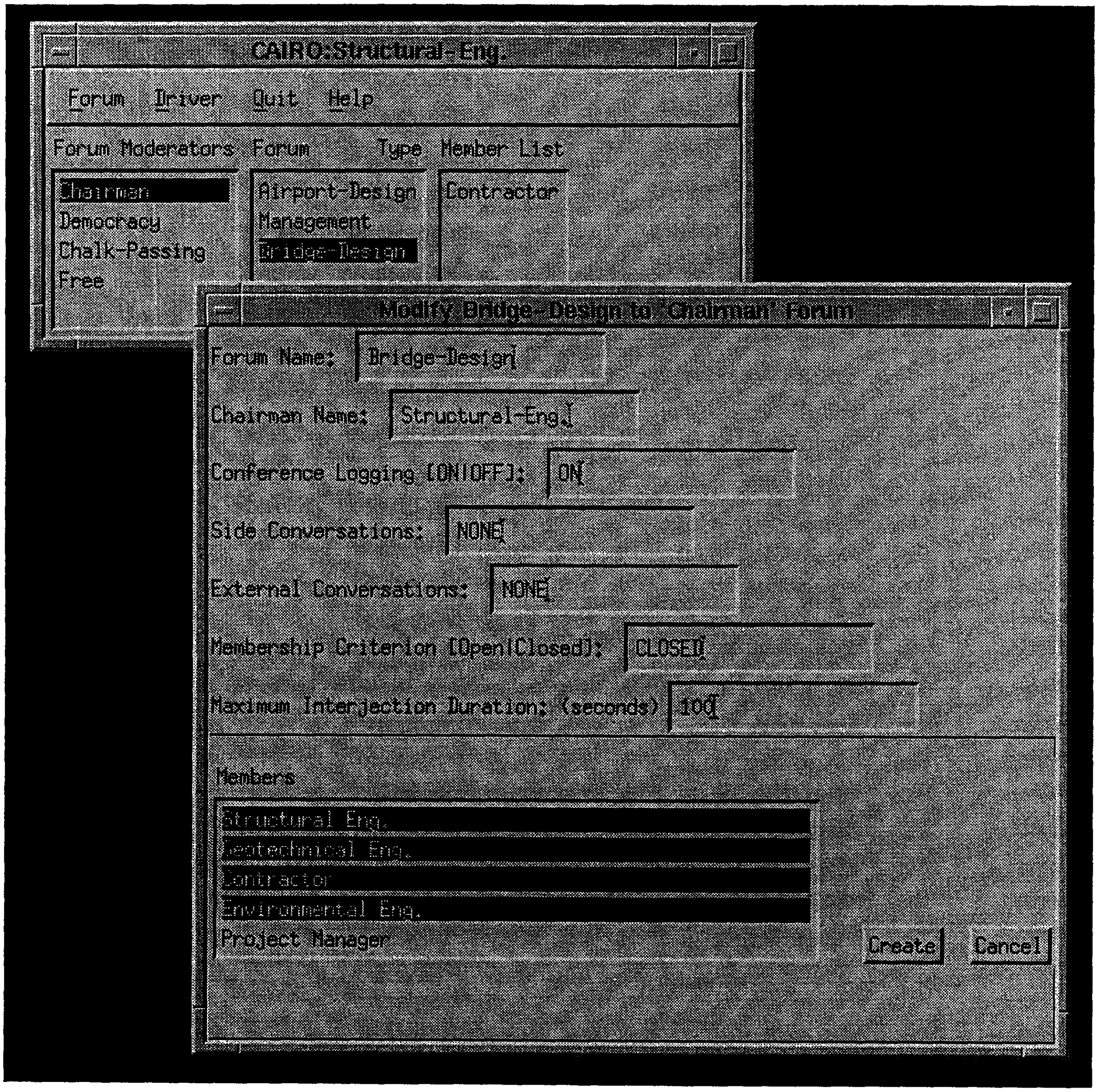

Figure 7-10: Forum Modification in CAIRO: Bridge Design Case (Structural Engineer Perspective)

in Section 7.2. All professionals are available on request throughout the design process thereby reducing unnecessary design effort on erroneous paths. Furthermore, structured and controlled conferencing enables more efficient interaction among the members since it aids in focusing the design process.

Chapter 8

Conclusions and Future Work

8.1 Concluding Remarks and Results

The CAIRO system provides computer support for a well structured conferencing mechanism with synchronized multimedia communication among the participants. Key features of the CAIRO architecture described in this thesis are enumerated below:

1. Media drivers can easily be added to the collaboration manager to support more media devices.

2. The multimedia synchronization engine is a generic implementation that can support any number of media drivers using any underlying network infrastructure.

3. Network protocols are decoupled from the system. Only the communication manager objects must be modified to comply to any network infrastructure.

4. The system supports any number of forums that can be organized in a hierarchical structure.

5. Multiple collaboration schemes can be used to control any given forum. Furthermore, new collaboration control methodologies can easily be coded and attached to the CAIRO system.

6. CAIRO supports a robust mechanism for controlling conversations among individuals in a conference. The token mechanism provides the forum with full control over each participant's communication connections without creating a communication bottleneck.

7. A name server is provided to help establish connections among participants in the CAIRO system.

Although, the CAIRO system was initially designed for the engineering domain, non of the mechanisms are engineering specific. Thus the system can be used for any distributed conferencing application (eg. business tele-conferencing).

The CAIRO conferencing tool has the potential to greatly enhance collaborative work, and could significantly reduce costs and increase the productivity of engineering firms. CAIRO allows professionals to hold efficient meetings with floor control strategies that can be modified as the group evolves. Furthermore, more complex floor control strategies can be implemented in CAIRO in accordance with the latest management dictums.

The CAIRO conference system facilitates the adoption of specified workflow management techniques. Minimal training of the employees will be required since much of the coordination control and process management will be maintained by the computer conference infrastructure.

Finally, corporate memory will be retained and proper documentation of decisions will be maintained since all team interaction will be recorded on computer. Corporate experience and expertise can then easily be queried at any later date. Employee effectiveness can also be easily evaluated by review of the proceedings of his/her interaction with the group.

The CAIRO system was implemented in C++ on Sun Sparc platform and complies to the OMT object design paradigm [Rumbaugh et al., 1991]. The user interface is written in X11R5/Motif and the audio drivers are based on the AudioFile extensions to X11. The CAIRO system is easily portable to most Unix environments.

8.2 Future Work

Future work includes identifying more complex moderation schemes. These schemes are dependent on the collaboration primitives currently defined within the forum model. Further research is required to derive the canonical collaboration primitives required by the CAIRO system. Tools to build forum moderation schemes could also be built to allow individuals to quickly design their own control strategy.

The current implementation has limited multi-media capabilities. As network bandwidth limits decrease and compression technologies are enhanced, live video driver could be incorporated within the CAIRO system. Furthermore, a generic X application sharing driver would also be a very useful addition to the system.

Currently, the informal design discussions supported by CAIRO are decoupled from the formal design specification process supported by DICE. Future extensions should include hooks between the informal design discussions recorded by CAIRO and the more formal design specifications encoded in the DICE work-space. This requires an exploration of the storage and representational structures required for the informal multimedia information exchanged during a conference. More intelligent repositories would allow effective archiving of conversations and easy retrieval of relevant information.

Appendix A

Object Hierarchy Diagram

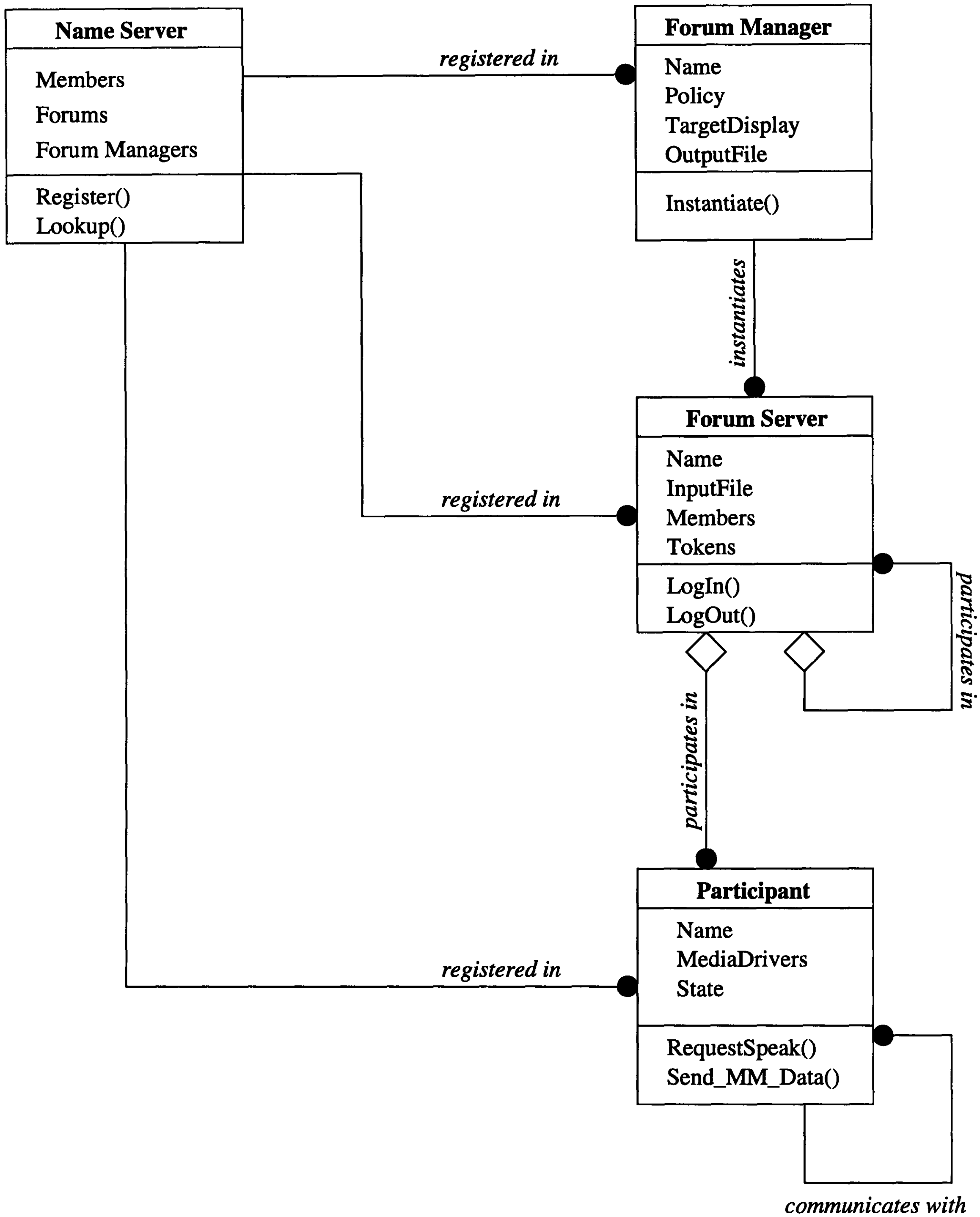

Figure A-1: Object Diagram: Overall Server Interconnectivity

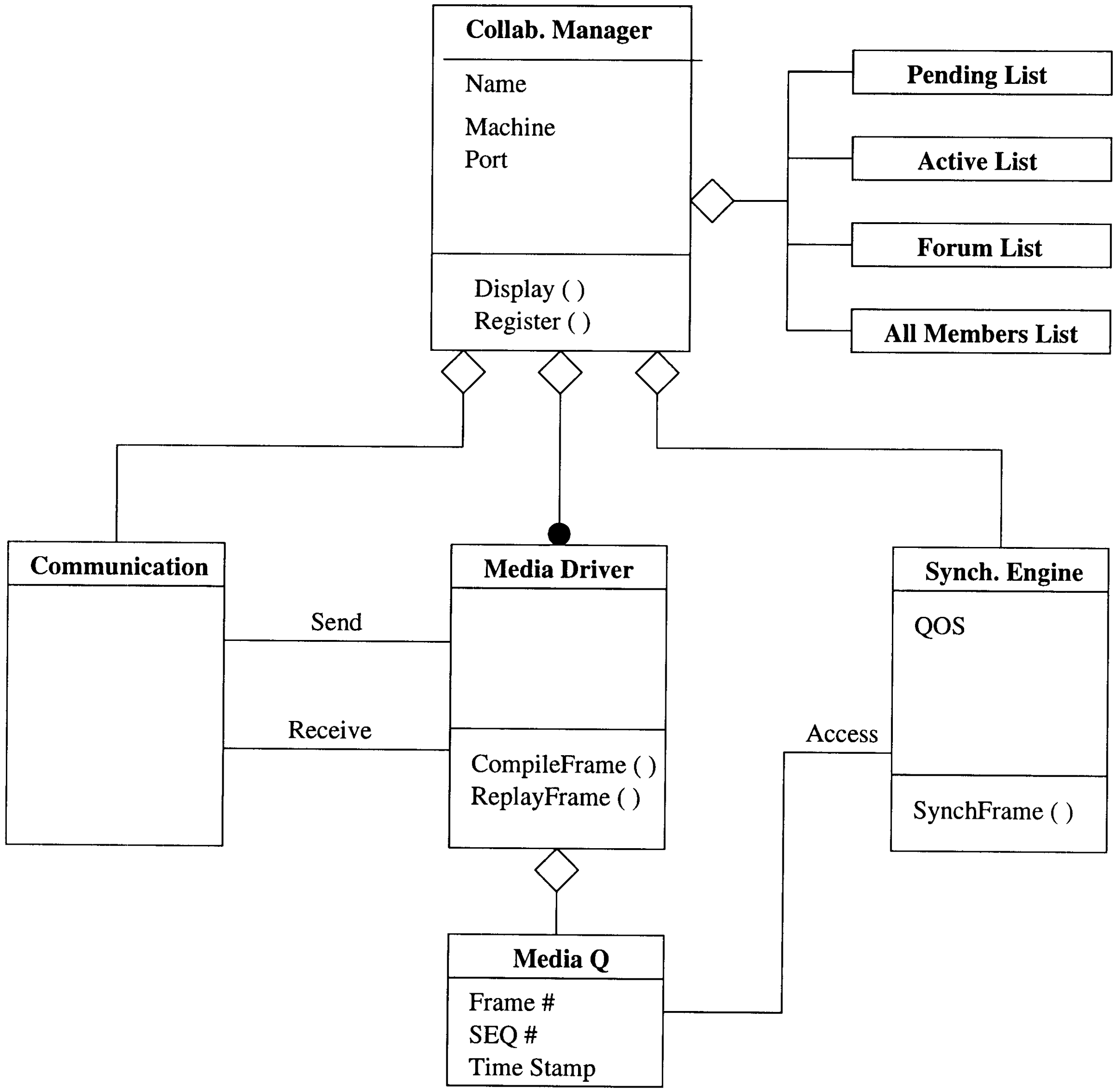

Figure A-2: Object Diagram: Components of a CAIRO participant

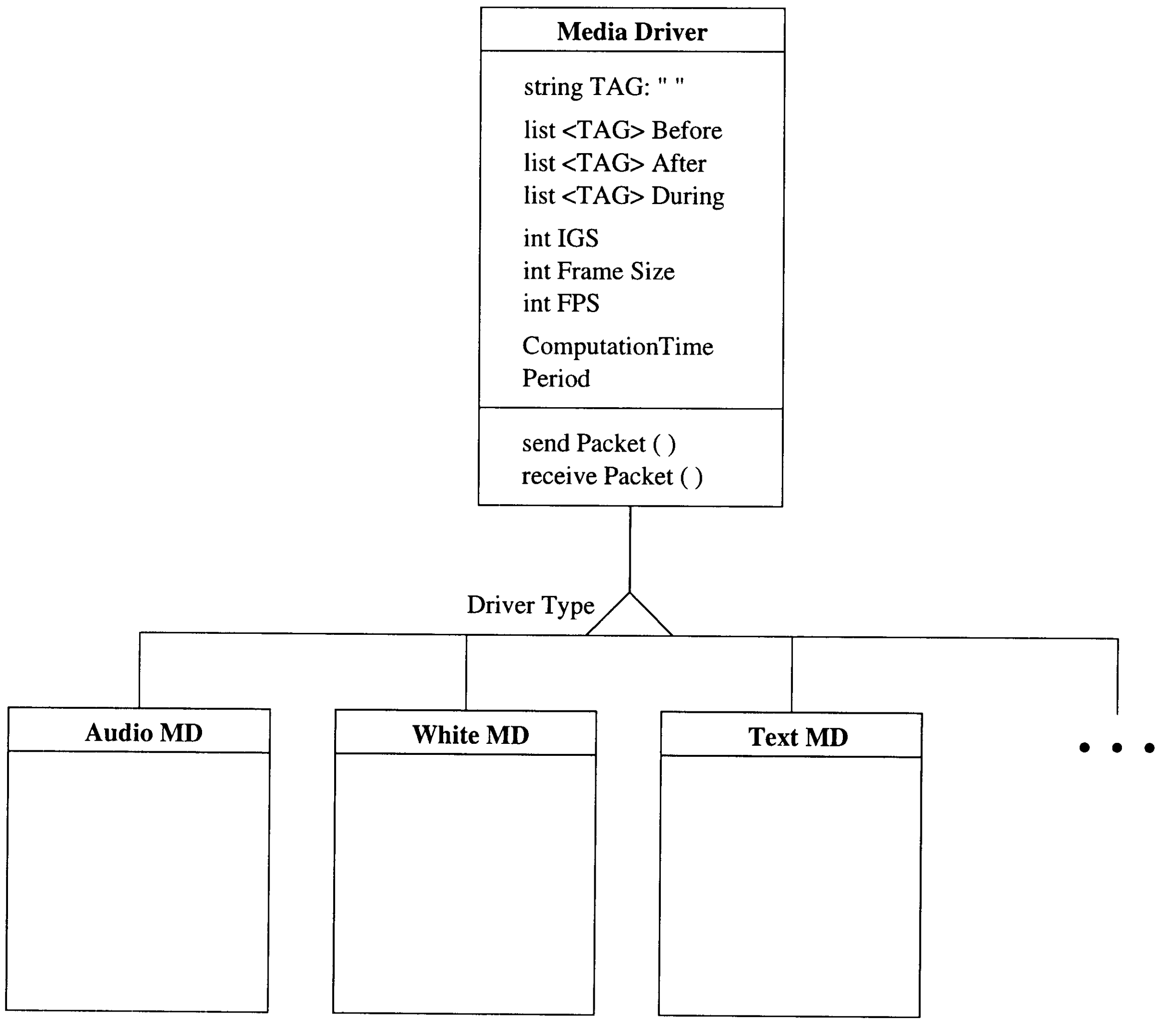

Figure A-3: Object Diagram: Media Driver classes

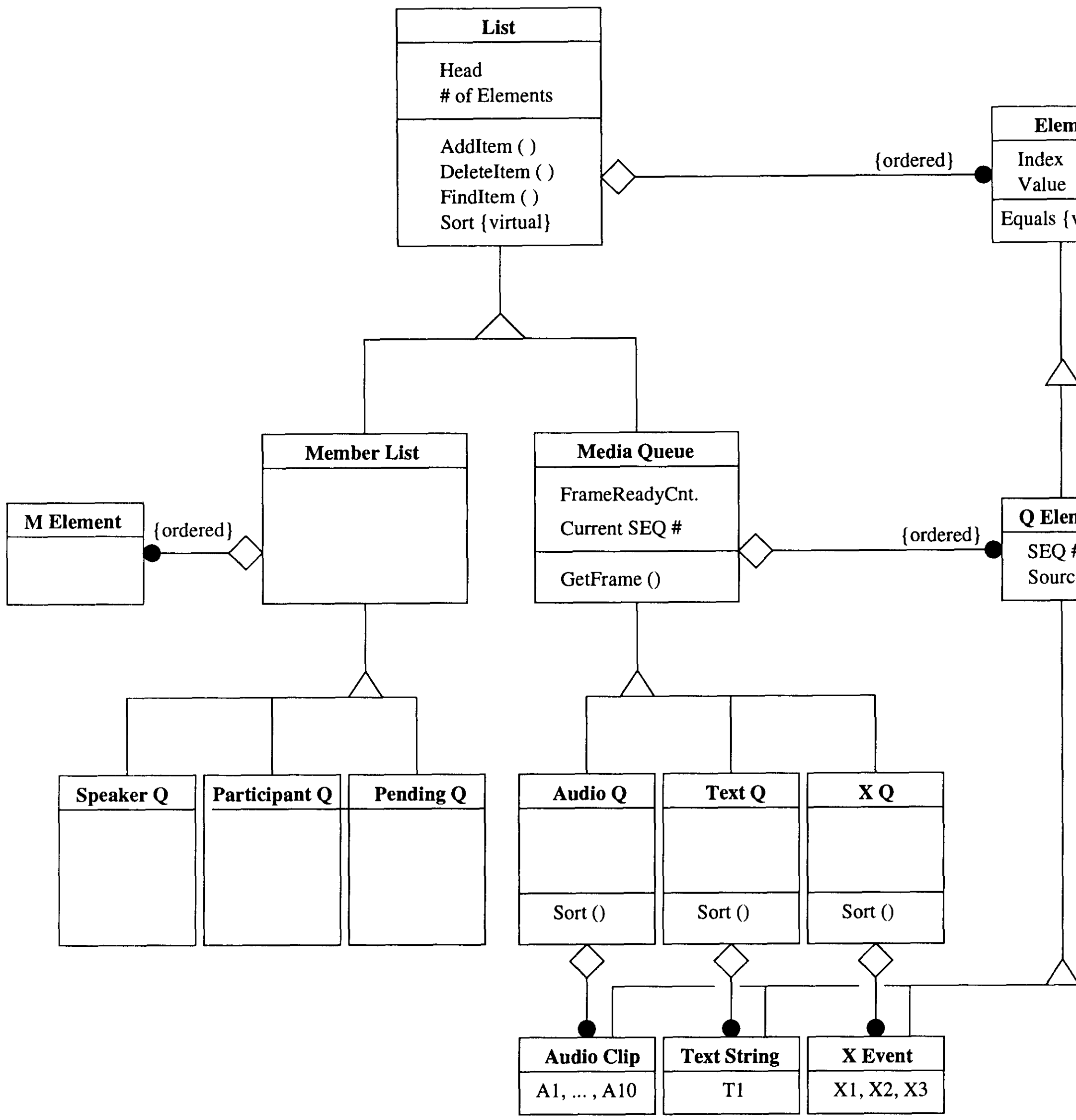

Figure A-4: Object Diagram: Queue Structure Hierarchy

Appendix B

Message Protocol

Table B.1: NameServer Output Messages

Prefix	ARG1	ARG2	ARG3	ARG4	Comment
FF	Name	Machine	Port		Sends a forum's complete directory information in response to a CAIRO user query
FU	Name	Machine	Port		Send a user's complete directory information in response to a CAIRO user query
GU	User #	Total #	Name	Machine, Port	Returns list of CAIRO users.
GF	Forum #	Total #	Name	Machine,Port	Returns list of active CAIRO forums.

Table B.2: NameServer Input Messages

Prefix	ARG1	ARG2	ARG3	Comment
RU	User Name	Machine	Port	Registers a user with the nameserver.
RF	Forum Name	Machine	Port	Registers a forum with the nameserver.
DU	User Name	Machine	Port	Removes a user from the nameserver.
DF	Forum Name	Machine	Port	Removes a forum from the nameserver.
LU	User Name			Logs in a registered user with the nameserver, the user is now actively using CAIRO.
LF	Forum Name			Logs in a registered forum with the nameserver, the forum is now actively using CAIRO.
OU	User Name			Logs out a registered user from the nameserver, the user is no longer actively using CAIRO.
OF	Forum Name			Logs out a registered forum from the nameserver, the forum is no longer actively using CAIRO.
GU	Machine	Port		Request List of all Active users from the Nameserver.
GF	Machine	Port		Request List of all Active forums from the Nameserver.
FU	Search Name	Machine	Port	Request machine and port number of the user Search Name from the Nameserver.
FF	Search Name	Machine	Port	Request machine and port number of the forum Search Name from the Nameserver.

Table B.3: Forum Server Output Messages

Prefix	ARG1	ARG2	ARG3	ARG4	Comment
RF	Name	Machine	Port		Sends a forum registration message to the Name Server.
DF	Name	Machine	Port		Sends a forum removal message to the Name Server.
AK	Name	Machine	Port		Acknowledge a user login.
RE	Name	Machine	Port	Reference	Refuses a user login and sends a reference e-mail address to request membership in the forum.
UA	User #	Total #	Name	Machine, Port	Sends a list of all active users.
UR	User #	Total #	Name	Machine, Port	Sends a list of all pending speakers.
K	From	To	Expiry	Type	Provides a conversation token to a user.
L	From	To	Type		Force a retrieve of a token from a CAIRO user.

Table B.4: Forum Server Input Messages

Prefix	ARG1	ARG2	ARG3	Comment
A	User Name	Machine	Port	Registers a user with the Forum.
D	User Name			Removes a user from the Forum.
R	From Name	To Name	Token Type	Request a speech token from the Forum. Enforces collaboration control.
N	From Name			Releases a speech token from the Forum so that it can be re-used.

Table B.5: Collaboration Manager Output Messages

Prefix	ARG1	ARG2	ARG3	Comment
M?	Frame #	SEQ #, Time	Data	Transmits messages to another participant's media drivers where ?=T,D,A and T = Text Media Driver, D = Whiteboard Media Driver and A=Audio Media Driver
RU	User Name	Machine	Port	Registers the user with the nameserver.
DU	User Name	Machine	Port	Requests removal of a user from the nameserver.
LU	User Name			Logs in a registered user with the nameserver, the user is now actively using CAIRO.
OU	User Name			Logs out a registered user from the nameserver, the user is no longer actively using CAIRO.
GU	Machine	Port		Request List of all Active users from the Nameserver.
GF	Machine	Port		Request List of all Active forums from the Nameserver.
FU	Search Name	Machine	Port	Request machine and port number of the user Search Name from the Nameserver.
FF	Search Name	Machine	Port	Request machine and port number of the forum Search Name from the Nameserver.
A	User Name	Machine	Port	Registers a user with the Forum.
D	User Name			Requests the forum server to removes the collaboration manager from the Forum user list.
R	From Name	To Name	Token Type	Request a speech token from the Forum Server. Enforces collaboration control.
N	From Name			Returns a speech token to the Forum Server once the user has completed his speech.

Table B.6: Collaboration Manager Input Messages

Prefix	ARG1	ARG2	ARG3	ARG4	Comment
M?	Frame #	SEQ #, Time	Data		Receives messages from another participant's media drivers where ?=T,D,A and T = Text Media Driver, D = Whiteboard Media Driver and A=Audio Media Driver
FF	Name	Machine	Port		Sends a forum's complete directory information in response to a Find Forum request to the name server
FU	Name	Machine	Port		Receive a user's complete directory information in response to a Find User request to the name server.
GU	User #	Total #	Name	Machine, Port	Receives a list of CAIRO users.
GF	Forum #	Total #	Name	Machine, Port	Receives a list of forums that are registered with the name server.
UA	User #	Total #	Name	Machine, Port	Receives a list of all active users.
UR	User #	Total #	Name	Machine, Port	Receives a list of all pending speakers.
AK	Name	Machine	Port		Acknowledge a user login.
RE	Name	Machine	Port	Reference	Refuses a user login and sends a reference e-mail address to request membership in the forum.
K	From	To	Expiry	Type	A token is received that allows conversation between the users and the person specified by To.
L	From	To	Type		Forces the collaboration manager to remove the conversation token associated with the (From,To) conversation.

Appendix C

Forum File Format

Line 0:<FORUM NAME> [Name of Forum]

Line 1:[Number of Members]

Line 2:

Line 3:<NAME> [Name of Member1]

Line 4:<MACHINE> [Member1 Machine]

Line 5:<PORT> [Member1 Port]

Line 6:<NUM DRIVER> [Number of Drivers Supported]

Line 7:<DRIVER NAME> [Name of Driver1]

Line 8:<DRIVER NAME> [Name of Driver2]

.

.

.

Line M+1:<MEMBER TYPE> [Type of Member1]

Line M+2:

Line M+3:<NAME> [Name of Member2]

Line M+4:<MACHINE> [Member2 Machine]

Line M+5:<PORT> [Member1 Port]

Line M+6:<NUM DRIVER> [Number of Drivers Supported]

Line M+7:<DRIVER NAME> [Name of Driver1]

Line M+8:<DRIVER NAME> [Name of Driver2]

```
.
.
.
Line   N:<FORUM TYPE> [Type of Forum]
Line N+1:<CHAIRMAN> [Name of Chairman]
Line N+3:<SPEECH DURATION> [Max. Speech Duration]
Line N+4:<INTER DURATION> [Max. Interjection Duration]
Line N+5:<NUM SPEAKER> [Max Number of SimultaneousSpeakers]
Line N+6:<SIDE CONVERSE> [Side Conversations Allowed?]
Line N+7:<EXT CONVERSE> [External Conversations Allowed?]
Line N+8:<LOG MODE> [Logging Mode]
Line N+9:<LOG FILE> [Name of Forum Log File]
Line N+10:
Line N+11: Any additional parameters to be specified for a Forum ......
```

Bibliography

[Abdel-Wahab, 1993] Abdel-Wahab, H. (1993). Reliable information service for internet computer conferencing. In *Second Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pages 128–142.

[Ahuja and Ensor, 1992] Ahuja, S. R. and Ensor, J. R. (1992). Coordination and Control of Multimedia Conferencing. *IEEE Communications Magazine*, pages 38–43.

[Babadi, 1990] Babadi, A. (1990). Cooperative multiuser interface to x-applications. Master's project, West Virginia University, Department of Computer Science.

[CMU S.E.I., 1993] Carnegie Mellon University Software Engineering Institute, (1993). *A Practitioner's Handbook for Real-Time Analysis – Guide to Rate Monotonic Analysis for Real-Time Systems.* Kluwer Academic Publishers, Boston.

[Cole and Nast-Cole, 1992] Cole, P. and Nast-Cole, J. (1992). A primer on group dynamics for groupware developers. In Marca, D. and Bock, G., editors, *Groupware: Software for Computer-Supported Collaborative Work*, pages 44–57. IEEE Computer Society Press.

[Comer and Stevens, 1993] Comer, D. E. and Stevens, D. L. (1993). *Internetworking with TCP/IP.* Prentice Hall, Englewood Cliffs, NJ.

[Craighill et al., 1992] Craighill, E., Lang, R., and Garcia-Luna, J. (1992). Environments to enable informal collaborative design processes. In *Proceedings of the CE & CALS Conference*, Washington D.C.

[Doyle and Straus, 1993] Doyle, M. and Straus, D. (1993). *How to Make Meetings Work*, Berkeley Books.

[Egido, 1988] Egido, C. (1988). Video conferencing as a technology to support group work: A review of its failures. In *Proceedings of the Second Conferenct on Computer-Supported Cooperative Work*, pages 13–24, Portland, Oregon.

[Ellis et al., 1991] Ellis, C., Gibbs, S., and Rein, G. (1991). Groupware: Some issues and experiences. *Communications of the ACM*, 34(1):38–58.

[Garcia-Luna et al., 1987] Garcia-Luna, J., Craighill, E., and Lang, R. (1987). Mosaic–a model for computer supported collaborative work. In *Proceedings of IEEE MILCOM '87*, Washington D.C.

[Garcia-Luna et al., 1989] Garcia-Luna, J., Craighill, E., and Lang, R. (1989). Floor management and control for multimedia computer conferencing. In *Proceedings of MULLTIMEDIA '89*, Ottawa, Ontario.

[Gomaa, 1993] Gomaa, H. (1993). *Software Design Methods for Concurrent and Real-Time Systems.* Addison-Wesley Publishing Company, Reading, MA.

[Gorti and Sriram, 1993] Gorti, S, R. and Sriram, D. (1993). CONGEN: An Integrated Approach to Conceptual Design. *International Journal of CAD/CAM and Computer Graphics*, 8(2):135–150. Special AI issue.

[Hussein et al., 1995] Hussein, K., Peña-Mora F. and Sriram D. CAIRO: A System for Facilitating Communication in a Distributed Collaborative Engineering Environment. Fourth IEEE Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, April 20, 1995.

[InVision, 1994] InVision (1994). *InVision User Manual.* InVision Systems Corporation, release 3.0 edition.

[Little and Ghafoor, 1993] Little, T. D. C. and Ghafoor, A. (1993). Interval-based conceptual models for time-dependent multimedia data. *IEEE Transactions on Knowledge and Data Engineering*, 5(4):551–563.

[Marshak, 1990] Marshak, D. S. (1990). Lotus notes: A platform for developing workgroup applications. In *Patricia Seybold's Office Computing Report.*

[NCSA Collage, 1994] NCSA Collage (1994). *NCSA Collage for the X Window System User's Guide.* National Center for Supercomputing Applications, release 1.2.1 edition.

[Nunamaker et al., 1991] Nunamaker, F. F., Dennis, A. R., Valacich, J., Vogel, D. R., and George, J. F. (1991). Electronic meeting systems to support group work. *Communications of the ACM*, 34(7):40–61.

[Peña Mora, 1994] Peña Mora, F. (1994). *Design Rationale for Computer Supported Conflict Mitigation during the Design-Construction Process of Large-Scale Civil Engineering Systems.* PhD thesis, Massachusetts Institute of Technology.

[Peña Mora et al., 1995] Peña Mora, F., Sriram, D., and Logcher, R. (1995). Design Rationale for Computer Supported Conflict Mitigation. *AI EDAM – Special Issue on Concurrent Engineering*, 9(2).

[Popek, 1974] Popek, G. J. (1974). Protection Structures. *Computer*, 7(6):22–23.

[Ravindran and Bansal, 1993] Ravindran, K. and Bansal, V. (1993). Delay compensation protocols for synchronization of multimedia data streams. *IEEE Transactions on Knowledge and Data Engineering*, 5(4):551–563.

[Rumbaugh et al., 1991] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., and Lorensen, W. (1991). *Object-Oriented Modeling and Design.* Prentice-Hall, Inc., Englewood Cliffs, NJ.

[Searle, 1969] Searle, J. (1969). *Speech Acts: an Essay in the Philosophy of Language.* Cambridge University Press, Cambridge, England.

[Silberschatz et al., 1991] Silberschatz, A., Peterson, J., and Galvin, P. (1991). *Operating System Concepts.* Addison-Wesley Publishing Co., Inc., New York, NY.

[Srinivas et al., 1992] Srinivas, K., Reddy, R., Babadi, A., Kamana, S., Kumar, V., and Dai, Z. (1992). Monet: A multimedia system for conferencing and application sharing in distributed systems. Technical Report CERC-TR-RN-91-009, West Virginia University, Concurrent Engineering Research Center, West Virginia University.

[Sriram et al., 1994] Sriram, D., Gupta, A., Wong, A., Vemulapati, M., Gorti, S., Fromont, B., Su, V., and Vaidya, V. (1994). An Object-Oriented Knowledge Based Building Tool for Engineering Applications. IESL Technical Report IESL-91, MIT.

[Sriram and Logcher, 1993] Sriram, D. and Logcher, R. (1993). The MIT Dice Project. *Computer*, 26(1):64–65.

[Sriram et al., 1991] Sriram, D., Wong, A., , and He, L. (1991). GNOMES: An Object-Oriented Non-manifold Geometric Engine. IESL Technical Report IESL-91, MIT.

[Stefik et al., 1987] Stefik, M., Foster, G., Bobrow, D., Kahn, K., Lanning, S., and Suchman, L. (1987). Beyond the chalkboard: Computer support for collaboration and problem solving in meetings. *Transactions of the ACM*, 30(1):32–47.

[White et al., 1972] White, R., Gergely, P., and Sexsmith, R. (1972). *Structural Engineering.* John Wiley & Sons, Inc., NY, USA.

[Wong, 1993] Wong, A. (1993). *SHARED Workspaces for Computer-Aided Collaborative Engineering.* PhD thesis, Massachusetts Institute of Technology.

[Wong and Sriram, 1993] Wong, A. and Sriram, D. (1993). SHARED: An Information Model for Cooperative Product Development. *Research in Engineering Design.* Fall 1993.

[Wood and Kochan, 1985] Wood, P. and Kochan, S. (1985). *Unix System Security.* Hayden, Hasbrouck Heights, NJ.