Session 1: Network and Operating System Support for Multimedia

Chair: Domenico Ferrari, University of California at Berkeley

The first session of the workshop consisted of four presentations that covered a wide variety of topics: from a new type of local-area network for multimedia applications to LAN support for video conferencing, from an implementation of an internetwork real-time protocol to priority inversion countermeasures in the context of multimedia applications.

Kevin Jeffay of the University of North Carolina described one of the many current projects intended to explore what can be done with present-day networks and protocols in the area of multimedia applications. Unlike other efforts, this one relied on a real-time operating system, YARTOS, developed by Jeffay and his coauthors. As long as their results are not interpreted in a much more general and long-term prospective than warranted by their here-and-now, ad-hoc scope (for example, if they are not used to "demonstrate" that admission control is not needed because "the system works even without it"), these studies are useful, as they often enhance our understanding of the requirements of continuous-media applications and our experience with them. The case described in the presentation was the one of desktop conferencing between two users across an Ethernet or another LAN. Particular attention in the talk was paid to the novel transport protocol designed to manage adaptively the scarce and time-variant available bandwidth, deliver audio packets reliably, reduce end-to-end delay and jitter, and deal satisfactorily with packet losses. The protocol runs on top of IP, and uses some more bandwidth than UDP, but is obviously better than UDP for conferencing applications.

The second presentation was given by Gerard Smit of the University of Twente in the Netherlands. It provided an overview of, and some of the details about, a new type of local-area network called Rattlesnake, which has been designed considering the needs of real-time applications as well as those of the more traditional ones. Like DEC's AN1, it is characterized by an aggregate bandwidth much greater than that of each link; however, Rattlesnake is based on a fixed topology, that of a Kautz graph, and is not self-reconfiguring; on the other hand, AN1 does not make any distinction between real-time and non-real-time traffic, whereas Rattlesnake relies on hybrid TDM to provide circuit switching to the former and packet switching to the latter. Also, cut-through is used for real-time packets, while the non-real-time ones are stored in the switches and then forwarded. The choice of hybrid TDM was, in the opinion of some of the attendees, not sufficiently motivated in the presentation, as the main argument seemed to ignore the existence of a number of schemes that can provide hard real-time guarantees even with ATM or, more generally, packet switching. The properties of Kautz topologies were considered interesting; in particular, the fixed degree, the fault tolerance, and the self-routing capabilities even in presence of link failures.

Ralf Herrtwich of the IBM European Networking Center in Heidelberg, Germany, talked about the implementation of the ST-II protocol that was done at ENC. He summarized the reasons for choosing ST-II as the internetwork protocol in the Heidelberg Transport System (HeiTS), which has been built to support a distributed multimedia
platform. He also discussed the additions and enhancements made by him and his co-author to ST-II to complete it and make it better suited to the needs of multimedia traffic: in particular, resource management mechanisms, characterized by optimistic (non-worst-case) reservation and graceful degradation, and a feedback scheme to allow rates to be controlled without having to keep all clocks in synchrony. This implementation of ST-II is probably the most complete and interesting to date. It is to be hoped that its authors will soon report on their experiments and experience with it.

The fourth and last paper of the session was presented by Akira Nakamura of Cambridge University in England. The connection between the two parts of the presentation was not easily grasped by the audience. In its first part, much longer than the second, the paper describes a new algorithm to deal with the priority inversion problem; the algorithm is less conservative than the priority ceiling protocol, which may cause waiting on a shared lock even when it is not necessary. In the second part, the question is raised whether the scheduling problems that may be caused by shared locks are really encountered in multimedia applications. The authors of the paper argue, unfortunately without providing any experimental evidence, that these applications may encounter lock conflicts, if any, between processing stages for the same stream rather than for access to data structures shared by several independent streams. Thus, in a sense, operating systems should not be extended, but rather reduced, and in any case, rethought and redesigned, for multimedia applications. The arguments presented in the second part of the talk sparked a heated debated between those who agreed and those who disagreed with the main thesis of the authors; the discussion was unfortunately cut short by the expiration of the session's deadline.