System Support for Efficient Dynamically-Configurable Multi-Party Interactive Multimedia Applications

Mark Moran[1] and Riccardo Gusella[2]

[1] The Tenet Group
Computer Science Division, University of California, Berkeley
and International Computer Science Institute
Berkeley, CA 94720

[2] Hewlett-Packard Laboratories
1501 Page Mill Rd.
Palo Alto, CA 94304

Abstract. An important new class of communication-oriented applications are described as multi-party interactive multimedia applications. In this paper, we examine the programming abstractions, system support and communication services required by these applications; evaluate existing support; and propose strategies for improving support by combining current systems with new designs. We have taken a holistic approach to this problem by considering the local system and communication architecture together in deciding where to implement each service, and in designing interfaces between components.

1 Introduction

Continuing advances in processing power, display capability, and communication capacity are enabling a class of demanding, communication-oriented applications that allow two or more geographically separated persons to interact as if they were at the same location. Escalating time and travel costs and the increasing need, in global markets, of personal and group communication, make these applications highly desirable. Applications that facilitate person-to-person collaboration by providing a variety of communication media—including video, voice, and shared drawing spaces—and support for shared applications—such as a spreadsheet, word processor, or designing tools—are classified as multi-party interactive multimedia (MIM) applications.

Current computer systems and communication architectures do not support MIM applications effectively for two major reasons: they do not provide abstractions and capabilities for MIM communication; and the device access, data copying, and data transport capabilities they do provide are inefficient and could potentially limit both the affordability and scalability of MIM applications. In this paper, we propose a holistic system approach to supporting MIM applications. The local computer system and the communication protocol stack are considered together in order to determine the best place to provide various services for MIM applications and to design interfaces among them. In Section 2 we discuss the requirements of MIM applications. Section 3 analyzes the strengths and weaknesses of current systems in meeting these requirements. Finally, in Section 4 we propose strategies for improving support for MIM applications by designing new abstractions and services to be combined with existing ones.

2 Requirements for Supporting MIM Applications

MIM applications require three types of system and communication support: programming and data abstractions that facilitate application development, a so-called real-time communication service for transmission of time-critical media, and efficient access to supporting services and multimedia I/O devices.

2.1 Programming abstractions

Development of multi-party interactive multimedia applications requires a richer set of programming abstractions and corresponding system support than traditional text- or graphics-based applications. Since MIM applications are inherently distributed, they must be able to operate across heterogeneous computing and communication environments. Therefore, abstractions must be provided to allow the flexible use of a variety of multimedia I/O devices and real-time communication services. We have identified the following abstractions as being fundamental:

Time-based media. Like general multimedia applications, MIM applications require support for time-based media, including device-independent functions to handle the acquisition, presentation, and transmission of time-based media. Presentation functions may perform synchronization between