Resource Sharing of High-Speed Optical Circuits for File Transfers

Malathi Veeraraghavan
University of Virginia
malathi@virginia.edu

Wu Feng
Los Alamos National Laboratory
feng@lanl.gov

A number of efforts are underway to use optical networks to create high-speed end-to-end circuits. These circuits can be used for transfers of large files such as those generated by large-scale scientific applications. At the present time, gigabit and 10-gigabit Ethernet signals from end hosts are typically mapped onto metro-/wide-area SONET circuits using Ethernet-over-SONET (EoS) technologies. While SONET circuits are realized by configuring electronic circuit switches, extension of this concept to all-optical circuits will be straightforward when all-optical switches become commonplace. A few hero demonstrations have shown that 1-Gb/ s circuits can be configured from hosts in Europe to hosts in Canada/US to transfer files at high speeds over these circuits. Efforts targeted at these demonstrations have focused on the implementation of two pieces of software: (i) control-plane modules to provision end-to-end circuits (referred to as "user-controlled lightpaths" in [1]), and (ii) transport protocols, such as SABUL [2], Tsunami [3], and RBUDP [4] suitable for these end-to-end circuits.

While the provisioning of such circuits is feasible on experimental research networks such as Starlight, Canarie, UKlight, and SURFnet, we need to address the issue of resource sharing if this concept is to be deployed economically on a wide basis. In a paper presented in PFLDN 2003 [5], we proposed a solution called Circuit-switched High-speed End-to-End Transport ArcHitecture (CHEETAH), which uses a *call-blocking mode* of operation by leveraging the presence of Internet paths already available between any two end hosts.

Here we extend the concepts presented in the PFLDN2003 paper by adding the notion of *call scheduling*. We arrived at the need for call scheduling by identifying a problem with using circuits for file transfers. As described in [6], once a circuit is established for a given transfer, it cannot take advantage of bandwidth that becomes available subsequent to the start of the transfer, unlike in packet-switched networks where an ongoing transfer can take advantage of such bandwidth. In response to this problem, we observe that a file transfer can be allocated varying bandwidth levels for the duration of a transfer if a circuit switch knows the size of the file to be transferred during circuit setup. We refer to such an allocation as a **Time-Range Capacity (TRC)** vector. A TRC vector is characterized as follows: (b_1,e_1,C_1), (b_2,e_2,C_2), …, (b_k,e_k,C_k), where b_i is the start of the i^{th} time range, e_i is the end of this time range, and C_i is the capacity allocated for in the i^{th} time range. The circuit switch knows exactly when each transfer will complete using its knowledge of the TRC allocation and file size of the transfer. It can thus keep track of available bandwidth on a time-varying basis (unlike current switches in connection-oriented networks, which only keep track of the currently available bandwidth). This knowledge allows it to make a TRC allocation for a newly arriving transfer request. We call this algorithm **Varying Bandwidth List Scheduling (VBLS)**. The cost of implementing VBLS entails that the circuit switch be reprogrammed at multiple time instances within a transfer unlike in the fixed-bandwidth allocation mode where the switch is only programmed at the start and end of a call [5]. We have run simulations to compare VBLS with Packet Switching (PS) and found that VBLS achieves almost the same performance as PS. These results will be reported in the presentation, if accepted.

The transport protocol implemented at end hosts in conjunction with VBLS must be able to make rate adjustments in accordance with the TRC allocation for the circuit. Many UDP-based transport protocols, such as SABUL and Tsunami, support rate control. Thus, we are currently experimenting with these protocol implementations to verify whether end hosts can indeed maintain the negotiated rates in spite of the variability induced by other processes.

In contrast to VBLS, every variant of TCP assumes an underlying network that is connectionless with no resources reserved prior to a file transfer. A TCP-based transfer begins at the slowest rate and adjusts its rate during the transfer with the goal of reaching the available bandwidth at any given time. Because the network is "dumb," constant measurements of available bandwidth are required to allow the sender to enjoy the advantage offered by packet-switched networks, i.e., resource sharing and the ability of a transfer to take advantage of bandwidth that becomes available after it starts.

On the other hand, in VBLS, the file sender determines what bandwidth will be available to it for the entire duration of the transfer during the call-setup phase. It can then adjust its rate as the transfer proceeds according to its TRC allocation. Available bandwidths predicted and allocated at call-setup time for a given transfer will not change if the only application supported on this VBLS network is file transfers. Thus, a transfer is able to take advantage of bandwidth released by other calls within its duration. If live applications, where the length of the communication session is unknown a priori, are to be supported, the circuit can use probabilistic estimates of sessions lengths in conjunction with the deterministic estimates needed for file transfers.

Regardless of the application, VBLS call scheduling must be supported by a rate-based, flow-control mechanism in order to verify whether end hosts can maintain the negotiated rates in spite of the variability induced by other processes.

We are currently undertaking an NSF-sponsored three-year project (starting in Jan 2004) to develop this concept for the Terascale Supernova Initiative (TSI) project. We will describe our preliminary results from this work in our presentation at the workshop.

References

[1] M. Sampson, "World's First Working Prototypes of User Control of Lightpaths Demonstrated," May 27, 2003, http://www.canarie.ca/canet4/obgp/index.html.

[2] Y. Gu, X. Hong, M. Mazzucco, R. Grossman, "SABUL: A High-Performance Data Transfer Protocol," http://www.dataspaceweb.net/index.htm.

[3] Tsunami, http://www.indiana.edu/~anml/anmlresearch.html.

[4] E. He, J. Alimohideen, J. Eliason, N. K. Krishnaprasad, J. Leigh, O. Yu, T. A. DeFanti, "Quanta: A Toolkit for High-Performance Data Delivery over Photonic Networks," Future Generation Computer Systems, 1005, 2003.

[5] M. Veeraraghavan, X. Zheng, H. Lee, M. Gardner, W. Feng, "Circuit-switched High-speed End-to-End Transport ArcHitecture (CHEETAH), Proc. of Opticomm 2003, Dallas, TX, Oct. 14-15, 2003, extended version of paper presented at the PFLDN 2003 workshop.

[6] D. Bertsekas and R. Gallager, "Data Networks," Prentice Hall, 1986.