

Multiple Crossbar Network Integrated Supercomputing Framework

Randy Hoebelheinrich

Richard Thomsen

Computer Network Engineering
Computing and Communications Division
Los Alamos National Laboratory
Los Alamos, New Mexico

Abstract

At Los Alamos National Laboratory, site of one of the world's most powerful scientific supercomputing facilities, a prototype network for an environment that links supercomputers and workstations is being developed. Driven by a need to provide graphics data at movie rates across a network from a supercomputer to a scientific workstation, the network is called the Multiple Crossbar Network (MCN). It is intended to be a coarsely-grained, loosely-coupled, general-purpose multicomputer framework that will vastly increase the speed at which supercomputers communicate with each other in large networks. The components of the network are described, as well as work done in collaboration with vendors who are interested in providing commercial products.

Introduction

The world of supercomputing is in transition, and researchers at Los Alamos National Laboratory are placing greater demands on computers than ever before. Because of their speed, supercomputers are essential tools for modern science. Supercomputers will become more beneficial when connected by networks that allow communication with each other and with workstations for multicomputer applications.

Los Alamos National Laboratory (LANL) has acted as the guiding influence in the development of a prototype high-speed network called the Multiple Crossbar Network (MCN)(Figure 1)[14]. The MCN will vastly increase the speed at which supercomputers communicate with each other in large networks. Information is transmitted between computers over high-speed channels through a richly connected set of special-purpose switches designed to replace general-purpose computer packet switches at Los Alamos.

The Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36. This work was performed under auspices of the U.S. Department of Energy.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1989 ACM 089791-341-8/89/0011/0713 \$1.50

The switches are composed of special-purpose protocol processors, channels, and a crossbar switch core. Services and protocols provided for the MCN include a data link protocol with a channel access capability, intranet routing and network access protocols, and both a network management and simple naming capability. The first prototype switch is expected to have an aggregate bandwidth of 12 Gbit/s. Future versions are intended to have bandwidths of 24 and 48 Gbit/s.

The MCN will be a hierarchy of interconnection networks [2,3,8,13]. The hosts themselves may include multistage interconnection networks in their architecture. Examples are Connection Machine's CM-2 [32], Cray's X-MP, IBM's 3090, BBN's Butterfly, or Intel's iPSC hypercube. The transport network between hosts will consist of interconnected switches. Each switch will use a crossbar interconnection network instead of a bus or ring based system. This will enable flexible and consistent addressing and routing in the MCN. The network topology will be general, regular, partitionable and reconfigurable. Software systems [25,26,27] could use this framework for applications with a resource allocation [22,23] and access control scheme.

Motivation

Stimulus for this new architecture came from diverse areas. Our primary needs were increased performance, reliability and

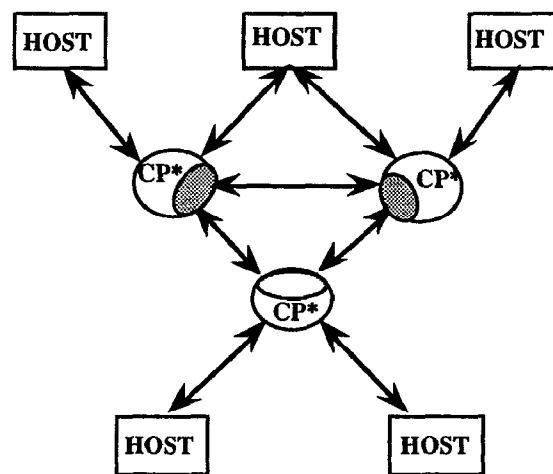


Figure 1. Multiple Crossbar Network

network management. Certain events and activities resulted in the initiation of efforts to define and solve new supercomputer networking requirements at Los Alamos. These activities could be summarized as efforts to explore, realize, and utilize the power of parallelism, distribution, and visualization.

One source of motivation came from a simple requirement to provide graphics data of hydrodynamic simulations at movie rates from a supercomputer to a frame buffer display [29]. The data transmission requirements were 30 frames of 24 bit pixels at 1024X1024 pixels per frame every second. This is a sustained rate of 755 Mbit/s and is needed to support visualization. If these rates were to be useful to the user community it would mean more than a dedicated host channel for one user at a time. We needed a network that could support this kind of bandwidth to any user's workstation. This meant a high-speed, high-bandwidth, general-purpose network.

Los Alamos has been networking heterogeneous supercomputers for almost 15 years. The core of the Integrated Computing Network (ICN)[5] is based on a network of high-speed point-to-point channels and general-purpose computers for packet switching. The configuration of each switch, currently Gould Concept 32/67s, includes a central controller and memory, a serially accessed bus and I/O devices (Figure 2). Each of these switches service 8-10 50Mbit/s channels called High-Speed Parallel Interfaces (HSPIs)[6] designed at Los Alamos. ICN protocols were also designed and developed at Los Alamos. The network protocol is a datagram service with additional security features and a maximum packet size of 32Kbytes. This architecture has served the Laboratory well, however new technology, requirements and applications are on the horizon. The computation speed of supercomputers has often outpaced the I/O bandwidth of their channels. Consequently, new supercomputer I/O channels have been developed that attempt to match the computational speed. As a result, bandwidth from supercomputers to user workstations will eventually outpace the bandwidth of our network.

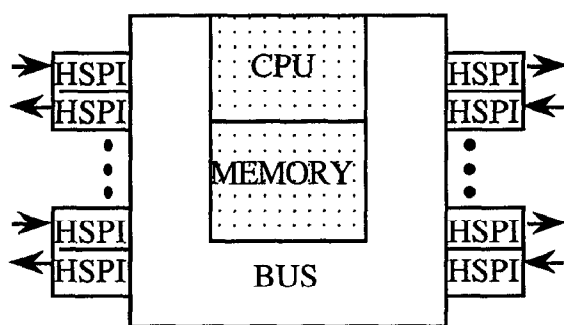


Figure 2. Centralized Packet Switch

Concurrently, increasing progress is being made in parallel computation. In fact, supercomputing is taking on new meaning. Multiprocessors have joined the supercomputer class machine. These systems are tightly coupled and fine-grained with multistage interconnection networks for interprocessor communication. In addition, some individuals think that one supercomputer working on a problem or simulation is simply not enough [7,10,29]. A system of multiple computers is necessary to provide the computing

power for applications using parallel algorithms, remote windows [31], distributed simulations [24], data flow computing [30] and nonlinear analysis. As specialized and general purpose machines play a greater role in large scale, distributed applications, the network will become a critical element of communications in the supercomputing arena as it did in multiprocessor systems [9,16,32,34]. The need is emerging for a loosely coupled, course-grained system of higher bandwidth and lower latency in a network environment [7,10,33]. Given these growing needs Los Alamos was motivated to take another look at supercomputer networking.

The Framework

The ICN had been originally designed with distributed utilities and shared supercomputers in mind. Now, we had to move even further toward a distributed, parallel, and very high-speed environment. For performance and LANL's physical extent, we required more than a serially accessed bus or ring. For security requirements, the central network could not be a shared broadcast network. Since LANL has experience with a high-speed switching network this model was a prime candidate for a second generation network. Network hardware changes centered on switches and channels. Software changes concentrated on streamlined protocol processing. We needed a high-speed channel, switching at the physical layer, and a special-purpose protocol processor on each channel. We also learned from years of experience that a standard interface to the network is essential. It is also necessary to design the means for detailed control of the network hardware and software for a cooperative computing environment. As people use supercomputers and workstations in a this environment, the network must be more reliable than our current network capability. An increasingly complex network indicates a need for comprehensive network management [19], extensive fault handling [21], and an integrated framework of switches, servers and applications.

The focus of the data transport framework was designing a switch. Several design characteristics were viewed as important for a new, high-bandwidth switch. Most important was distributing the processing overhead and minimizing decision overhead at the switch junction. This could be accomplished by having a processor memory unit dedicated to each channel or dual simplex set of channels. This is designated as PE and M in Figure 3. At that point, the devices could communicate over an internal interconnection network. Rather than use a serial shared-bus system, a crossbar interconnect was used. This is designated as Crossbar in Figure 3. In this way, arbitration was confined to those channels needing access to a given destination, and contention was limited to a control processor in the interconnection network. This crossbar interconnect needed to exist on some sort of channel between source and destination devices. Los Alamos already had the HSPI [6]. This parallel concept was extended to be higher speed with a wider data path. It was revised with a different connect scheme incorporating a multiple access mechanism for transparent physical switching of the channel. Using this switch will provide the framework for cooperative supercomputing as illustrated in Figure 1 and later in Figure 9. Combining switches into a multistage interconnection network will provide scale and extent for growth. Fiber optic media will provide switching for remote sites at Los Alamos.

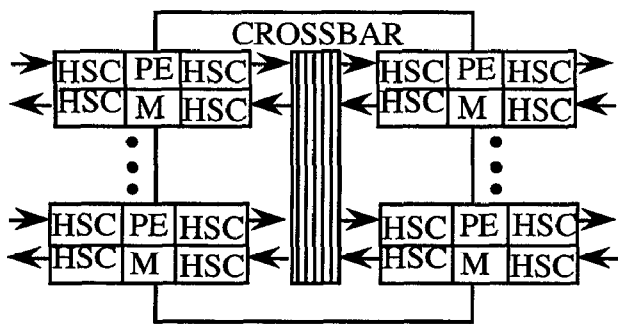


Figure 3. Distributed Packet Switch

The Switch

The resulting switch was the CrossPoint Star (CP*) (Figure 4). CP* is made up of three major elements: 1) the High-Speed Channel (HSC) [1], 2) the CrossBar Switch (CBS), and 3) the CrossBar Interface (CBI) [12]. CP* is designed to increase performance by having distributed special-purpose protocol processing on each channel and incorporating physical layer switching between these processors. A comparison of Figures 2-4 will illustrate the concept. Note that the HSC links between CBIs and a CBS can be 25 meters each with independent packaging for CBIs and the CBS. The physical layer switching is accomplished on an HSC with the aid of an intermediate CBS controller. This CBS is strictly dedicated to switching links to minimize switching latency and provide fast packet switching. The distributed protocol processors or CBIs provide streamlined optimal packet throughput. By using this design, we will, in a broader sense, have distributed the functions of the traditional packet-switching node over many processors and controllers at the channel end points as well as "on the wire." This is replicated on all links for parallel simultaneous transfers at any CP* node.

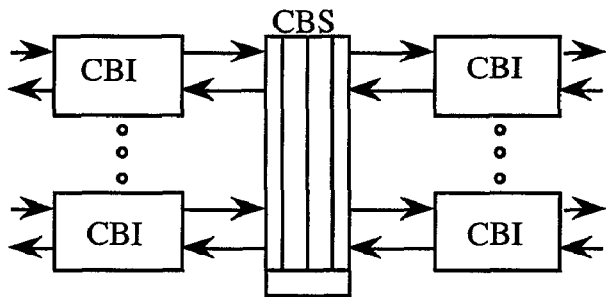


Figure 4. Crosspoint Star

The Channel

The most important goal for the HSC was high speed. Another goal was to keep it simple. An 800-Mbit/s channel already existed on the Cray computers, so matching that data rate seemed appropriate. We also knew from experience with the HSPI at Los Alamos that standardizing the HSC interface for vendor implementations was highly desirable. The question was, would industry see sufficient need for such a high-speed point-to-point channel to standardize it? History will show there was considerable interest. Another goal was to move some data link functionality into the channel. In

specifying a new parallel channel, we had the opportunity to incorporate this data link functionality into the signalling. These functions included framing and flow control. Flow control was handled using a Ready signal at the destination HSC. Framing was specified by signals defined for multi-word burst(s), packet(s) and a physical connection. Error detection and notification were also functions the HSC could perform for us. For ease in implementation a VRC/LRC error detection scheme with a length field was chosen with the ability to notify a controlling entity of data errors. Finally, and this is the most interesting goal, physical layer switching by means of an intermediate controller on the HSC was needed.

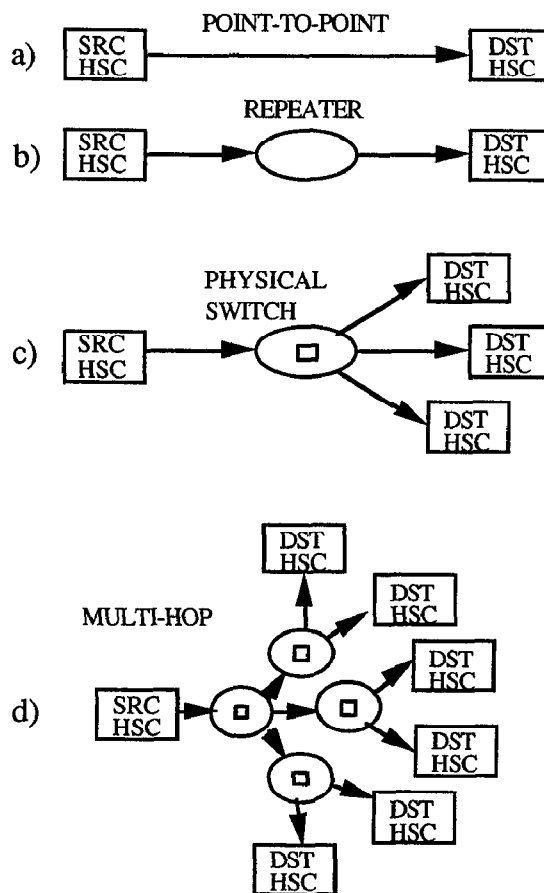


Figure 5. HSC Configurations

This would provide an integrated service of packet transfers on a directed connection and prove crucial in the overall architecture of a new high-speed network. Figure 5 illustrates HSC configurations.

The Switch Core

The sole purpose of the CBS is to connect a Source HSC to a requested Destination HSC minimizing switch latency (Figure 6). A request for a connection is made by a source HSC with a REQUEST signal and single parameter on the data lines. The CBS controller need only interpret this

parameter or I-Field of the HSC connect sequence to select a destination HSC. The first prototype CBS controller polls HSC REQUEST signals. Upon seeing a REQUEST for connection the CBS controller commands the switch to connect the source HSC to the destination HSC. If the destination is busy it is possible to specify and select alternate or redundant paths due to a regular network topology [9,10] and understanding of a structure of redundant paths by the intranet routing protocol. There are potential problems for deadlock with this method [17]. An alternative is for the CBI to back off and try again after an HSC REQUEST timeout. Minimum latency through the switch from this process is 350ns. All channels are polled except channel 0. Channel 0 is used by the controller and unavailable for connections. One of the more difficult problems was designing the interface connector and pin arrangement for the switch. The unit consists of an array of crossbar chips mounted 1 chip to a board with an HSC interface using 1 pin on each of the chips. The first prototype is a 48 bit 16X16 crossbar with plans for a 32X32 crossbar after that. Fiber optics will be used in future implementations.

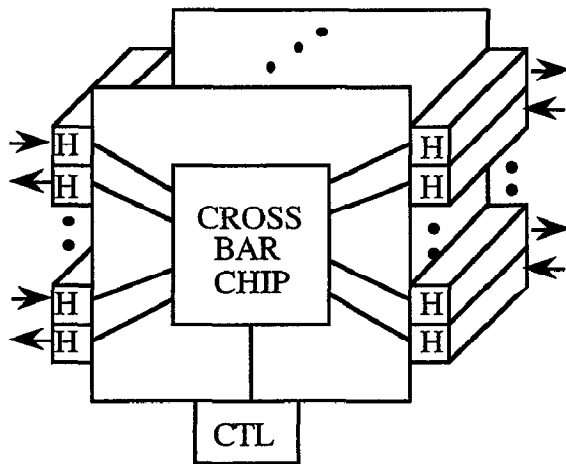


Figure 6. Crossbar Switch Core

The Protocol Processor

The CBI hardware (Figure 7) [12], which has been designed and built by Digital Equipment Corporation (DEC), is a protocol processor for CP*. The CBI's primary purpose is to act as a specialized processor and buffer for store and forward packets traversing the network. Although the prototype CBI will implement store and forward packet switching, this is not a requirement. We specifically wanted to allow variations of packet, circuit and hybrid switching such as cut-through [10] routing. Use of an AM29K RISC processor, four register VRAM and hardware FIFOs streamlines packet processing by distributing the workload to two HSC packet streams. This will lower protocol processing overhead on the switches and offload protocol processing from hosts using the network.

The Software

The Services

Services of the MCN prototype will be data transfer provided by our communication protocols, network management and a simple naming service.

We saw three types of service for data transfers over the MCN. These were 1) an external, connection-oriented, host-to-host data transfer, 2) an internal, connectionless, communications subnet data transfer and 3) a channel select and access capability. This latter service could be likened to the multiple access or media access control of packet broadcast networks. Specifically, a data link protocol to control the HSC was required. It's functions were to transfer data over any HSC configuration (Figure 5) and access a destination HSC for that transfer. An intranet protocol was necessary to provide routing of packet or message data from one boundary of the MCN to the other over a general topology of links connected by intermediate switching nodes. We wanted the routing to be dynamic to react to changing traffic patterns and allow network reconfiguration. Fault tolerant routing will take advantage of alternate or redundant routes around busy or failed links. Flexible routing will use characteristics of regular network topologies and capability information for partitioning and multicomputer routing decisions. The final protocol we required was a lightweight network access and transport protocol for end-to-end transfers between hosts. It had to be separate and distinct from the routing protocol as well as sensitive to current and evolving access control techniques and computer architectures.

The functions of the naming service are twofold. First, it is used to provide a name to logical address translation database for name translations on the network access portion of a host and logical to physical address translations for the intranet protocol. The latter capability is designed to allow flexibility for reconfiguration by adding a level of indirection to addresses. Second, the naming service will allow introduction, maintenance, access control, and accountability of objects that become part of the network. These objects include, hosts, processes, users and user sessions.

Network management will consist of the capability to address CBIs in the MCN for purposes of diagnostics, fault isolation,[21] configuration management, gathering statistics and tracing or logging activities. Network management is considered central to the architecture of the MCN, but will not be dealt with further in this discussion.

The link configurations in the MCN are of two general types. One is a point-to-point simplex HSC. The second includes an intermediate CBS between the source and destination HSC entities. It is helpful to view this latter configuration from the perspective of three different elements. These elements are the HSC, the source data link entity, and the intermediate crossbar switching core. The HSC is viewed as a point-to-point link where the switching core is transparent. The data link entity will view the link as a multipoint configuration with physical switching between several HSCs. The crossbar switching core will also view the set of HSCs as a multipoint topology. In all cases, the duplexity of the link is simplex.

The Protocols

Protocols provided for the MCN include the physical HSC protocol [1], a data link protocol over the HSC with a channel access capability, intranet routing and network access protocols, and both a network management and name-server capability. See Figure 8 for a software view of the CBI. Before discussing the individual protocols it is

necessary to point out some of the design decisions for the protocol stack in the MCN.

The data link for the MCN provides a connectionless data transfer service using HSC physical connections between data link entities. The data link will use physical address to channel mapping with flexibility for alternate or redundant channels. The HSC connection sequence and information will act as a channel access control mechanism. This is accomplished by utilizing the underlying HSC, HSC I-Field and intermediate CBS to access one of many possible destination HSCs. The data link entity should be viewed as contending for an available HSC in a multipoint configuration of data link entities. The data link entity does not control multiple HSCs simultaneously. It does not, therefore, provide a downward multiplexing or splitting capability. The data link is able to use the framing capability of the HSC to provide a hierarchy of data transfer units ranging from a short burst of less than 1Kbyte to multi-packet connections. The data link is capable of multi-hop paths or cascade switching over multiple CBSs with the aid of a binary routing word [9] utilized by the CBS controller. See Figure 5. Note that an implementation may or may not have a CBI between each CBS. There is a tradeoff between cost for extra CBIs and reliability in terms of network fault isolation. Providing this routing word and knowing the network topology is a function of the intranet protocol.

The intranet accepts packets from a network access entity at the boundary of the network and transfers these packets over a series of HSC links to a destination network access entity. The intranet is a local network connectionless data transfer service. Its primary function is to take each packet and determine the routing for the packet. The intranet receives a destination physical network address from the source network access entity when it is given a packet to transmit. This address is used for link selection by the CBS. Selecting these links may involve more than knowing the physical route. It may also mean knowing which portions of the network may be traversed [19]. Partitioning [9] and masking [20] for capability routing and resource allocation will be important areas of investigation for this protocol. These latter points address our reason for having distinct data link and intranet protocols. Each protocol has a very different scope. Routing based on capability will require a close relationship with the overlying directory and access control services of the network.

The purpose for a distinct network access protocol in the MCN is to provide secure memory-to-memory transfer of data between local hosts using very large blocks of data. We will also have the flexibility of experimenting with protocol issues at the host and network boundary. For our present purposes the network access accepts packets from a transport or internet entity and transfers these packets across the network to a local, logical destination using the connectionless service of the intranet. Transport protocols will be used for reliable end-to-end transfers. A separate network access protocol will provide the means to hide the workings of the communications subnet from communicating hosts and allow an explicit security and access boundary for our network. Security and access information are checked after entry to, and before exit from, the MCN. This has been and will continue to be a fundamental characteristic of our network. This will provide insulation between the attached

host and network [18]. We also want to provide a local reliable data transfer which could take advantage of future enhancements in computer architectures and network interfaces. The host-network interface is an important area of investigation from a performance and architectural point of view. Considerable work will be needed at this interface to address issues of performance and reliability from the physical media aspect to the application layer.

Initially, we considered implementing protocols up to and including the transport layer on the CBIs. For reasons of expediency and resources we chose not to implement these protocols on the CBI. We have provided the flexibility to allow any internet or transport protocol to access the MCN through the network access service interface.

Implementation

There are two areas of performance of immediate interest to us. These are throughput and latency of a CBI and CBS. In collaboration with DEC, we attempted to get an idea of what kind of performance to expect. An initial calculation using some simple traffic statistics from the current ICN suggested a possible CBI throughput of 600 Mbit/s. Later calculations [12] suggested a minimum time for a CBI to process a packet at 42us. CP* latency would include two CBIs and an intermediate CBS. With a minimum switch latency of 350ns, minimum CP* latency was about 84 us. A maximum CP* latency depends on the particular channel access method used. The current CBS controller services 31 ports. This could result in a CBS latency of 9us increasing CP* latency to 93us. Three tables were also compiled for expected CBI throughput. Throughput for typical packet sizes varied from 207 Mbit/s for a 1Kbyte packet to 790 Mbit/s for a 64Kbyte packet. Using two different packet mixes, a sustained aggregate throughput is expected to be between 470-645 Mbit/s. We have very preliminary results of data transfer rates looping between a CBI and CBS. Depending on the number of bursts in a packet and the packet mix of a loop, we have seen data transfer rates on a single channel of 280-720 Mbit/s. We have no switch latency statistics at this time. Actual rates will be compiled as equipment and the MCN testbed develop.

What has been accomplished at this point? The HSC physical specification is in public review in the ANSI process of standardizing the channel. HSC implementations have been completed and, in some cases, announced by commercial interests. There are at least two instances of HSC implementations in VLSI at this time. A fiber optic HSC extension and commercial HSC token ring network are in production.

DEC has built and installed four CBIs at Los Alamos where they are undergoing testing with the Los Alamos CBS and HSC interface. The first level of testing included passing data without checking by the RISC processor and HSC interfaces on the CBI. The CBS has also been included in the loop and demonstrated to switch and pass data. DEC has implemented data link, intranet and network access protocols on the CBI. These protocols still need to undergo extensive testing. The network management and naming services that reside above the intranet and network access protocols are being designed and expected to be implemented at a later date.

We are now working with a prototype CP* to integrate the

various pieces a level at a time. This CP* will become the framework for an MCN testbed of vendor equipment. This testbed will include three supercomputers, frame buffers, and two workstations. All of these systems are from different vendors. With a proposed standard network access channel, vendors have some hope of utilizing this testbed.

Commercial vendors are in various stages of design, implementation and testing of HSCs and HSC drivers for the MCN testbed. Implementations of the data link, and network access protocols are in preliminary stages and will be included in the MCN testbed. All vendor equipment will access the MCN with HSCs.

Possibly the most interesting accomplishment so far, has been transfers of 64Kbyte graphics image packets from an IBM 3090-600 through a DEC CBI to a frame buffer. This exercise demonstrated connectivity and interoperability by two major vendors. This was done without major problems or changes to the respective HSC implementations.

Our next milestone is setting up the MCN testbed network as shown in Figure 9. The first step will be the connection of an IBM 3090 and frame buffer to a CP* in September 1989. We expect to make considerable progress now through 1990, when we plan to have CBIs with a full set of protocols communicating to attached hosts. Efforts in the future will include connecting a small set of hosts to the MCN and developing tests and applications to verify and improve the protocols and services at all levels. In particular, the network access connections in both hardware and software need to be investigated to begin reducing data copy [28], share network address space with users, and verify security and access control.

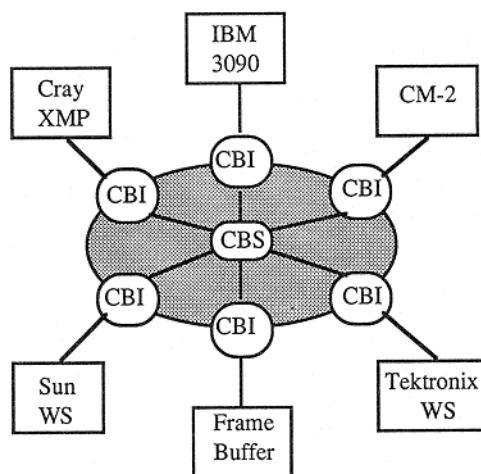


Figure 9. Supercomputer Framework

The following is a chronology of events leading up to the current stage of this project.

September 1985	Network Modernization Project
April 1986	Ultra-High-Speed Graphics Project
July 1986	Initial HSC Standard Proposed by LANL
July 1986	CP* Concept Proposed

January 1987	MCN Concept Developed
December 1987	LANL/DEC CP* Collaboration
January-July 1988	Service and Protocol Specification
January 1988	Crossbar Interface (CBI) Design Initiated
May 1988	Initial HSC Data Link Proposed by LANL
July 1988	CBS Project Initiated
November 1988	Fiber HSC Standard Initiated
January 1989	CBS Assembly
February 1989	HSC Interface for CBI
February 1989	CBI Delivery
May 1989	HSC Public Review
March-May 1989	Base Level 0 CBI-CBS Testing
May-June 1989	Base Level 1 Data Link Testing
June-September 1989	Base Level 2-5 Intranet, Network Access Testing
September-December 1989	MCN Testbed Setup

Conclusion

We have presented hardware and software elements of a network architecture for supercomputer data transfers. The MCN has been described as a framework and testbed for investigations into architecture, protocol and service issues for a new generation of high-speed networking. As a supercomputer framework we expect the MCN to evolve to a highly integrated state. In fact, the future MCN may be viewed as a very large scale collective computer with special purpose machines effectively used as peripherals to an overall system controlled by a distributed operating system. Figure 10 illustrates a possible evolution from Figure 9. In Figure 10 the DKernal represents a host front-end. This front-end is tightly coupled by a high-speed, high-bandwidth link or shared memory to a processor or Compute Engine (CE). It is advantageous to think of the DKernal as a peer or coprocessor to the CE itself. The DKernal is a distributed operating system kernel and handles session and communication tasks for the CE. The CE handles the computation or specific task it was designed to do. This is only conjecture and presented as a possibility. It is interesting and may open a whole new area of investigation. The time appears right for significant efforts into an integrated system of computing and communications.

Acknowledgments

We would like to acknowledge the efforts of Michael McGowen for the basic concepts for the HSC, the CP*, and for the concept and design of the CBS. The efforts of the Network Modernization Project and Ultra-High-Speed Graphics Project also played a role in early motivation of this project. Valuable contributions were made by Don Tolmie, Gene Dornhoff, Steve Tenbrink, John Morrison, Dave Dubois, Allan Meddles and Eric Vandevere. We are indebted to Norm Morse for supporting this project and Karl-Heinz Winkler for demonstrating the need. We are grateful to the members of DEC's Southwest Engineering group in Albuquerque who contributed greatly to this effort. They

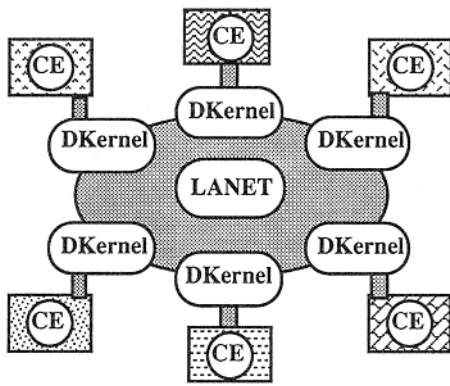


Figure 10. LaNet : MetaComputer

include: Bruce Ellsworth, Paul Brooks, Rich Lewis, Bill Hedberg, Marty Halvorson, Mike Martinez, Joel Kaufmann, Win Quigley and contractors Gary Mendelsohn and Mary Monson. Finally, credit goes to all companies and attendees of the HSC Working Group for their work in developing the HSC standard. This has truly turned into an industry-wide effort.

References

- [1] High-Speed Channel (HSC) Mechanical, Electrical, and Signalling Protocol Requirements, ANSI X3T9.3/88-023, Rev. 6.4, BSR X3.7/83 198x.
- [2] T. Y. Feng, "A Survey of Interconnection Networks," *Computer* 14, 12, December 1981, pp. 12-27.
- [3] G. J. Lipovki and M. Malek, *A Theory for Multicomputer Interconnection Networks*, Tech. Rep. TRAC-40, University of Texas at Austin, March 1981.
- [4] Richard F. Rashid, Avadis Tevanian, Jr., Michael W. Young, et al., "Machine-Independent Virtual Memory Management for Paged Uniprocessor and Multiprocessor Architectures," *IEEE Trans. Comp.*, C-37(8):896-908, August 1988.
- [5] H. Ferman Kelso, *Overview of LASL Integrated Computer Network* Los Alamos Scientific Laboratory report LA-6756-MS (January 1977).
- [6] D. E. Tolmie, A. G. Dornhoff, and S. C. Tenbrink, *Interconnecting Computers with a High-Speed Parallel Interface*, Los Alamos National Laboratory report LA-9503-MS (August 1982).
- [7] Emmanuel A. Arnould, Francois J. Bitz, Eric C. Cooper, et al., *The Design of Nectar: A Network Backplane for Heterogeneous Multicomputers*, Third Int. Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS III), April 1989.
- [8] Mark A. Franklin and Sanjay Dhar, "Interconnection Networks: Physical Design and Performance Analysis," *J. Parallel and Distributed Comp.* 3 (1986), pp. 352-372.
- [9] Howard Jay Siegel and Robert J. McMillen, "The Multistage Cube: A Versatile Interconnection Network," *IEEE Comp.* 14, 12 (December 1981), pp. 65-76.
- [10] William C. Athas and Charles L. Seitz, "Multicomputers: Message-Passing Concurrent Computers" *Computer*, 21(8):9-24, August 1988.
- [11] Howard Jay Siegel, "Interconnection Networks for Large-Scale Parallel Processing" *Lexington Books*, 1985
- [12] William F. Hedberg, "Multiple Crossbar Network: The Crossbar Interface", *Proceedings of the 14th Local Computer Network Conference*, Minneapolis, MN. October 1989
- [13] Laxmi N. Bhuyan, Qing Yang and Dharma P. Agrawal, "Performance of Multiprocessor Interconnection Networks" *Computer*, 22(2):25-37, February 1989
- [14] Randy Hoebelheinrich and Richard Thomsen, "Multiple Crossbar Network, A Switched High-Speed Local Network", *LA-UR 89-2234*, Los Alamos National Laboratory
- [15] Jonathan S. Turner, "Design of a Broadcast Packet Switching Network", *IEEE Transactions on Communications*, 36(6):734-743, June 1988
- [16] Charles L. Seitz, "The Cosmic Cube", *Communications of the ACM*, 28(1):22-33, January 1985
- [17] William J. Dally and Charles L. Seitz, "Deadlock-Free Message Routing in Multiprocessor Interconnection Networks", *IEEE Transactions on Computers*, 36(5):547-553, May 1987
- [18] H. Kanakia and D. Cheriton, "The VMP Network Adaptor Board (NAB): High-Performance Network Communication for Multiprocessors", *Computer Communications Review*, 18(4), August 1988
- [19] Gigabit Working Group, B. Leiner, Editor, "Critical Issues in High Bandwidth Networking", *RFC 1077*, Stanford University, November 1988
- [20] Howard Jay Siegel, "Analysis Techniques for SIMD Machine Interconnection Networks and the Effects of Processor Address Masks", *IEEE Transactions on Computers*, 26(2):153-161, February 1977
- [21] George B. Adams III, Dharma P. Agrawal and Howard Jay Siegel, "Fault-Tolerant Multistage Interconnection Networks", *Computer*, 20(6):14-27, June 1987
- [22] Eli Oppen and Mirosław Malek, "On Resource Allocation in Multistage Interconnection Network-Based Systems", *Journal of Parallel and Distributed Computing* 1, 206-220 (1984)
- [23] O. Kramer and H. Muhlenbein, "Mapping strategies in message-based multiprocessor systems", *Parallel Computing* 9 (1988/89) 213-225
- [24] Jayadev Misra, "Distributed Discrete Event Simulation", *Computing Surveys*, 18(1), 39-65, March 1986

- [25] Leonard Kleinrock, "Distributed Systems", *Communications of the ACM*, 28(11):1200-1213, November 1985
- [26] Richard W. Watson and John G. Fletcher, "An Architecture for Support of Network Operating System Services", *Computer Networks*, 4(1980) 33-49
- [27] Richard W. Watson, "The Architecture of Future Operating Systems" *UCRL-Preprint Lawrence Livermore National Laboratory*, Cray Users Group Meeting, Tokyo, September 1988
- [28] Willy Zwaenepoel, "Protocols for Large Data Transfers over Local Networks", *Technical Report TR-85-23*, Department of Computer Science, Rice University
- [29] K.H.Winkler, et al., "A Numerical Laboratory", *Physics Today*, Vol. 40 No. 10 Oct.,1987,pp.28-37.
- [30] P.C.Treleaven, D.R. Brownbridge, and R.P. Hopkins, "Data-driven and demand-driven computer architectures," *ACM Comput. Surveys*, vol. 14, Mar. 1982
- [31] Alan Katz and Stephen Casner, "Supercomputer Workstation Communication", *Information Sciences Institute Special Report*, IDI/SR-89-235, June 1989, USC
- [32] W. Daniel Hillis, "The Connection Machine", *Thesis (Ph.D.)*, MIT, 1985, *ACM Distinguished Dissertation*, 1985
- [33] Daniel A. Reed and Dirk C. Grunwald, "The Performance of Multicomputer Interconnection Networks", *Computer*, 20(6):63-73, June 1987.
- [34] Mark A. Franklin and Sanjay Dhar, "Interconnection Networks: Physical Design and Performance Analysis", *J. Parallel and Distributed Comp.* 3 (1986), pp. 352-372.