Massachusetts Institute of Technology
Laboratory for Computer Science

Advanced Network Architecture

from January 1, 1989 to December 31, 1991

submitted to the

Defense Advanced Research Projects Agency

David D. Clark
Co-Principal Investigator and
Head, Advanced Network
Architecture Group

Michael L. Dertouzos
Co-Principal Investigator and
Director, Laboratory for Computer
Science

George H. Dummer, Director
Office of Sponsored Programs

Volume I

1. Points of Contact

Advanced Network Architecture

Principal Investigators:
Dr. David D. Clark
MIT Laboratory for Computer Science
545 Main Street
Cambridge MA 02139
(617)253-6003
ddc@lcs.mit.edu

Prof. Michael L. Dertouzos
(617)253-2145
dertouzos@xx.lcs.mit.edu

Contract Administrator:
Mr. Paul C. Powell
Coordinator
Office of Sponsored Programs
(617) 253-3856

2. Innovation of Proposed Research

While currently popular network protocols (for example the Internet suite earlier developed by DARPA) function well under today's assumptions, they will not prove sufficient for the networks of tomorrow.

- As transmission speeds go up, the processing requirements in the host, switch and gateway go up accordingly. Extrapolation of current trends show that straight-forward speedup of the switch and host will not be sufficient to match expected trunking speeds; new protocols and structures will be required to reduce the processing burden.
- At the same time, more sophisticated functional requirements are being developed, such as enforcement of policy concerns. This added overhead directly and adversely impacts performance, and increases the need for an architecture that reduces overhead.
- As highly parallel processors become more important, the serial nature of today's networks and protocols will prove increasingly incompatible.
- Finally, as the existing Internet grows larger, it becomes clear that issues of scale are as pressing as issues of speed. The existing tools for routing, fault isolation or resource control are not able to deal with systems of the current size. As networks grow by another order of magnitude, as they certainly will over the next decade, these tools will prove totally inadequate. Again, new approaches are needed.

Our research is based on several key concepts and innovations:

- The architectural source of processing overhead is that the unit of multiplexing on the network (the packet) has also been the unit of processing in the host and switch. To achieve a performance breakthrough, we propose to separate these ideas, and to introduce a new abstraction, the "flow", which represents the unit of processing. This abstraction will group together larger units of data than the unit of multiplexing, and thus reduce processing overhead. This separation of processing and multiplexing is a key idea for the networks of tomorrow.
- The flow concept defines a new form of basic network service, more powerful than the datagram and more flexible than the traditional connection. It has the power to support a wide range of services, including new services such as data streams, and will permit the effective creation of an integrated services network.
- Once multiplexing has been separated from processing, we are free to introduce new techniques for multiplexing (wavelength division or highly parallel mini-packet schemes, for example) without impacting the network structure. This will permit a network architecture that can encompass a number of simultaneous multiplexing techniques.
- To support processing of flows, state is required in the switch to remember the processing done at the flow level and apply this to each multiplexing element. This state must be properly managed if we are not to lose the robustness of the current stateless gateways. We introduce the idea of "soft state" to describe the idea that the state information must not be critical to flow survivability, and the idea of "flow labelling" by the end node as a means to restore flow state.
- At the host interface, flows must be mapped directly to user buffers, to reduce the host overhead. We introduce the idea of "named messages" as end-to-end transmission elements to permit efficient buffer management.
- As networks get faster, traditional congestion control will not be possible, because the duration of most messages will be too short to permit useful feedback. We propose to study a new class of "congestion constraint" techniques that are compatible with future speeds. We are currently demonstrating a new flow control technique, rate controls, which better matches the requirements of high-speed trunks.

3. Deliverables of Proposed Research

The core of this research is a number of architectural studies and simulations which will define a new architecture, suitable for a next generation network. We identify the following areas which, at minimum, must be considered:

- Flow and congestion control.
- Support of multiple types of service.
- Addressing.
- Accounting and access control.
- Autonomous Administration and policy enforcement.
- Network interfacing.
- Security.

The integration of these studies will result in several key specifications. These are:

- The functional specification of the new network service based on flows.
- The functional specification of a network management architecture.
- The design of several key application services based on the flow interface to the network.

To validate these studies, we will use our existing simulation tools, and develop other modelling techniques as appropriate.

It is highly desirable to construct prototype demonstrations of our concepts. These will permit experiments to be performed, support early development of applications, and provide a more convincing argument for our approach than paper studies and simulations.

These demonstrations have been organized as a number of optional projects that can be performed as funding is available.

- Option 1: To demonstrate the idea of flow processing, and to explore our approach to congestion constraints, rate controls, support for multiple services and addressing, we propose to develop a prototype flow switch and network interface module. This network interface will also provide a platform for experiments in high-speed transport protocols, and will demonstrate the support for a variety of network clients, including a traditional host computer and a data stream source such as video. To avoid the re-implementation of the management software for the switch, we will use the relevant components from the operational MIT Internet gateway.
- Option 2: To demonstrate our ideas in policy routing, we propose to develop a policy gateway, a policy controller, and the matching host software, or to perform such other demonstrations as seem most appropriate.
- Option 3: In conjunction with some developer of network switching technology, we will implement a prototype high-speed network incorporating the above components.

If these demonstrations are brought to a point where they can be made operational for a small group of users, a final aspect of the research would be the support of an experiment involving some advanced application, such as access to parallel computers, and the packaging of our modules in a form where they can be exported to a small number of interested research labs.

4. Schedules and Milestones

The core proposal will permit the architectural studies outlined in the previous section to be completed within two years. The key specifications of the project, the series of functional specifications outlined in the previous section, will be completed in the third year of the effort.

To perform the various demonstrations, software and possibly hardware development will be required. The rate of progress of these projects will depend on the level of staffing possible. In addition, some aspects of the core effort will be accelerated to provide the needed specifications in a timely manner.

For each of the demonstrations, we provide an estimate of the effort required for a sequence of incremental milestones.

Option 1: Flow switch:

- Hardware development: To be determined based on needed performance.
- Port of existing software: 3 months.
- Integration of flow model: 4 months.
- Implementation of rate-based flow control: 4 months.
- Implementation of related host-based protocols (Unix): 6 months.

Network interface unit: (Assuming use of the same hardware as flow switch)

- Packet oriented transport protocol: 4 months.
- Application service support software: 4 months.
- Host (Unix) support: 3 months.
- Traffic generator and other test programs: 3 months.

This plan would permit an initial demonstration of flow processing by Q3 89, a demonstration of rate based flow control by Q2 90, and a demonstration of host based transport in 1991. We would hope to demonstrate another sort of service, such as video data streams, in 1991 as well.

Option 2: Policy routing facility:

- Add policy enforcement to gateway: 4 months.
- Implement policy controller: 12 months.
- Migration support in controller: 5 months.
- Implement host support for policy: 3 months.

This plan would permit a first demonstration of policy forwarding in Q3 89, a migration demonstration in Q2 90, and a full demonstration of the policy controller in 1991. Short term experiments will be scheduled as appropriate.

Option 3: Prototype network:

This project is less well defined at this time, and it is premature to make manpower estimates. The effort for this project will be proposed at a later time, after current negotiation with collaborators is complete.

5. Proprietary Claims

There are no proprietary claims for any material in this proposal.

Industrial collaboration in this effort may result in devices that are covered by industrial claims of ownership. These matters will be reviewed with DARPA as part of defining any such relationships.

6. Statement of Work

Under this proposal we intend to do the following:

- Perform a series of studies in key architectural areas.
- Write a functional specification for the new network service based on flows.
- Write a set of functional specifications for a number of application services, such as bulk data transfer, data streams, and request/response exchange.
- Provide a set of protocols and methods to realize these services.
- Describe the requirements for network management in this architecture.
- Present the above results in papers, reports and presentations to working groups as appropriate.
- Provide leadership and participation in Internet working groups as appropriate.
- Provide limited development effort in mid-term Internet evolution in key areas.
- Develop key relationships with academic and industrial organizations with the goal of a national effort in high-speed networking.

Additionally, as funding permits:

- Demonstrate our ideas in a fast prototype of a flow switch and network interface.
- Demonstrate a number of devices attached to the interface, such as a computer (high performance work station or an advanced computational engine) and a video element or other image generator.
- Demonstrate the simultaneous support for the necessary services for the above attachments over one network.
- Demonstrate our model of policy routing in a fast prototype of a policy gateway and policy controller.
- Collaborate with selected users to provide pilot support of specific applications such as access to parallel or supercomputers.

7. Technical Rationale

7.1. Introduction

It is our belief that existing technology and protocols will not prove sufficient to move us to the next generation of network. To achieve the quantum jump in speeds to the gigabit rates, the issues of speed, scaling and management all must be re-evaluated, and some of the basic assumptions of networking must be challenged.

Today there is great interest in high performance, large scale networking. Image transmission, integration of video into the normal computing environment, and distributed information retrieval, for example, might well increase by two orders of magnitude the volume of data transferred to a single work station.

Given that, it is critical that we begin to consider the next generation of architecture for networks and protocols. If we start this effort now, results may be available in time to be exploited as we reach the limits of the current technology. Work performed now can be relevant. Work performed in a few years will be too late.

If the research community can demonstrate the feasibility and utility of large scale, high performance networking, we believe that we can create a new market for network interconnect (using the fiber now being installed so commonly) and we can create the infrastructure necessary for the development of a whole new class of applications, based on access to remote information. We are launching the Advanced Network Architecture project in the hope that we can be part of this achievement.

7.2. The Real Problem

Superficially, it might seem that the problem of advanced networking is one of technology. That is, to support high speed networking what is needed is faster links and faster switches. These technological components are certainly needed. But there has already been much successful research on fiber transmission and switch architecture. What we see as the missing component is not technology but architecture. Broadly, the goal of our research is to understand the key issues which underlie the next generation of network. Based on our experience with current networks, we believe that the following issues are central:

Flow and congestion control -- As traditional networks become faster, two related problems arise. First, existing flow control mechanisms such as windows do not work well, because the window must be opened to such an extent to achieve desired bandwidth that effective flow control cannot be achieved. Second, especially for long-haul networks, the larger number of bits in transit at one time becomes so large that most computer messages will fit into one window. This means that traditional congestion control schemes will cease to work well.

What is needed is a combination of two approaches, both new. First, for messages which are small (most messages generated by computers today will be small, since they will fit into one round-trip time of future networks) open-loop controls on flow and congestion are needed. We call these controls "congestion constraints" to distinguish them from the traditional feedback control techniques. For longer messages (voice or video streams, for example) some explicit resource commitment will be required.

Qualities of service -- The goal of ISDN is to deliver a suite of services from one network architecture. Our experience is that this goal has not been met, either with ISDN, or with the popular connectionless

network technologies. The goal of delivering multiple qualities of service over one network will be critical, as the number of distinct services goes up, and the need arises to deliver them simultaneously to the work station of the future.

What is needed is some new form of basic network service, with the power to serve as a building block for a range of diverse higher-level services. Neither the datagram service, which lacks the necessary controls, or the connection service, which lacks the necessary flexibility, will suffice. A new and radical proposal is needed.

To utilize multiple qualities of service, several problems in the application must be solved. First, most computer applications of today are not structured to understand what service they actually need. They are written on operating systems which provide a virtual view of time and space, so they have few tools to discover the real rates of operation. Second, if applications are allowed to specify needed qualities of service, they must be constrained from optimizing their particular access by requesting unneeded services. An obvious control is the use of billing policy: billing for the requested service rather than the utilized one. Third, useful parameters for quality of service must be identified. The simple ideas of rate and latency do not capture effectively the complex patterns of bursty traffic, transient overload, and modes of service degradation.

What is thus needed is a proposal for a new form of basic network service, and a proposal for a number of higher level services which can be built from it and multiplexed over a single network.

Addressing -- Current networks, both voice (telephone) and data, use addressing structures which closely tie the address and the physical location on the network. That is, the address identifies a physical access point, rather than the higher level entity (computer, process, human) attached to that access point. In future networks, this physical aspect of addressing must be removed.

Consider, for example, finding the desired party in the telephone network of today. For a person not at his listed number, finding the number of the correct telephone may require preliminary calls, in which forwarding advice is given to the person placing the call. This works well when a human is placing the call, since humans are well equipped to cope with arbitrary conversations. But if a computer is placing the call, the process of "redirection" will have to be architected as a core service of the network. Since it is reasonable to expect mobile hosts, hosts that are connected to multiple networks, and replicated hosts, the issue of mapping to the physical address must be properly resolved.

To permit the network to maintain the dynamic mapping to current physical address, it is necessary that high-level entities have an address (some call it a name or a logical address) which identifies them independently of location. Such an address must be maintained by the network, and mapped to the current physical location as a core network service. The needed function is somewhat similar to the 800 service now provided by the telephone system, but must be considered in the context of the mobility and address resolution rates if all addresses in a global data network were of this sort.

Interfacing -- As networks get faster, the most significant bottleneck will turn out to be the packet processing overhead in the host. While this does not restrict the aggregate rates we can achieve over trunks, it prevents delivery of high data rate flows to the host-based applications, which will prevent the development of new uses for the available bandwidth. The host bottleneck is thus a serious impediment to the expansion of networking.

For highly parallel computers, the problem will be even more severe. Networks of today are intrinsically

serial. This requires that all data to be sent from a computer be gathered at one point in the computer for transmission to the network. This implies a single hot-spot, the sort of congestion that parallel computers attempt to avoid. If data transmission at significant rates is required for a parallel computer, a more parallel access method may be required, in which several parallel data paths are provided to different regions of the computer.

For devices other than traditional computers, such as video or image generators, a similar set of problems must be solved. Some interface technology must be developed which supports the needed services with acceptable overhead.

Accounting and Access Control -- The current connectionless transport architectures such as TCP/IP or the connectionless ISO configuration using TP4 do not have good tools for accounting for traffic, or for restricting traffic from certain resources. Building these tools is difficult in a connectionless environment, because an accounting or control facility must deal with each packet in isolation, which implies a significant processing burden as part of packet forwarding. This burden is an increasing problem as switches are expected to operate faster.

The lack of these tools is proving a significant problem for network design. Not only are accounting and control needed to support management requirements, they are needed as a building block to support enforcement of such things as multiple qualities of service, as discussed above.

Autonomous Administration -- Most current protocols do not properly support the idea that a large network will be composed of regions which must be separately administered: private segments (e.g. campus networks), regional networks (e.g. metropolitan area networks) and more than one transit service for long distance interconnect. We deal today, as a practical problem, with a very large network which has this structure while lacking the tools to manage it; it is clear to us that this is a pressing problem.

Explicit tools for policy routing are needed, which permit human specification of policy concerns, and proper translation of these concerns into routes that can be efficiently managed.

Security -- Any new network architecture must take into account the needed aspects of network security. Attempts to add these features after the fact are not generally effective.

Today we suffer from a lack of a proper model of requirements, as well as a lack of mechanism. The term security can be used to imply protection of network assets (bandwidth), end-node access control, or data integrity and disclosure control. Each of these is required in some form. but there is no consistent view of division of responsibility between network and end-node in providing the services.

7.3. New Network Architecture

These problems interact in a problematic way. If the only goal of a new network architecture was high speed, reasonable solutions would not be difficult to propose. But if one must achieve higher speeds while supporting multiple service, and at the same time support the establishment of these services across administrative boundaries, so that policy concerns (e.g. access control) must be enforced, the interactions become complex.

We believe that to address the above issues, we must revisit some of the basic assumptions which now underlie data networking. In particular, we must rethink the role of packets in the network, and we must revisit the debate between connection and connectionless services.

Central to our research approach is the concept we call a flow. A flow is a sequence of packets passing from a source to a destination with the same set of requirements. That is, a flow is associated with a certain quality of service, a set of policy decisions about access and routing, a set of accounting records, and so on. The effort of setting up a flow is intended to be more or less what would be done for each packet in a connectionless network; the idea of a flow is that the results of this effort can be cached to good advantage.

A flow is thus somewhat less than a virtual circuit. It need not require advance reservation to create it, and the state associated with it can be lost without causing unrecoverable disruption to the sequence of packets.

The flow is central to the solution of most of the problems outlined above. Once the network understands the concept of a flow, it can be equipped with tools to associate a particular flow with a type of service, and controls to insure that the usage of the flow matches the service requested. This is an example of the critical role of the flow in network control. In a connectionless network with no flow concept, there is no way to monitor the sequence of packets to determine if its usage matches the proposed type of service. Without this information, control is impossible.

Similarly, by associating with a flow a set of access control or policy routing decisions, it becomes possible to reduce the overhead of these sort of controls to the point where reasonable performance is possible. If a control decision must be computed for each packet, increased performance will not be possible.

The other component of our research approach is the message, which we take to mean the unit of data which the host-based application wishes to transmit across the network. To control host-based overhead, it must be the message, rather than individual packets, which the host passes into the network. It must be the job of the network to break down the message into the units of multiplexing across the links. The message thus plays for the host a role analogous to the flow; it is an abstraction that permits processing to be performed on larger elements than the units of multiplexing.

To support a range of applications, such as bulk data transfer, video streaming or voice, a variety of message types must be defined. By mapping these onto the common network service, the flow, the network can integrate these services, and provide to each the necessary support, while permitting the different applications the needed variation in message level functions.

The isolation of the host from the overhead of packets is even more important as we see techniques other than classic packet switching proposed for high speed networks. There are several proposals now for high speed integrated voice and data networks, which use very small packets to permit voice transmission. If these are to be used for data, it is clear that the small packets must be reassembled into some larger aggregate before being passed to the host. The message captures the idea of the host-visible aggregate, as distinct from the unit of network multiplexing.

The overall goal of our research, then, is to explore various aspects of network design, using the core concept of flow and message as the basic architectural elements.

7.4. Research Approach

Our group has a tradition of practical engineering studies, as opposed to more formal analysis. We intend to propose new approaches, and to evaluate these by simulation, and by actual implementation in hosts and switches. Through involvement in certain collaborative efforts, we hope to influence and lead the evolution of network architecture for experimental networks which could be prototypes for the future.

7.5. Future Projects

A number of specific projects have been identified as contributing to our overall understanding of network architecture. These will be undertaken as studies, and as actual prototype demonstrations as permitted by the level of funding.

Further extensions to transport protocols -- We wish to explore algorithms for dynamic setting of rate controls, and the extension of the rate models to service classes other than bulk data transfer. In particular, a new transport protocol is needed to support the "remote procedure call" or "request-response" class of interaction.

In high-speed long-haul networks, almost all interactions of the above nature will require a transmission time less than the round-trip time of the network. However, if we are forced to use transport protocols of today, the delays caused by the end-to-end communication of control information will effectively preclude taking advantage of the bandwidth. In other words, the latency of the control function will dominate the interaction time.

What is needed is a transport protocol which operates "open loop" under normal conditions. To cope with congestion, our concept of a "congestion constraint" must be employed, rather than feedback. For this class of traffic -- medium to small messages being sent back and forth, the best constraint may be a very simple one: congested data gets buffered at the point of congestion.

For current networks, it is generally recognized that buffering is not an effective technique for congestion control. However, this insight does not directly map to the networks of tomorrow because of the great differences in speed and design. Buffering works for this service class precisely because it is only expected to deal with those transients of traffic too short to control by feedback. This bounds the amount of buffering needed (but the bound may be very high.)

To make this scheme work, one requirement is a transport protocol that can deal with highly variable delays across the network. While data will normally be delivered at network speeds, congestion may cause queueing, which will increase the delay by orders of magnitude. Traditional transport protocols, which use an estimate of the round trip delay as a basis for retransmission, will fail utterly under this circumstance. What is needed is a transport protocol that does not directly depend on an accurate round-trip estimate for proper operation.

The NETBLT protocol, developed by this group under the current DARPA contract, is an example of a protocol with exactly this feature. The design principles of NETBLT can be applied to create a new transport protocol that will work well for this class of service. We are thus in a position to design and demonstrate a system that deals with one of the hardest problems of high-speed networking: regulation of congestion for bursty traffic over high-speed long-delay links.

More generally, we propose to explore the use of alternative models of flow and congestion control, such as our rate based flow models, to determine their applicability to high speed networking. This research is

particularly relevant at this time, since the emerging standards for high-speed transit networking in the common carriers have a form of rate-based control in them.

Network interface architecture -- To achieve high speed, it is clear that the interface between the network and the attached device (host or other data source) must be reconsidered. The hosts and operating systems of today cannot achieve high speeds if they must deal with individual packets, so it is necessary to divorce the technique used to multiplex the bandwidth on the network (circuit or packet switching) from the network model which the network interface presents to the host. Protocols will have to be moved outboard to the interface, but this will require a restructuring of the protocols.

The network interface must permit the host to transmit messages with the same overhead that would arise from a disk interface. To prevent the need for a copy of the data in main memory, the interface must be able to direct incoming messages to the correct location in memory. This will require that messages have names, which are associated at the receiving end with buffer areas into which the message is to be transferred. In this way, the processing currently done by the host to deliver the packet is eliminated, and the host need not perform any copy operation to move the data to the proper region of memory.

We have an initial specification for the architecture for the host interface to the network, which describes our idea of named messages, and discusses the relation of messages to buffers and to flows. See "Host-Network Interface Architecture", attached in section 10.

Policy architecture -- We believe that an explicit language is required for specifying policies related to routing, access restrictions, and accounting. This language would permit operators of regions of the network to specify the services they are willing to offer, and would in turn permit the identification of global paths that are consistent with each region's policies.

The major problems with policy is not the specification, but the testing for consistency and the selection of the preferred route. Since policies reflect the desires of persons, disagreements about policy may arise, and the system cannot be expected to resolve them. The persons involved must resolve them. But the system must detect these disagreements, and continue to function in some effective way in the presence of disagreement. This is in contrast to most current routing algorithms, which must be consistent in a global way to avoid routing loops.

Second, if several routes are permitted by the existing policy assertions, one among them must be selected. The procedure for selecting is not clear. Traditional routing algorithms use a minimization of something (e.g. some cost metric) to find the best routes. But policy concerns are not susceptible to summation and minimization. Some other metric will have to be devised to permit competing routes to be ranked.

Note that the evaluation of policy concerns can be arbitrarily complex. It is not reasonable to imagine performing this evaluation for each packet, or even each message. Only by creating some form of flow, and binding the policy decision to the flow, can we expect to manage policies in an efficient manner. Policy enforcement is an excellent example of the conflict between the increasing complexity which is required in the network to meet external requirements, and at the same time the increasing simplicity which is required by the higher switching speeds.

We have a preliminary proposal for policy routing. See "Policy Routing in Internet Protocols", attached in section 10. We propose to refine this model, through better identification of policy requirements and implementation alternatives, and (if funding is provided under Option 2) to demonstrate these ideas.

Our model of policy routing provides an effective means for accounting for network usage, which we believe is a central part of any set of policy controls for network assets.

Techniques for flow management -- Our research in rate based flow control will, if successful, provide a model of how one aspect of flows, bandwidth, can be managed. In parallel with this, it is necessary to develop a management architecture for flow state. A simple model for flows is that they must be explicitly created before data can be sent. This model, rather similar to the connection model of network service, is not desirable for several reasons. It requires a setup delay before sending, and it is not robust in the face of switch failure.

Our view is that a flow comes into existence by being used, rather like a path in the woods. An isolated packet may just be too small to account for or control. But it lays down a trace in the switches it passes through, and as more and more packets on the same flow pass by, the significance of the trace increases. As the flow starts, and periodically, the end point of the flow will label the flow, by sending out some control information which describes the desired quality of service, the billing authorization, and so on. This information is captured by the switches through which the flow is passing, where it is associated with the flow identifier so that it can be quickly found for subsequent packets. If no packets pass over the flow, the switch will gradually forget about it.

If a flow has not been labeled, the host may not obtain the desired quality of service. Indeed, a failure of the net to deliver the desired quality of service is the signal that relabeling is needed. Just as a host detects non-delivery of packets and retransmits, it should detect other failures of the desired quality of service and take corrective action. This technique will permit the reconstitution of flows after a switch has crashed, which permits this architecture to have the robustness of the connectionless network model which having the control state of the connection model. The flow model thus represents a potential hybrid of the connection and connectionless models with the good features of both.

We propose to explore these ideas of flow management by producing a more detailed design, by simulation, and by eventual implementation in an experimental gateway or switch element. By means of prototype code development, it should be possible to get a preliminary idea of the cost of processing flow state information in the switch.

Support for service types -- Once we have effective algorithms for flow management, we can then integrate into the switches support for various types of service, and then add to our host interfaces support for the new applications which require these services. We are currently beginning the simulation of algorithms for management of network resources in the presence of multiple types of service, using our ideas of rate based flow control, soft state in switches, and explicit interaction with the application to select the needed service.

Addressing -- We believe that the next generation of network should support an address space of end-node identifiers that are not related to the physical location of the end-node. These identifiers should be bound to the physical location as a flow is created, and the physical location should be cached by the switch. A hierarchy of servers should be provided to store the current location of each end-node, which the switch can query as needed.

This design can have severe problems with reliability (failed servers) and performance (excessive lookups overloading the servers.) But a very preliminary assessment suggests that a proper design could be made to work, even with a network the size of the current phone system. We propose to explore such a design

in more detail.

An addressing scheme such as this would solve several problems with the current Internet protocols, such as mobile hosts, multiple attachment of hosts, partitioned networks, and so on. With the increasing number of small and portable hosts, an addressing structure such as this is critical.

Security -- The problem of security (in its various forms) is being widely explored, and we do not believe that we should launch a major effort in this area. We must work with those now exploring the various issues of security to integrate the proper tools into our proposals.

There is one aspect of security in which basic architectural work seems to be missing. Most models of security have one of two basic architecture. Either the end-node performs the security functions, and the network is transparent, or the network has a centrally managed security function.

Neither of these is satisfactory for the network of tomorrow. Because the network of tomorrow is a global network, composed of parts connected together as peers, there can be no central, globally trusted security function. At the same time, it is not acceptable for all security functions to be pushed out to the boundaries of the network. Inside a secure environment, the expense of complex security checks will not be tolerable. What is needed is some model of security mediators, which are interposed into the network as a part of the policy architecture. These mediators will define regions of "unequal trust", between which explicit security procedures will be completed.

We propose to study the needs of security in the global network, and to propose mechanisms as part of our overall architecture.

7.6. Flow Performance

The key element of our research approach is that the flow concept will permit the design of forwarders and interfaces at speeds of a gigabit or greater. To demonstrate the power of the flow model, we provide a simple analysis of switch performance, using a traditional Internet gateway as a comparison.

Detailed analysis of a current gateway, for example the MIT Internet gateway, shows that at the Internet level there are four significant tasks. These are:

1. Checking the packet for errors.
2. Modifying the IP header.
3. Looking up the route.
4. Formatting the outgoing local network header.

The flow model essentially eliminates all of these tasks.

There are a number of error checks that are performed, such as testing the version number, and verifying the header checksum. These checks need not be performed if the state of the flow is cached in the switch, since the stored values in the switch can be checked once and then trusted. The only check required for each packet is to compare the actual length of the packet with the value in the header, to insure that a DMA error has not truncated the packet.

The changes which the switch makes to the IP header are also a form of error detection. The gateway decrements the time to live field to detect routing loops. This test is not needed on each packet, only if the

flow is disrupted and reconstituted. But so long as the flow is stable, the route will be consistent, and loop detection is not needed. If there were no change to the time to live, there would be no change to the IP checksum, and the header would not be changed in any way by the switch.

Looking up the route need be done only if the route has changed, which is uncommon. The result of the lookup can be saved, and only recomputed if a routing update occurs. The result of the lookup can be saved in the form of an actual local network header, which in most cases will eliminate any processing requirements. The only required processing is to find the preformed header and add it to the packet.

In the flow switch, the following steps are required.

1. Find the flow identifier in the packet and look up the flow state. This could be done using a CADM (a content addressable data memory, an available chip) or in software using a hash table. A demonstration with a small number of flows could perform a software lookup in under 10 instructions.
2. Attach the output local header (a DMA chaining operation).
3. Thread the packet onto the proper queue (as found in the state information). By pre-selecting the output queue, a wide variety of resource allocation decisions can be made with no per-packet processing.
4. Check that queue to see if congestion control is needed.

These four steps, if carefully coded, could easily be fewer than 50 instructions, even without special hardware help. Using a RISC chip at 10 mips, which is conservative, would permit a demonstration without special forwarding hardware at 200,000 packets per second.

Of course, there are many other limits to packet forwarding. In the current gateway, more instructions are allocated to restarting the I/O device than to the IP level processing. This odd imbalance results from the excessive generality of the devices and the gateway. Interfaces for fast networks will have to be assigned their own control processor, and this will require rethinking such basic tasks as buffer allocation. However, a preliminary design suggests that these problems can be solved with careful design.

The other performance problem is the speed of the bus and memory. Actual performance of these elements is always less than one would hope. Details of timing can conspire to reduce efficiency, sometimes by an order of magnitude.

The throughput problem is real, but not fundamental and not central to the innovative aspects of the research. That is, any box that goes fast, whether workstation or gateway, must carry through a careful integration of bus, memory and device. Such computers do exist, and serve as a demonstration that the needed bit rates can be achieved. The interaction of the data flow with the processing at the flow level create some additional problems, but it is not hard to propose a design for a flow forwarder which has data rates of a gigabit or more.

For our proposed demonstration, the key decision will be to determine how much effort to put into the bit rates of the units, as opposed to the processing rates. Careful balancing of cost against the achieved performance will be required, as discussed below.

7.7. The Demonstrations

Our proposals will not be very compelling if they are only paper studies. What is needed is quick prototypes, which can validate our designs and provide convincing evidence to the community.

Our goal is to define experiments which strike a balance between benefit and complexity. For example, the development of special hardware may permit a higher actual performance to be demonstrated, which is more compelling; on the other hand the effort of hardware development may be intolerable for a demonstration. We must pick a design point for our experimental platform based on our best estimate of the experiments to be done during its useful lifetime.

To demonstrate our ideas in flows and messages, and the various related ideas (congestion control, multiple service types, policy, addressing, etc.) we require two demonstration elements, and switch and a host interface. We discuss each in turn.

Flow Switch -- Our proposed switch is intended to demonstrate the idea of flow state, and to indicate the sort of performance improvement which this idea can yield.

The emphasis of the demonstration would be on the flow and packet processing, rather than on a high degree of cross-connect. This would make the switch resemble in configuration a current gateway. That is, it would connect together a limited number of packet forwarding media, and deal with the issues of conversion between different technologies.

A first demonstration, with only modest performance requirements, would be in terms of classic packet switching as the multiplexing technique, so that we can take advantage of existing networks. We would connect to emerging LANs such as FDDI, and long-haul technology such as DS-3 (in particular the recently announced Bellcore SMDS service). However, a higher performance demonstration is preferable. We are currently developing a plan for a joint development of a network using the small or "fast-packet" multiplexing techniques. If this can be arranged, the switch would also be used to interconnect to that technology.

We believe that an effective demonstration can be done using a collection of board level elements with a high-speed backplane. Because of the wide availability of units, a popular bus such as a VME is most obvious, but does not meet the more demanding of our demonstration targets, which require a bus with a gigabit speed to permit flows at 500 mbps. With a VME bus, it will be difficult to achieve much over 100 mbps data throughput. In fact, depending on details, 50 mbps may be more reasonable. We would prefer a faster platform, but must be careful to define an approach that does not require excessive hardware development. Through collaboration with other researchers and product developers, we hope to define a suitable approach.

We believe that a commercial RISC processor such as the SPARC or the M-8800 would be a suitable first processing element. The 8800 is now available on a VME board, but the suitability of the memory architecture has yet to be determined. The analysis in the previous section suggested that such a processor could perform the flow level processing in under 50 instructions. Additional processors should be allocated to manage the I/O devices. Overall, a target of 100 instructions, or 100,000 packets a second, is not unreasonable. Gateways today can forward perhaps 1,000 packets a second, without the use of special hardware.

The cost estimates for Option 1 (which covers this demonstration) are based on the assumption that minimal hardware development is required. We propose to review our project plans with DARPA as the

targets for demonstration and the options for collaboration become more clear.

Whatever bus is used for our flow forwarder, it will be an open bus which permits the development of new interfaces. This will allow other DARPA efforts that develop alternative high-speed network technologies to use this unit to perform a gateway function consistent with the flow model. In this way, we can show the generality of the flow model, and support practical interconnection and interfacing of other technologies.

Host Interface -- The same platform proposed for a switch can also perform as a host interface. Instead of connecting to a number of networks, it would connect to one network, and to the bus or I/O interface of the host. For a host with a VMEbus, for example, this connection might be a VME to VME converter, which is a commercial product. (The performance constraints of such a device must be explored.)

The hardware platform would execute rather different code in this case, dealing with message reassembly and delivery. However, the support code (I/O drivers and operating system functions such as timer management) would be shared between the switch and the host interface.

The overall performance of a demonstration based on a VME bus would be perhaps 50 mbps. Again, a higher speed platform would permit a more interesting and diverse set of demonstrations.

Policy demonstration -- The demonstration of policy routing requires two components, a policy gateway and a policy controller. The policy gateway connects two autonomous regions, and checks each packet for suitability. The policy controller exists in each region, and supports control tasks, synthesis of routes, the exchange protocols with other regions, human interfaces and so on. In addition to these components, there is host software needed.

Each of these components could be built on the hardware platform proposed above, or on a current gateway platform such as the MIT gateway. The gateway is just another example of a flow-based forwarder, and would be easily implemented on the proposed flow switch.

The controller is a more complex element, because of its diverse functions. Because (in the migration step where host support is not available) it must process every packet leaving the region, it must have the performance of a gateway. In addition, it must run background programs of some complexity for such things as route synthesis.

For a complete demonstration, it is also necessary to set up several autonomous regions. For this purpose, one of the T1 paths in the "RIB" link could be used to implement a transit region.

As part of developing a plan for Policy controls in the Internet, it may be desirable to perform other experiments in this area, for example to explore other shorter term options for control. Under this funding, we are prepared to perform such experiments as appropriate.

Fast packet network demonstration -- We are currently in negotiation with Bellcore to perform a joint demonstration of a transit network based on the idea of "fast packet space division" networking; a network based on the Batcher-Banyon sorting fabric transporting small fixed-size packets among a number of trunks. The proposed fabric can switch among 256 streams, each at 140 mbps, for a total switch capacity of 35 Gbps. This technology is sometimes called Asynchronous Transfer Mode, or ATM.

We believe that this technique is the most promising for high speed networks of tomorrow. It matches the

infrastructure now being installed by the telephone companies, it avoids the need for exotic high-speed technology by means of parallelism, and it has been demonstrated in the laboratory. What is now needed is an experiment in the field.

Such an experiment can only be done effectively in concert with the Bell Operating Companies or other carriers, because of the need for very expensive trunking. For this reason, there are procedural complexities associated with setting up such a program, and we are not in a position to make a specific proposal to DARPA in support of such a collaboration at this time. However, our intention to perform such an experiment is an integral part of this proposal, and we have included this outline of the effort to convey the total scope of the intended research.

The hardware platform proposed for the flow switch and host interface is intended to be compatible with this fast packet demonstration. As this network project becomes more refined, the performance requirements on the platform will be re-examined before committing to a particular hardware approach.

8. Expected Results

Our long range goal is to factor together a model for resource allocation in the network, speculatively our rate control algorithms, together with a new host interface structure, and propose a network architecture which hides the details of the switching paradigm, e.g. packet switching or circuit switching. If successful, our architecture will permit these various sorts of switching technologies to be used collectively to provide end-to-end service, with the switching details hidden behind the host interface architecture. In this way, existing and new network technologies can be combined to provide a new generation of services, with better performance and scaling.

There is a parallel between our goal and the goal of the original Internet project 15 years ago. Internet attempted to find a structure within which a variety of network technologies could be interconnected. Its success can to a large extent be attributed to this achievement. Advanced network research today has not addressed the issue of heterogeneity of approach. Techniques for switching and modulation have been proposed as if they are to be the only technology in the next generation network. Our project, by addressing architecture in a more abstract way, divorced from the details of switching and multiplexing, will provide a way to intermix various new and existing network technologies, even those as diverse as packet switching and circuit switching, as a part of an overall network infrastructure.

Our primary result is thus an architecture, which is embodied in a set of protocols, and a set of assumptions about how these protocols are used. This architecture will permit the integration of a number of diverse techniques for switching and multiplexing, will permit operation in a new and higher range of speeds, and will support new classes of applications.

The second result will be the demonstration of this architecture in a form that permits validation of the concept and trial support of applications.

Depending on the funding and the interest, we would hope that a quasi-operational capability might be put in place, rather like the early ARPANET, in support of a restricted class of users. We look in particular to the users of supercomputers or the advanced parallel computers being developed by DARPA. We believe that remote access to these machines would represent a demonstration critical both to the network and the parallel computing efforts.

We do not imagine that our demonstrations will be directly transferable to the commercial world as products. They will represent advanced capabilities which may not be economically viable in the short run, and since they are viewed as demonstrations, additional effort would be required to develop equivalent products.

However, our group has a strong history of successful technology transfer in this area, and we will work, as we have in the past, to find an effective path to transfer our demonstration technology to the commercial world.

The policy routing effort can have an impact in the existing Internet-based protocol context. We expect that work, if carried to a demonstration level, might have a direct transfer to the operational Internet and the vendors who support it.

9. Previous Work

9.1. Present Activities

Currently, a number of projects are underway, which present starting points for the proposed research. We summarize the more relevant of these.

New transport protocols -- As speeds increase, window flow controls become ineffectual. Additionally, as networks grow larger, we are seeing congestion oscillation effects which seem to result from the use of windows for flow control. Based on these observations we have designed, implemented and tested a new transport protocol, which uses an alternative flow control model, rate controls. Initial experiments indicate that this approach, especially when coupled with a sophisticated error recovery scheme, can achieve high data rates, even over long-delay paths, without causing the same degree of congestion oscillation.

The long range goal of this effort is to understand how flows might better be modeled in the network. We believe that rates are the most basic measure of flow capacity, rather than windows, and that by modeling rates directly, we can get a more effective measure to characterize flows. This particular project is a short range effort to explore the limits of rates as a flow control model.

Multiple service types -- A PhD student is exploring techniques for allocating resources in a network based on the rate based flow model described above. The thesis of the research is that by identifying explicit rate controlled flows, resources can be allocated among the flows in an effective manner.

Virtual circuit technology -- As part of a joint experiment with ATT Bell Labs, we expect to receive a Datakit packet switch. This will be used to explore the virtual circuit model of networking, and to understand the interactions between this model and the connectionless model of Internet. This experiment is part of a nationwide network of Datakit switches being deployed by Bell Labs.

Interactive network simulation -- The group has developed a network simulator which, in addition to the normal gathering of statistics, permits the visual inspection of the network state as the simulation proceeds. It is our experience that such a visualization permits an intuitive understanding of the problem being simulated.

9.2. Past results

The proposed work will be carried out in the Laboratory for Computer Science at M.I.T. Both in this lab and elsewhere at M.I.T. there is a strong tradition of research in communications and networking.

For over ten years, this group has been a major contributor to the DARPA Internet project, providing leadership of the IAB, and participation on various working groups. It has provided architectural guidance to the project, and performed a number of demonstrations and evaluation efforts.

The group has a strong history of technology transfer of DARPA-funded research. In the mid-seventies it worked on the concept of the token ring LAN. This led directly to commercial products, it led to the current IEEE standard for ring-based LANs, and it laid the groundwork for the higher speed LANs such as FDDI now being proposed as standards.

The group developed one of the early gateway implementations. This software was used for a number of

Internet experiments, and is still in use on the M.I.T. campus as the basis for the campus network. In the commercial world, it formed the basis of the router product sold by Proteon, Inc., one of the most successful vendors of Internet gateways.

The group developed an implementation of TCP/IP for the IBM PC. Initially done as a quick demonstration to prove the feasibility of running this class of program on a PC, it became a very popular package for the PC, and brought the PC into the Internet world. Four companies sold or are selling a supported version of this code, including FTP Software, Inc, which was founded by a former member of this group.

9.3. Information on Investigators

David Clark (Principal Investigator for this proposal and leader of the Advanced Network Architecture group) received his Ph.D. from M.I.T. in 1973, and is currently a Senior Research Scientist at the Laboratory for Computer Science. His research interests include networks, protocols, operating systems, distributed systems and computer and communications security.

After receiving his Ph.D., he worked on the early stages of the ARPAnet, and managed the development of one of the first host implementations of the ARPA network protocols. Following this effort, he worked on local area network technology, and was one of the developers of the token ring LAN. This effort directly led to current commercial products, and was the origin of the IEEE 802.5 token ring standard.

Since the mid 70s, Dr. Clark has been involved in the development of the DoD network protocol suite: TCP/IP or Internet. For several years he was Chief Protocol Architect for the Defense Advanced Research Projects Agency (DARPA) in this development, and currently chairs the Internet Activities Board, a steering committee sponsored by several agencies and with ties to industry, which attempts to guide the evolution of TCP/IP as it has entered the commercial world.

As a part of this work in protocols, Dr. Clark has made an extensive study of protocol efficiency. He has presented tutorials on problems of protocol performance, and wrote an implementation guide for TCP. Under his guidance, an operating system was designed and implemented at MIT, to show that a major impediment to effective data throughput is the internal structure of existing systems.

In the security area, Dr. Clark consulted on the development of a secure version of the Internet architecture, for the protection of DoD classified information. He has also recently written a paper describing a security model derived from commercial as opposed to military practices. This model, which stresses integrity of data rather than classification, was the subject of a recent workshop, and may well provide a formal basis for commercial security.

Relevant publications:

Refereed Articles:

1. D. Clark, M. Lambert, L. Zhang, "NETBLT: A High Throughput Transport Protocol", *Frontiers in Computer Communications Technology: Proceedings of the ACM-SIGCOMM '87*, Association for Computing Machinery, Stowe, VT, August 1987, pp. 353-359.

2. D.D. Clark, D.R. Wilson, "A Comparison of Commercial and Military Computer Security Policies", *Proceedings of the 1987 IEEE Symposium on Security and Privacy*, IEEE, Oakland, CA, April 1987, pp. 184-194.

3. K.R. Sollins, D.D. Clark, "Distributed Name Management", *Proceedings of the IFIP WG 6.5 International Computer Message Systems Working Conference*, IFIP WG 6.5, Munich, Germany, April 1987, pp. 2.3.1-1.3.19.

4. David D. Clark, "The Structuring of Systems Using Upcalls", *Proceedings of the 10th ACM Symposium on Operating Systems Principles*, Association for Computing Machinery, Oakland, CA, December 1985, pp. 171-180.

5. Jerome H. Saltzer and David D. Clark, "Why A Ring", *Proceedings of the Seventh Data Communications Symposium*, IEEE, Mexico City, Mexico, October 1981, pp. 211-217.

6. J.H. Saltzer, Reed, D.P., Clark, D.C., "End-to-End Arguments in System Design", *ACM Transactions on Computer Systems*,Vol. 2,No. 4,November 1984, pp. 277-288.

7. J.H. Saltzer, D.D. Clark, J.L. Romkey, and W.C. Gramlich, "The Desktop Computer as a Network Participant", *IEEE Journal on Selected Areas in Communications*,Vol. SAC-3,No. 3,May 1985, pp. 468-478.

8. D.D. Clark, K.T. Pogran, D.P. Reed, "An Introduction to Local Area Networks", *Proceedings of the IEEE*, IEEE, November 1978, pp. 1497-1517.

9. D.Clark, B.Halstead, S.Keohan, J.Sieber, J.Test, S.Ward, "The Trix 1.0 Operating System", *Distributed Processing Quarterly: Special Issue on Distributing Operating Systems*,Vol. 1,No. 2,December 1981, pp. 3-5, Published by the IEEE Computer Society Technical Committee on Distributed Processing

10. Saltzer, J.H., Reed, D.P., and Clark, D.D., "Source Routing for Campus-Wide", *Proceedings of the IFIP Working Group 6.4, International Workshop on Local Networks*, , Zurich, Switzerland, August 1980, published in "Local Networks for Computer Communications," pp.1-25 by North Holland Publishing Company

11. Clark, D., Svobodova, L., "Design of Distributed Systems Supporting Local Autonomy", *Proceedings of COMPCON '80*, IEEE, San Francisco, CA, February 1980, Invited Paper

12. Schroeder, M.D., Clark, D.D., and Saltzer, J.H., "The Multics Kernel Design Project", *Proceedings of the ACM Sixth Symposium on Operating Systems Principles*, ACM, Purdue University, Lafayette, IN, November 1977, Published in ACM Operating System Review, Vol. II No. 5 pp. 43-56

Other publications and talks:

1. David D. Clark, "Window and Acknowledgment Strategy in TCP NIC-RFC-813", *DDN Protocol Handbook*,Vol. 3,July 1982, pp. 3-5 to 3-26.

2. David D. Clark, "Name, Addresses, Ports, and Routes NIC-RFC-814", *DDN Protocol Handbook*,Vol. 3,July 1982, pp. 3-27 to 3-40.

3. David D. Clark, "IP Datagram Reassembly Algorithms NIC-RFC-815", *DDN Protocol Handbook*,,Vol. 3,July 1982, pp. 3.41-3.49.

4. David D. Clark, "Fault Isolation and Recovery NIC-RFC-816", *DDN Protocol Handbook*,No. 3,July 1982, pp. 3.51-3.62.

5. David D. Clark, "Modularity and Efficiency in Protocol Implementation NIC-RFC 817", *DDN Protocol Handbook*,Vol. 3,July 1982, pp. 3.63-3.88.

David Tennenhouse (Assistant Professor with the Advanced Network Architecture project) will receive his PhD from the University of Cambridge, England, and join the project in September.

His work has concentrated on the development of a model for site interconnection and design of a related network architecture based on Asynchronous Transfer Mode (ATM) communication. The ATM approach is characterized by the use of very small packets to support rate adaption, service integration, and dynamic bandwidth allocation. The packets are switched between sites with throughput, delay and jitter attributes that are competitive with conventional circuit switching. He has been working towards an overall site interconnection architecture which address issues such as addressing, routing, bandwidth management, network interconnection and subscriber attachment. His focus is in contrast to the other work in the area, which seems confined to a narrower scope, concentrating on either ATM switching or the format of the subscriber interface. His PhD thesis is based on experience gained in the design and implementation of a pilot network. It describes an overall service model, the Exchange site interconnection architecture, the implementation of a switch fabric based on a fast slotted ring, the extension of ATM by means of a packet overlay operating over primary rate ISDN carriers, and the analysis of experimental results derived from the network.

He has been active in a number of Standards activities, including ISO TC97/SC16/WG1, Open Systems Interconnection Architecture.

Relevant publications:

1. Tennenhouse, D.L., *Site Interconnection and the Exchange Architecture,* PhD dissertation, Cambridge University, UK, 1988.

2. Tennenhouse, D.L. et al, "Exploiting Wideband ISDN: The Unison Exchange", *Proceedings of the IEEE INFOCOM*, IEEE, San Francisco, CA, March 1987.

3. Burren, J., Adams, C., Pitura, H., Tennenhouse, D., and Leslie, I., "Variable Data Rate Channel for Digital Network". UK Patent Application No. 8618424, London. (Also subject of further UK, US, and European patents.)

4. Leslie, I.M. and Tennenhouse, D.L., "Cambridge Networks and Distributed Systems", *Proceedings of COMPINT*, COMPINT, Montreal, September 1985.

5. Patkau, B.H. and Tennenhouse, D.L., "The Implementation of Secure Entity-Relationship Databases", *1985 IEEE Symposium on Computer Security and Privacy*, IEEE, Oakland, CA, 1985.

6. Tennenhouse, D.L., "Permute: A Language Control Structure", Master's thesis, University of Toronto, 1981.

Michael L. Dertouzos is Professor of Computer Science and Electrical Engineering at the Massachusetts Institute of Technology, and Director of the Laboratory for Computer Science. His research interest is the development and application of multiprocessor systems. He will be available to provide advice and guidance to this project.

10. Bibliography

Attached are working papers and publications which provide background for the proposed research.

"Designing a New Architecture for Packet Switching Communication Networks" Lixia Zhang, IEEE Communications Magazine September 1987

This paper, by a graduate student in the group, provides another view of resource management in networks.

"Host-Network Interface Architecture", David Clark, MIT LCS RFC 310, December 1987.

This group working paper describes our proposal for host interfacing, and describes the concept of the "named message".

"Policy Routing in Internet Protocols", David Clark, Unpublished.

This draft NIC RFC, now in limited circulation, is our proposal for policy routing. It describes the proposed demonstration in detail, including the policy gateway and the policy controller.

"Improving Gateway Performance With A Routing-Table Cache", David Feldmeier, Infocom 88

This paper, by a graduate student in the group, provides experimental support for the flow model.

Volume IIA

Project Management

11. Organization

Since the Massachusetts Institute of Technology's Laboratory for Computer Science has been a contractor to DARPA/ONR for the last 24 years, and its general management procedures are well known to DARPA, we shall confine our remarks in this section to the management issues that are specific to the proposed project.

The proposed work will be carried out in the Advanced Network Architecture group of the MIT Laboratory for Computer Science. Project direction will be the responsibility of the Senior Research Scientist in charge of that group, who will work in consultation with the other faculty members involved. He will set the project goals and priorities, manage the allocation of tasks to staff, monitor progress, and serve as the primary point of contact for other DARPA principal investigators.

12. Personnel

The group is currently 75% of the Principal Investigator, one staff member, and 5 graduate students. We have recently hired a new faculty member, Dr. David Tennenhouse, who will work in this area. This addition to our faculty represents a major commitment by the Department of Electrical Engineering and Computer Science, and by the Laboratory for Computer Science, to research in the area of advanced networking.

We anticipate that in 1990, Prof. Jerome Saltzer, currently on leave to the Athena project at M.I.T., will return to LCS to work in this area.

The funding level for the core proposal permits support for this level of staff. It should permit the studies and specifications to be completed within the term of the proposal.

For each of options 1 and 2, support at a minimum for 1 PhD level staff and one programmer plus hardware is required. The PhD staff will participate in and help supervise the demonstration, as well as help complete the architectural studies in a more timely manner.

If fully funded (Options 1 and 2), this proposal would thus support the following:

- 1 Faculty 50%
- 1 Faculty 50% starting 1990
- 1 Senior Research Scientist 80%
- 1 Post-Doctoral Associate
- 1 Research Scientist
- 3 Research staff
- 5 Graduate students

13. Collaboration

We believe that the proposed effort can best be carried out in the context of a coordinated national effort in high-speed networking. While that coordination may require the adjustment of individual research goals on occasion, the benefit that comes from collaboration is the ability to build on the work of others.

We stand ready to help lead such a collaboration as appropriate, ar d to participate toward mutual goals as best we can. Based on our long experience in the DARPA Internet project, we believe we understand the issues and pitfalls in such coordination, and we are sensitive to the problems that may arise. We hope that that experience will prove valuable if a significant effort is started in the area.

In support of collaboration on high-speed networking, we are actively exploring options for joint research. We are currently investigating the extent of joint interest with communications companies, computer companies, and companies dealing in relevant applications.

Because the exact nature of the collaborative activities will continue to evolve, we expect that there may be significant changes in the detailed emphasis of the project. We will keep DARPA informed, through regular status reports, of the relation of this effort to any larger activities.

We will pursue additional sources of funding as part of this collaboration. Such funding will be used to address the general goals outlined in this proposal, either broadening some aspects of the project or permitting more rapid progress in the studies and demonstrations.

14. Demonstrations

The pilot demonstrations identified in the proposal will require the development of software and hardware, and will result in operational units which could be replicated for use in a small number of interested research labs.

Our group is well equipped for the development of software, both for interface and switch elements, and for the supporting software in hosts. Our normal program development environment is UNIX on microVax work stations. We will continue to use this environment except as required to run specialized tools, such as RISC compilers.

To support hardware development, it will be necessary to equip a group facility, or obtain the use of a suitable facility by some arrangement. The proper approach to this problem will be determined after the scope of such an effort has been determined.

15. Technology Transfer

As discussed in the section on previous results above, this group has an excellent record of technology transfer. Our general approach to the goal will remain unchanged.

We do not believe that our demonstration elements will be suitable for direct recasting as products. We propose to demonstrate ideas which will require support from the communications carriers which may not be generally available at the time of the demonstrations. We believe that the general concepts will be applicable to the developers of switching software, and we will make our work available in that context.

Our work in policy routing is cast in the context of the current Internet protocols, and would be directly applicable. We will make this work available in whatever ways best facilitate its transfer into the

operational community.

The most direct form of technology transfer may be through the participation of industry in collaborative research.