Final Technical Report

DOE Award Number: DE-FG02-04ER25640

Name of recipient: University of Virginia

Project title: Enabling Supernova Computations by Integrated Transport and Provisioning Methods Optimized for Dedicated Channels

Principal Investigator: Malathi Veeraraghavan

Collaborators: Nageswara Rao, ORNL

Reporting Period Start Date: June 1, 2004

Reporting Period End Date: Sept. 14, 2007

Date of Report: October 31, 2007

Executive summary: [Discuss (i) how the research adds to the understanding of the area investigated; (ii) the technical effectiveness and economic feasibility of the methods investigated; and (iii) how the project is otherwise of benefit to the public. The discussion should be written in terms understandable by an educated layman][1]:

The original problem consists of developing transport protocols for high-speed optical circuit networks, developing internetworking solutions to create a "connection-oriented" Internet, and to interconnect two circuit networks, NSF-funded CHEETAH and DOE-funded UltraScience Net. A high-speed optical circuit network is one that offers users rate-guaranteed connectivity between two endpoints, unlike today's IP-routed Internet in which the rate available to a pair of users fluctuates based on the volume of competing traffic. An analogy in the physical world is that circuit networks are comparable to airline transport where a user makes a reservation for a seat prior to taking a flight, while the IP-routed Internet is comparable to road transport where travel time is dependent on other concurrent traffic.

This particular research project advanced our understanding of circuit networks in *two* ways. *First*, transport protocols were developed for circuit networks. A transport protocol serves to achieve reliable data transfer on an end-to-end basis. In the Internet, a commonly used reliable transport protocol is TCP. TCP includes (i) error control functions for acknowledging segments (data packets), and retransmitting segments for which acknowledgments are not received before the sender's retransmission timer runs out, (ii) flow control functions that prevent the sender from sending data at a rate faster than the receiver's capacity for processing and storing received data, and (iii) congestion control functions to prevent the sender from sending data so fast that the router and switch buffers on the end-to-end path fill up leading to packet loss while simultaneously sending data fast enough to achieve the best possible throughput. In a circuit network, since bandwidth resources are reserved for each circuit on an end-to-end basis (much like how a person reserves a seat on every leg of a multi-segment flight), and the sender is limited to send at the rate of the circuit, there is no possibility of congestion during data transfer. Therefore, no congestion control functions are necessary in a transport protocol designed for circuits. However, error control and flow control are still required because bits can become errored due to noise and interference even on highly reliable optical links, and receivers can, due to multitasking or other reasons, not deplete the receive buffer fast enough to keep up with the sending rate (e.g., if the receiving host is multitasking between receiving a

[1] Description taken from the Federal Assistance Reporting Instructions for the Final Scientific/Technical Report

file transfer and some other computation). In this work, we developed two transport protocols for circuits, both of which are described below.

Second, this project developed techniques for internetworking different types of connection-oriented networks, which are of two types: circuit-switched or packet-switched. In circuit-switched networks, multiplexing on links is “position based,” where “position” refers to the frequency, time slot, and port (fiber), while connection-oriented packet-switched networks use packet header information to demultiplex packets and switch them from node to node. The latter are commonly referred to as virtual circuit networks. Examples of circuit networks are time-division multiplexed Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH) and Wavelength Division Multiplexing (WDM) networks, while examples of virtual-circuit networks are MultiProtocol Label Switched (MPLS) networks and Ethernet Virtual Local Area Network (VLAN) networks. A series of new technologies have been developed to carry Ethernet VLAN tagged frames on SONET/SDH and WDM networks, such as Generic Framing Procedure (GFP) [1] and ITU G.709 [2], respectively. These technologies form the basis of our solution for connection-oriented internetworking. The benefit of developing such an architecture is that it allows different providers to choose different connection-oriented networking technologies for their networks, and yet be able to allow their customers to connect to those of other providers. As Metcalfe, the inventor of Ethernet, noted, the value of a network service grows exponentially with the number of endpoints to which any single endpoint can connect [3]. Therefore internetworking solutions are key to commercial success.

The *technical effectiveness* of our solutions was measured with proof-of-concept prototypes and experiments. These solutions were shown to be highly effective. Economic feasibility requires business case analyses that were beyond the scope of this project.

The project results are *beneficial to the public* as they demonstrate the viability of simultaneously supporting different types of networks and data communication services much like the variety of services available for the transportation of people and goods. For example, Fedex service offers a deadline based delivery while the USPS offers basic package delivery service. Similarly, a circuit network can offer a deadline based delivery of a data file while the IP-routed network offers only basic delivery service with no guarantees.

Two *project Web sites* [W2] and [W2], 13 publications, 7 software programs, 21 presentations resulted from this work. This report provides the complete list of publications, software programs and presentations.

As for student education and training (*human resources*), this DOE project, along with an NSF project, jointly supported two postdoctoral fellowships, three PhDs, three Masters, and two undergraduate students. Specifically, two of the Masters students were directly funded on this DOE project.

Comparison of the actual accomplishments with the objectives of the project:

Table 1: Original work plan with deliverables [13], and accomplishments [1,2]

Work item	Deliverable	Accomplishments
Integrate a high-throughput *transport protocol* suitable for large file transfers on dedicated circuits (we call this Fixed-Rate Transport Protocol, or FRTP, a modified version of SABUL) into the Secure-FTP application	Transport protocol specification Transport protocol implementation	Specified, implemented, and published papers [1] [2]
	Secure-FTP with FRTP implementation	Integrated with file transfer programs called BWdetail [F1], WebFT [10] [F7], and HTTP [6] [F6]
Design and implement mechanisms for *peering* the CHEETAH network control-plane solution with the UltraScience Net's *centralized* control-plane	Software modules to enable this peering	CHEETAH Client System Agent (CCSA) provided to ORNL [4] [F4] Centralized book-ahead algorithms published [3] and [5]
	Demonstrate wide-area test across CHEETAH and UltraScience Net (with centralized control)	Demonstrations completed [9]
Design and implement mechanisms for *peering* UltraScience Net and CHEETAH *distributed* GMPLS-enabled control-planes	Software modules to enable this peering	CHEETAH Client System Agent (CCSA) provided to ORNL [4] [F4] and control-plane security designed [12]
	Demonstrate wide-area test across CHEETAH and UltraScience Net (with distributed control)	This was not done because USN used only centralized control
Extend the CHEETAH concept of end-to-end Ethernet/Ethernet-over-SONET circuits to a "*connection-oriented internet*" with segments of the end-to-end connection traversing packet-switched networks such as Ethernet VLANs and MPLS networks	Design document	Completed; papers were published [7][8]
Implement control modules to support connections through Ethernet LANs using VLAN technologies such as IEEE 802.1q	VLAN provisioning software modules	CHEETAH Control Plane Module (CCPM) completed [F4] and tested on Hybrid Optical Packet Infrastructure (HOPI) testbed [P17]
Implement control modules to support connections through MPLS networks	MPLS provisioning software modules	CHEETAH Control Plane Module (CCPM) included MPLS submodule [P17]
Integrate CHEETAH segments with VLAN and MPLS segments	Demonstration of a "connection-oriented internet"	Tests completed at UVA laboratory with Cisco GSRs [7]

Project activities for the entire period of funding (hypotheses, approaches, problems, assessment):

Track 1: Transport Protocols for Dedicated Circuits

From a theoretical protocol design perspective, we developed a protocol called *Fixed-Rate Transport Protocol (FRTP)* with the philosophy that if the network and the end hosts cooperatively agree on a certain fixed rate prior to the start of the user data transfer, then there is no potential for receiver buffer overflows, nor is there a possibility of losses within network switches (since these switches are circuit switches). Hence there is no flow control (“null” flow control) or congestion control built into FRTP. However a problem arose when we implemented this ideal solution. Disk access at the sending and receiving end hosts, an important component of file transfers, proved to be the major stumbling block because of the variable rate nature of this access. In other words, there is a mismatch between the variable rate access of disks and the fixed rate nature of circuits. To avoid all losses, a user can choose a pessimistic rate for the circuit using the smaller of the worst-case sending and receiving disk access rates. While this achieves high circuit utilization (with no losses and hence no retransmissions), it also results in high file transfer delays. Therefore, we added a selective-ACK based error control procedure to FRTP. With this solution, one can trade-off circuit utilization for lower transfer delays by selecting a more-aggressive circuit rate, allowing for losses and recovering from these losses with retransmissions. While this implementation works, it is not entirely satisfactory from a research perspective. To begin with, the answer of using null flow control in FRTP is not satisfactory. By holding the sending rate fixed at the circuit rate (which is set to a high value to reduce latencies), even though the circuit appears to be used all the time, a significant portion of the time is spent in retransmissions. In other words, the utilization is lowered. Therefore, adding a window-based flow control scheme seems attractive. Ironically, while the window-based flow control scheme of TCP is inadequate for connectionless networks (because it does not provide information on router buffer states), it is ideal for dedicated circuits since the only buffer about which the sender needs information is located at the receiver. It is better to send this feedback and have the sender stop sending data rather than allowing the sender to keep sending even when the receiver buffer is full. The latter causes sender and receiver CPU utilizations to be high.

While this solution (with window-based flow control) is better than the basic FRTP with null flow control, it is still not satisfactory. This is because it does not address the main problem of how to select the circuit rate and receiver buffer size to maximize circuit utilization and minimize file transfer delays within the constraints of the end host hardware. This problem of determining what circuit rate and receiver buffer size to use is more difficult if we allow the end hosts to multitask. Every time the file sending task is scheduled out of the sending end host processor, data transfer on the circuit stops, which lowers circuit utilization. Similarly, every time the file receiving task is scheduled out of the receiving end host processor, the receive buffer overflows, leading to losses, retransmissions, and lower utilization and increased delays. At a fundamental level, even the task that reads data from the disk competes with the task that sends the data on to the network at the sending host and correspondingly the task that writes data to the disk competes with the task that receives data from the network at the receiving host. Thus multitasking enhances the unpredictability of end host performance, originally created by disk access.

Next, we experimented with rate-based schedulers for Linux systems to test whether we can execute these disk read/write tasks and network send/receive tasks as soft real-time tasks. This will provide some control at the end hosts for how often and for how long these tasks get scheduled. The goal was to combine a solution for O/S scheduling with improved file-systems and file preprocessing for increased predictability of disk access rates. While this could yield an ideal answer, it adds overhead to the average user because it requires patches to Linux. Therefore, we also pursued more practical solutions that are easier to transition to scientists. For example, by leveraging the Pause feature in Ethernet interface implementations, we can use a coarse-grained timer to periodically schedule the file sending task. We coupled this with a receiver implementation that sends back flow control data (on receive window size) to the sender.

We successfully tested GridFTP on a cluster of 22 nodes at the University of Virginia, and experimented with striping. To enable striping, we needed a parallel file system. So we downloaded PVFS to this cluster and experimented with GridFTP striping. This work helped us understand disk access constraints better.

Next, we fixed the CPU utilization issue in the first release of FRTP software. The sender implements a "busy-wait" in order to send packets with a small inter-packet time (for high sending rates). This make the CPU utilization high. We found two possible solutions. The first is to use a combination of the Linux 10ms timer with Linux signals to awaken the network thread to send out as many packets as needed to achieve the desired rate. For example, to achieve a 400Mbps sending rate, the network thread needs to send 500KB every time it is awakened. This data will be sent out over the 1Gbps NIC within 4ms. If the first switch on the path is an MSPP operated in circuit mode, it cannot buffer the excess packets if the outgoing SONET circuit is only 400Mbps. But the PAUSE feature in GbE NICs stops the sender from sending packets and holds up the packets in the UDP buffer at the host. The UDP buffer should be set to a large enough value. This solution required the network thread to be broken up into two threads: a data thread and a control thread. The data thread is the one that is signaled every 10ms when it promptly sends data frames and exits the processor. The control thread handles ACKs, ERRs and other control messages received from the far-end. The FRTP work was published in [1]. The second solution is to use kernel-based transport protocols.

After the above-described experimentation with user-space implementation of transport protocols on UDP sockets, we decided to turn instead to kernel-based TCP implementations. We designed, implemented, tested and evaluated a new transport protocol called *Circuit-TCP (C-TCP)*, which resulted in a publication [2]. Our solution uses Net100/Web100.

To implement C-TCP in Linux we used the Web100 instrumented TCP stack. The Web100 instrumented stack provides an interface for user space programs to access many of TCP's internal state variables. The interface also allows some fields (control parameters), in the internal data structure that Linux maintains for each TCP socket, to be set from the user space. We added 2 control parameters to the Web100 stack, modified TCP sender code to ignore the congestion window *cwnd*, and instead maintain a minimum of a set sending window size (set equal to the bandwidth-delay product) and the receiver's flow control window, *rwnd*, of unacknowledged data in the network throughout the transfer. Further we set the additive increase and multiplicative decrease factors to values such that the cwnd does not change. Linux uses a slow start like scheme to update *rwnd* too. This makes *rwnd* a bottleneck during the initial part of the transfer and defeats the purpose of the changes made at the sender. Therefore, we modified the TCP receiver code to advertise the maximum possible *rwnd* when the socket is being used over a CHEETAH circuit. Here again we will need the PAUSE feature because the TCP implementation in the kernel will simply send a whole cwnd worth of packets at the Ethernet NIC speed of 1Gb/s. This will cause packets to be held up in the TCP buffer. Again this buffer should be sized correctly.

We tested C-TCP across the CHEETAH network using an end-to-end 1Gbps circuit on a 13-ms round-trip-delay path. Data transfers on the order of a few KB to 100MB will be served much faster with C-TCP than with TCP on a dedicated circuit because of TCP's Slow Start mechanism (see relative delay plot of Fig. 1). For larger data transfer sizes, as long as the TCP send and receive buffers are properly sized for the bandwidth-delay product of the path, the utility offered by C-TCP over the dedicated circuit instead of TCP will decrease. We also show with iperf that a sustained data transfer rate is better maintained with C-TCP than TCP. Finally, for disk-to-disk transfers, we show how the disk receive rate can be determined with a disk-write program and then used to set the circuit rate. With C-TCP, as the sender maintains a constant sending rate equal to the disk-write rate, as long as there is no multitasking on the sender or receiver, which will cause the circuit to be under-utilized, delay can be reduced to propagation delay plus transmission delay. For further details, the reader is referred to [2] and [11].

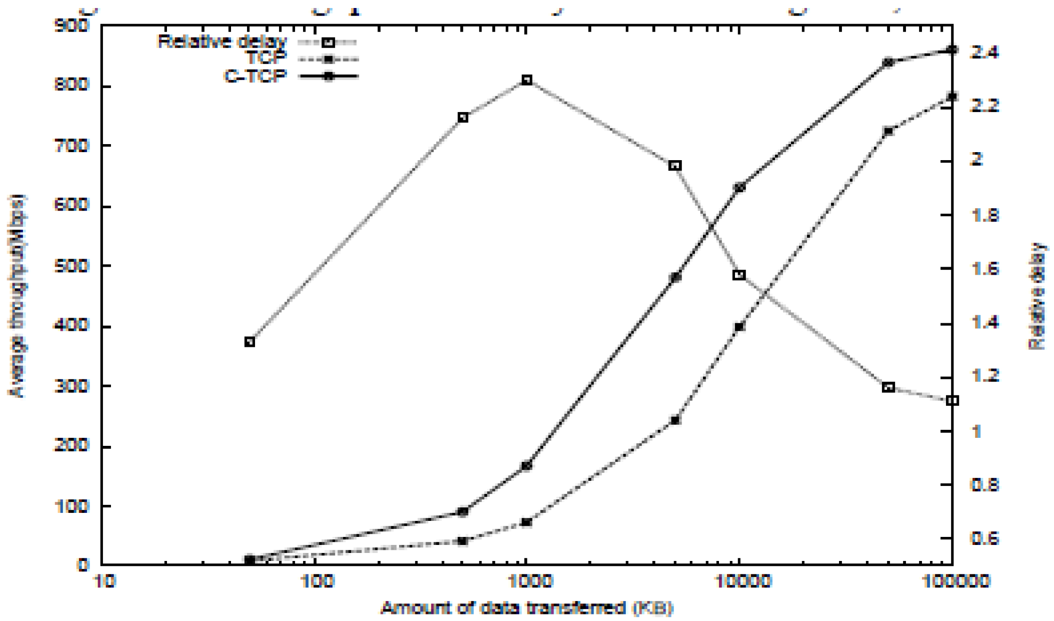

Fig. 1 Comparison of TCP and C-TCP for different file sizes

Assessment of the impact of the transport protocol results: The impact of this work will be significant when virtual circuit services are offered by enterprises and regional networks allowing for the creation of end-to-end circuits. In the current deployment, only core networks offer circuit services, and therefore, without end-to-end circuits, these transport protocols cannot be used.

Track 2: Peering of the CHEETAH network and UltraScience Net

We presented an architecture at the DOE Office of Science High-Performance Network Research PI Meeting, posted on the Presentations page of the project web site of http://www.ece.virginia.edu/cheetah/DOE [P1], which shows the details of the peering architecture. The key problems we identified include (i) security, and (ii) scheduling.

On *security*, we developed a solution to provide authentication and integrity checks on control-plane messages exchanged between the CHEETAH network and UltraScience Net (USN). Our goal was keep delays low without sacrificing the extent of protection. The control-plane security architecture is based on the solution developed for USN, and is documented in [12]. A brief summary of the architecture is as follows.

The CHEETAH control-plane network design uses the Internet to create out-of-band channels between end hosts and GMPLS (Generalized MultiProtocol Label Switching) systems as well as between neighboring switches. First, we consider the question of what type of IP addresses, static or dynamic, public or private, to assign to control-plane interfaces on switches and end hosts. Our conclusion is that we require static public IP addresses if the goal is to create scalable GMPLS networks. Given the shortage of such IPv4 addresses, we recommend the use of IPv6. Second, we note that the Router ID/Switch IP loopback interface addresses assigned to GMPLS switches should be advertised through routing protocols, allowing them to be reachable through at least one interface on the Internet. Third, to secure the control-plane channels, we use IPsec tunnels. Using open-source Linux software called Openswan on the end hosts and Juniper NS-5XT devices to protect control ports of switches, we use host based authentication and encryption of RSVP-TE and OSPF-TE messages. Finally, we used a mechanism to handle IP and MAC addressing on the data-plane in GMPLS networks. When an end-to-end circuit/VC is established, conventional IP networking dictates that the two ends of the Ethernet connection should be in the same IP subnet. But this leads to an unscalable solution requiring the data-plane interfaces of all hosts on a GMPLS network to be assigned addresses within one subnet. Our solution is to assign IP addresses to these interfaces in different subnets, based on the enterprise within which hosts are located, and to then use IP routing table and ARP table updates to add host-specific entries when circuits/VCs are setup. This architecture was prototyped, and demonstrated.

On *scheduling*, a presentation was given at the 2nd Intl. Optical Control Planes for the Grid Community, which is also posted on the Presentations web site of http://www.ece.virginia.edu/cheetah/DOE [P2]. There were two tracks of work on this issue. In one track, CHEETAH focused on immediate-request calls. This is comparable to the plain old telephone system, in which a user dials a called number and a 64kbps circuit is established end-to-end. The duration is unspecified. The goal in CHEETAH was to enable a high-speed version of POTS with which a user could request and obtain multi-Gbps circuits end-to-end. The second track was to support book-ahead scheduling (advance reservation) of high speed circuits for specified durations. DOE's UltraScience Net (USN) project pursued this goal. A thorough analysis of these two types of circuit services was carried out and published in [8]. Our conclusion was as follows. The immediate-request (unspecified duration) service is not well suited for applications in which the required per-circuit bandwidth is high, on the order of one-tenth the shared link capacity, or low, on the order of one-thousandth the shared link capacity. Ideal is the one-hundredth range. Further for file transfers, we show that the best range of operation to achieve high utilization and low call blocking rates with an acceptable

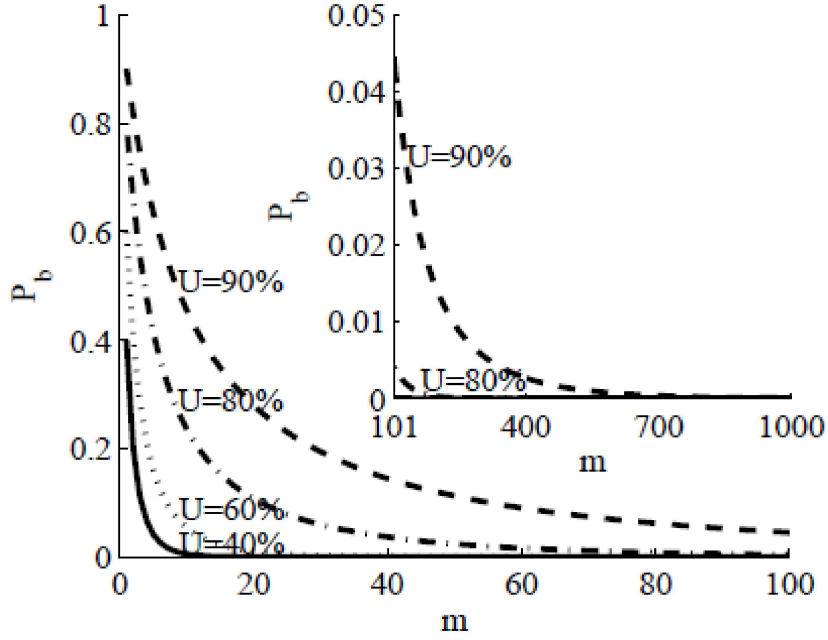

Fig. 2 Plot of call blocking probability (P_b) vs. m, the number of channels at different values of Utilization U.

number of ports on switches for traffic aggregation, requires call holding time to be small, in the range of seconds. For the high-rate calls, book-ahead service is required, and for the calls requiring low bandwidth, connectionless service is sufficient. As shown in Fig. 2, when $m = 10$, e.g., 1 Gb/s circuits are allocated on a 10Gbps link, to achieve 80% link utilization, a concomitant call blocking probability of 23.6% is to be expected. This high call blocking probability can be reduced through the use of book-ahead scheduling. This was shown in another publication [3] that with a flexible book-ahead scheme, the link can be operated at 95% utilization with a call blocking probability of just 1%.

The DOE UltraScienceNet (USN) and NSF CHEETAH networks were peered to realize dedicated circuits that span the United States, from the East Coast to the West Coast. The data-plane connectivity was achieved by connecting a GbE port on a CHEETAH Sycamore SN16000 switch at ORNL to a GbE port on the ORNL UltraScience Net Force10 Ethernet switch. Later on, as shown in Fig. 3, the Ciena Core Director CI of USN was connected by an OC192 link to the CHEETAH SN16000. This allowed us to test multi-Gb/s circuits.

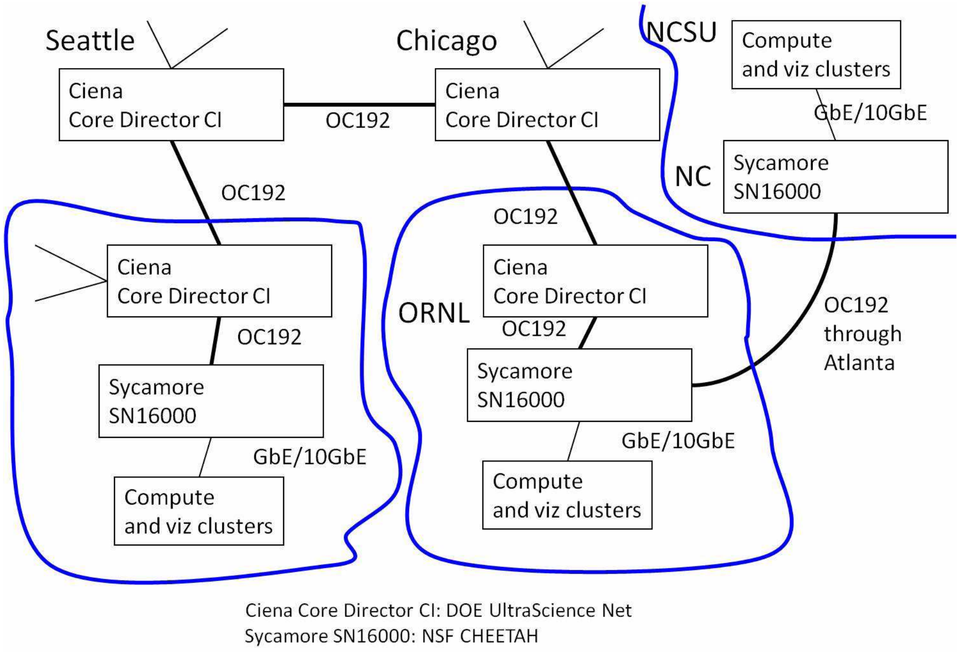

Fig. 3 Peering of DOE UltraScience Net and NSF CHEETAH network

On the control-plane, we completed the implementation of distributed GMPLS based signaling (software is available at the web sites [F4]-[F5]), and provided this software to ORNL. A journal paper was published on our GMPLS signaling implementation [4]. The key findings reported were that using our RSVP-TE software package, which successfully interoperated with an off-the-shelf commercial SONET switch from Sycamore networks, SN16000, circuits could be setup end-to-end. We presented measurements for typical end-to-end circuit setup delays across the CHEETAH network. For example, end-to-end circuit setup delay from a Linux end host in NC to a host in Atlanta is 166ms.

Track 3: Creation of a Connection-Oriented Internet

The goal is to design a connection-oriented internet to complement the existing connectionless Internet. The term "connectionless" refers to the fact that no admission control is executed prior to transferring data making these networks comparable to roadway transportation networks, while in connection-oriented

(CO) networks, an advance reservation or immediate bandwidth allocation is made on every hop of the end-to-end path before data transfer can commence. CO networks are packet- or circuit-switched.

Many of the packet switches offered by equipment vendors today implement varying degrees of support for connection-oriented networking. The degree of rate "guarantee" depends upon the type of CO network used. A circuit-switched network, such as a TDM-based SONET solution, offers hard rate guarantees. On the other hand, a network of IP routers with built-in MPLS engines can offer a "softer" rate guarantee. There are three dimensions associated with providing quality-of-service guarantees in connection-oriented packet-switched networks, Connection-Admission Control (CAC), scheduling of packets from various connections at the switches, and traffic shaping to ensure that the traffic meets the profile specified during connection admission. Some of the packet switches offered by vendors support only one or two of the three dimensions. CAC and traffic shaping can be implemented outside the switches, CAC more easily because it can be readily handled in user-level software, while traffic shaping may require kernel-level software at the end hosts. Many Ethernet switches support the IEEE 802.1q standard by which two ports can be programmed as belonging to the same "untagged VLAN." Ethernet frames arriving on any port other than the two untagged VLAN ports will not be forwarded to one of the two untagged VLAN ports. Frames received on one port of this untagged VLAN will be forwarded to the second port of the same untagged VLAN, and vice versa. In effect, we have created a "connection," i.e., a virtual circuit through this Ethernet switch. Such a capability enables us to create connections through campus LANs to reach the edge of a campus LAN from any scientist's office or laboratory. Dedicated wide-area circuits between SONET switches, as required in the CHEETAH solution, can be quite expensive. Therefore, Internet2, ESNet, NSF TeraGrid, and other IP based high-performance networks often use the MPLS capability in already deployed IP routers to support CO services. Using label-switched paths through these networks allows for a less-expensive realization (perhaps with a "softer" rate guarantee) than SONET circuits.

Given this variety of options for CO networking, we concluded that creating a wide-area "homogeneous" Ethernet/Ethernet-over-SONET solution is difficult, if not impossible. With ATM networks, such attempts at creating a homogeneous CO network was made in the nineties, but the efforts failed.

Our goal is to develop methods by which GMPLS protocols can be used to set up and released (dynamically using distributed control) heterogeneous connections whenever needed. In other words, end-to-end paths between two hosts may traverse two or more of these types of networks, Ethernet VLAN based network, MPLS network, SONET network, WDM network. Inter-area intra-domain and inter-domain scenarios are considered.

An 80+ slide presentation and audio files listed under "Track 3" on the Presentations Web site [P3] presents our solution to CO internetworking. We used two key ideas: (i) the creation of abstracted links between gateways and the spreading of this topological information through OSPF-TE to other switches/gateways, and (ii) the use of this topological information to select the signaling message parameters to create heterogeneous connections. We have applied the ideas developed here to two situations in implementing the CHEETAH network itself: one, to connect NCSU campus to MCNC campus through VLANs, and the second, to connect the SOX facility to ORNL through MPLS tunnels. Therefore, the CHEETAH network itself is a heterogeneous connection-oriented internet. Data-plane interworking is done through Ethernet interfaces. Solutions for routing and signaling interworking are also included. This work is published in an IEEE Magazine paper [7].

Technical details in our CO internetworking solution include a comparison of nested vs. contiguous vs. stitched Label Switched Paths (LSPs), methods for configuring temporary private IP addresses and ARP tables to avoid wide-area MAC address resolution, etc. Control-plane message parameters such as switching type, LSP encoding, etc. were carefully chosen for the three cases of nested, contiguous and stitches

LSPs. An example scenario is shown in Fig. 4. In Space Division Multiplexed (SDM) networks, a whole port is crossconnected to another port, in VLAN networks, the 12-bit VLAN ID is used for switching frames, and in MPLS, a 20-bit label is used for switching frames. To create an end-to-end rate-guaranteed virtual circuit, GW1 in Fig. 4 needs to support port-mapped LSPs (which means the entire signal, without demultiplexing, is sent on a VLAN), and GW2 needs to support VLAN-mapped LSPs (which means frames tagged with a particular VLAN ID are extracted and mapped on to an MPLS LSP). This is an illustration of data-plane internetworking. The control-plane is more complex and involves the Virtual Label Switched Routers (VLSR), which are software programs run on external hosts. The VLSR code, originally developed by the NSF-funded DRAGON team, was modified to create the CHEETAH CCPM [F4].

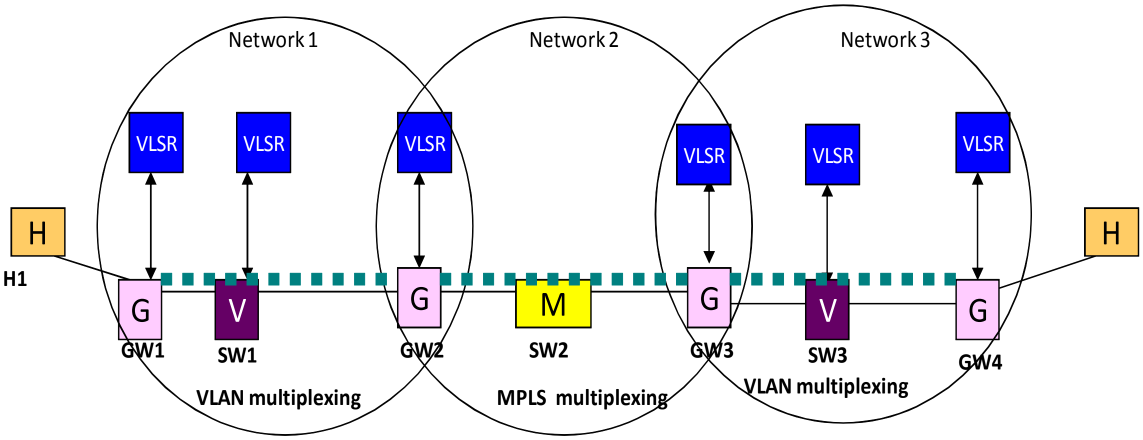

Fig. 4 An SDM-(VLAN)-(MPLS)-(VLAN)-SDM internetworking scenario;
SDM: space division multiplexing; VLSR: Virtual Label Switched Router

This solution, though complex, is necessary because different service providers deploy different types of connection-oriented networks. Inter-domain virtual circuit service is still in its infancy as most providers are first focusing on offering intra-domain virtual circuit service. But as more providers offer such services, as stated earlier, per Metcalfe's law the need for such internetworking will become stronger, and our solutions offer providers a template for how to internetwork connection-oriented networks.

The impact to specific DOE science applications:

The Terascale Supernova Initiative (TSI) is a multidisciplinary collaboration of one national laboratory (ORNL) and eight universities (NCSU, etc.) to develop models for core collapse supernovae and enabling technologies in radiation transport, radiation hydrodynamics, nuclear structure, linear systems and eigenvalue solution, and collaborative visualization. Communications applications in this TSI project include transfers of large datasets, distributed and collaborative remote visualization, and remote computational steering. These TSI applications place the following requirements on the network: (i) high throughput for data downloads, (ii) low latency and jitter for remote visualization and computational steering, and (iii) protocol/middleware support for collaborative work environments.

The Terascale Supernova Initiative (TSI) project scientists used the CHEETAH network to move large datasets (TB sized) from ORNL to NCSU. They used the leadership-class computing facility at ORNL to run large-scale simulations of supernova. The generated datasets were transferred to NCSU where the scientists maintained compute and storage clusters to further analyze and visualize the data. We also provided these scientists remote visualization tools to enable them to use multi-LCD panels to visualize the complex data sets generated from the simulations.

Products:

Publications:

[1] X. Zheng, A. P. Mudambi, and M. Veeraraghavan, “FRTP: Fixed Rate Transport Protocol - A modified version of SABUL for end-to-end circuits,” Pathnets Workshop, held in conjunction with Broadnets 2004, October 25-29, 2004, San Jose, CA.

[2] A. P. Mudambi, X. Zheng, M. Veeraraghavan, “A Transport Protocol for dedicated end-to-end circuits,” *Proc. of IEEE ICC 2006*, June 11-15, 2006, Istanbul, Turkey.

[3] X. Zhu, M. Veeraraghavan, “Analysis and design of a book-ahead bandwidth-sharing mechanism,” *IEEE Trans. on Communications*, vol. 56, no. 12, Dec. 2008, pp. 2156-2165.

[4] X. Zhu, X. Zheng, M. Veeraraghavan, “Experiences in implementing an experimental wide-area GMPLS network,” *IEEE Journal on Selected Areas in Communication*, vol. 25, issue 3, part supplement, April 2007, pp. 82-92.

[5] X. Zhu, M. E. McGinley, T. Li, M. Veeraraghavan, “An Analytical Model for a Book-ahead Bandwidth Scheduler,” *IEEE Globecom 2007*, Nov. 26-30, Washington, DC.

[6] X. Fang, M. Veeraraghavan, M. E. McGinley, R. W. Gisiger, “An overlay approach for enabling access to dynamically shared backbone GMPLS networks,” *Proc. of IEEE ICCCN’07*, Aug. 13-16, 2007, Honolulu, HI.

[7] M. Veeraraghavan, X. Zheng and Z. Huang, “On the use of connection-oriented networks to support Grid computing,” *IEEE Communications Magazine*, Volume 44, No. 3, March 2006, pp. 118-123.

[8] M. Veeraraghavan, X. Fang, X. Zheng, “On the suitability of applications for GMPLS networks,” *Proc. of IEEE Globecom 2006*, San Francisco, Nov. 27 - Dec. 1, 2006.

[9] X. Zhu, X. Zheng, M. Veeraraghavan, Z. Li, Q. Song, I. Habib, N. S. V. Rao, “Implementation of a GMPLS-based Network with End Host Initiated Signaling,” *Proc. of IEEE ICC 2006*, June 11-15, 2006, Istanbul, Turkey.

[10] X. Fang, X. Zheng, and M. Veeraraghavan, “Improving web performance through new networking technologies,” *IEEE ICIW’06*, Feb. 23-25, 2006, Guadeloupe, French Caribbean.

[11] Mark McGinley, Helali Bhuiyan, Tao Li, Malathi Veeraraghavan, An in-depth cross-layer experimental study of transport protocols over virtual circuits, IEEE ICCCN 2010, Aug. 2-5, Zurich, Switzerland.

[12] Malathi Veeraraghavan, Xuan Zheng, Xiangfei Zhu, “Addressing and secure control-plane network design in GMPLS networks,” April 7, 2006, http://www.ece.virginia.edu/cheetah/documents/dcn/dcn-design.pdf

[13] Project Statement of Work, Sept. 2004, http://www.ece.virginia.edu/cheetah/DOE/documents/revised-sow.pdf.

Web sites

[W1] Circuit-switched High-speed End-to-End Transport Architecture (CHEETAH) web site, http://www.ece.virginia.edu/cheetah/

[W2] DOE SCiDAC: Enabling Supernova Computations by Integrated Transport and Provisioning Methods Optimized for Dedicated Channels, http://www.ece.virginia.edu/cheetah/DOE

Software

[F1] BWdetail, http://www.ece.virginia.edu/cheetah/software/software.html#bwdetail

[F2] CTCP, http://www.ece.virginia.edu/cheetah/software/software.html#ctcp

[F3] Web100 based CTCP, http://www.ece.virginia.edu/cheetah/software/software.html#web100-ctcp

[F4] CHEETAH Control Plane Module (CCPM), CHEETAH Client System Agent (CCSA), and CHEETAH RSVP-TE client, http://www.ece.virginia.edu/cheetah/software/software.html#ccpm-etc

[F5] CHEETAH-lite for the Sycamore SN16000, http://www.ece.virginia.edu/cheetah/software/software.html#cheetah-lite

[F6] Circuit-aware squid, http://www.ece.virginia.edu/cheetah/software/software.html#squid

[F7] WebFT, http://www.ece.virginia.edu/cheetah/software/software.html#webft

Presentations (available on project Web sites)

P1. DOE Office of Science High-Performance Network Research PI Meeting, Sept. 15-17, 2004, Fermi Lab, Chicago; “Enabling Supernova Computations by Integrated Transport and Provisioning Methods Optimized for Dedicated Channels,” talk by Nagi Rao, ORNL, and “UVA Work Items,” talk by Malathi Veeraraghavan

P2. 2nd Intl. Optical Control Planes for the Grid Community, sponsored by MCNC, collocated with SC2004, Nov. 12, 2004, “Immediate-request vs. Scheduled Calls and Short-duration vs. Long-duration calls,” by Malathi Veeraraghavan

P3. Track 3: Creation of a connection-oriented internet presentation, Presentation and Audio files

P4. Hosted an exhibit at the SCInet Xnet booth in Super Computing 2004, Nov. 8-11, 2004.

P5. Served as Panelist on Optical Networks and Grid Computing Panel, Broadnets 2004, Oct. 25-29, 2004.

P6. Duke University, “Building a connection-oriented internet,” Feb. 11, 2005, http://cheetah.cs.virginia.edu/DOE Presentations.

P7. G. Tech University, “Building a connection-oriented internet,” March 30, 2005, http://cheetah.cs.virginia.edu/ DOE Presentations.

P8. NSF workshop, Santa Barbara, Apr 12-13, 2005, “Enabling a complementary connection-oriented internet,” M. Veeraraghavan

P9. JET meeting, Apr 20, 2005, “CHEETAH (Circuit-Switched High-speed End-to-End Transport Architecture),” by Malathi Veeraraghavan

P10. DOE meeting, Sep 29, 2005, “Enabling Supernova Computations by Integrated Transport and Provisioning Methods Optimized for Dedicated Channels,” M. Veeraraghavan

P11. MCNC Applications Symposium, April 10, 2006, Research Triangle Park, NC, Applications and Cheetah, by Malathi Veeraraghavan

P12. MCNC Meeting of the Board of Directors, April 27, 2006, “Remote visualization over the CHEETAH network Demo testbed”

P13. Fairfax County Economic Development Meeting between UVa SEAS faculty and Northern VA businesses, July 13, 2006, “CHEETAH Applications,” by Malathi Veeraraghavan and Xiuduan Fang

P14. Meetings with Tom Lehman, Dragon, ISI East, Arlington, VA, Aug. 25, 2006, and with Jerry Sobieksi, MAX, MD, Aug. 12, 2006

P15. Dragon Users' Group Presentation, Reston, VA, Aug. 30, 2006, “CHEETAH's use of DRAGON,” by Malathi Veeraraghavan

P16. Globecom 2006, IEEE Communications Society, San Francisco, CA, Dec.1, 2006, “Generalized MultiProtocol Label Switched (GMPLS) Networks,” all-day tutorial by Malathi Veeraraghavan

P17. HOPI Meeting, Internet2 Joint Techs Meeting, Minneapolis, Feb. 13, 2007, “Proposals for HOPI testing,” by Malathi Veeraraghavan

P18. UVA SEAS Open House, Charlottesville, VA, Feb. 24, 2007, “CHEETAH: A high-speed optical network,” by Malathi Veeraraghavan, Tao Li, Mark Eric McGinley, Xiuduan Fang, and Xiangfei Zhu

P19. MCNC Meeting, Raleigh, NC, May 8-9, 2007, “HOPI applications,” by Malathi Veeraraghavan

P20. ESCC/Internet2 Joint Techs Meeting, Batavia, IL, July 16-18, 2007, “CHEETAH applications and control-plane testing on HOPI (with demonstrations),” by Malathi Veeraraghavan, Tao Li, Xiangfei Zhu, Mark Eric McGinley, and Xiuduan Fang

P21. Presentation at Ciena, Sept. 24, 2007, “Applications for dynamically shared GMPLS networks,” by Malathi Veeraraghavan

Networks or collaborations fostered:

1. GridFTP developers.
2. Middleware researchers at UVA on GridFTP and security issues.
3. Astro-physicist, John Blondin, NCSU, to test out our file transfer software on his clusters at NCSU as well as at ORNL.
4. We obtained the rate-based scheduler from Prof. Scott Brandt, UC Santa Cruz, and may soon experiment with his object-based file systems.
5. We used GMPLS RSVP-TE code from Jerry Sobieski on the Dragon team.
6. We used SABUL code from Robert Grossman for the FRTP implementation.

1. References

[1] IEEE Communications Magazine, May 2002, Special issue on “Generic Framing Procedure (GFP) and Data over SONET/SDH and OTN,” Guest Editors, Tim Armstrong and Steven S. Gorshe

[2] ITU-T G. 872 and G.709/Y.1331 Specifications

[3] B. Metcalfe. Metcalfe’s law: A network becomes more valuable as it reaches more users. *Infoworld*, Oct. 1995.